













4D Write Pro Handbuch

-  Einführung
-  Einen 4D Write Pro Bereich erstellen
-  4D Write Pro Dokumente in 4D Objektfeldern speichern
-  4D Write Pro Bereich verwenden
-  4D Write Pro Dokumente drucken
-  4D Write Dokumente importieren
-  Ausdrücke in einem 4D Write Pro Dokument filtern
-  4D Write Pro Standardaktionen verwenden
-  Bilder verwalten
-  4D Write Pro Programmiersprache
-  Neuerungen der Version
-  Alphabetische Liste der Befehle

Einführung

Überblick

4D Write Pro bietet 4D Benutzern ein fortschrittliches Werkzeug für Textverarbeitung, das vollständig in ihre 4D Anwendung integriert ist. Mit 4D Write Pro können Sie vorformatierte E-Mails oder Briefe mit Bildern, eingescannter Unterschrift, formatiertem Text und Platzhaltern für dynamische Variablen verfassen. Sie können auch dynamisch Rechnungen oder Berichte mit formatiertem Text und Bildern erstellen.

4D Write Pro bietet folgende Schlüsselfunktionen:

- 4D Write Kompatibilität: Ein Objekt 4D Write Pro kann frühere 4D Write Dokumente öffnen und konvertieren und unterstützt dabei die meisten seiner spezifischen Eigenschaften.
- Textverarbeitung: Ein Objekt 4D Write Pro, das in ein Formular eingebunden ist, bietet standardmäßige Funktionen der Textverarbeitung, wie Text und Stil bearbeiten, Bilder einfügen, Import und Export, u.v.m.
- Integration in die Anwendung:
 - Ein Objekt 4D Write Pro kann dynamische Teile anzeigen, die mit Daten aus der Anwendung oder mit von 4D berechneten Werten gefüllt werden.
 - 4D Write Pro Dokumente lassen sich in einem Feld der Datenbank oder auf der Festplatte speichern.

Installation und Aktivierung

4D Write Pro ist kein Plug-In mehr, sondern direkt in 4D integriert. So lässt es sich leichter einsetzen und verwalten. Es ist keine zusätzliche Installation erforderlich, Sie können 4D Write Pro Bereiche in Ihre Formulare einfügen und 4D Write Pro Variablen direkt in Ihren 4D Anwendungen verwalten.

Beachten Sie jedoch, dass 4D Write Pro dieselbe Lizenz wie 4D Write verwendet, d.h. diese Lizenz muss in Ihrer Anwendung installiert sein, damit es funktioniert.

Anforderungen: Unter Windows basieren 4D Write Pro Features auf Direct2D. Auf Rechnern mit Windows 7 oder Windows Server 2008 müssen Sie sicherstellen, dass das *Plattform Update für Windows* installiert ist, damit die erforderliche Version Direct2D verfügbar ist.

Über dieses Handbuch

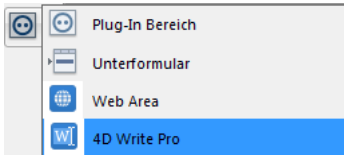
Dieses Handbuch beschreibt die Features der Benutzeroberfläche und Befehle der Programmiersprache von 4D Write Pro.

4D Write Pro Objekte lassen sich über spezifische **4D Write Pro** Befehle, aber auch über Befehle der 4D Programmiersprache steuern (siehe Kapitel ("**Objekte (Formulare)**" und "**Mehrfachstil Text**").

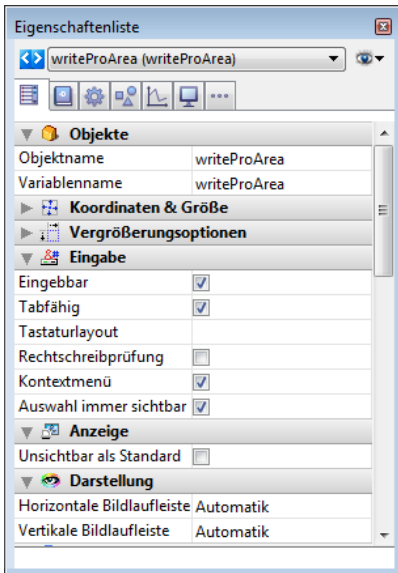
■ Einen 4D Write Pro Bereich erstellen

Bereich erstellen

In 4D lassen sich 4D Write Pro Dokumente in einem 4D Formularobjekt, genannt **4D Write Pro**, anzeigen und manuell bearbeiten. Es ist im Formularbereich in der Objektleiste unter dem Tool "Plug-In" verfügbar.



Ein 4D Write Pro Formularbereich wird über standardmäßige Eigenschaften der Eigenschaftensliste konfiguriert, wie **Objektname** und **Variablenname**, **Koordinaten**, **Eingabe**, **Anzeige**, **Darstellung**, bzw. **Ereignisse**.



Die Eigenschaft **Variablenname** lässt sich in der Programmiersprache als Referenz auf den 4D Write Pro Bereich verwenden. Beachten Sie, dass die Variable vom Typ *Objekt* sein muss. Weitere Informationen dazu finden Sie unter dem Befehl **C_OBJECT**. Die Eigenschaften für "Eingabe" bieten gängige Features zur Textbearbeitung:

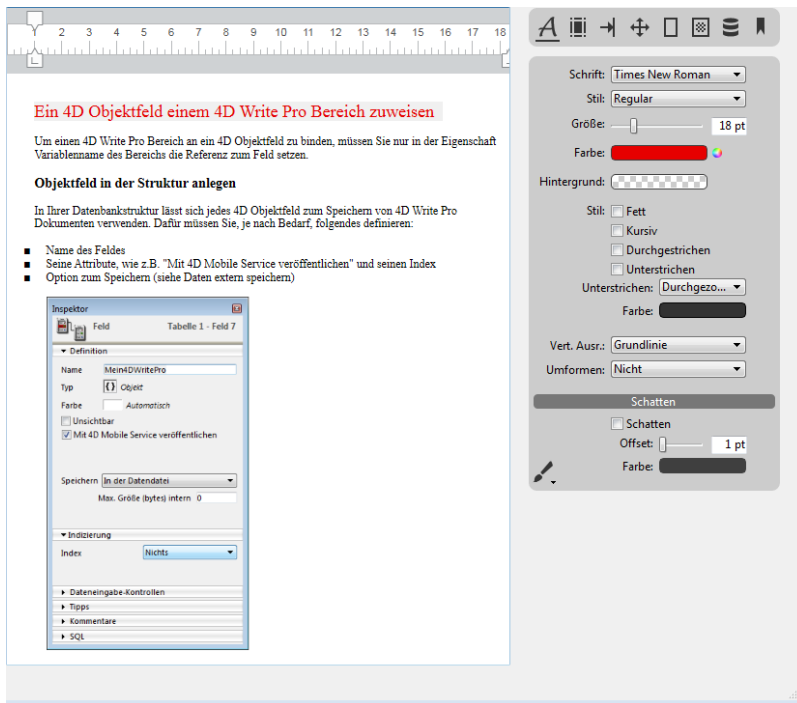
- **Eingebbar**: Sperrt bzw. entsperrt den Bereich, um die Bearbeitung zuzulassen bzw. zu verhindern
- **Auto Rechtschreibhilfe**: Ist für 4D Write Pro Bereiche verfügbar
- **Kontextmenü**: Aktiviert bzw. deaktiviert das Kontextmenü im Anwendungsmodus (siehe **4D Write Pro Oberfläche**)
- **Auswahl immer sichtbar**: Verwaltet die Textauswahl wie in standardmäßigen Textbereichen

4D Write Pro Widget der Objektbibliothek verwenden

Über das Objekt **4D Write Pro** in der Objektbibliothek (Kategorie "Eingabe Bereiche") können Sie einen vorkonfigurierten 4D Write Pro Bereich erstellen.



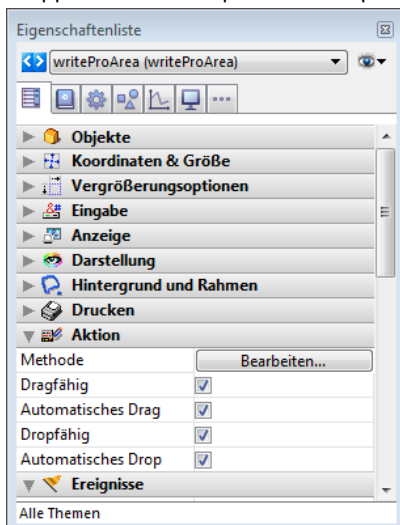
Er enthält ein Steuer-Panel zum Verwalten der Attribute, wie Schrift, Farbe, Stil, etc.:



Hinweis: Weitere Informationen dazu finden Sie im Abschnitt **4D Write Pro Area**

Drag and Drop konfigurieren

Um die Drag and Drop Features für Ihre 4D Write Pro Bereiche einzurichten, müssen Sie in der Eigenschaftensliste unter der Gruppe "Aktion" die passenden Optionen auswählen:



4D Write Pro Bereiche unterstützen zwei Modi für Drag and Drop:

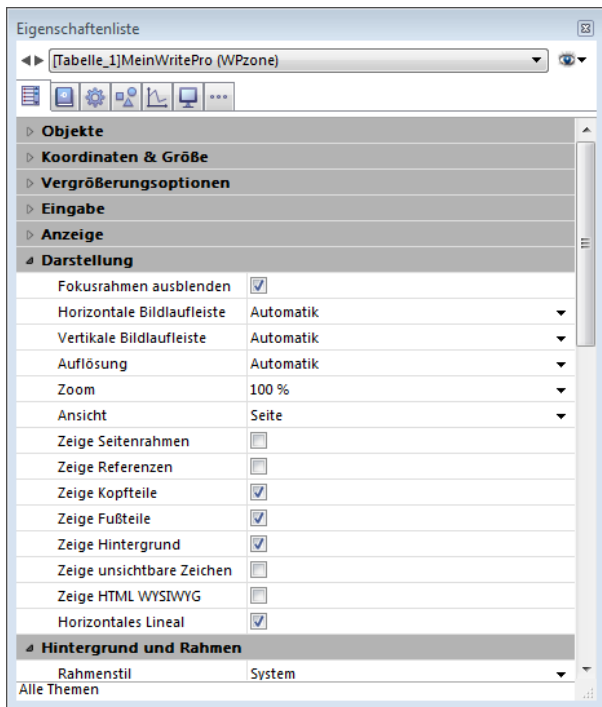
- **Eigener Modus:** Nur die Optionen "Dragfähig" und "Dropfähig" sind markiert. In diesem Modus können Sie Text auswählen und das Bewegen starten. Die Objektmethode wird mit dem Ereignis On Begin Drag Over aufgerufen, so dass Sie die Drop-Aktion über eigenen Code definieren können.
- **Automatischer Modus:** Die Optionen "Dragfähig", "Dropfähig", "Automatisches Drag" und "Automatisches Drop" sind markiert. In diesem Modus können Sie den gewählten Text automatisch bewegen oder kopieren (mit der Tastenkombination **Alt/Option**). Das Ereignis On Begin Drag Over wird nicht ausgelöst.

Hinweis: Wählen Sie nur die Optionen "Automatisches Drag" und "Automatisches Drop", hat das im 4D Write Pro Bereich keine Auswirkung.

Darstellungseigenschaften konfigurieren

Die Darstellungseigenschaften für 4D Write Pro Bereiche lassen sich jetzt direkt über die Eigenschaftensliste definieren. Damit können Sie z.B. einstellen, ob 4D Write Pro Dokumente genauso angezeigt werden, wie sie gedruckt oder wie sie eingebunden werden. Sie können im gleichen Formular verschiedene Ansichten desselben 4D Write Pro Dokuments setzen.

Die Eigenschaften für 4D Write Pro Formularobjekte sind in der Eigenschaftensliste unter der Gruppe **Darstellung** verfügbar:



- **Auflösung:** Setzt die Bildschirmauflösung für den Inhalt des 4D Write Pro Bereichs. Standardmäßig ist **72 dpi (Mac OS)** eingestellt, das ist die Standardauflösung für 4D Formulare auf allen Plattformen. Wird die Einstellung **Automatik** gewählt, ist das Rendern des Dokuments zwischen Mac OS und Windows Plattformen unterschiedlich. Wird ein spezifischer dpi Wert gesetzt, ist das Rendern des Dokuments für Mac OS und Windows Plattformen gleich.
- **Zoom:** Setzt das Zoomen in Prozent für den Inhalt des 4D Write Pro Bereichs. Standard ist 100%.
- **Ansicht:** Setzt den Modus zum Anzeigen des 4D Write Pro Dokuments im Formularbereich. Es gibt drei Optionen:
 - **Seite** (Standard): Zeigt die komplette Ansicht mit Seitenrand, Ausrichtung, Rahmen, Umbrüchen, Kopf- und Fußteilen, etc. Weitere Informationen dazu finden Sie im Abschnitt **Einstellungen für Ansicht als Seite**.
 - **Vorschau:** Zeigt die Vorschau mit den allgemeinen Dokumenteigenschaften
 - **Eingebunden:** Passende Ansicht für eingebundene Bereiche; sie zeigt keine Ränder, Fußteile, Kopfteile, Rahmen, etc. Dieser Modus eignet sich auch für die Darstellung im Web. Dafür müssen als Auflösung 96 dpi und die Option **Zeige HTML WYSIWYG** ausgewählt werden.

Hinweis: Die Eigenschaft **Seitenansicht** wird nur für das Rendern auf dem Bildschirm verwendet. Für Druckeinstellungen werden automatisch spezifische Render-Regeln angewandt. Weitere Informationen dazu finden Sie im Abschnitt **4D Write Pro Dokumente drucken**.

- **Zeige Seitenrahmen:** Blendet den Seitenrahmen ein/aus, wenn "Seite" als Seitenansicht gewählt ist. Standard ist ausgeblendet
- **Zeige Referenzen:** Zeigt alle eingefügten 4D Ausdrücke im Dokument als *Referenzen*. Diese Option ist standardmäßig nicht markiert, d.h. 4D Ausdrücke erscheinen als *Werte*. Fügen Sie ein 4D Feld oder Ausdruck ein, berechnet und zeigt 4D Write Pro seinen aktuellen Wert. Markieren Sie diese Option, können Sie sehen, welches Feld bzw. welcher Ausdruck angezeigt wird. Sie erscheinen im Dokument mit einem grauen Hintergrund. Haben Sie z.B. das aktuelle Datum mit einem Format eingefügt, erscheint das Datum z.B. wie folgt:
26. Juli 2016
Markieren Sie die Option **Zeige Referenzen**, wird die Referenz angezeigt:
`String(Current date;Internal date long)`
- **Hinweis:** 4D Ausdrücke lassen sich über den Befehl **ST INSERT EXPRESSION** einfügen.
- **Zeige Kopfteile/Fußteile:** Blendet die Kopf- und Fußteile ein/aus, wenn "Seite" als Seitenansicht gewählt ist (standardmäßig angezeigt). Weitere Informationen dazu finden Sie im Abschnitt .
- **Zeige Hintergrund:** Blendet die Hintergrundbilder bzw. -farbe ein/aus (standardmäßig angezeigt).
- **Zeige unsichtbare Zeichen:** Blendet unsichtbare Zeichen ein/aus (standardmäßig ausgeblendet).
- **Zeige HTML WYSIWYG:** Aktiviert/deaktiviert die HTML WYSIWYG Ansicht. Hier werden alle erweiterten 4D Write Pro Attribute entfernt, die nicht mit allen Browsern kompatibel sind (standardmäßig deaktiviert).
- **Horizontales Lineal:** Blendet das Lineal ein/aus (standardmäßig eingeblendet). Weitere Informationen dazu finden Sie im Abschnitt **Lineale verwalten**.

Hinweise zur Kompatibilität:

- 4D Write Pro Dokumente, die mit Versionen bis zu 4D v15 R5 erstellt wurden, verwenden für diese Eigenschaften die Standardwerte, mit Ausnahme der Eigenschaft **Auflösung** (auf **Automatik** gesetzt).
- Horizontale Lineale sind in Anwendungen, die mit 4D v16 R2 erstellt wurden, standardmäßig verfügbar; für Anwendungen, die aus früheren Versionen konvertiert wurden, ist diese Einstellung standardmäßig **nicht** markiert.

4D Write Pro Dokumente in 4D Objektfeldern speichern

Sie können Ihre 4D Write Pro Dokumente automatisch in der 4D Datendatei speichern. Haben Sie einen 4D Write Pro Bereich in einem Formular angelegt und ein Feld vom Typ Objekt zum Speichern des Bereichsinhalt erstellt, wird der im Bereich eingegebene Text automatisch mit jedem Datensatz gesichert, wenn dieser bestätigt wird. Über den Befehl **QUERY BY ATTRIBUTE** können Sie Datensätze nach dem Wert seiner internen Attribute auswählen. Sie können auch Ihre eigenen Attribute in 4D Write Pro Bereichen hinzufügen oder danach suchen.

Dieser Abschnitt beschreibt folgendes:

- Ein 4D Objektfeld an einen 4D Write Pro Bereich in einem Formular binden
- Über die standardmäßigen Objektbefehle **OB SET**, **OB Get** und **QUERY BY ATTRIBUTE** eigene Attribute für gespeicherte 4D Write Pro Dokumente setzen, erhalten und danach suchen.

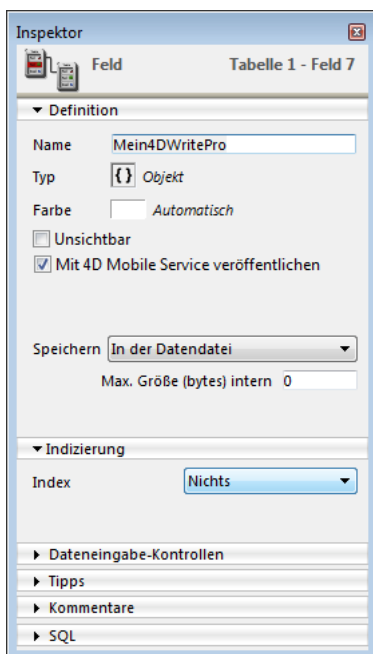
Ein 4D Objektfeld einem 4D Write Pro Bereich zuweisen

Um einen 4D Write Pro Bereich an ein 4D Objektfeld zu binden, müssen Sie nur in der Eigenschaft **Variablenname** des Bereichs die Referenz zum Feld setzen.

Objektfeld in der Struktur anlegen

In Ihrer Datenbankstruktur lässt sich jedes 4D Objektfeld zum Speichern von 4D Write Pro Dokumenten verwenden. Dafür müssen Sie, je nach Bedarf, folgendes definieren:

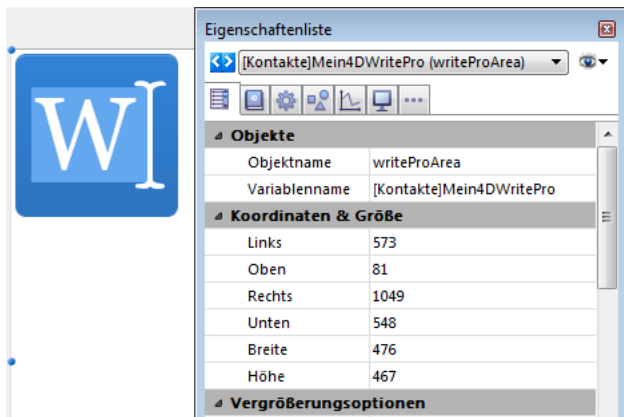
- Name des Feldes
- Seine Attribute, wie z.B. "Mit 4D Mobile Service veröffentlichen" und seinen Index
- Option zum Speichern (siehe **Daten extern speichern**)



Objektfeld dem 4D Write Pro Bereich zuweisen

Ist das Objektfeld zum Speichern Ihres 4D Write Pro Dokuments definiert, müssen Sie es im Formular mit dem 4D Write Pro Bereich zuweisen. Sie können eine beliebige Tabelle oder ein Projektformular verwenden.

Im Formulareditor geben Sie den Feldnamen in der Standardform "[Tabelle]Feld" im Bereich **Variablenname** der Eigenschaftenliste für den 4D Write Pro Bereich ein:



Ihr 4D Write Pro Bereich wird dann dem Feld zugewiesen und stellt sicher, dass sein Inhalt automatisch mit jedem Datensatz gesichert wird. Falls Sie keine automatischen 4D Schaltflächen verwenden, müssen Sie den Bereich manuell über 4D Befehle sichern.

Eigene Attribute verwenden

Werden 4D Write Pro Bereiche in Objektfeldern gespeichert, können Sie mit dem 4D Write Pro Dokument eigene Attribute sichern und einlesen, wie z.B. Name des Autors, Kategorie des Dokuments oder weitere hilfreiche Information. Dann können Sie nach Ihren eigenen Attributen suchen, um Datensätze mit passenden Kriterien auszuwählen.

Eigene Attribute werden mit dem Befehl **WP EXPORT DOCUMENT** oder **WP EXPORT VARIABLE** exportiert. Sie werden auch beim Konvertieren eines 4D Write Pro Objektfeldes in JSON über den Befehl **JSON Stringify** exportiert (zusammen mit den 4D Write Pro Hauptattributen des Dokuments).

Sie erhalten oder setzen eigene Attribute über die Objektnotation oder die Befehle **OB Get** und **OB SET**.

In der Formularmethode können Sie beispielsweise schreiben:

```

if(Form event=On Validate)
  [MyDocuments]My4DWP["myatt_Last edition by"]:=Current user
  [MyDocuments]My4DWP.myatt_Category:="Memo"
  [MyDocuments]My4DWP:=[MyDocuments]My4DWP //um die Bearbeitung zu speichern
End if

```

oder:

```

if(Form event=On Validate)
  OB SET([MyDocuments]My4DWP;"myatt_Last edition by";Current user)
  OB SET([MyDocuments]My4DWP;"myatt_Category";"Memo")
End if

```

Sie können eigene Attribute des Dokuments auch einlesen:

```

vAttrib:=OB Get([MyDocuments]My4DWP;"myatt_Last edition by")

```

Haben Sie eigene 4D Write Pro Attribute in Ihrer Datendatei gesichert, können Sie nach diesen Attributen suchen, um eine Auswahl Datensätze mit den passenden Attributwerten zu erstellen. Das folgende Beispiel sucht nach der Tabelle mit dem Objektfeld, um Datensätze auszuwählen:

```

QUERY BY ATTRIBUTE([MyDocuments];[MyDocuments]My4DWP;"myatt_Category";="Memo")
//Wählt alle Datensätze in MyDocuments, in denen das eigene Attribut "myatt_Category" den Wert "Memo" hat
//im Objektfeld My4DWP (mit einem 4D Write Pro Bereich verbunden)

```

Hinweis zu Namen für eigene Attribute: Da eigene Attribute den gleichen Namensbereich wie interne Attribute von 4D Write Pro benutzen, empfehlen wir dringend, die Namen Ihrer eigenen Attribute mit Vorsilben zu kennzeichnen, um Konflikte zwischen internen und eigenen Attributen zu vermeiden. Namen ohne Vorsilbe sind für interne Attribute in 4D Write Pro reserviert. Sie können eine beliebige Vorsilbe verwenden. Im obigen Beispiel haben wir "myatt_" verwendet.

Hinweis: Ab 4D v15 R4 können Sie auch per Programmierung auf die internen Attribute in 4D Write Pro zugreifen: Über die Standardbefehle **OB Get** und **OB SET** sowie **WP SET ATTRIBUTES**, **WP GET ATTRIBUTES**, **WP RESET ATTRIBUTES**. Weitere Informationen dazu finden Sie im Abschnitt **4D Write Pro Attribute**.

4D Write Pro Bereich verwenden

Dokumente in 4D Write Pro Bereichen verwalten

In 4D Anwendungen lassen sich 4D Write Pro Dokumente mit spezifischen Befehlen aus dem Kapitel **4D Write Pro** erstellen, importieren und exportieren (**WP EXPORT DOCUMENT**, **WP EXPORT VARIABLE**, **WP Import document**, **WP New**).

Sie können einen 4D Write Pro Bereich auch einem Objektfeld in einem Formular der Anwendung zuweisen. Auf diese Weise wird jedes 4D Write Pro Dokument automatisch mit dem Datensatz gesichert und in den Daten der Anwendung gespeichert (siehe **4D Write Pro Dokumente in 4D Objektfeldern speichern**).

Dokumentformat .4wp

Sie können 4D Write Pro Dokumente über das native **.4wp** Format auf die Festplatte sichern und erneut öffnen.

Das **.4wp** Format besteht aus einem Zip Ordner, der den Namen des Dokumenttitels hat und HTML Text und Bilder enthält:

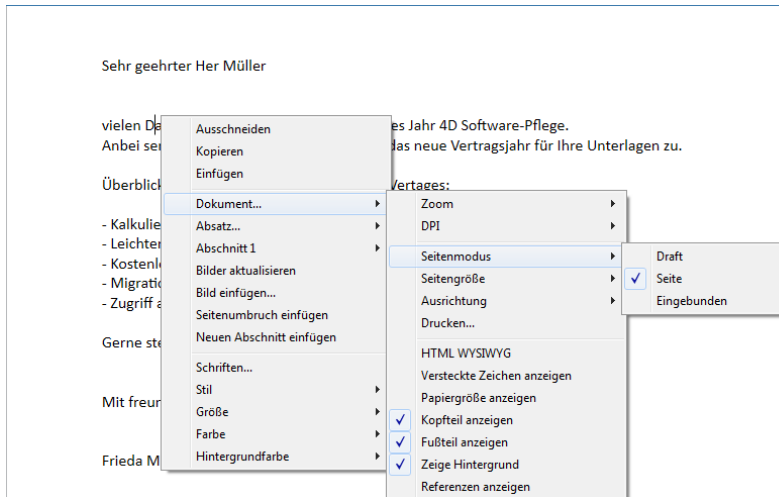
- HTML Text kombiniert reguläres HTML mit 4D Ausdrücken (nicht-berechnet) und 4D-spezifischen Tags
- Bilder werden in einem Ordner neben der HTML Datei gespeichert, der den gleichen Namen wie das Dokument hat

Da **.4wp** Dokumente auf HTML basieren, lassen sie sich in jede externe Anwendung, die HTML unterstützt, importieren bzw. öffnen.

Hinweis: Das interne Dokumentformat von 4D Write Pro ist eine proprietäre HTML Extension, die mit XHTML5 oder HTML5 kompatibel ist, jedoch ihre eigene Untermenge von HTML/CSS Attributen und Tags unterstützt. Deshalb können nur die von 4D Write Pro exportierten HTML Dokumente ohne Risiko von Datenverlust wieder von 4D Write Pro geöffnet werden. Beim Import von extern erstellten HTML Dokumenten können Fehler auftreten.

Benutzeroberfläche

Ist im 4D Write Pro Bereich die Eigenschaft **Kontextmenü** markiert (siehe **Einen 4D Write Pro Bereich erstellen**), steht Benutzern im Anwendungsmodus ein leicht verständliches Kontextmenü zur Verfügung:



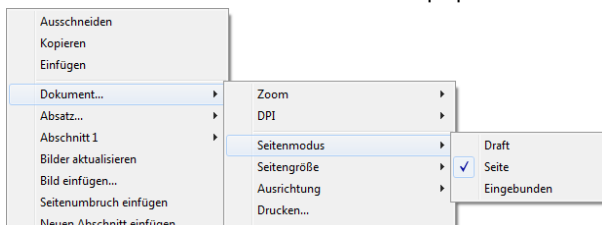
Dieses Menü bietet Zugriff auf alle Features für Benutzer von 4D Write Pro.

Seitenmodus wählen

Für 4D Write Pro Dokumente gibt es drei Ansichtsmodi:

- **Draft:** Entwurfsmodus mit den grundlegenden Eigenschaften
- **Seite** (Standard): Modus "Druckansicht"
- **Eingebunden:** Modus für eingebundene Bereiche; zeigt keine Rahmen, Fußteile, Kopfteile, Spalten, Seitenränder, etc. Dieser Modus eignet sich auch für die Darstellung im Web. Dafür müssen als Auflösung 96 dpi und die Option **Zeige HTML WYSIWYG** ausgewählt werden.

Der Seitenmodus lässt sich über das PopUp-Menü des Bereichs einstellen:



Hinweis: Der Seitenmodus wird nicht mit dem Dokument gespeichert.

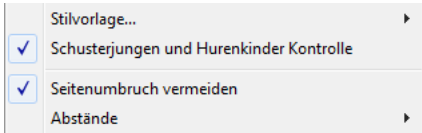
Für eingebundene Bereiche in 4D Formularen lässt sich der Seitenmodus auch standardmäßig über die Eigenschaftenliste setzen. Er wird dann als Eigenschaft des 4D Write Pro Formularobjekts gespeichert. Weitere Informationen dazu finden Sie im Abschnitt **Darstellungseigenschaften konfigurieren**.

Einstellungen für Ansicht als Seite

Ist das Dokument im Seitenmodus **Seite**, werden für den Benutzer folgende Dokumenteigenschaften angezeigt:

- Seitenbegrenzung zur Darstellung des druckbaren Bereichs
- Seitenbreite und Seitenhöhe (Standard: 21x29,7 cm)
- Seitenausrichtung (Standard: Hochformat)
- Seitenrand (Standard: 2,5 cm)

Darüberhinaus gibt es im Untermenü **Absatz...** spezifische Absatzzeigenschaften:



- **Schusterjungen und Hurenkinder Kontrolle:** Ist diese Option für einen Absatz markiert, erlaubt 4D Write Pro im Dokument keine Schusterjungen (letzte Zeile des Absatzes erscheint isoliert am Seitenanfang) und Hurenkinder (erste Zeile des Absatzes erscheint isoliert am Seitenende). Im ersten Fall wird auch die vorangehende Zeile an den Seitenanfang gesetzt, so dass hier zwei Zeilen erscheinen. Im zweiten Fall wird die einzelne letzte Zeile auf die nächste Seite gesetzt.
- **Seitenumbruch vermeiden:** Ist diese Option für einen Absatz markiert, verhindert 4D Write Pro Umbrüche im Absatz, so dass er nicht auf zwei oder mehr Seiten verteilt wird.

Sie können auch Befehle für die Seite aus dem Kontextmenü nutzen, wie **Seitenumbruch/Spaltenumbruch/Neuen Abschnitt einfügen** bzw. **Dokument->Seitengröße** oder **->Ausrichtung** verwenden.

Hinweis: Ist das Dokument im Seitenmodus **Eingebunden** oder **Draft**, lassen sich Seiteneigenschaften setzen, auch wenn die Auswirkung nicht sichtbar ist. Im Modus **Draft** sind folgende Effekte für Absatzzeigenschaften sichtbar:

- Begrenzung der Seitenhöhe (angezeigte Zeilen)
- Spalten
- Seitenumbruch vermeiden
- Schusterjungen und Hurenkinder Kontrolle

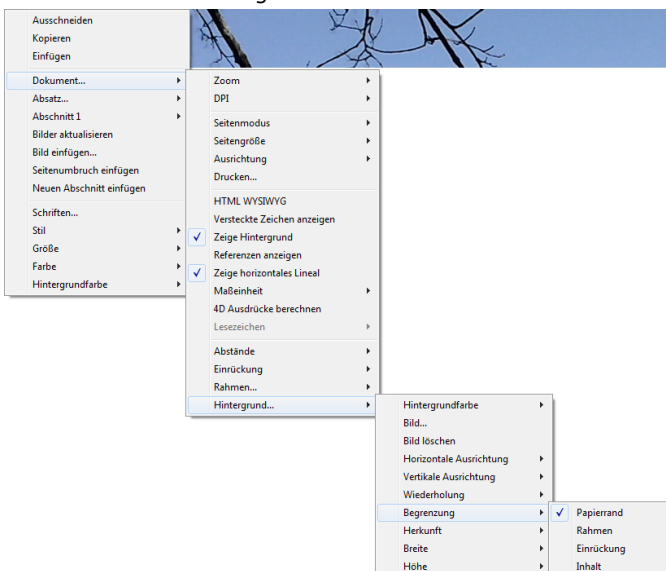
Hintergrund

Der Hintergrund von 4D Write Pro Dokumenten bzw. Dokumentteilen, wie Tabellen, Absätze, Ausschnitte, Kopf-/Fußteile lässt sich mit folgenden Effekten gestalten:

- Farben
- Rahmen
- Bilder
- Originalpositionierung oder horizontal/vertikal versetzt
- Zeichenbereich
- Wiederholen

Diese Attribute lassen sich mit dem Befehl **WP SET ATTRIBUTES** oder **4D Write Pro Standardaktionen verwenden** für Hintergründe einzelner Teile auf der Seite bzw. für das gesamte Dokument setzen. Die komplette Liste der Hintergrundattribute mit Beschreibung finden Sie im Kapitel **4D Write Pro Attribute**.

Benutzer können Hintergrundattribute über das Kontextmenü verändern:



Background image clipped to Border box.

Background image clipped to Paper box.

Unter [How Do I \(HDI\) Demos](#) sehen Sie ein Beispiel, um ein Bild als Hintergrund zu setzen.

Kopfteile, Fußteile und Abschnitte verwalten

4D Write Pro Dokumente unterstützen Kopf- und Fußteile. Sie sind Abschnitten zugeordnet.

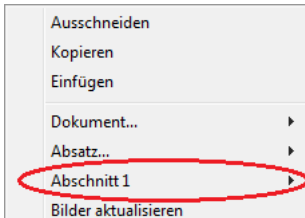
Ein Abschnitt ist Teil eines Dokuments, der durch eine Seitenfolge definiert ist und eine eigene Seitennummerierung und Attribute haben kann. Ein Dokument kann eine beliebige Anzahl Abschnitte haben (einer bis zur Gesamtzahl der Seiten). Jede Seite kann nur zu einem Abschnitt gehören.

Sie können für jeden Abschnitt ein Kopf- und Fußteil definieren.

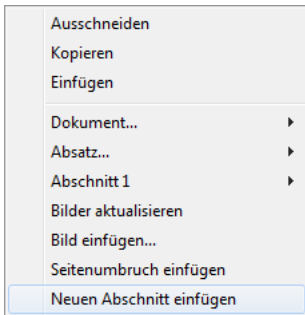
Abschnitt definieren

Ein Abschnitt ist eine Untermenge von fortlaufenden Seiten in einem 4D Write Pro Dokument. Ein Dokument kann einen oder mehrere Abschnitte enthalten. Ein Abschnitt kann mehrere Seiten enthalten, von einer einzelnen Seite bis zur Gesamtanzahl der Seiten des Dokuments. Ein Abschnitt kann eine bis zu 20 Spalten enthalten.

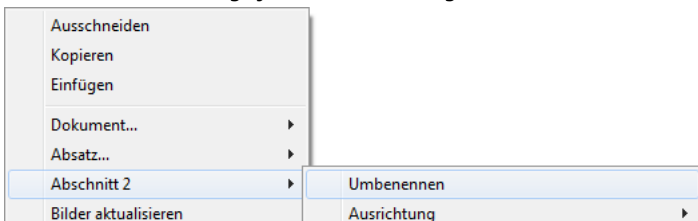
Ein Dokument enthält standardmäßig einen einzigen Abschnitt mit Namen **Abschnitt 1**. Klicken Sie in das Dokument, erscheint das Kontextmenü von 4D Write Pro und zeigt diese Abschnittsnummer:



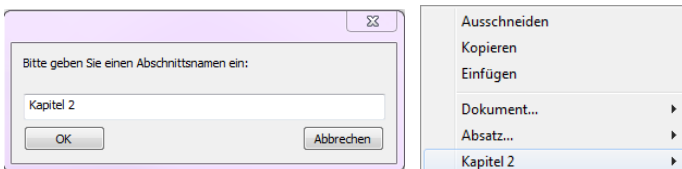
Sie erstellen einen neuen Abschnitt, indem Sie im Textfluss einen neuen Abschnitt hinzufügen:



Das Kontextmenü zeigt jeweils die weitergezählte Nummer des Abschnitts an. Sie können den Abschnitt auch umbenennen:



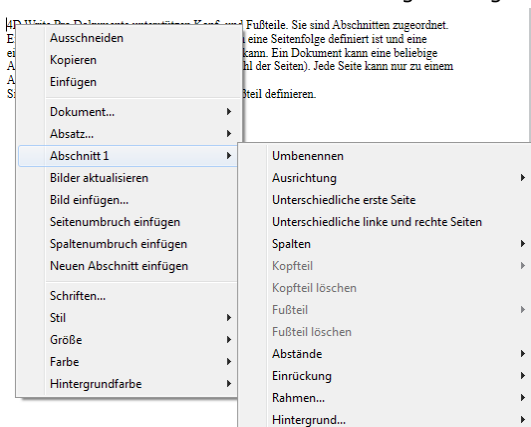
Dann wird überall im Dokument der eingegebene Name für den Abschnitt verwendet:



Haben Sie für einen Abschnitt eine unterschiedliche erste Seite oder unterschiedliche linke und rechte Seiten definiert, erscheint im Kontextmenü auch der Seitentyp (siehe unten).

Attribute für Abschnitt

Abschnitte erben die Attribute des Dokuments. Gängige Attribute, inkl. Kopf- und Fußteile, lassen sich jedoch für jeden Abschnitt einzeln ändern. Das Kontextmenü zeigt die Eigenschaften und Attribute, die für Abschnitte verfügbar sind:



- **Ausrichtung:** Setzt die Seitenausrichtung pro Abschnitt (Hochformat oder Querformat)
- **Unterschiedliche erste Seite:** Damit lassen sich für die erste Seite des Abschnitts andere Attribute setzen; so können Sie z.B. lose Blätter erstellen. Ist dieses Attribut aktiviert, wird die erste Seite des Abschnitts als Unterabschnitt behandelt, die

eigene Attribute haben kann.

Kapitel 2 (Erste Seite)

- **Unterschiedliche linke und rechte Seiten:** Damit lassen sich für die linken und rechten Seiten des Abschnitt unterschiedliche Attribute setzen. Ist dieses Attribut aktiviert, werden die linken und rechten Seiten des Abschnitts als Unterabschnitte behandelt, die eigene Attribute haben können.

Kapitel 2 (Rechte Seite)

- **Spalten:** Damit lassen sich Anzahl und Eigenschaften von Spalten für den Abschnitt definieren. Weitere Informationen siehe unten.
- **Kopfteil** und **Fußteil:** Setzt eigene Kopf- und Fußteile für den Abschnitt. Weitere Informationen siehe unten.
- **Abstände / Einrückung / Rahmen / Hintergrund:** Diese Attribute lassen sich für jeden Abschnitt separat definieren. Weitere Informationen dazu finden Sie im Abschnitt **4D Write Pro Attribute**.

Kopf- und Fußteile einfügen

Jeder Abschnitt kann eigene Kopf- und Fußteile haben. Sie erscheinen nur, wenn **Seite** als Seitenmodus des Dokuments gewählt ist.

Sie können je nach aktivierter Option in einem Abschnitt drei verschiedene Kopf- und Fußteile einrichten:

- erste Seite
- linke Seite(n)
- rechte Seite(n)

Um einen Kopf- oder Fußteil zu erstellen:

1. Stellen Sie sicher, dass das Dokument im Seitenmodus **Seite** ist.
2. Doppelklicken Sie in den Bereich Kopfteil oder Fußteil des gewünschten Abschnitts oder der Seite, um in den Modus Bearbeiten zu wechseln.
 - Der Kopfteilbereich ist am Anfang der Seite:

Abschnitt 1 Kopfteil
Handbuch 4D Write

- Der Fußteilbereich ist am Ende der Seite:

Abschnitt 1 Fußteil

Sie können dann beliebigen statischen Inhalt eingeben, der dann automatisch auf jeder Seite des Abschnitts wiederholt wird (außer für die erste Seite, sofern aktiviert).

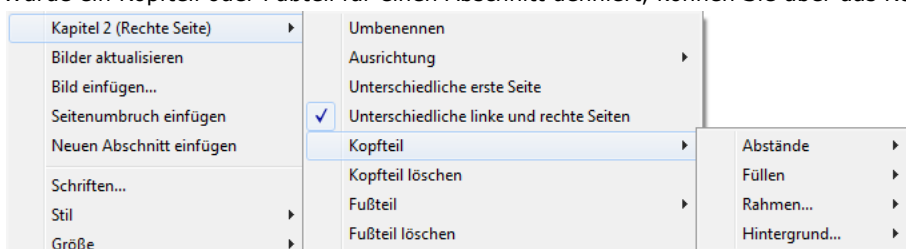
Kapitel 2 Fußteil (Rechte Seite)

Kapitel 2 - Referenzen

Über den Befehl **ST INSERT EXPRESSION** können Sie dynamischen Inhalt wie Seitennummer oder Seitenanzahl eingeben. Weitere Informationen dazu finden Sie im Abschnitt **Ausdrücke für Dokument oder Seite einfügen**.

Hinweis: Sie können Kopf- und Fußteile auch per Programmierung über Befehle wie **WP Get header** und **WP Get footer** verwalten.

Wurde ein Kopfteil oder Fußteil für einen Abschnitt definiert, können Sie über das Kontextmenü allgemeine Attribute festlegen:



Weitere Informationen zu **Abstände**, **Füllen**, **Rahmen** und **Hintergrund** finden Sie im Abschnitt **4D Write Pro Attribute**.

Über die Einträge **Kopfteil löschen** oder **Fußteil löschen** im Kontextmenü können Sie die gesamte Definition für Kopf- oder Fußteil (Inhalt und Attribute) auf einmal entfernen.

Kompatibilität

4D Write Pro verwaltet Kopf- und Fußteile von konvertierten Dokumenten des Plug-In 4D Write.

Auch die folgenden Ausdrücke und Eigenschaften werden unterstützt und aus den Kopf- und Fußteilen des 4D Write Plug-Ins konvertiert:

- Variablen für Seitennummer und Seitenanzahl
- Unterschiedliche erste Seite
- Unterschiedliche linke/rechte Seiten

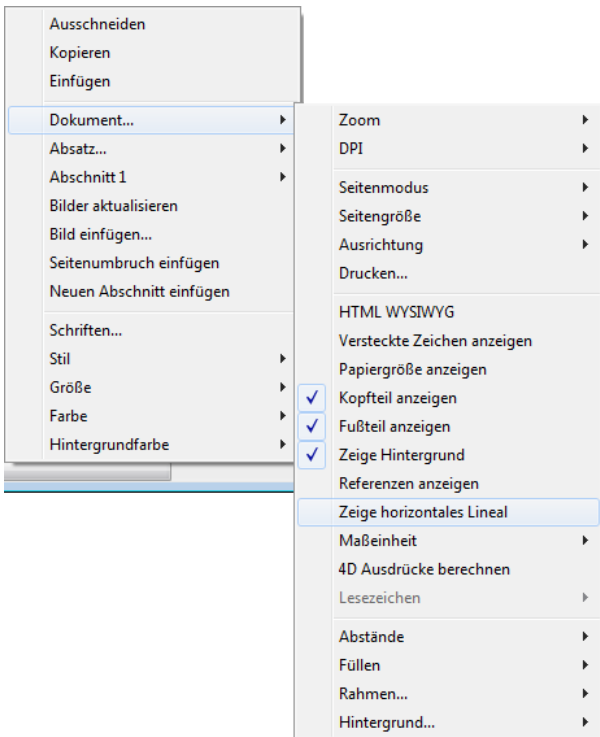
Lineale verwalten

Lineale sind in jedem Ansichtsmodus von 4D Write Pro verfügbar. Sie haben folgende Merkmale:

- Einteilungen in cm, mm, Zoll oder pt gemäß der für das aktuelle Layout definierten Einheit im 4D Write Pro Dokument. Sie können Maßeinheiten über das Kontextmenü oder durch Ändern des Attributs `wk layout unit` definieren.
- Symbol für erste Zeile eingerückt

- Symbol für Absatz linker Rand
- Symbol für Absatz rechter Rand
- Definierte Tabs am unteren Rand des Lineals
- Graustufung zur Anzeige der Seitenränder links und rechts

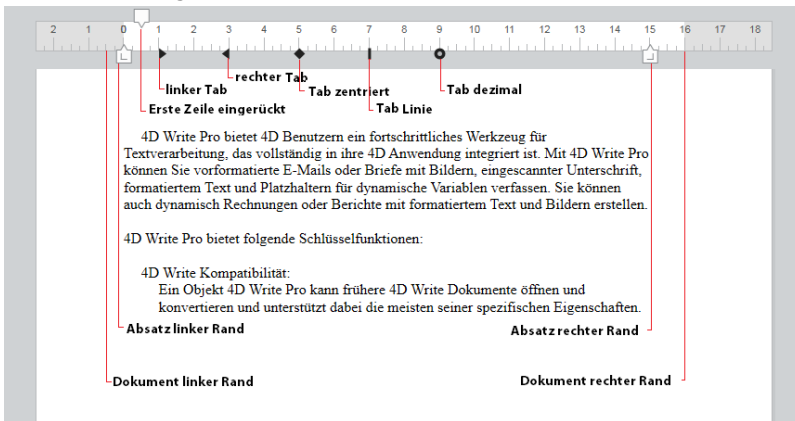
Sie können die Anzeige des Lineals über den Eintrag **Zeige horizontales Lineal** im Kontextmenü des 4D Write Pro Bereichs ein- oder ausblenden:



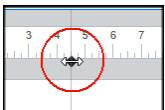
Hinweis: Die Standardanzeige des Lineals lässt sich über eine spezifische Eigenschaft des 4D Write Pro Bereichs definieren. Weitere Informationen dazu finden Sie im Abschnitt **Darstellungseigenschaften konfigurieren**.

Ränder, Einrückungen und Tabulatoren

Sie können Ränder, Einrückungen und Tab-Positionen verändern. Dazu klicken Sie einfach auf das entsprechende Symbol und ziehen es in die gewünschte Position:



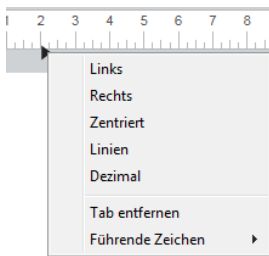
Gehen Sie mit der Maus über ein Symbol, ändert sich der Cursor, um anzuzeigen, dass es sich verschieben lässt. Während dem Ziehen erscheint eine vertikale Hilfslinie:



Sind mehrere Abschnitte ausgewählt, gilt die Hilfslinie für Rand oder Einrücken für alle ausgewählten Abschnitte. Ziehen Sie diese Symbole mit gedrückter Shift-Taste, bleiben die vorhandenen Abstände zwischen Einrückungen und Rändern in den ausgewählten Abschnitten erhalten.

Tags verwalten

Über das Kontextmenü des Lineals können Sie Tags erstellen, ändern oder löschen:



Zum Erstellen eines Tab klicken Sie mit rechter Maustaste direkt auf das Lineal und wählen im Kontextmenü den gewünschten Typ. Ein einfacher linker Mausklick erstellt automatisch einen linksbündigen Tab. Durch rechten Mausklick auf einen vorhandenen Tab können Sie seinen Typ über das Kontextmenü ändern.

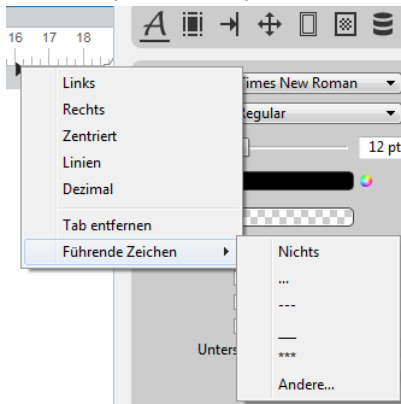
Hinweis: Tab entfernen ist nur verfügbar, wenn Sie mit rechter Maustaste direkt auf einen vorhandenen Tab klicken; Sie können Tabs auch entfernen, wenn Sie diese außerhalb des Lineals ziehen.

Führende Zeichen definieren

Für führende Zeichen vor den Tabs gibt es fünf vordefinierte Arten. Standardmäßig ist die Option "Nichts" vorgegeben, d.h. es erscheinen keine Zeichen:

- Nichts (= Standard)
- (Punkte)
- --- (Striche)
- ___ (Unterstriche)
- *** (Sterne)

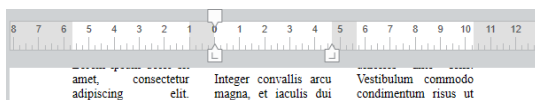
Sie erscheinen immer vor dem Tab und folgen der Textrichtung (links-nach-rechts oder rechts-nach-links). Sie lassen sich per Programmierung über den Selektor `wk tab stop_leadings` der Befehle **WP SET ATTRIBUTES**, **WP GET ATTRIBUTES** und **WP RESET ATTRIBUTES** definieren, oder über die Benutzeroberfläche durch rechten Mausklick auf den Tab, um das Kontextmenü zu öffnen (siehe unten).



Ist **Andere...** ausgewählt, erscheint ein Dialogfenster zum Definieren eigener führender Zeichen.

Lineale bei mehreren Spalten

Wenn zwei oder mehr Spalten für das Dokument oder den Abschnitt definiert sind, zeigt das Lineal für jede Spalte einen eigenen Bereich an:



Hinweis: Die Mehrspalten-Funktion ist im Seitenmodus **Eingebunden** nicht verfügbar.

Ereignis On After Edit

Für ein Formularobjekt im 4D Write Pro Bereich wird ein Ereignis On After Edit ausgelöst, wenn ein Tabulator oder ein Rand verschoben, hinzugefügt oder gelöscht wird, unabhängig, ob sie gezogen oder über das Kontextmenü verändert werden.

Spalten verwalten

Mit 4D Write Pro können Sie Spalten in Ihren Dokumenten verwalten. Spalten werden von der linken Spalte bis zur rechten Spalte verkettet. Mit anderen Worten, wenn Sie Text eingeben, beginnt der Textfluss die linke Spalte zu füllen und fährt mit der Spalte rechts daneben fort, bis er das Ende der Seite erreicht und durchläuft dann die nächste Seite. Um die Seiteneinstellungen ausgleichen zu können, können Sie in 4D Write Pro Spaltenumbrüche einfügen.



Spalten lassen sich auf Dokumentenebene (sie werden im gesamten Dokument angezeigt) bzw. auf Abschnittebene (jeder Abschnitt kann eine eigene Spaltenkonfiguration haben) definieren.

Hinweis: Spalten werden nur im Seitenmodus **Seite** und **Draft view** unterstützt (Sie erscheinen nicht im Seitenmodus **Eingebunden** und werden nicht im Format **WP EXPORT DOCUMENT** und `wk web page complete` exportiert).

Spalten lassen sich setzen über:

- das Untermenü Spalten im Kontextmenü des 4D Write Pro Bereichs,
- 4D Write Pro-Attribute (siehe **4D Write Pro Attribute**),
- 4D Write Pro Standard-Aktionen (siehe **4D Write Pro Standardaktionen verwenden**).

Sie können die folgenden Eigenschaften und Aktionen für Spalten festlegen oder abrufen:

Eigenschaft	Beschreibung	Dokument Attribute	Standardaktionen
Anzahl der Spalten	Sie können bis zu 20 Spalten für das Dokument/den Abschnitt definieren	wk column count	columnCount
Spaltenabstand	Abstand zwischen den Spalten in pts, inch oder cm. Beachten Sie, dass alle Spalten die gleiche Größe haben. Jede Spaltenbreite wird von 4D Write Pro automatisch berechnet, entsprechend der Anzahl der Spalten, der Seitenbreite und dem Abstand	wk column spacing	columnSpacing
Spaltenbreite	(read-only Attribut) Aktuelle Breite für jede Spalte, d.h. berechnete Breite	wk column width	-
Spaltenregel für Stil, Farbe und Breite	Sie können einen vertikalen Trenner (eine dekorative Linie) zwischen den Spalten hinzufügen. Mit diesen Optionen können Sie Stil, Farbe und Breite des Trenners festlegen. Um den vertikalen Trenner zu entfernen, wählen Sie Nichts als Stil aus.	wk column rule style , wk column rule color , wk column rule width	columnRuleStyle , columnRuleColor , columnRuleWidth
Spaltenumbruch einfügen	Fügt einen Spaltenumbruch ein	wk column break , siehe auch WP INSERT BREAK	insertColumnBreak
Spalten-Menü	Erstellen Sie ein Spalten-Untermenü	-	columns

Tabellen verwalten

4D Write Pro Dokumente können Tabellen enthalten. Sie werden per Programmierung erstellt und definiert. Ihr Inhalt kann jedoch vom Benutzer geändert werden. Folgende 4D Write Pro Tabellenattribute sind editierbar: Zeilenhöhe, Ausrichtung, Rahmen, Textstil, Farbe oder Ränder.

Name	Date	Value
Alpha	03/04/2017	867
Bravo	05/04/2017	579
Charlie	02/06/2017	501
Delta	06/06/2017	354
Echo	09/04/2017	463
Foxtrot	31/05/2017	912
Golf	17/03/2017	756
Hotel	11/04/2017	437
India	20/04/2017	64
Juliett	05/04/2017	684
Total		5617

Eine Tabelle erstellen

Sie erstellen 4D Write Pro Tabellen mit der Funktion **WP Insert table**. Dann können Sie mit der Funktion **WP Table append row** Zeilen hinzufügen.

Hinweis: Ein Benutzer kann eine Tabelle durch Kopieren und Einsetzen eines Zellenausschnitts anlegen:

Golf	17/03/2017	756
Hotel	11/04/2017	437
India	20/04/2017	64
Juliett	05/04/2017	684
Total		5617

Golf	17/03/2017
Hotel	11/04/2017

Tabellen bearbeiten

Zelleninhalt können Sie per Programmierung mit der Funktion **WP Table append row** hinzufügen.

Ist eine Tabelle angelegt, können Benutzer den Zelleninhalt direkt ändern. Sie können in Zellen klicken, Text oder Bilder genauso wie in regulären Absätzen auswählen, bearbeiten, kopieren/einsetzen oder löschen. Mit der Tabulatortaste können sie durch Zellen navigieren (mit **Shift+Tab** in der Gegenrichtung).

Beachten Sie, dass die Zellenbreite fest ist: Ändert ein Benutzer Text oder setzt ein Bild ein, wird die Zeilenhöhe bei Bedarf automatisch erweitert und der Text automatisch umgebrochen:

Charlie	20/05/2017	844
Delta is the fourth letter of the most known spelling alphabet	06/05/2017	188
Echo	29/05/2017	579

Benutzer können auch Spalten, Zeilen oder Zellen auswählen und verfügbare Attribute, wie Textstil, Farben, Ausrichtung, etc. über die eingebauten PopUp-Menüs oder eine angepasste Oberfläche anwenden. 4D Write Pro bietet verschiedene Funktionen zum Auswählen von Tabellenausschnitten:

- **WP Table get rows** für einen Zeilenausschnitt
- **WP Table get columns** für einen Spaltenausschnitt (*)
- **WP Table get cells** für einen Zellausschnitt

(*) Spalten haben keine Entsprechung in html. In 4D Write Pro ist ein Spaltenausschnitt dasselbe wie ein Zellausschnitt, d.h. Spalten haben dieselbe Priorität wie Zellen.

Haben Sie einen Ausschnitt gewählt, können Sie mit dem Befehl **WP SET ATTRIBUTES** passende Attribute setzen. Innerhalb von Zellen werden Attribute je nach Inhalt auf Absätze, Zeichen oder Bilder angewandt. Sie können z.B. die Höhe, Schriftgröße, Ränder, etc. von Tabellen oder Tabellenzellen setzen. Weitere Informationen dazu finden Sie im Abschnitt **4D Write Pro Attribute**.

Werden verschiedene Attribute auf konkurrierende Elemente einer Tabelle angewandt, gilt für das Rendern folgende Reihenfolge:

1. Zuerst wird die Tabelle gerendert
2. Dann werden Zeilen gerendert (überschreibt Tabellenattribute)
3. Dann werden Zellen/Spalten gerendert (überschreibt Zeilenattribute)

4D Write Pro Dokumente drucken

4D Write Pro Dokumente lassen sich auf zwei Arten drucken:

- Als Teil von 4D Formularen
- Als eigenständige Dokumente

Dokumente drucken

Mit standardmäßigen 4D Druckbefehlen wie **PRINT SELECTION** oder **PRINT RECORD** können Sie eingebundene 4D Write Pro Objekte als Teil jedes Formulars (Projekt, Tabelle, Eingabe oder Ausgabe) drucken.

Die Standardoption **Drucke variable Größe** wird auch für 4D Write Pro Bereiche unterstützt (*). Damit können Sie unterschiedliche Größen beim Drucken verwalten. Ist diese Option markiert, wird der Rahmen (außen und innen) und oberer Rand nur auf die erste Seite angewandt, Rahmen (außen und innen) und unterer Rand nur auf die letzte Seite. Eigenschaften für Seitennummerierung werden ignoriert: Steuern von Schusterjungen/Hurenkindern ist deaktiviert und Seitenumbrüche werden nicht angewandt (sie gelten nur zum Rendern der Seite auf dem Bildschirm oder für eigenständiges Drucken des Dokuments). Ist die Option **Drucke variable Größe** markiert, werden nur Objekte über dem Bereich im Formular gedruckt. Weitere Informationen dazu finden Sie im Abschnitt **Drucke variable Größe** des Handbuchs *4D Designmodus*.

(*) gilt nicht für die Funktionen **Print object** und **Print form**.

Darstellungsmodus zum Drucken

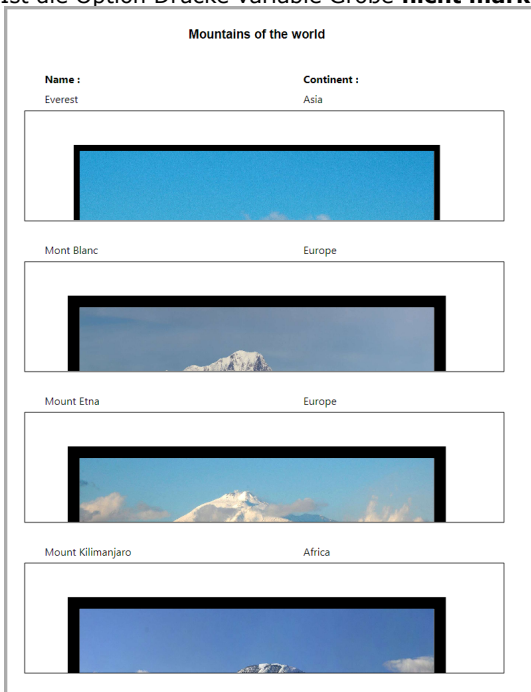
Unabhängig vom eingestellten **Seitenmodus** für den 4D Write Pro Bereich (siehe **Darstellungseigenschaften konfigurieren**) wird bei 4D Druckbefehlen wie **Print form** immer im Modus **Eingebunden** gedruckt. Außerdem werden folgende Darstellungseigenschaften für 4D Write Pro Formularobjekte nicht berücksichtigt: Seitenmodus (immer "Eingebunden"), Kopfteil anzeigen, Fußteil anzeigen, Seitenrand zeigen (immer "Nein"), Versteckte Zeichen anzeigen (immer "Nein").

Beispiel

Nachfolgendes Beispiel zeigt die Auswirkung der Option **Drucke variable Größe** für einen 4D Write Pro Bereich, der in ein Standard Ausgabeformular eingebunden ist. Es wird folgender Code ausgeführt:

```
ALL RECORDS([Movies])
ORDER BY([Movies]Title)
PRINT SELECTION([Movies])
```


- Ist die Option **Drucke variable Größe** **nicht markiert** (aus), erhalten Sie folgendes Ergebnis:



- Ist die Option Drucke variable Größe markiert (an), erhalten Sie folgendes Ergebnis:

Mountains of the world


Name : Everest **Continent :** Asia



Mount Everest, also known in Nepal as Sagarmāthā and in Tibet as Chomolungma, is Earth's highest mountain. It is located in the Mahalangur mountain range in Nepal.[8][9] Its peak is 8,848 metres (29,029 ft) above sea level.[1] It is not the furthest summit from the centre of the Earth. That honour goes to Mount Chimborazo, in the Andes.[10] The international border between China and Nepal runs across Everest's precise summit point. Its massif includes neighbouring peaks Lhotse, 8,516 m (27,940 ft); Nuptse, 7,855 m (25,771 ft) and Changtse, 7,580 m (24,870 ft).

Mountains of the world

Name : Mont Blanc **Continent :** Europe



Mont Blanc (French pronunciation: [mɔ̃.blɑ̃] or Monte Bianco (Italian pronunciation: [ˈmonte ˈbjɑŋko]), both meaning "White Mountain", is the highest mountain in the Alps and the highest peak in Europe outside of the Caucasus range.[2] It rises 4,809 m (15,778 ft)[1][3] above sea level and is ranked 11th in the world in topographic prominence.[4]

(sample text source: Wikipedia)

Eigenständige Dokumente drucken

4D Write Pro enthält ab 4D v15 R5 Funktionalitäten zum Drucken eigenständiger 4D Write Pro Dokumente und zum Steuern standardmäßiger Druckereinstellungen wie Format, Ausrichtung und Seitennummern.

4D Write Pro Befehle

Im wesentlichen werden die Optionen zum Drucken über die 4D Write Pro Befehle **WP PRINT** und **WP USE PAGE SETUP** verwaltet.

- WP PRINT** startet einen Druckauftrag für das 4D Write Pro Dokument oder fügt das Dokument zum aktuellen Druckauftrag hinzu.
- WP USE PAGE SETUP** ändert die aktuellen Seiteneinstellungen des Druckers gemäß den 4D Write Pro Attributen für Seitengröße und Ausrichtung.

Hinweise zur Konfiguration:

- Druckbefehle in 4D Write Pro basieren auf einer neuen internen Architektur, die für 64-bit Versionen von 4D eingerichtet wurde. Sie werden aber weiterhin in 32-bit Versionen unterstützt, nur nicht in Druckaufträgen, die mit **OPEN PRINTING JOB** gestartet wurden. Weitere Informationen dazu finden Sie im Abschnitt **WP PRINT**.
- Stellen Sie auf Rechnern mit Windows 7 oder Windows Server 2008 R2 sicher, dass das *Plattform Update für Windows 7* installiert wurde, damit die Features zum Drucken unterstützt werden.

Reguläre 4D Befehle

Folgende 4D Befehle unterstützen die Druckereinstellungen für 4D Write Pro:

- SET PRINT OPTION** und **GET PRINT OPTION** unterstützen alle Optionen für 4D Write Pro Dokumente, die mit **WP PRINT** gedruckt werden. Für [Paper option](#) und [Orientation option](#) kann es effizienter sein, **WP USE PAGE SETUP** aufzurufen, um diese Attribute auf einfache Weise mit den Einstellungen für 4D Write Pro Dokumente zu synchronisieren. Mit [Page range option](#) (15) können Sie die erste und letzte Seite zum Drucken übergeben.

- **PRINT SETTINGS** setzt die Druckeinstellungen für den aktuellen Drucker; wird **WP PRINT** danach aufgerufen, verwendet es die modifizierten Druckeinstellungen, falls sie über Dialogfenster für Druckeinstellungen abgeändert wurden (außer Einstellungen für Seitenränder, die immer vom 4D Write Pro Dokument übernommen werden).
- **OPEN PRINTING JOB** und **CLOSE PRINTING JOB**: **WP PRINT** lässt sich zwischen diesen Befehlen aufrufen, um ein oder mehrere 4D Write Pro Dokumente in einem Druckauftrag einzufügen.

4D Write Dokumente importieren

Ein grundlegendes Feature des neuen Objekts 4D Write Pro ist, dass sich vorhandene 4D Write Dokumente importieren und konvertieren lassen. So können Sie Applikationen migrieren, die derzeit das Plug-In 4D Write verwenden.

Hinweise zur Kompatibilität:

- Nur 4D Write Dokumente der neuesten Generation ("4D Write v7") werden unterstützt.
- Kopieren-Einsetzen von einem 4D Write Dokument in einen 4D Write Pro Bereich wird derzeit nicht unterstützt. Ein 4D Write Dokument lässt sich nur über Programmiersprache-Befehle von 4D Write Pro importieren.

Ein 4D Write Dokument importieren

Es gibt zwei Wege zum Importieren von 4D Write Dokumenten:

- Funktion **WP Import document** für 4D Write Dateien, die auf der Festplatte gespeichert sind
- Funktion **WP New** für 4D Write Bereiche, die in BLOB Feldern gespeichert sind

Weitere Informationen dazu finden Sie in der Beschreibung zu diesen Befehlen.

Welche Eigenschaften von 4D Write werden übernommen?

Um Ihre Migration vom Plug-In 4D Write zu 4D Write Pro zu vereinfachen, wollen wir möglichst viele 4D Write Features in 4D Write Pro Objekten unterstützen.

Dieser Abschnitt listet die Eigenschaften des Plug-In 4D Write auf, die nach einem Import über die Funktionen **WP Import document** oder **WP New** in einen 4D Write Pro Bereich übernommen werden.

Beachten Sie jedoch, dass es ein paar Unterschiede geben kann, die nicht als Fehler gewertet werden. Das gilt z.B. für die Standardschrift von Aufzählungszeichen (Bullets) in 4D Write Pro oder minimale Konvertierung beim Typ Unterstrichen.

Dokument Info

Plug-In 4D Write	4D Write Pro
Erstelldatum & Zeit	Verfügbar
Änderdatum & Zeit	Verfügbar
Gesperrt	Nicht verfügbar (Objekteigenschaft Nur-Lesen verwenden)
Titel	Verfügbar
Betreff	Verfügbar (nur Volltext)
Autor	Verfügbar
Firma	Verfügbar
Notizen	Verfügbar

Parameter für Anzeigen im Dokument

Plug-In 4D Write	4D Write Pro
Anzeigen Seite	Nicht importiert (im Kontextmenü Dokument->Seitenmodus verwenden)
Anzeigen Lineal	Nicht importiert (Objekteigenschaft verwenden)
Anzeigen Rahmen	Nicht importiert (Objekteigenschaft verwenden)
Anzeigen Kopfteil	Nicht importiert (Objekteigenschaft verwenden)
Anzeigen Fußteil	Nicht importiert (Objekteigenschaft verwenden)
Anzeigen erste Seite Kopfteil	Nicht verfügbar
Anzeigen erste Seite Fußteil	Nicht verfügbar
Anzeigen Bilder	Nicht verfügbar
Anzeigen Horizontale Bildlaufleiste	Nicht importiert (Objekteigenschaft Horizontale Bildlaufleiste verwenden)
Anzeigen Vertikale Bildlaufleiste	Nicht importiert (Objekteigenschaft Vertikale Bildlaufleiste verwenden)
Anzeigen Unsichtbare Zeichen	Nicht importiert (Objekteigenschaft verwenden)
Anzeigen Referenzen	Nicht importiert (ST SET OPTIONS verwenden)
Anzeigen Spaltentrenner	Verfügbar (ab v17)
Anzeigen H Splitter	Nicht verfügbar
Anzeigen V Splitter	Nicht verfügbar
Anzeigen Wysiwyg	Nicht verfügbar
Anzeigen Zoom	Nicht importiert (Im Kontextmenü Dokument->Zoom verwenden)

Parameter für Dokument

Plug-In 4D Write	4D Write Pro
Einheit	Nicht importiert (im Kontextmenü Dokument->Maßeinheit verwenden)
Sprache	Nicht verfügbar
Spaltenanzahl	Verfügbar (ab v17)
Spaltenabstand	Verfügbar (ab v17)
Schusterj. und Hurenk.	Verfügbar
Standardtabulatoren	Verfügbar
Führende Tabulatoren	Verfügbar
Farbe URL	Nicht verfügbar
Farbe angeklickte URL	Nicht verfügbar

Parameter für Paginierung des Dokuments

Plug-In 4D Write	4D Write Pro
Seitenbreite	Verfügbar
Seitenhöhe	Verfügbar
Erste Seitennummer	Ab v16 verfügbar
Kopf- und Fußzeilen: Unterschiedliche Erste Seite	Ab v16 verfügbar
Kopf- und Fußzeilen: Unterschiedliche Linke/Rechte Seite	Ab v16 verfügbar
Seitenbindung	Ab v16 verfügbar
Linke/Rechte Seite	Ab v16 verfügbar
Seitenränder	Verfügbar
Kopfteil Rand oben	Ab v16 verfügbar
Kopfteil Rand unten	Ab v16 verfügbar
Fußteil Rand oben	Ab v16 verfügbar
Fußteil Rand unten	Ab v16 verfügbar
Erste Seite Rand oben	Ab v16 verfügbar
Erste Seite Rand unten	Ab v16 verfügbar
Kopfteil erste Seite Rand oben	Ab v16 verfügbar
Kopfteil erste Seite Rand unten	Ab v16 verfügbar
Fußteil erste Seite Rand oben	Ab v16 verfügbar
Fußteil erste Seite Rand unten	Ab v16 verfügbar
Erste Seite ist rechte Seite	Ab v16 verfügbar

Druckparameter für Dokument

Plug-In 4D Write	4D Write Pro
Papiertyp	Nicht verfügbar
Querformat	Nicht verfügbar
Breite	Nicht verfügbar
Höhe	Nicht verfügbar
Benutzerdefinierte Ränder	Nicht verfügbar
Skalierung	Nicht verfügbar
X Auflösung	Nicht verfügbar
Y Auflösung	Nicht verfügbar

Bilder

Hinweis zur Kompatibilität: Unter Windows werden in 4D Write eingefügte Bilder im Mac OS PICT Format importiert, lassen sich aber nicht in 4D Write Pro (64-bit und 32 bit Version) rendern. Wenn Sie Dokumente mit diesem Bildtyp importieren, müssen Sie diese erst in ein anderes Format konvertieren. Beachten Sie, dass das PICT Format überholt ist und nicht länger verwendet werden darf. Weitere Informationen dazu finden Sie unter [Bilder in PICT Format](#).

Plug-In 4D Write	4D Write Pro
X (links)	(& Position: absolut) (nur für Bilder in Seite)
Y (oben)	(& Position: absolut) (nur für Bilder in Seite)
Breite	Verfügbar
Höhe	Verfügbar
Seitennummer	Nicht verfügbar
Nach hinten	Nicht verfügbar
Nicht auf erster Seite	Nicht verfügbar
Bildformat (Größe angepasst, etc.)	Verfügbar
Ist Ausdruck	Verfügbar (ab v16 R5)
Größe beibehalten	Nicht importiert (in 4D Write Pro automatisch)

Eigenschaften für Zeichen

Plug-In 4D Write

Kursiv
Fett
Durchgestrichen
Unterstrichen
Schatten
Exponent (hoch- oder tiefgestellt)
Großbuchstaben oder Kapitälchen
Schriftfamilie
Schriftgröße
Farbe Text
Farbe Hintergrund
Farbe Unterstrichen
Farbe Durchgestrichen
Farbe Schatten
Benutzerdefiniert
Rechtschreibprüfung (Syntax & Grammatik an oder aus)
Darstellung URL
Stilvorlage

4D Write Pro (span HTML Tags)

Verfügbar
Verfügbar
Verfügbar
Verfügbar
Verfügbar
Verfügbar
Verfügbar
Verfügbar
Verfügbar
Verfügbar
Verfügbar
Verfügbar
Verfügbar
Nicht verfügbar
Nicht verfügbar
Nicht verfügbar
Nicht importiert (Stilarten werden importiert, aber Stilvorlagen sind nicht verfügbar)

Eigenschaften für Absatz

Plug-In 4D Write

Text ausrichten
Zeilenhöhe
Bullet/Punkte
Abstand links
Abstand rechts
Text einrücken
Rahmenlinie Stil
Rahmenlinie Farbe
Rahmen Hintergrundfarbe
Linker Rand
Rechter Rand
Oberer Rand & Oberer Rand innen
Unterer Rand & Unterer Rand innen
Abstand Rand
Stilvorlage
Tabulatoren

4D Write Pro

Verfügbar
Verfügbar
Verfügbar
Verfügbar
Verfügbar
Verfügbar
Verfügbar
Verfügbar
Verfügbar
Verfügbar
Verfügbar
Verfügbar
Verfügbar
Verfügbar
Verfügbar
Verfügbar
Verfügbar

Hyperlink Aktion

Plug-In 4D Write

URL Link (*)
4D Methode Link
Öffne Dokument Link

4D Write Pro

Verfügbar
Nicht verfügbar
Verfügbar (wird in eine URL lokale Datei konvertiert)

(*) Ab 4D v16 R4 werden URL Links in editierbare Attribute umgewandelt. Weitere Informationen dazu finden Sie im Abschnitt [URLs verwalten](#).

4D Ausdrücke

Plug-In 4D Write

4D Ausdruck
Datum & Zeit
HTML Ausdruck
RTF Ausdruck

4D Write Pro

Verfügbar
Verfügbar
Nicht verfügbar
Nicht verfügbar

Textdaten

Plug-In 4D Write

Hauptteil Textdaten
Kopfteil Textdaten
Fußteil Textdaten

4D Write Pro

Verfügbar
Verfügbar
Verfügbar

■ Ausdrücke in einem 4D Write Pro Dokument filtern

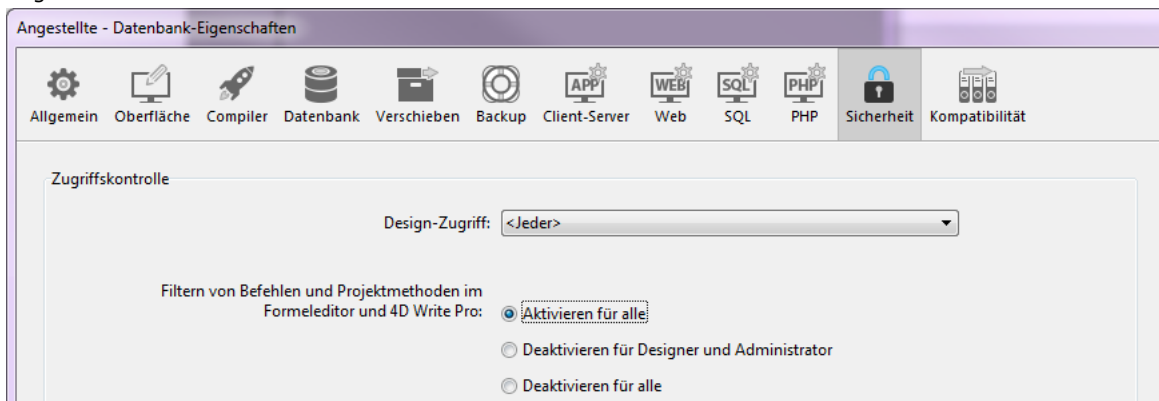
Überblick

4D Write Pro Dokumente können Referenzen auf dynamische 4D Ausdrücke enthalten, wie Variablen und Felder, aber auch Formeln, Projektmethoden oder 4D Befehle. Diese Referenzen werden beim Anzeigen oder Drucken des Dokuments bewertet. Aus Sicherheitsgründen muss der Entwickler die Bewertung von dynamischen Ausdrücken steuern, um die Verwendung unpassender Ausdrücke, unvorhergesehene Änderungen oder Nebeneffekte in der Anwendung zu verhindern. Das unterbindet Aufrufen von Befehlen wie **DELETE SELECTION** oder von Projektmethoden wie "LöscheRechnungen".

Hinweis: Ausdrücke lassen sich über den Befehl **ST INSERT EXPRESSION** oder durch Bearbeiten von 4D Write Pro Dokumenten mit einem HTML Editor einfügen.

Standardeinstellung 4D Filtern

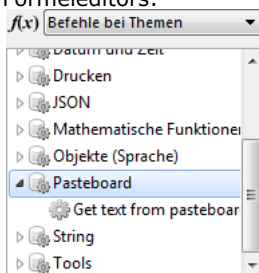
Ab 4D v16 greift beim Bewerten von 4D Befehlen und Projektmethoden, die in 4D Write Pro Dokumente eingefügt sind, die Einstellung zum Filtern auf Datenbankebene. In den Datenbank-Eigenschaften auf der Seite **Sicherheit** gibt es drei Möglichkeiten:



Standardmäßig ist die Option **Aktivieren für alle** markiert, d.h. Befehle und Methoden müssen explizit zugelassen sein, sonst werden Bewertungsfehler zurückgegeben. Sie können das Filtern für alle Benutzer oder teilweise deaktivieren (für Designer und Administrator). Die Option gilt für die gesamte 4D Anwendung und steuert alle Bewertungen von Benutzerformeln. Weitere Informationen dazu finden Sie im Handbuch *Designmodus*.

Ist diese Option aktiviert:

- Können Benutzer nur Befehle aufrufen, für die Formeln möglich sind. Diese Liste erscheint im rechten Teil des Formeleditors:



- Können Benutzer keine Projektmethode ausführen. Sollen Projektmethoden in 4D Write Pro Dokumenten ausführbar sein, müssen sie explizit mit dem Befehl **SET ALLOWED METHODS** deklariert werden.

Wird beim Bewerten eines Ausdrucks in einem 4D Write Pro Dokument eine nicht-zulässige 4D Methode bzw. ein Befehl gefunden, wird der Wert durch " #command_5#command_5 Error #command_5 48" ersetzt.

Beispiel

Sie haben in Ihrem 4D Write Pro Dokument folgenden Ausdruck eingefügt:

```
ST INSERT EXPRESSION(*;"WriteProArea";"Gender")
```

Standardmäßig ist die Option zur Sicherheit markiert, d.h. die Methode **Gender** gibt einen Fehler zurück:

```
Dear ## Error # 48 Smith,
```

Führen Sie folgenden Code aus:

```
ARRAY TEXT(aTallow;1)  
aTallow{1}:="Gender"
```

SET ALLOWED METHODS(aTallow)

wird die Methode **Gender** bewertet und gibt folgenden Wert zurück:

Dear **Mr.** Smith,

Kompatibilität

In bisherigen Releases war das Filtern für 4D Write Pro Dokumente nicht aktiviert. Enthalten Ihre 4D Write Pro Dokumente Referenzen auf 4D Methoden, werden sie nach Konvertieren in 4D v16 oder höher nicht mehr bewertet. Stattdessen erscheint die Meldung "#command_5_command_5 Error #command_5 48".

In diesem Fall müssen Sie die Methoden mit dem Befehl **SET ALLOWED METHODS** in die Liste der zugelassenen Methode aufnehmen.

4D Write Pro Standardaktionen verwenden

Überblick

Die Benutzeroberfläche von 4D Write Pro Bereichen lässt sich über einen umfangreichen Satz an **Standardaktionen** verwalten. Sie können sie verwenden:

- für Menübefehle über die **Menüleiste** oder die Funktion **Dynamic pop up menu**
- Listeneinträge (hierarchische Liste), PopUp/DropDown Listen, hierarchische PopUp-Menüs,
- für Schaltflächen, Optionsfelder
- oder als Parameter des Befehls **INVOKE ACTION**

Standardaktionen, die Objekten der Oberfläche zugewiesen sind, verwalten die Aktivierung/Deaktivierung des Objekts je nach Kontext. Weitere Informationen dazu finden Sie im Abschnitt **Standardaktionen** des Handbuchs *4D Designmodus*.

Es gibt zwei Arten von Aktionen:

- Spezifische 4D Write Pro Aktionen, die sich nur mit 4D Write Pro Bereichen einsetzen lassen
- **Andere Aktionen**, wie Schriften, Ausdrücke, Rechtschreibprüfung und Operationen zum Bearbeiten, die sich in 4D Write Pro Bereichen und anderen 4D Bereichen verwenden lassen (siehe **Standardaktionen**).

4D Write Pro Aktionen

Mit 4D Write Pro Bereichen sind folgende Standardaktionen verfügbar.

Hinweise:

- 4D Write Pro Aktionen haben keine entsprechenden Konstanten.
- Aktionen, die ein automatisches Menü/Liste anzeigen, lassen sich nur Menübefehlen, PopUp/DropDown-Listen oder hierarchischen PopUp-Menüobjekten zuweisen (siehe *)
- Optionsfelder und 3D Optionsfelder sollten Status-Aktionen, wie "section/differentFirstPage" oder "visibleHorizontalRuler" zugewiesen werden. Die Option "Three-states" wird nur mit Standard Optionsfeldern unterstützt.
- Die Aktion für einen Ausschnitt ist immer gleich mit dem aktuellen Status des gewählten Ausschnitts (vererbter Status, wenn das Attribut von einem übergeordneten Abschnitt oder von einem Standardattribut geerbt wurde), da sie den Status des gewählten Ausschnitts reflektieren muss (Status, der aktuell auf den Ausschnitt angewandt ist). Wird das Attribut eines Ausschnitts mit einer Standardaktion verändert, überschreibt sie dagegen nur das Attribut des gewählten Ausschnitts.

(*) **submenu:** Wird diese Aktion ohne Parameter verwendet und einem Menübefehl, einer PopUp/DropDown-Liste oder einem hierarchischen PopUp-Menüobjekt zugeordnet, zeigt sie automatisch ein Untermenü zur übergeordneten Aktion. Weisen Sie z.B. die Aktion "backgroundColor" einem Menübefehl zu, zeigt er beim Auswählen in Echtzeit die Einträge des Untermenüs

backgroundColor.

Weisen Sie einem hierarchischen PopUp-Menü die Aktion "zoom" zu, enthält es automatisch eine Liste der vordefinierten Zoom-Werte.

Beachten Sie, dass diese spezifischen Aktionen sich nicht für Schaltflächen verwenden lassen.

(**) **/showDialog:** Wird der String **/showDialog** an den Aktionsnamen angefügt, lässt sich das zugeordnete Standard Dialogfenster für die Aktion anzeigen. Sie können z.B. "paragraph/styleSheet/showDialog" zum Öffnen eines Eingabedialogs verwenden, um den Namen einer neuen Stilvorlage einzugeben.

Aktion	Dokument	Absatz	Bild	Ausschnitt	Kopfteil	Fußteil	Submenu(*)	showDialog
anchorHorizontalAlign			X				X	
anchorLayout			X				X	
anchorOrigin			X				X	
anchorPage			X				X	
anchorSection			X				X	
anchorVerticalAlign			X				X	
avoidPageBreakInsideEnabled		X						
background	X	X	X	X	X	X	X	
backgroundColor	X	X	X	X	X	X	X	X
backgroundClip	X	X	X	X	X	X	X	
backgroundImage	X	X	X	X	X	X	X	X
backgroundOrigin	X	X	X	X	X	X	X	
backgroundPositionH	X	X	X	X	X	X	X	
backgroundPositionV	X	X	X	X	X	X	X	
backgroundRepeat	X	X	X	X	X	X	X	
backgroundSizeH	X	X	X	X	X	X	X	
backgroundSizeV	X	X	X	X	X	X	X	
bookmark							X	
borderCollapse		X						
borderColor	X	X	X	X	X	X	X	X
borderRadius	X	X	X	X	X	X	X	
borders	X	X	X	X	X	X	X	
borderStyle	X	X	X	X	X	X	X	
borderWidth	X	X	X	X	X	X	X	
columnCount	X			X			X	
columnRuleColor	X			X			X	X
columnRuleStyle	X			X			X	
columnRuleWidth	X			X			X	
columns	X			X			X	
columnSpacing	X			X			X	
deleteStyleSheet		X	X				X	
direction		X					X	
doc	X						X	
dpi							X	
fontStyleWrite							X	
fontSubscript		X						
fontSuperscript		X						
footer							X	
footer/remove						X		
header							X	
header/remove					X			
htmlWYSIWIGEnabled	X							
image			X				X	
image/height			X				X	
image/verticalAlign			X				X	
insertColumnBreak	X			X				
insertImage	X							
insertPageBreak	X							
insertSectionBreak	X							
lineHeight		X					X	
listStartNumber							X	
listStyleImage								X
listStyleType		X					X	
margin	X	X	X	X	X	X	X	
merge		X						
minHeight		X	X				X	
minWidth		X	X				X	
moveToFront			X					
moveToBack			X					
newLineStyleSheet		X					X	
padding	X	X	X	X	X	X	X	
pageMode							X	
pageOrientation				X			X	
pageSize							X	
paragraph		X					X	
print	X							
refreshImages			X					

section				X					X	
section/differentFirstPage				X						
section/differentLeftRightPages				X						
section/name/showDialog				X						X
section/reset				X						
split		X								
styleSheet		X	X					X		X
styleSheets		X	X					X		
textAlign								X		
textIndent								X		
textLinethrough								X		
textLinethroughColor								X		X
textLinethroughStyle								X		
textShadow								X		
textShadowColor								X		X
textShadowOffset								X		
textTransform								X		
textUnderline								X		
textUnderlineColor								X		X
textUnderlineStyle								X		
updateStyleSheet		X	X					X		
userUnit								X		
verticalAlign		X						X		
visibleBackground	X	X	X	X		X	X			
view								X		
visibleFooters	X									
visibleHeaders	X									
visibleHiddenChars	X									
visibleHorizontalRuler	X									
visiblePageFrames	X									
widowAndOrphanControlEnabled	X									
width		X	X					X		
zoom	X							X		

anchorHorizontalAlign

Syntax: anchorHorizontalAlign?value={left | center | right}

Definiert die horizontale Ausrichtung des Bildes in Bezug auf den Ankerursprung für Bilder auf der Seite, oder dem Layoutrahmen für Bilder im eingebundenen Modus. Diese Aktion setzt den horizontalen Versatz auf 0 zurück (ist für Bilder im Textfluss nicht aktiviert).

anchorLayout

Syntax: anchorLayout?value={inline | front | behind}

Definiert die Ankerebene für ein Bild. Dient zum Verschieben eines Bildes im Textfluss von vorne nach hinten oder umgekehrt. **Hinweis:** Wird ein Bild vor/hinter dem Text in den Textfluss gesetzt, wird es an den Anfang des aktuell ausgewählten Textes gesetzt.

anchorOrigin

Syntax: anchorOrigin?value={paper-box | header-box | footer-box}

Definiert den Bereich in Bezug auf die absolute Position eines verankerten Bildes. Bezieht sich das Bild auf einen auf der Seite nicht sichtbaren Kopf-/Fußteil, wird das Bild nicht angezeigt. Diese Aktion setzt horizontalen und vertikalen Versatz auf 0.

Gilt nur für Seitenmodus, ist für Bilder im Textfluss nicht aktiviert.

anchorPage

Syntax: anchorPage?value={all | current | currentSubSection}

Definiert die Seite(n), wo das ausgewählte Bild erscheint (ist für Bilder im Textfluss nicht aktiviert).

Gilt nur im Seitenmodus.

anchorSection

Syntax: anchorSection?value={all | current}

Definiert den Ausschnitt bzw. Ausschnitte, wo das ausgewählte Bild erscheint (ist für Bilder im Textfluss nicht aktiviert).

Gilt nur im Seitenmodus.

anchorVerticalAlign

Syntax: anchorVerticalAlign?value={top | center | bottom}

Definiert die vertikale Ausrichtung des Bildes in Bezug auf den Ankerursprung für Bilder auf der Seite oder dem Layoutrahmen für Bilder im eingebundenen Modus. Diese Aktion setzt den vertikalen Versatz auf 0 (ist für Bilder im Textfluss nicht aktiviert).

avoidPageBreakInsideEnabled

Syntax: avoidPageBreakInsideEnabled
Erlaubt/Verweigert Seitenumbruch in Abschnitten.

background

Syntax: {doc | paragraph | image | section | header | footer}/background
Nur für Menübefehle. Standard Untermenü für alle Hintergrundattribute.

backgroundClip

Syntax: {doc | paragraph | image | section | header | footer}/backgroundClip?value={paper-box | border-box | padding-box | content-box}
Ändert Ziel für Zeichenbereich des Hintergrunds (Standardziel ist Absatz) Papierrand gilt nur für Dokument und Absatz.

backgroundColor

Syntax: {doc | paragraph | image | section | header | footer}/backgroundColor?value={<Css_color> | transparent}
Ändert Ziel für Hintergrund.

backgroundImage

Syntax: {doc | paragraph | image | section | header | footer}/backgroundImage?value=none
Hebt Ziel für Hintergrundbild auf (Standardziel ist Absatz).

backgroundOrigin

Syntax: {doc | paragraph | image | section | header | footer}/backgroundOrigin?value={paper-box | border-box | padding-box | content-box}
Ändert Ziel für Positionierung des Hintergrundbilds (Standardziel ist Absatz), Ziel Papierrand gilt nur für Dokument und Absatz

backgroundPositionH

Syntax: {doc | paragraph | image | section | header | footer}/backgroundPositionH?value={left | right | center}
Ändert Ziel für Hintergrundbild horizontale Ausrichtung (Standardziel ist Absatz).
Beispiel: *paragraph/backgroundPositionH?value=left*

backgroundPositionV

Syntax: {doc | paragraph | image | section | header | footer}/backgroundPositionV?value={top | bottom | center}
Ändert Ziel für Hintergrundbild vertikale Ausrichtung (Standardziel ist Absatz).
Beispiel: *paragraph/backgroundPositionV?value=top*

backgroundRepeat

Syntax: {doc | paragraph | image | section | header | footer}/backgroundRepeat?value={no-repeat | repeat | repeat-x | repeat-y}
Ändert Ziel für Hintergrundbild Wiederholungsmodus (Standardziel ist Absatz).

backgroundSizeH

Syntax: {doc | paragraph | image | section | header | footer}/backgroundSizeH?value={<Css_length> | <Css_percentage> | auto | cover | contain}
Ändert Ziel für Hintergrundbild Breite (Standardziel ist Absatz).
Beispiel: *paragraph/backgroundSizeH?value=100%. section/backgroundSizeH?value=cover*

backgroundSizeV

Syntax: {doc | paragraph | image | section | header | footer}/backgroundSizeV?value={<Css_length> | <Css_percentage> | auto}
Ändert Ziel für Hintergrundbild Höhe (Standardziel ist Absatz).
Beispiel: *paragraph/backgroundSizeV?value=50%. Beispiel: section/backgroundSizeV?value=40pt*

bookmark

Syntax: bookmark?index=<number>
Wählt die nte Bookmark.
Beispiel: *bookmark?index=2 // wählt die zweite Bookmark*

borderCollapse

Syntax: borderCollapse
Schließt ausgewählte Ränder und Rahmen innen für Absatz.

borderColor

Syntax: {doc | paragraph | image | section | header | footer}/{inside | outside/}{borderColor | borderColorLeft | borderColorRight | borderColorTop | borderColorBottom | borderColorLeftRight | borderColorTopBottom}?value=<Css_color>

Ändert Ziel für Rahmenfarbe (Standardziel ist Absatz).
Beispiel: `paragraph/borderColorLeft?value=green`

borderRadius

Syntax: {doc | paragraph | image | section | header | footer}/borderRadius?value=<Css_length>

Ändert Ziel für Rahmenradius (Standardziel ist Absatz).
Beispiel: `paragraph/borderRadius?value=4pt`

borders

Syntax: {doc | paragraph | image | section | header | footer/}{inside | outside/}borders
Standard Untermenü für Ziel Rahmen.

borderStyle

Syntax: {doc | paragraph | image | section | header | footer/}{inside | outside/}{borderStyle | borderStyleLeft | borderStyleRight | borderStyleTop | borderStyleBottom | borderStyleLeftRight | borderStyleTopBottom}?value={none | hidden | dotted | dashed | solid | double | groove | ridge | inset | outset}

Ändert Ziel für Rahmenstil (Standardziel ist Absatz).

Beispiel: `paragraph/borderStyleLeft?value=double`.

Beispiel: `outside/borderStyle?value=solid //setzt Rahmen außen für gewählte Absätze auf Stil kompakt`.

Beispiel: `inside/borderStyle?value=none //entfernt Rahmen innen für gewählte Absätze`.

borderWidth

Syntax: {doc | paragraph | image | section | header | footer/}{inside | outside/}{borderWidth | borderWidthLeft | borderWidthRight | borderWidthTop | borderWidthBottom | borderWidthLeftRight | borderWidthTopBottom}?value=<Css_length>

Ändert Ziel für Rahmenbreite (Standardziel ist Absatz).

Beispiel: `paragraph/borderWidthLeft?value=4pt`

columnCount

Syntax: {section/}columnCount?value={1<=number<=20}

Anzahl Spalten im aktuellen Dokument bzw. Abschnitt.

Beispiel: `section/columnCount?value=3`

columnRuleColor

Syntax: {section/}columnRuleColor?value={CSS color}

Farbe für Zierlinie zwischen Spalten im Dokument oder Abschnitt.

Beispiel: `columnRuleColor?value="#command_5FFFFFFF"`

columnRuleStyle

Syntax: {section/}columnRuleStyle?value={none | dotted | dashed | solid | double | groove | ridge | inset | outset}

Stil für die Zierlinie zwischen Spalten im Dokument oder Abschnitt.

Beispiel: `columnRuleStyle?value="solid"`

columnRuleWidth

Syntax: {section/}columnRuleWidth?value={CSS length}

Breite der Zierlinie zwischen Spalten im Dokument oder Abschnitt.

Beispiel: `columnRuleWidth?value="2pt"`

columns

Syntax: {section/}columns

Nur für einen Menüeintrag: Erstellt automatisch ein komplettes Untermenü **Spalten** mit allen Untermenüs für Spaltenaktionen (außer **insertColumnBreak**)

columnSpacing

Syntax: {section/}columnSpacing?value={CSS length}

Abstand zwischen zwei Spalten im aktuellen Dokument bzw. Abschnitt.

Beispiel: `columnSpacing?value="1cm"`

deleteStyleSheet

Syntax: {paragraph | image}/deleteStyleSheet?index={1<= number <= number of target style sheets}

Entfernt die nte Stilvorlage aus den gespeicherten Stilvorlagen für das gewählte Ziel (Standardziel ist Absatz). Name des Menüeintrags wird in Echtzeit durch Name der Stilvorlage ersetzt.

direction

Syntax: direction?value={ltr | rtl}

Ausrichtung des Absatzes

doc

Syntax: doc

Standard Untermenü für Layout des Dokument Hauptteils (Attribute eingebundener Modus oder Standardattribute für Abschnitte – für Rahmen nur für eingebundenen Modus)

dpi

Syntax: Ändert die dpi der aktuellen Ansicht (unabhängig vom intern verwendeten Dokumentattribut [wk_dpi](#) zum Konvertieren von Pixel < - > Punkte).

Ändert dpi für Ansicht.
Beispiel: *dpi?value=72*

fontStyleWrite

Syntax: fontStyleWrite

Nur für Menübefehle. Zeigt Untermenü für Standard Schriftstile für 4D Write Pro.

fontSubscript

Syntax: fontSubscript

Schaltet um auf Attribut Schrift tiefgestellt.

fontSuperscript

Syntax: fontSuperscript

Schaltet um auf Attribut Schrift hochgestellt.

footer

Syntax: footer

Zeigt Untermenü für Fußteil.

footer/remove

Syntax: footer/remove

Entfernt den gewählten Fußteil.

header

Syntax: header

Zeigt Untermenü für Kopfteil.

header/remove

Syntax: header/remove

Entfernt den gewählten Kopfteil.

htmlWYSIWIGEnabled

Syntax: htmlWYSIWIGEnabled

Schaltet um auf html wysiwyg Modus.

image

Syntax: image

Zeigt Untermenü für Bildlayout.

image/height

Syntax: image/height?value={<Css_length> | auto}

Zielhöhe (nur für Bild).

Beispiel: *image/height?value=50pt*. For image/width, see *width*.

image/verticalAlign

Syntax: image/verticalAlign?value={top | middle | bottom | baseline | super | sub}

Bild vertikale Ausrichtung.

Beispiel: *image/verticalAlign?value=super*

insertImage

Syntax: insertImage

Öffnet ein Dialogfenster für Bildauswahl und fügt das gewählte Bild (wenn vorhanden) als Zeichen in den Bereich ein.

insertColumnBreak

Syntax: insertColumnBreak

Fügt an der Auswahl einen Spaltenumbruch ein.

insertPageBreak

Syntax: insertPageBreak

Fügt an der Auswahl einen Seitenumbruch ein.

insertSectionBreak

Syntax: insertSectionBreak

Fügt an der Auswahl einen Abschnittumbruch ein.

lineHeight

Syntax: lineHeight?value={<Css_length> | <Css_percentage>}

Zeilenhöhe für Absatz.

Beispiel: *lineHeight?value=120%*

listStartNumber

Syntax: listStartNumber?value={<number> (longint) | auto}

Startnummer Liste.

Beispiel: *listStartNumber?value=10*

listStyleImage

Syntax: listStyleImage/showDialog

Öffnet Dialog, um ein Bild für Bullet einer Liste auszuwählen; wird kein Bild gefunden, erscheint Bullet als Kreis.

listStyleType

Syntax: listStyleType?value={none | disc | circle | square | hollow-square | diamond | club | decimal | decimal-leading-zero | lower-latin | lower-roman | upper-latin | upper-roman | lower-greek | decimal-greek | armenian | georgian | hebrew | hiragana | katakana | cjk-ideographic}

Bullettyp für Liste in Absatz.

margin

Syntax: {doc | paragraph | image | section | header | footer/}{inside | outside/}{margin | marginLeft | marginRight | marginTop | marginBottom | marginLeftRight | marginTopBottom}?value={<Css_length> | auto}

Ändert Zielrand (Standardziel ist Absatz).

Beispiel: *margin?value=4pt //setzt alle Ränder des Absatzes auf 4pt.*

Beispiel: *outside/margin?value=4pt //setzt Außenränder für Absatz auf 4pt.*

Beispiel: *doc/marginLeft?value=1cm //setzt linken Rand auf 1 cm für das Dokument wie im Modus eingebunden.*

Beispiel: *section/marginLeft?value=1cm //setzt linken Rand auf 1 cm für die Seiten des ersten gewählten Abschnitts.*

merge

Syntax: merge

Führt Absätze zusammen.

minHeight

Syntax: {paragraph | image}/minHeight?value=<Css_length>

Mindesthöhe Ziel (Standardziel ist Absatz).

Beispiel: *paragraph/minHeight?value=50pt*

minWidth

Syntax: {paragraph | image}/minWidth?value=<Css_length>

Mindestbreite Ziel (Standardziel ist Absatz).

Beispiel: *paragraph/minWidth?value=50pt*

moveToBack

Syntax: moveToBack

Setzt das Bild nach hinten. (ist für Bilder im Textfluss nicht aktiviert)

moveToFront

Syntax: moveToFront

Setzt das Bild nach vorne. (ist für Bilder im Textfluss nicht aktiviert)

newLineStyleSheet

Syntax: newLineStyleSheet?index={1<=number<=number of paragraph stylesheets | newLineStyleSheet?value=auto}

Gewählte Absatzelemente verwenden nte Stilvorlage für neue Absätze, die mit Zeilenschaltung oder durch Splitten von Absätzen entstehen; Bei Wert=auto verwenden neue Absätze dieselbe Stilvorlage (Standard). (Name des Menüeintrags wird in Echtzeit mit Namen der Stilvorlage ersetzt)

padding

Syntax: {doc | paragraph | image | section | header | footer/}{inside | outside/}{padding | paddingLeft | paddingRight | paddingTop | paddingBottom | paddingLeftRight | paddingTopBottom }?value=<Css_length>

Ändert Ziel für Einrückung (Standardziel ist Absatz).

pageMode

Syntax: pageMode?value={embedded | page | draft}

Ändert Ansichtsmodus. Beispiel: *pageMode?value=page*

pageOrientation

Syntax: {section/}pageOrientation?value={landscape | portrait}

Ändert Seitenausrichtung des Dokuments oder des ersten gewählten Abschnitts (ist kein Abschnitt angegeben, gilt die Ausrichtung für alle Abschnitte außer dem ersten). Beispiel: *pageOrientation?value=portrait //setzt Hochformat für das ganze Dokument*

Beispiel: *section/pageOrientation?value=landscape //setzt Querformat für den ersten gewählten Abschnitt*

pageSize

Syntax: pageSize?index=<number>

Ändert Seitengröße des Dokuments: Nummer ist nte Seitengröße in der Liste der verfügbaren Seitengrößen im aktuellen Drucker.

Beispiel: *pageSize?index=1*

paragraph

Syntax: paragraph

Standard Untermenü für Absatzlayout.

print

Syntax: print

Druckt Dokumente wie im Seitenmodus in der aktuellen Ansicht des 4D Write Pro Bereichs gerendert.

refreshImages

Syntax: refreshImages

Lädt erneut Bilder aus dem Netzwerk für Bilder mit URLs.

section

Syntax: section

Standard Untermenü für Seitenlayout für den ersten gewählten Abschnitt.

section/differentFirstPage

Syntax: section/differentFirstPage

Abschnitt hat unterschiedliche erste Seite oder nicht.

section/differentLeftRightPages

Syntax: section/differentLeftRightPages

Abschnitt hat unterschiedliche linke und rechte Seiten oder nicht.

section/name/showDialog

Syntax: section/name/showDialog

Zeigt Dialog zur Eingabe des ersten gewählten Namen für Abschnitt.

section/reset

Syntax: section/reset

Setzt alle Attribute für ersten gewählten Abschnitt zurück auf Standardattribute - sie werden vererbt von Doc Attributen für Hintergrund, Rahmen und Füllen (gleich wie für eingebundenen Modus) und 2,5 cm für Seitenränder. Die Spalteneigenschaften des ersten ausgewählten Abschnitts werden ebenfalls entfernt (es gelten also nur die Standardeigenschaften des Abschnitts).

split

Syntax: split

Splittet Abschnitte.

styleSheet

Syntax: {paragraph | image}/styleSheet?value={1<=number<=number of target style sheets}

Wendet die nte Stilvorlage auf die gewählten Zielelemente an. Mit Dialog verwendet die neue Stilvorlage aktuelle Attribute aus dem ersten gewählten Ziel (Standardziel ist Absatz).

Beispiel: *styleSheet?value=1 //Stilvorlage erster Absatz auf die ausgewählten Absätze anwenden und als Vorlage für Absatz setzen.*

styleSheets

Syntax: {paragraph | image}/styleSheets
Standard Untermenü zum Bearbeiten von Ziel Stilvorlagen.

textAlign

Syntax: textAlign?value={left | right | center | justify | initial}
Textausrichtung für Absatz (initial=verwende rechts für rechtsbündig bzw. links für linksbündig).

textIndent

Syntax: textIndent?value=<Css_length> (kann negativ sein)
Öffnet Texteinrückung für Absatz (Einrückung erste Zeile).
Beispiel: *textIndent?value=2cm*

textLinethrough

Syntax: textLinethrough
Nur für Menübefehle. Untermenü Durchgestrichen (mit Stil und Farbe).

textLinethroughColor

Syntax: textLinethroughColor?value={<Css_color> | currentColor}
Beispiel: *textLinethroughColor?value=red*

textLinethroughStyle

Syntax: textLinethroughStyle?value={none | solid | dotted | dashed | double | semi-transparent | word}
Beispiel: *textLinethroughStyle?value=dotted*

textShadow

Syntax: textShadow
Nur für Menübefehle. Standard Untermenü für Textschatten (mit Farbe und Versatz)

textShadowColor

Syntax: textShadowColor?value={<Css_color> | none}
Ändert Schattenstatus oder -farbe für Zeichen.
Beispiel: *textShadowColor?value=green*

textShadowOffset

Syntax: textShadowOffset?value=<Css_length>
Nur Länge in pt wird unterstützt.
Beispiel: *textShadowOffset?value=2pt*

textTransform

Syntax: textTransform?value={none | capitalize | uppercase | lowercase | small-uppercase}
Formatiert Text.
Beispiel: *textTransform?value=capitalize*

textUnderline

Syntax: textUnderline
Nur für Menübefehle. Untermenü Unterstreichen (mit Stil und Farbe)

textUnderlineColor

Syntax: textUnderlineColor?value={<Css_color> | currentColor}
Beispiel: *textUnderlineColor?value=red*

textUnderlineStyle

Syntax: textUnderlineStyle?value={none | solid | dotted | dashed | double | semi-transparent | word}
Beispiel: *textUnderlineStyle?value=dotted*

updateStyleSheet

Syntax: {paragraph | image}/updateStyleSheet
Aktualisiert erstes gewähltes Ziel Stilvorlage aus dem ersten gewählten Ziel der aktuellen Attribute (Standardziel ist Absatz).

userUnit

Syntax: userUnit?value={cm | mm | in | pt}
Ändert Einheit des Dokumentlayouts (wird auf Lineale übertragen).

verticalAlign

Syntax: verticalAlign?value={top | middle | bottom}

Absatz vertikale Ausrichtung.

Beispiel: *verticalAlign?value=top*. Die Auswirkung richtet sich nach dem Wert *minHeight* für den Absatz.

view

Syntax: view

Nur für Menübefehle. Standard Untermenü für Einstellungen Ansicht.

visibleBackground

Syntax: visibleBackground

Blendet Hintergrund ein/aus.

visibleFooters

Syntax: visibleFooters

Blendet Fußteile ein/aus.

visibleHeaders

Syntax: visibleHeaders

Blendet Kopfteile ein/aus.

visibleHiddenChars

Syntax: visibleHiddenChars

Blendet Zeichen ein/aus.

visibleHorizontalRuler

Syntax: visibleHorizontalRuler

Blendet horizontales Lineal ein/aus.

visiblePageFrames

Syntax: visiblePageFrames

Blendet Seitenrahmen ein/aus.

widowAndOrphanControlEnabled

Syntax: widowAndOrphanControlEnabled

Schaltet um zwischen Kontrolle Schusterjungen und Hurenkinder.

width

Syntax: {paragraph | image}/width?value=<Css_length> | auto

Breite Ziel (nur für Absatz und Bild - Standardziel ist Absatz).

Beispiel: *image/width?value=50pt*

zoom

Syntax: zoom?value={25% <= percentage in Css <= 400%}

Ändert Zoom für Ansicht des Dokuments.

Beispiel: *zoom?value=120%*

Andere Aktionen

Folgende Standardaktionen für 4D Formulare sind auch für 4D Write Pro Bereiche verwendbar:

- **Aktionen Bearbeiten**, z.B. copy/paste
- Aktionen für **Schrift**, wie fontBold oder fontSize
- Aktionen für **Dynamische Ausdrücke** zum Verwalten eingefügter Ausdrücke
- Aktionen für **Rechtschreibprüfung**

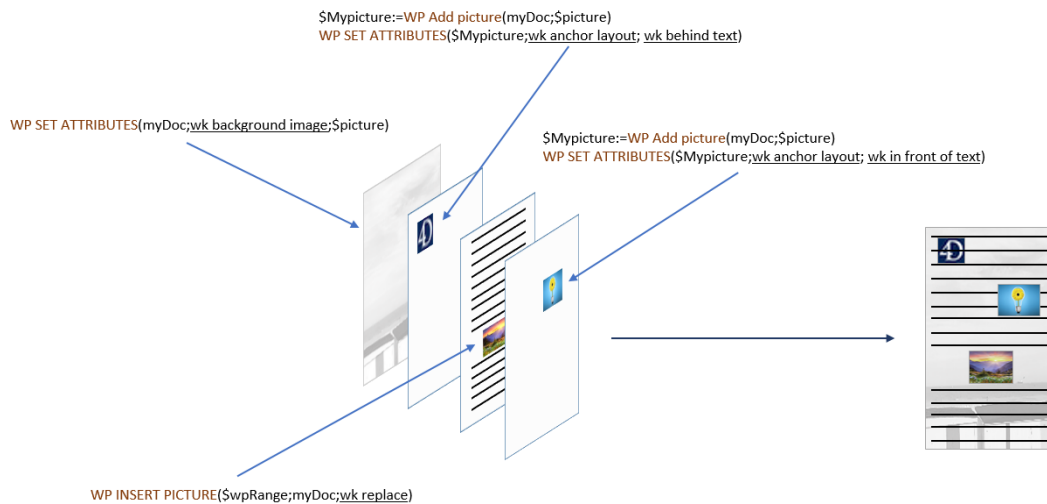
Bilder verwalten

Bilder hinzufügen

Es gibt verschiedene Möglichkeiten, um Bilder in einem 4D Write Pro Dokument einzufügen:

- als **Hintergrundbild**. Dazu verwenden Sie den Befehl **WP SET ATTRIBUTES** und setzen das Attribut `wk background image` oder `wk background image url`.
- als **Bild im Textfluss**, d.h. genauso wie ein Zeichen im Text. Dazu verwenden Sie den Befehl **WP INSERT PICTURE** oder **ST INSERT EXPRESSION**.
- als **verankertes Bild** auf der Seite (hinter oder vor dem Text). Dazu verwenden Sie den Befehl **WP Add picture** und setzen das Attribut `wk anchor layout`.

Nachfolgende Darstellung zeigt die verschiedenen Ebenen:



Hintergrundbilder

Bilder lassen sich als Hintergrund von 4D Write Pro Dokumenten und Dokumentelementen setzen, wie Tabellen, Absätze, Ausschnitte, Kopf-/Fußteile, etc..

Dieses Beispiel zeigt zwei Arten, um ein Bild in einem Dokument als Hintergrund zu setzen:



Hintergrundbilder lassen sich per Programmierung oder über das Kontextmenü setzen. Weitere Informationen dazu finden Sie im Abschnitt **Hintergrund** des Kapitels **4D Write Pro Bereich verwenden**.

Verankerte Bilder positionieren und anzeigen

Verankerte Bilder werden mit einer absoluten Position vor oder hinter dem Text hinzugefügt und mit der Seite oder bestimmten Teilen des Dokuments verankert (z.B. Kopfteil, Fußteil, Ausschnitt). Die absolute Position für ein Bild wird über die Befehle **WP Add picture** und **WP SET ATTRIBUTES** gesetzt.

Die Position des verankerten Bildes lässt sich über folgende Attribute bzw. Standardaktionen verändern:

Eigenschaft (Konstante)	Standardaktion
<u>wk anchor layout</u>	anchorLayout
<u>wk anchor horizontal offset</u>	
<u>wk anchor horizontal align</u>	anchorHorizontalAlign
<u>wk anchor vertical offset</u>	
<u>wk anchor vertical align</u>	anchorVerticalAlign
<u>wk anchor origin</u>	anchorOrigin
<u>wk anchor page</u>	anchorPage
<u>wk anchor section</u>	anchorSection
	moveToBack
	moveToFront

Alle verankerten Bilder erscheinen nur im Ansichtsmodus Seite. Sie erscheinen nicht, wenn:

- Sie mit einem Kopf- oder Fußteil verankert sind, der nicht sichtbar ist
- Der Ansichtsmodus Draft aktiv ist
- Sie zentriert oder mit Ausschnitten verankert sind und die Option **HTML WYSIWYG** markiert ist
- Die Option "Hintergrund" **nicht** ausgewählt ist

Bildeigenschaften

Alle Bilder haben Eigenschaften (Attribute) wie Höhe, Breite, Rahmen, etc., die sich über die 4D Write Pro Befehle **WP GET ATTRIBUTES** und **WP SET ATTRIBUTES** oder Standardaktionen setzen lassen.

- Die komplette Liste der verfügbaren Bildeigenschaften finden Sie auf der Seite **4D Write Pro Attribute**
- Der Abschnitt **Bilder** erläutert die Attribute, die nur für Bilder gelten
- Die Seite **4D Write Pro Standardaktionen verwenden** zeigt die Eigenschaften für Bilder, die über Standardaktionen verfügbar sind

Bildreferenz oder Bild URL

Sie können mit Bildreferenzen (Bildvariablen, Felder, Ausdrücke) oder Bild URLs (Text mit lokaler oder Netzwerkadresse des Bildes) arbeiten.

Es gibt zwei Sätze von Attributen, um entweder eine Bildreferenz oder eine Bild URL zu setzen oder erhalten:

- wk image, wk background image, wk list style image: Setzen oder erhalten Bildreferenzen. Erhalten Sie ein Bild über eines dieser Attribute, empfangen Sie ein 4D Bild, unabhängig wie das Bild definiert wurde.
- wk image url, wk background image url, wk list style image url: Setzen oder erhalten Bild URLs. Erhalten Sie ein Bild über eines dieser Attribute, empfangen Sie einen Text. Wurde das Bild über eine Referenz definiert (z.B. eine Bildvariable), erhalten Sie eine lokale URI, gefolgt von dem Bild in Base64.

Beispiel:

```
$range:=WP Get selection(WPArea) //vom Benutzer ausgewähltes Bild erhalten
$range:=WP Create picture range($range) //Ausschnitt erstellen
$url:="http://doc.4d.com/image/logo/poweredby4D_web.png"

WP SET ATTRIBUTES($range;wk image;$url) //Bildreferenz von einer URL setzen

//das Bild erhalten
C_PICTURE(vPictureGet)
WP GET ATTRIBUTES($range;wk image;vPictureGet) //vPictureGet enthält ein Bild
C_TEXT(vPictureURLGet)
WP GET ATTRIBUTES($range;wk image url;vPictureURLGet) //vPictureURLGet=$url
```

Bilder finden

Über folgende Funktionen lassen sich Bilder zurückgeben:

- **WP Create picture range** - nur für Bilder im Textfluss
- **WP Get selection** - nur für vom Benutzer ausgewählte Bilder










































Aktionen per Mausklick oder Tastenkürzel

Bilder lassen sich mit der Maus oder der Tastatur verwalten. Es gibt folgende Operationen:

- **Auswählen**
 - Bilder auf der obersten Ebene lassen sich durch Anklicken auswählen.
 - Hintergrundbilder lassen sich durch Anklicken mit gedrückter STRG-Taste (Windows) oder Befehlstaste (Mac) auswählen.
 - Es lässt sich nur ein einzelnes Bild mit absoluter Position auswählen: mehrfache Auswahlen sind nicht möglich.
- **Verschieben**
 - Bilder lassen sich mit der Maus verschieben und in der Größe verändern (Bilder im Textfluss lassen sich nicht außerhalb des Dokuments ziehen).

- **Einfügen**
 - Wird ein Bild mit absoluter Position in den Draft-Modus gesetzt, erscheint es im Textfluss am Beginn der aktuellen Tastaturlauswahl.
- **Ausschneiden/Einsetzen/kopieren**
 - Bilder lassen sich in die Zwischenablage kopieren oder während der Auswahl ausschneiden/einsetzen.
 - Wird ein Bild mit absoluter Position in den Seitenmodus übertragen, wird es auf der ersten ausgewählten Seite verankert, unabhängig von den Einstellungen der Originalseite; übertragene Bilder behalten ihren Versatz nur im gleichen Seitenmodus wie das Original. Wird ein Bild z.B. in den Seitenmodus kopiert und in den eingebundenen Modus übertragen, wird der Versatz auf 0 zurückgesetzt.
- **Löschen** - Ist ein Bild markiert, lässt es sich löschen mit:
 - der Taste **Löschen** oder
 - der Tastenkombination für die Aktion **Löschen**

4D Write Pro Programmiersprache

-  Über 4D Write Pro Objekte
-  Befehle aus dem Kapitel Objekte (Formulare) verwenden
-  Befehle aus dem Kapitel Mehrfachstil Text verwenden
-  Per Programmierung auf Dokumentinhalte zugreifen
-  WP Add picture
-  WP CREATE BOOKMARK
-  WP Create paragraph range
-  WP Create picture range
-  WP Create range
-  WP Create table range Neu 17.0
-  WP DELETE BOOKMARK
-  WP EXPORT DOCUMENT
-  WP EXPORT VARIABLE
-  WP GET ATTRIBUTES
-  WP Get body
-  WP Get bookmark range
-  WP GET BOOKMARKS
-  WP Get element by ID Neu 17.0
-  WP Get elements Neu 17.0
-  WP Get footer
-  WP Get frame
-  WP Get header
-  WP Get page count
-  WP Get position Neu 17.0
-  WP Get selection
-  WP Get text Neu 17.0
-  WP Import document
-  WP INSERT BREAK Updated 17.0
-  WP INSERT DOCUMENT
-  WP INSERT PICTURE
-  WP Insert table
-  WP Is font style supported
-  WP New
-  WP PRINT
-  WP RESET ATTRIBUTES
-  WP SELECT
-  WP SET ATTRIBUTES
-  WP SET FRAME
-  WP SET TEXT Neu 17.0
-  WP Table append row
-  WP Table get cells Updated 17.0
- WP Table get columns Updated 17.0
- WP Table get rows Updated 17.0
- WP USE PAGE SETUP
- 4D Write Pro Konstanten
- 4D Write Pro Attribute

Über 4D Write Pro Objekte

4D Write Pro Objekte

In 4D Applikationen werden 4D Write Pro Dokumente über Variablen vom Typ *Objekt* verwaltet. Weitere Informationen dazu finden Sie unter dem Befehl **C_OBJECT**.

Ein einmal gespeichertes 4D Write Pro Objekt lässt sich:

- mit dem Formularobjekt "writeProBereich" in einem 4D Formular anzeigen und verändern. Dabei muss der Formularbereich denselben Namen wie die Objektvariable haben (siehe **Einen 4D Write Pro Bereich erstellen**).
- per Programmierung mit Befehlen aus verschiedenen Kapiteln verwalten (siehe unten).
- Über den Befehl **WP EXPORT DOCUMENT** in ein HTML Dokument auf der Festplatte exportieren.

Befehle zum Verwalten von 4D Write Pro Objekten

Sie können 4D Write Pro Objekte mit 4D Befehlen aus verschiedenen Kapiteln verwalten:

- Spezifische **4D Write Pro** Befehle; sie werden in diesem Kapitel näher erläutert.
- Befehle aus dem Kapitel **Objekte (Formulare)** für Formatierungen und Objekteigenschaften; weitere Informationen dazu finden Sie im Abschnitt **Befehle aus dem Kapitel Objekte (Formulare) verwenden**.
- Befehle aus dem Kapitel **Mehrfachstil Text** für Funktionalitäten, die den Inhalt betreffen; weitere Informationen dazu finden Sie im Abschnitt **Befehle aus dem Kapitel Mehrfachstil Text verwenden**.
- Befehle aus dem Kapitel **Objekte (Sprache)** zum Arbeiten mit Attributen für 4D Write Pro Objekte; weitere Informationen dazu finden Sie im Abschnitt **4D Write Pro Dokumente in 4D Objektfeldern speichern**.

Befehle aus dem Kapitel Objekte (Formulare) verwenden

Folgende Befehle unterstützen 4D Write Pro Formularobjekte:

Befehl	Kommentar
OBJECT DUPLICATE	
OBJECT Get auto spellcheck/OBJECT SET AUTO SPELLCHECK	
OBJECT Get border style/OBJECT SET BORDER STYLE	
OBJECT Get context menu/OBJECT SET CONTEXT MENU	
OBJECT GET COORDINATES /OBJECT SET COORDINATES	
OBJECT Get data source/OBJECT SET DATA SOURCE	
OBJECT GET DRAG AND DROP OPTIONS/OBJECT SET DRAG AND DROP OPTIONS	
OBJECT Get enabled/OBJECT SET ENABLED	
OBJECT Get enterable/OBJECT SET ENTERABLE	
OBJECT GET EVENTS/OBJECT SET EVENTS	
OBJECT Get focus rectangle invisible/OBJECT SET FOCUS RECTANGLE INVISIBLE	
OBJECT Get font/OBJECT SET FONT	Wird auf die aktuelle Auswahl angewendet (falls vorhanden)
OBJECT Get font size/OBJECT SET FONT SIZE	Wird auf die aktuelle Auswahl angewendet (falls vorhanden)
OBJECT Get font style/OBJECT SET FONT STYLE	Wird auf die aktuelle Auswahl angewendet (falls vorhanden)
OBJECT Get horizontal alignment/OBJECT SET HORIZONTAL ALIGNMENT	Wird auf die aktuelle Auswahl angewendet (falls vorhanden). Unterstützt die Konstante <u>wk_justify</u> für 4D Write Pro Bereiche
OBJECT GET RESIZING OPTIONS/OBJECT SET RESIZING OPTIONS	
OBJECT SET COLOR	Wird auf die aktuelle Auswahl angewendet (falls vorhanden)
OBJECT GET RGB COLORS/OBJECT SET RGB COLORS	Wird auf die aktuelle Auswahl angewendet (falls vorhanden)
OBJECT Get type	
OBJECT Get vertical alignment/OBJECT SET VERTICAL ALIGNMENT	Vertikale Ausrichtung von Absätzen: Ist nur relevant, wenn die Höhe des Absatzes größer als die Höhe des Absatztextes ist.
OBJECT Get visible/OBJECT SET VISIBLE	
OBJECT Is styled text	Gibt wahr zurück
OBJECT MOVE	
OBJECT GET SCROLL POSITION/OBJECT SET SCROLL POSITION	
OBJECT GET SUBFORM CONTAINER SIZE	
OBJECT Get name	
OBJECT Get pointer	

Weitere OBJECT Befehle, die hier nicht aufgelistet sind, sind für 4D Write Pro Bereiche nicht verwendbar.

Beispiel

Die horizontale Ausrichtung eines 4D Write Pro Bereichs auf mittig setzen:

```
Case of
  :(Form event=On Clicked)
    OBJECT SET HORIZONTAL ALIGNMENT(*;"myObject";Align_center)
    hAlignLeft:=0 // Eigenschaft für die nächste Anzeige speichern
    hAlignRight:=0
End case
```

Befehle aus dem Kapitel Mehrfachstil Text verwenden

Folgende Befehle unterstützen 4D Write Pro Objekte. Sie können diese in 4D Write Pro mit einem Formularobjektnamen oder einer Dokumentreferenz (eine Variable vom Typ *Objekt*) verwenden:

Befehl	Kommentar
ST COMPUTE EXPRESSIONS	
ST FREEZE EXPRESSIONS	
ST GET ATTRIBUTES/ST SET ATTRIBUTES	
ST Get content type	Für den Content-Typ Bild wurde der neue Typ (6) hinzugefügt
ST Get expression / ST INSERT EXPRESSION	Ab 4D v16 werden Ausdrücke in 4D Write Pro Dokumenten standardmäßig gefiltert und müssen explizit erlaubt sein. Weitere Informationen dazu finden Sie im Abschnitt Filtern von Ausdrücken in einem 4D Write Pro Dokument .
ST GET OPTIONS/ST SET OPTIONS	
ST Get plain text/ST SET PLAIN TEXT	Ab 4D v17 können Sie Plain Text in 4D Write Pro Dokumenten mit den Befehlen WP Get text und WP SET TEXT erhalten/setzen.
ST Get text / ST SET TEXT	
ST GET URL / ST INSERT URL	Ab 4D v16 R4 können Sie editierbare URLs über das Attribut <u>wk link url</u> der Befehle WP SET ATTRIBUTES und WP GET ATTRIBUTES direkt in jedem 4D Write Pro Ausschnitt verwalten (inkl. Bilder und Zellen).

Beispiel

Die Auswahl in einem 4D Write Pro Bereich mit dem Inhalt einer Variablen ersetzen:

```
C_TEXT(ganzerName)

Case of
:(Form event=On Clicked)
  ST INSERT EXPRESSION(myArea;"ganzerName";ST Start highlight;ST End highlight)
End case
```

Ausdrücke für Dokument oder Seite einfügen

Über den Befehl **ST INSERT EXPRESSION** können Sie spezielle Ausdrücke für Dokument- oder Seitenattribute einfügen.

Syntax des Ausdrucks	Verfügbar für	Typ	Beschreibung
\$wp_title	alle Teile im Dokument	Text	Titel definiert im Attribut <u>wk title</u>
\$wp_author	alle Teile im Dokument	Text	Autor definiert im Attribut <u>wk author</u>
\$wp_subject	alle Teile im Dokument	Text	Betreff definiert im Attribut <u>wk subject</u>
\$wp_company	alle Teile im Dokument	Text	Firma definiert im Attribut <u>wk company</u>
\$wp_notes	alle Teile im Dokument	Text	Hinweise definiert im Attribut <u>wk notes</u>
\$wp_dateCreation	alle Teile im Dokument	Date	Erstellungsdatum definiert im Attribut <u>wk date creation</u>
\$wp_dateModified	alle Teile im Dokument	Date	Änderungsdatum definiert im Attribut <u>wk date modified</u>
\$wp_pageNumber	Kopf- und Fußteil - an anderer Stelle Fehler	Lange Ganzzahl	Seitennummer wie vorgegeben: Von Beginn des Dokuments (Standard) oder ab Seitenabschnitt, wenn von Start Seitenabschnitt definiert ist.
\$wp_pageCount	Kopf- und Fußteil - an anderer Stelle Fehler	Lange Ganzzahl	Anzahl Seiten: Gesamtanzahl der Seiten

Achten Sie beim Einfügen eines Ausdrucks darauf, dass der Cursor im passenden Bereich steht (Kopfteil, Fußteil oder Hauptteil des Dokuments) und rufen dann den Befehl **ST INSERT EXPRESSION** auf. Um beispielsweise die Seitennummer im ausgewählten Fußteilbereich einzufügen:

```
ST INSERT EXPRESSION(*;"4DWPArea";"$wp_pageNumber")
```

Hier ein Beispiel für Gestaltung eines Dokuments:

4D Partner

Abschnitt 1 Kopfteil (Erste Seite)
Sehr geehrter Herr Müller,

Wir freuen uns, Sie in 2016 als SILVER-Partner zu begrüßen
Mit Ihrer Entscheidung zur Teilnahme am 4D Partner Programm haben Sie sich die Lizenz zum Erfolg gesichert.

Neben den aktuellen 4D Entwickler-Lizenzen erhalten Sie viele weitere Vorteile zur Unterstützung

Gerne steht Ihnen das Team von 4D Deutschland für Fragen zur Verfügung.

Mit freundlichen Grüßen

Sabine Meierhofer

Abschnitt 1 Fußteil
Seite 1/10

Vertragsbedingungen
28.07.16
Abschnitt 2 Kopfteil

Bildausdrücke einfügen

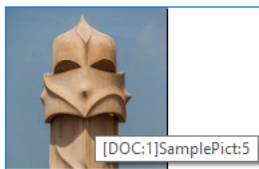
Mit dem Befehl **ST INSERT EXPRESSION** können Sie 4D Ausdrücke einfügen, die Bilder in Ihren 4D Write Pro Bereichen zurückgeben. Das können Variablen, Felder oder 4D Methoden sein (*).

(*) Rufen Sie eine 4D Methode als Ausdruck auf, müssen Sie vorher sicherstellen, dass die Methode explizit erlaubt ist (siehe **Filtern von Ausdrücken in einem 4D Write Pro Dokument**).

Beispiele:

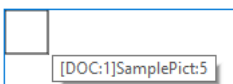
```
//Eine Bildvariable einfügen  
ST INSERT EXPRESSION(*;"WParea";"vPicture";ST_Start_highlight;ST_End_highlight)  
  
//Ein Feld einfügen  
ST INSERT EXPRESSION(*;"WParea";"[DOC]SamplePict";ST_Start_highlight;ST_End_highlight)  
  
//Eine 4D Methode einfügen  
ST INSERT EXPRESSION(*;"WParea";"M_ComputeChart";ST_Start_highlight;ST_End_highlight)
```

Im Bildtipp können Sie die Referenz des Ausdrucks sehen:



Alle Bildattribute lassen sich auf Bildausdrücke anwenden (Attribut [wk_image](#) ist nur lesbar). Da Bilder spezifische Attribute haben, müssen Sie jedoch beachten, dass 4D Write Pro den Ausdruck mindestens einmal bewerten muss, um festzustellen, dass das Ergebnis ein Bild ist und als Bildausdruck behandelt werden soll. Deshalb muss zuerst der Befehl **ST COMPUTE EXPRESSIONS** zum Setzen von Bildattributen aufgerufen werden, und dann der Befehl **ST INSERT EXPRESSION** zum Einfügen eines Bildausdrucks.

Lässt sich das Bild als Ergebnis eines Ausdrucks nicht berechnen oder laden, zeigt 4D Write Pro standardmäßig einen leeren Bildrahmen für undefinierte oder nicht gefundene url:

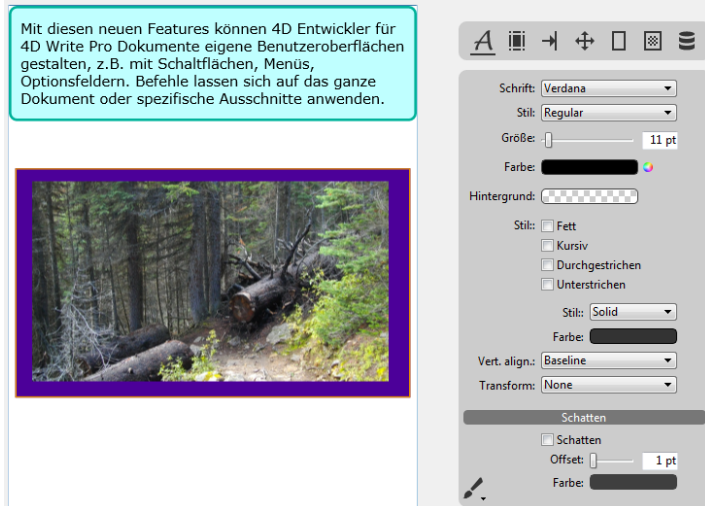


Per Programmierung auf Dokumentinhalte zugreifen

4D Write Pro bietet einen ganzen Satz Befehle, um Attribute für Text und Bilder in Ihren Dokumenten zu verwalten. Damit können 4D Entwickler für 4D Write Pro Dokumente eigene Benutzeroberflächen gestalten, z.B. mit Schaltflächen, Menüs, Optionsfeldern. Befehle lassen sich auf das ganze Dokument oder spezifische Ausschnitte anwenden, die entweder durch eine Benutzerauswahl oder über eigene Werte definiert werden. Verfügbare Eigenschaften sind Einheit des Dokuments, Ränder, Füllen, Hintergrund, Einstellungen für Absatz, Farben, Schriften, Schriftstile und Bildeigenschaften.

Das Objekt der Objektbibliothek **4D Write Pro Area** beispielsweise verwendet einige dieser Funktionalitäten für eine ansprechende Benutzeroberfläche in 4D Write Pro:

Mit diesen neuen Features können 4D Entwickler für 4D Write Pro Dokumente eigene Benutzeroberflächen gestalten, z.B. mit Schaltflächen, Menüs, Optionsfeldern. Befehle lassen sich auf das ganze Dokument oder spezifische Ausschnitte anwenden.



The image shows a screenshot of the 4D Write Pro interface. On the left, there is a rectangular area containing a photograph of a forest with a fallen log. This area is surrounded by a thick purple border. To the right of the image is a text formatting panel. At the top of the panel are several icons for text manipulation. Below the icons, there are dropdown menus for 'Schrift:' (set to 'Verdana') and 'Stil:' (set to 'Regular'). There is a slider for 'Größe:' (set to '11 pt') and a color picker for 'Farbe:'. Below these are options for 'Hintergrund:' (a checkerboard pattern), 'Stil:' (checkboxes for 'Fett', 'Kursiv', 'Durchgestrichen', 'Unterstrichen'), 'Stil:' (set to 'Solid'), 'Farbe:', 'Vert. align.:' (set to 'Baseline'), and 'Transform:' (set to 'None'). At the bottom, there is a 'Schatten' section with a checkbox for 'Schatten', a slider for 'Offset:' (set to '1 pt'), and a color picker for 'Farbe:'.

Befehle zum Verwalten des Ausschnitts

Mit 4D Write Pro können Sie Inhalte des Dokuments per Programmierung auswählen und verwalten. Da der ausgewählte Inhalt Text, Bilder, Tabellen, etc. enthalten kann, sowie nicht-sichtbare Tags für Formatierung, arbeitet 4D Write Pro mit *Ausschnitten*. Ein Ausschnitt ist ein Objekt, das einen Teil in einem 4D Write Pro Dokument darstellt. Er dient zum Festlegen von Elementen, die ausgewählt werden sollen oder zum Setzen von Attributen in einem Teil des Dokuments (über die Funktionen **WP GET ATTRIBUTES** und **WP SET ATTRIBUTES**).

Es gibt verschiedene Typen von Ausschnitten. Sie können den jeweiligen Typ über das Attribut `wk_type` (nur lesbar) bestimmen. Jeder Ausschnitt wird über verschiedene private Attribute definiert:

Konstante	Wert	Kommentar
wk end	end	(Nur lesbares Attribut für Ausschnitt). Ausschnitt Ende Versatz. Wert: Lange Ganzzahl
wk id	id	(Nur lesbares Attribut) ID des Elements. Wert: String
wk owner	owner	(Nur lesbares Attribut) Eigentümer des Ausschnitts/Objekts. Wert: Objekt
wk start	start	(nur lesbares Attribut für Ausschnitt). Ausschnitt Start Versatz. Wert: Lange Ganzzahl (Nur lesbares Attribut für Ausschnitt) Typ des 4D Write Pro Objekts. Mögliche Werte:
wk type	type	<ul style="list-style-type: none">• <code>wk_type_default</code>: Typ Standardausschnitt• <code>wk_type_paragraph</code>: Typ Absatzausschnitt• <code>wk_type_image</code>: Verankertes Bild• <code>wk_type_container</code>: z.B. Kopf- oder Fußteil• <code>wk_type_table</code>: Referenz Tabelle• <code>wk_type_table_row</code>: Referenz Tabellenzeile• <code>wk_type_table_cell</code>: Referenz Tabellenzelle• <code>wk_type_table_column</code>: Referenz Tabellenspalte (nur für Spaltenausschnitt)• <code>wk_type_body</code>: Referenz Hauptteil

Hinweis: Mit Arrays verbundene Ausschnitte (Ausschnitte von Zellen, Spalten und Zeilen) enthalten spezifische Attribute (siehe Abschnitt unten).

Zum Definieren von Ausschnitten gibt es verschiedene Funktionen:

- **WP Create range**(*wpArea* ; *StartAusschnitt* ; *EndeAusschnitt*) -> *objAusschnitt*: Gibt einen neuen Ausschnitt mit den Parametern für Start und Ende zurück.
- **WP Get selection**({*;} *wpArea*) -> *objAusschnitt*: Gibt einen neuen Ausschnitt gemäß der aktuellen Benutzerauswahl zurück.
- **WP Create picture range**(*objAusschnitt*) -> *objAusschnitt*: Gibt einen neuen Ausschnitt zurück, der nur die Bilder enthält.
- **WP Create paragraph range**(*objAusschnitt*) -> *objAusschnitt*: Gibt einen neuen Ausschnitt zurück, der nur die Absätze enthält.

Mit dem Befehl **WP Get position** erhalten Sie Angaben zu Position des Ausschnitts in einem Dokument (Seitennummer, Spaltennummer...).

Befehle für Tabellen

In 4D Write Pro lassen sich Tabellen per Programmierung über folgende Funktionen erstellen und einrichten:

- **WP Insert table** zum Einfügen einer Tabelle und Erhalten ihrer Referenz
- **WP Table append row** zum Hinzufügen von Zeilen in der Tabelle und zum Erhalten des entsprechenden Ausschnitts
- **WP Table get rows** zum Erhalten eines Zeilenausschnitts
- **WP Table get columns** zum Erhalten eines Spaltenausschnitts (*)
- **WP Table get cells** zum Erhalten eines Zellausschnitts

(*) Spalten haben keine Entsprechung in html. In 4D Write Pro ist ein Spaltenausschnitt tatsächlich ein Zellausschnitt.

Ausschnitte von Zeilen, Spalten und Zellen werden jeweils über spezifische private Attribute definiert:

Konstante	Wert	Kommentar
		(verfügbar für Tabellen, Dokumente und Ausschnitte) Anzahl der Spalten. Wert: Lange Ganzzahl
wk column count	columnCount	<ul style="list-style-type: none">• Für eine Tabelle: Attribut nur lesbar• Für ein Dokument oder einen Ausschnitt: Attribut im Lese-/Schreibmodus. Standardwert=1 (einzelne Spalte). Maximumwert=20
wk first column	firstColumn	(Nur lesbares Attribut für Ausschnitt) Nummer der ersten Spalte im Ausschnitt. Wert: Lange Ganzzahl
wk first row	firstRow	(Nur lesbares Attribut für Ausschnitt) Nummer der ersten Zeile im Ausschnitt. Wert: Lange Ganzzahl
wk row count	rowCount	(Nur lesbares Attribut für Ausschnitt) Gesamtanzahl der Zeilen im Ausschnitt. Wert: Lange Ganzzahl
wk table ID	tableID	(Nur lesbares Attribut für Ausschnitt) ID der übergeordneten Tabelle des Ausschnitts. Wert: String

Befehle für Lesezeichen

4D Write Pro ermöglicht, dynamische Referenzen, genannt Lesezeichen, zu Ausschnitten anzulegen und einzusetzen. Sie werden einem spezifischen Ausschnitt in einem 4D Write Pro Dokument zugeordnet.

Lesezeichen sind dynamisch, d.h. wenn der Benutzer Text, der zum Lesezeichen gehört, verschiebt, erweitert und verkleinert, wird der zugeordnete Bereich automatisch aktualisiert und das Lesezeichen bezieht sich weiterhin auf denselben Ausschnitt im Dokument. Hierzu ein Beispiel:

- Sie erstellen ein Lesezeichen mit Namen "MyBM" zum Text "Hallo Welt" auf Seite 20 im Dokument.
- Dann fügen Sie zu Beginn des Dokuments 50 Seiten hinzu
- Über das Lesezeichen "MyBM" können Sie weiterhin automatisch auf den gleichen Text "Hallo Welt" zugreifen, jetzt auf Seite 70 des Dokuments

Ein Dokument kann eine unbegrenzte Zahl an Lesezeichen haben. Im gleichen Ausschnitt sind mehrere Lesezeichen möglich und Lesezeichen Ausschnitte können sich auch überlappen. Lesezeichen werden nicht mit dem Befehl **WP INSERT DOCUMENT** importiert (Lesezeichen im Zieldokument lassen sich nicht überschreiben).

Ein angelegtes Lesezeichen wird im Dokument gespeichert. Lesezeichen werden mit dem Dokument gesichert und lassen sich über verschiedene Befehle verwalten. Sie können auch als Referenzteile einer Vorlage dienen. Diese Teile werden dann automatisch mit Daten aus der Anwendung zusammengefügt, um dynamische Ausgabedokumente, wie z.B. Rechnungen oder Kataloge zu erstellen.

Es gibt verschiedene Befehle zum Verwalten von Lesezeichen:

- **WP CREATE BOOKMARK** erstellt ein neues Lesezeichen von einem Ausschnitt
- **WP GET BOOKMARKS** erhält alle definierten Lesezeichen in einem Dokument
- **WP Get bookmark range** findet einen Ausschnitt über ein vorhandenes Lesezeichen
- **WP DELETE BOOKMARK** löscht ein Lesezeichen

Befehle zum Verwalten von Attributen

Diese Befehle können die verschiedenen Attribute des Dokuments erhalten oder setzen:

- **WP SET ATTRIBUTES**(*objAusschnitt* | *wpDoc* ; *attribName* ; *attribWert* { ; *attribName2* ; *attribWert2* ; ... ; *attribNameN* ; *attribWertN* }): setzt ein oder mehrere Attribut/Wert Paare im Dokument oder Ausschnitt.
- **WP GET ATTRIBUTES**(*objAusschnitt* | *wpDoc* ; *attribName* ; *attribWert* { ; *attribName2* ; *attribWert2* ; ... ; *attribNameN* ; *attribWertN* }): erhält den aktuellen Wert der Attribute im Dokument oder Ausschnitt.
- **WP RESET ATTRIBUTES**(*objAusschnitt* ; *attribName* { ; *attribName2* ; ... ; *attribNameN* }): setzt die Attributwerte im Dokument oder Ausschnitt erneut.

Weitere Informationen dazu finden Sie im Abschnitt **4D Write Pro Attribute**.

Funktion für Schriftstil

Diese Funktion meldet, ob ein bestimmter Stil für den Ausschnitt unterstützt wird:

- **WP Is font style supported**(*objAusschnitt* ; *wpSchriftStil*) -> *wahr* oder *falsch*: Meldet, ob ein Ausschnitt einen bestimmten Stil unterstützt (hilfreich zum Gestalten einer Oberfläche).

URLs verwalten

In 4D Write Pro können Sie jedem Ausschnitt Ihres Dokuments Hyperlinks zuweisen; das gilt für Textausschnitt, Absatz, Bild, Tabellenzeilen oder -zellen gleichermaßen. Sie können z.B. ein Hyperlink in einen Bildbereich setzen; wird das 4D Write Pro Dokument dann in HTML exportiert, können Benutzer auf das Bild klicken, um eine Seite mit einer spezifischen Adresse zu öffnen.

Hinweis: Mit der Kombination **Strg-Taste+Klick** (Windows) oder **Befehlstaste+Klick** (macOS) können Sie Hyperlinks direkt in 4D Write Pro Dokumenten aktivieren.

Hyperlinks werden wie Attribute für Ausschnitte behandelt und über die Konstante `wk link url` der Befehle **WP SET ATTRIBUTES** und **WP RESET ATTRIBUTES** gesetzt oder erhalten. Wollen Sie z.B. den vom Benutzer ausgewählten Text in eine URL umwandeln:

Please refer to the [4D documentation](#)
for more information

schreiben Sie folgenden Code:

```
$range:=WP Get selection(*;"WParea")  
WP SET ATTRIBUTES($range;wk link url;"http://doc.4d.com")
```

Please refer to the [4D documentation](#)
for more information

Hinweis: 4D Write Pro übersetzt nicht evtl. vorhandene spezielle Zeichen der URL. Solche Zeichen müssen Sie bei Bedarf codieren. Ist die URL nicht gültig, wird ein Fehler generiert.

Eingefügte URLs erhalten das standardmäßige Hyperlink Format (blau und unterstrichen), außer es wird bereits ein Textstil auf den Ausschnitt angewendet.

Beim Verwalten von URLs in Ihren Dokumenten wendet 4D Write Pro folgende Regeln an:

- Setzen Sie einen Link auf einen Ausschnitt mit mehreren Absätzen, werden mehrere separate Links definiert.
- Setzen Sie einen Link auf einen Ausschnitt, der bereits Links enthält, gilt folgendes:
 - vorhandene Links, die komplett innerhalb des Ausschnitts liegen, werden entfernt und der neue Link wird auf den Bereich angewandt.
 - Ist ein Link teilweise ausgewählt (er überschneidet den Ausschnitt), wird er gekürzt, so dass der externe Teil den vorigen Link behält und der neue Link auf den Bereich angewandt wird.
- Beim Lesen von Links mit dem Befehl **WP GET ATTRIBUTES** können folgende Fälle auftreten:
 - Der Ausschnitt enthält einen einzelnen Hyperlink: der Befehl gibt den URL String zurück
 - Der Ausschnitt enthält mehrere Hyperlinks: der Befehl gibt den ersten URL String zurück
 - Der Ausschnitt enthält keinen Hyperlink: der Befehl gibt einen leeren String zurück
- Um einen Link aus einem Ausschnitt zu entfernen, schreiben Sie folgenden Code:

```
WP RESET ATTRIBUTES($range;wk link url)
```

oder

```
WP SET ATTRIBUTES($range;wk link url;"")
```

WP Add picture

WP Add picture (wpDok ; Bild) -> Funktionsergebnis

Parameter	Typ	Beschreibung
wpDok	Objekt	→ 4D Write Pro Dokument
Bild	Bild, Text	→ Bild (Feld oder Variable) oder Bildpfad (Text)
Funktionsergebnis	Objekt	↪ Objekt mit Bildreferenz

Beschreibung

Die Funktion **WP Add picture** setzt *Bild* an eine vorgegebene Stelle in *wpDok* und gibt seine Referenz zurück. Diese Referenz lässt sich dann an den Befehl **WP SET ATTRIBUTES** übergeben, um *Bild* an eine andere Stelle in *wpDoc* (z.B. Seite, Ausschnitt, Kopfteil) mit einer definierten Ebene, Größe, etc. zu versetzen.

In *wpDok* übergeben Sie den Namen eines 4D Write Pro Dokuments.

In *Bild* können Sie folgendes übergeben:

- entweder ein 4D Bild
- oder ein String in der Syntax des Systems, der den Pfad zu einer Bilddatei auf der Festplatte angibt. Sie können entweder den vollständigen Pfadnamen angeben oder einen Pfadnamen in Bezug auf die Strukturdatei der Anwendung. Sie können auch einen Dateinamen übergeben. Dann muss die Datei neben der Strukturdatei der Anwendung liegen und eine Endung haben.

Hinweis: Sie können alle von 4D unterstützten Bildformate verwenden (siehe Abschnitt **Überblick über Bilder**). Die Liste der verfügbaren Bildformate erhalten Sie über den Befehl **PICTURE CODEC LIST**. Sind mehrere Formate in das Bild eingebunden (Codecs), behält 4D Write Pro im Dokument nur ein Format zur Anzeige und eins zum Drucken bei (falls sie sich unterscheiden); es werden automatisch die "besten" Formate ausgewählt.

Das hinzugefügte Bild wird standardmäßig:

- hinter dem Text eingefügt
- in die obere linke Ecke des Papierrands gesetzt
- in allen Seiten angezeigt

Position, Ebene (im Textfluss, vor/hinter dem Text), Sichtbarkeit und andere Eigenschaften von *Bild* lassen sich mit dem Befehl **WP SET ATTRIBUTES** oder über Standardaktionen verändern. Weitere Informationen dazu finden Sie im Abschnitt **4D Write Pro Standardaktionen verwenden**.

Hinweis: Die Funktion **WP Get selection** gibt ein Objekt *Bildreferenz* zurück, wenn ein verankertes Bild ausgewählt wurde, ein Objekt *Ausschnitt*, wenn ein Bild im Textfluss ausgewählt ist. Das Attribut *wk type* definiert, ob ein ausgewähltes Objekt ein Bildobjekt (Wert = 2) oder ein Objekt Ausschnitt (Wert = 0) ist.

Beispiel 1

Ein Bild mit Standardeinstellungen über einen Dateipfad hinzufügen.

```
C_OBJECT($obPict)
$obPict:=WP Add picture(myDoc;"C:\\Users\\John.Doe\\Pictures\\Sunrise.jpg")
```

Das Ergebnis ist:



Beispiel 2

Ein angepasstes Bild mittig und mit dem Kopfteil verankert hinzufügen:

```
C_OBJECT($oblImage)
$oblImage:=WP Add picture(myDoc;"C:\\Users\\John.Doe\\Pictures\\Saved Pictures\\Sunrise.jpg")
WP SET ATTRIBUTES($oblImage;wk anchor origin;wk header box)
WP SET ATTRIBUTES($oblImage;wk anchor horizontal align;wk center)
WP SET ATTRIBUTES($oblImage;wk anchor vertical align;wk center)
WP SET ATTRIBUTES($oblImage;wk width;"650px";wk height;"120px")
```

Das Ergebnis ist:



WP CREATE BOOKMARK

WP CREATE BOOKMARK (ZielObj ; bkName)

Parameter	Typ		Beschreibung
ZielObj	Objekt	→	Ausschnitt oder Referenz auf Element
bkName	String	→	Name des zu erstellenden Lesezeichens

Beschreibung

Der Befehl **WP CREATE BOOKMARK** erstellt ein neues Lesezeichen mit Namen *bkName* basierend auf *ZielObj* von 4D Write Pro im Eltern-Dokument des Ausschnitts.

Lesezeichen sind Referenzen mit Namen auf Ausschnitte, über die Sie auf bestimmte Teile des Dokuments zugreifen und diese wiederverwenden können, z.B. für Vorlagen. Weitere Informationen dazu finden Sie im Abschnitt **Befehle für Lesezeichen**.

In *ZielObj* können Sie folgendes übergeben:

- Ausschnitt, oder
- Referenz auf ein Element (Hauptteil, Tabelle, Zeile, Absatz, Bild im Textfluss)

Hinweis: Haben Sie in *ZielObj* eine Referenz auf ein Element übergeben, enthält das Lesezeichen nur das angegebene Element. In *bkName* übergeben Sie den Namen des neuen Lesezeichens. Er muss sich nach den Kriterien für HTML/CSS Namen richten, z.B. darf er nur alphanumerische Zeichen enthalten; ungültige Zeichen wie Leerzeichen werden automatisch entfernt. Lesezeichennamen müssen innerhalb des Dokuments einmalig sein. Ist bereits ein gleichlautendes Lesezeichen vorhanden, wird es überschrieben.

Sie können im gleichen Dokument beliebig viele Lesezeichen erstellen, auch im gleichen Ausschnitt sind mehrere Lesezeichen möglich. Ist ein Lesezeichen erstellt, wird es automatisch im Elterndokument gespeichert und ist so im Dokument selbst gesichert.

Beispiel 1

Ein neues Lesezeichen mit Referenz auf den aktuell ausgewählten Text im Dokument erstellen. Sie schreiben folgenden Code:

```
C_OBJECT($range)
$range:=WP Get selection(*;"WPDocument")
WP CREATE BOOKMARK($range;"my_bookmark")
```

Beispiel 2

Ein vorhandenes Lesezeichen umbenennen. Dazu müssen Sie ein neues Lesezeichen mit derselben Reichweite erstellen und dann das alte löschen:

```
C_TEXT($bookmarkOldName)
C_TEXT($bookmarkNewName)
C_LONGINT($p)
C_OBJECT($wpRange)

$bookmarkOldName:="MyBookmark"
$bookmarkNewName:="MyNewBookmark"

ARRAY TEXT($_bookmarks;0)
WP GET BOOKMARKS(WParea;$_bookmarks)

$p:=Find in array($_bookmarks;$bookmarkOldName)
if($p>0)
    $wpRange:=WP Get bookmark range(WParea;$bookmarkOldName)
    WP DELETE BOOKMARK(WParea;$bookmarkOldName)
    WP CREATE BOOKMARK($wpRange;$bookmarkNewName)
End if
```

⚙ WP Create paragraph range

WP Create paragraph range (ZielObj) -> Funktionsergebnis

Parameter	Typ		Beschreibung
ZielObj	Objekt	→	Ausschnitt oder Referenz auf Element
Funktionsergebnis	Objekt	↩	Ausschnitt betrifft nur Absätze

Kompatibilität

WP Create paragraph range hieß in früheren Versionen von 4D Write Pro **WP Get paragraphs**. Er wurde für mehr Klarheit umbenannt.

Beschreibung

Die Funktion **WP Create paragraph range** gibt einen spezifischen Objektausschnitt zurück, der nur die im Parameter *Zielobj* übergebenen Absätze bewertet. Der zurückgegebene Objektausschnitt kann von **WP GET ATTRIBUTES** und **WP SET ATTRIBUTES** zum Verwalten von Absatzattributen verwendet werden.

In *Zielobj* übergeben Sie:

- einen Ausschnitt oder
- eine Referenz auf ein Element (Tabelle, Zeile, Absatz, Kopf-, Fuß-, Hauptteil)

Beispiel

Füllen nur für die Absätze definieren:

```
$oParagraphs:=WP Create paragraph range($oSelection)  
WP SET ATTRIBUTES($oParagraphs;wk padding;20)
```


⚙️ WP Create picture range

WP Create picture range (Zielobj) -> Funktionsergebnis

Parameter	Typ		Beschreibung
Zielobj	Objekt	→	Ausschnitt oder Referenz auf Element
Funktionsergebnis	Objekt	↩	Objektausschnitt, der nur Bilder enthält

Kompatibilität

WP Create picture range hieß in früheren Versionen von 4D Write Pro **WP Get pictures**. Er wurde für mehr Klarheit umbenannt.

Beschreibung

Die Funktion **WP Create picture range** gibt einen spezifischen Objektausschnitt zurück, der nur die im Parameter *Zielobj* übergebenen Bilder bewertet. Das zurückgegebene Objekt Bildausschnitt kann von den Befehlen **WP GET ATTRIBUTES** und **WP SET ATTRIBUTES** zum Verwalten von Bildattributen verwendet werden.

In *Zielobj* übergeben Sie:

- Ausschnitt oder
- Referenz auf ein Element (Tabelle, Zeile, Absatz, Kopf-, Haupt-, Fußteil)

Beispiel

Nur die Rahmenfarbe von Bildern ändern:

```
$oPicts:=WP Create picture range($oSelection)  
WP SET ATTRIBUTES($oPicts;wk border color;"blau")
```

WP Create range

WP Create range (ZielObj ; StartAusschnitt ; EndeAusschnitt) -> Funktionsergebnis

Parameter	Typ	Beschreibung
ZielObj	Objekt	→ Ausschnitt oder Referenz auf Element oder 4D Write Pro Dokument
StartAusschnitt	Lange Ganzzahl	→ Start neuer Textausschnitt
EndeAusschnitt	Lange Ganzzahl	→ Ende neuer Textausschnitt
Funktionsergebnis	Objekt	→ Objektausschnitt

Geänderter Name

WP Create range hieß in bisherigen Versionen von 4D Write Pro **WP Get range**. Dieser Name wurde für mehr Klarheit geändert.

Beschreibung

Die Funktion **WP Create range** gibt einen neuen Objektausschnitt (*objAusschnitt*) mit der Auswahl zwischen *StartAusschnitt* und *EndeAusschnitt* im 4D Write Pro Bereich *wpArea* zurück.

Ein **4D Write Pro Objektausschnitt** dient zum Verwalten von Attributen in einer Textauswahl (insbesondere über die Befehle **WP GET ATTRIBUTES** und **WP SET ATTRIBUTES**). Weitere Informationen dazu finden Sie im Abschnitt **Befehle zum Verwalten des Ausschnitts**.

In *Zielobj* übergeben Sie:

- Ausschnitt oder
- Referenz auf ein Element (Tabelle, Zeile, Absatz, Bild im Textfluss, Kopf-, Haupt-, Fußteil) oder
- 4D Write Pro Dokument

Ist im Parameter *Zielobj* kein gültiges Objekt übergeben, wird ein leerer *objAusschnitt* zurückgegeben.

In *StartAusschnitt* und *EndeAusschnitt* übergeben Sie Werte für die Position des ersten und letzten Zeichens der Auswahl im Dokument. Sie können wk_start_text in *StartAusschnitt* übergeben, um den Anfang des Dokuments zu definieren und wk_end_text in *EndeAusschnitt*, um das Ende des Dokuments zu definieren. Beachten Sie, dass ein 4D Write Pro Dokument nicht nur sichtbaren Text, sondern auch Tags zur Formatierung im Ausschnitt enthält.

Haben Sie in *Zielobj* eine Referenz auf ein Element Tabelle, Zeile, Absatz, Bild im Textfluss, Kopf-, Haupt- oder Fußteil übergeben, beziehen sich *StartAusschnitt* und *EndeAusschnitt* auf das jeweilige Element. Ist z.B. die Referenz eines Kopfteils übergeben, gibt wk_start_text das startende Zeichen und wk_end_text das endende Zeichen des Kopfteils an. Bei Referenz auf eine Tabelle gibt die Funktion den Textausschnitt in dieser Tabelle zurück.

Hinweis: Haben Sie in *Zielobj* eine Referenz auf ein Element übergeben, bezieht sich der zurückgegebene Ausschnitt auf den Textinhalt von Haupt-, Kopf- oder Fußteil. Der Objektausschnitt behält eine interne Referenz auf den übergeordneten Haupt-, Kopf- oder Fußteil.

Beispiel

Einen Ausschnitt mit 12 Zeichen auswählen, der mit dem Anfang eines 4D Write Pro Feldes startet. Das Feld erscheint in einem Formularobjekt:

Description

The DIALOG command presents the form form to the user. This command is often used to get information from the user through the use of variables, or to present information to the user, such as options for performing an operation. It is common to display the form inside a modal window created with the Open window command.

Der Code lautet:

```
$range2:=WP Get range([SAMPLE]WP;wk_start_text;12)  
WP SELECT(*;"WParea";$range2)
```

Description

The DIALOG command presents the form form to the user. This command is often used to get information from the user through the use of variables, or to present information to the user, such as options for performing an operation. It is common to display the form inside a modal window created with the Open window command.

⚙ WP Create table range

WP Create table range (ZielObj) -> Funktionsergebnis

Parameter	Typ	Beschreibung
ZielObj	Objekt	→ Ausschnitt oder Referenz auf Element oder 4D Write Pro Dokument
Funktionsergebnis	Objekt	↩ Objektausschnitt, der nur Tabellen enthält

Beschreibung

Die Funktion **WP Create table range** gibt einen spezifischen Objektausschnitt zurück, der nur die im Parameter *ZielObj* übergebenen Tabellen bewertet. Sie gibt einen Ausschnitt von wk type table an, selbst wenn der übergebene Parameter *ZielObj* keine Tabelle enthält. Das zurückgegebene Objekt Tabellenausschnitt kann von den Befehlen **WP GET ATTRIBUTES** und **WP SET ATTRIBUTES** zum Verwalten von Attributen für Tabellen und Tabelleninhalt verwendet werden.

In *ZielObj* übergeben Sie:

- Ausschnitt oder
- Referenz auf ein Element (Kopf-, Haupt-, Fußteil, Absatz) oder
- 4D Write Pro Dokument

Beispiel

Die Darstellung der Tabellen ändern, die im Tabellenausschnitt enthalten sind:

```
tableRange:=WP Create table range(myDoc)
//Tabellenattribute
WP SET ATTRIBUTES(tableRange;wk border style;wk solid;wk border color;"Blue";wk border width;"3px")
WP SET ATTRIBUTES(tableRange;wk table align;wk center)
//Textattribute
WP SET ATTRIBUTES(tableRange;wk font size;12)
WP SET ATTRIBUTES(tableRange;wk text transform;wk capitalize)
WP SET ATTRIBUTES(tableRange;wk font bold;wk true)
```

WP DELETE BOOKMARK

WP DELETE BOOKMARK (wpDok ; bkName)

Parameter	Typ		Beschreibung
wpDok	Objekt	⇒	4D Write Pro Dokument
bkName	String	⇒	Lesezeichenname zum Löschen

Beschreibung

Der Befehl **WP DELETE BOOKMARK** entfernt das Lesezeichen mit Namen *bkName* aus *wpDok*.
Existiert kein Lesezeichen *bkName* in *wpDok*, führt der Befehl nichts aus.

Beispiel

Ein vorhandenes Lesezeichen umbenennen. Dazu müssen Sie ein neues Lesezeichen mit derselben Reichweite erstellen und dann das alte löschen:

```
C_TEXT($bookmarkOldName)
C_TEXT($bookmarkNewName)
C_LONGINT($p)
C_OBJECT($wpRange)

$bookmarkOldName:="MyBookmark"
$bookmarkNewName:="MyNewBookmark"

ARRAY TEXT($_bookmarks;0)
WP GET BOOKMARKS(WParea;$_bookmarks)

$p:=Find in array($_bookmarks;$bookmarkOldName)
If($p>0)
    $wpRange:=WP Get bookmark range(WParea;$bookmarkOldName)
    WP DELETE BOOKMARK(WParea;$bookmarkOldName)
    WP CREATE BOOKMARK($wpRange;$bookmarkNewName)
End if
```

WP EXPORT DOCUMENT

WP EXPORT DOCUMENT (wpDoc ; DateiPfad {; Format {; Option}})

Parameter	Typ		Beschreibung
wpDoc	Objekt	→	Objekt 4D Write Pro
DateiPfad	String	→	Pfad der exportierten Datei
Format	Lange Ganzzahl	→	Ausgabeformat des Dokuments
Option	Lange Ganzzahl	→	Exportoptionen

Beschreibung

Der Befehl **WP EXPORT DOCUMENT** exportiert das Objekt 4D Write Pro *wpDoc* mit dem Parameter *DateiPfad* sowie optionalen Parametern in ein Dokument auf der Festplatte.

In *wpDoc* übergeben Sie das Objekt 4D Write Pro zum Exportieren.

In *DateiPfad* übergeben Sie den Zielpfad und den Namen des zu exportierenden Dokuments. Übergeben Sie nur den Dokumentnamen, wird es auf derselben Ebene wie die 4D Strukturdatei gesichert. Standardmäßig, d.h. ohne den Parameter *Format*, wählt der Befehl das Dateiformat anhand der Endung des Dokuments aus.

Im Parameter *Format* können Sie auch eine Konstante unter dem Thema **4D Write Pro Konstanten** übergeben. Dann fügt 4D bei Bedarf die passende Endung an die Datei an. Folgende Formate werden unterstützt:

Konstante	Typ	Wert	Kommentar
wk 4wp	Lange Ganzzahl	4	Das 4D Write Pro Dokument wird in einem nativen Archivformat gesichert (gezipptes HTML und Bilder in einem eigenen Ordner abgelegt). 4D spezifische Tags sind enthalten, 4D Ausdrücke werden nicht berechnet. Dieses Format ist besonders geeignet, um 4D Write Pro Dokumente ohne Verluste auf der Festplatte zu sichern und archivieren.
wk web page complete	Lange Ganzzahl	2	Endung .htm oder .html. Das Dokument wird als Standard HTML gesichert, seine Ressourcen werden getrennt gesichert. 4D Tags werden entfernt und Ausdrücke berechnet. Dieses Format ist besonders geeignet, um ein 4D Write Pro Dokument in einem Web Browser anzuzeigen.

Hinweise:

- "4D spezifische tags" bedeutet 4D XHTML mit einem 4D Namensbereich und 4D CSS Stilarten
- Ausdrücke lassen sich jederzeit vor dem Export mit dem Befehl **ST FREEZE EXPRESSIONS** einfrieren
- Weitere Informationen zum 4D Write Pro Format finden Sie im Abschnitt **Dokumentformat .4wp**

Im Parameter *Optionen* übergeben Sie Optionen zum Konfigurieren des Exports. Sie können einen Wert *Lange Ganzzahl* übergeben, um den Stil des HTML Codes zu definieren. Es gibt folgende Konstanten:

Konstante	Typ	Wert	Kommentar
wk html debug	Lange Ganzzahl	1	Formatierter HTML Code ("pretty print"), leichter zu debuggen
wk normal	Lange Ganzzahl	0	Standard HTML Code

- *HTML debug option aus (Standard):*

```
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<title>New 4D Write Pro Document</title>
<style type="text/css">
body { background-color:#FFFFFF }
ul, ol { margin:0;padding:0 }
</style>
</head>
```

- *HTML debug option an:*

```
<html xmlns="http://www.w3.org/1999/xhtml"><head><title>New 4D Write Pro Document</title><style
type="text/css">body { background-color:#FFFFFF }ul, ol { margin:0;padding:0 }p,li { white-space:pre-
wrap;margin:0pt;padding:0pt;font-family:Times New Roman }p.Normal,li.Normal { text-align:left }p._p1,li._p1
{ font-family:Arial;font-size:18pt;color:#1D1D1D }img._img1 { width:51pt;height:51pt }</style></head>
<body><p class="Normal_p1"><span>23/12/14</span></p><p class="Normal_p1">Dear <span
style="color:#BE0E00">Shaun Stoltz</span><span style="color:#800062"> <span style="color:#000000">from
```

Beispiel 1

Den Inhalt des 4D Write Pro Objekts *myArea* in ein Dokument in Ihrem Datenbankordner exportieren. Sie können die Option zum Debuggen mit einer Schaltfläche 'pprint' setzen:

```
C_TEXT($filePath)
$filePath:=Get 4D folder(Database folder)+"Exportierte Dateien"+Folder separator+"WriteProExport.html"
If(pprint=0) //Ist die die Option debug aus
WP EXPORT DOCUMENT(myArea;$filePath;wk web page complete;wk normal)
Else
WP EXPORT DOCUMENT(myArea;$filePath;wk web page complete;wk html debug)
End if
```

Beispiel 2

Den Inhalt des 4D Write Pro Objekts *myArea* im .4wp Format exportieren:

```
C_TEXT($path)
```

```
C_LONGINT($docRef)
```

```
Case of
```

```
:(Form event=On Clicked)
```

```
  $path:=Get 4D folder(Database folder)+"Export"+Folder separator
```

```
  $path:=Select document($path;".4wp";" title";File name entry)
```

```
  If($path#command_5")
```

```
    WP EXPORT DOCUMENT(myArea;document;wk 4wp;wk normal)
```

```
  Else
```

```
    ALERT("Es ist ein Fehler aufgetreten.")
```

```
  End if
```

```
End case
```

WP EXPORT VARIABLE

WP EXPORT VARIABLE (wpDoc ; Ziel ; Format {; Option})

Parameter	Typ		Beschreibung
wpDoc	Objekt	→	4D Write Pro Variable
Ziel	Textvariable	←	Variable für den exportierten Inhalt
Format	Lange Ganzzahl	→	Ausgabeformat der Variablen
Option	Lange Ganzzahl	→	Exportoptionen

Beschreibung

Der Befehl **WP EXPORT VARIABLE** exportiert das 4D Write Pro Objekt *wpDoc* in die 4D Variable *Ziel* im angegebenen *Format*. In *wpDoc* übergeben Sie das Objekt 4D Write Pro für den Export.

In *Ziel* übergeben Sie die Variable, die Sie mit dem exportierten Objekt 4D Write Pro füllen wollen. Der Variablentyp richtet sich nach dem Exportformat, das im Parameter *Format* angegeben ist:

- Übergeben Sie das native .4wp Format, ist die Variable vom Typ Blob
- Übergeben Sie ein HTML Format, ist die Variable vom Typ Text

Im Parameter *Format* übergeben Sie eine Konstante unter dem Thema **4D Write Pro Konstanten**, um das gewünschte Exportformat zu setzen. Jedes Format dient für eine spezifische Verwendung. Folgende Formate werden unterstützt:

Konstante	Typ	Wert	Kommentar
wk 4wp	Lange Ganzzahl	4	Das 4D Write Pro Dokument wird in einem nativen Archivformat gesichert (gezipptes HTML und Bilder in einem eigenen Ordner abgelegt). 4D spezifische Tags sind enthalten, 4D Ausdrücke werden nicht berechnet. Dieses Format ist besonders geeignet, um 4D Write Pro Dokumente ohne Verluste auf der Festplatte zu sichern und archivieren.
wk mime html	Lange Ganzzahl	1	Das 4D Write Pro Dokument wird als standard MIME HTML mit html Dokumenten und Bildern, eingebunden als MIME Teile (codiert in base64), gesichert. Ausdrücke werden berechnet und 4D spezifische Tags werden entfernt. Dieses Format eignet sich besonders, um E-Mails in HTML mit dem Befehl SMTP_QuickSend zu senden.
wk web page html 4D	Lange Ganzzahl	3	Das 4D Write Pro Dokument wird als HTML mit spezifischen 4D Tags gesichert; jeder Ausdruck wird als nicht-umgebrochener Bereich eingefügt. Da dieses Format verlustlos ist, eignet es sich zum Speichern in einem Textfeld.

Hinweise:

- "4D spezifische Tags" bedeutet 4D XHTML mit einem 4D Namensbereich und 4D CSS Stilarten
- Ausdrücke lassen sich mit dem Befehl **ST FREEZE EXPRESSIONS** jederzeit vor dem Export einfrieren
- Weitere Informationen zum Format des 4D Write Pro Dokuments finden Sie im Abschnitt **Dokumentformat .4wp**

Im Parameter *Option* übergeben Sie Optionen zum Konfigurieren des Exports. Sie haben folgende Möglichkeiten:

- Einen Wert *Lange Ganzzahl* zum Definieren der Art des HTML Code; es gibt folgende Konstanten:

Konstante	Typ	Wert	Kommentar
wk html debug	Lange Ganzzahl	1	Formatierter HTML Code ("pretty print"), leichter zu debuggen
wk normal	Lange Ganzzahl	0	Standard HTML Code

HTML debug option aus (Standard):

```
<html xmlns="http://www.w3.org/1999/xhtml"><head><title>New 4D Write Pro Document</title><style type="text/css">body { background-color:#FFFFFF }ul, ol { margin:0;padding:0 }p,li { white-space:pre-wrap;margin:0pt;padding:0pt;font-family:'Times New Roman' }p.Normal,li.Normal { text-align:left }p._p1,li._p1 { font-family:'Arial';font-size:18pt;color:#1D1D1D }img_img1 { width:51pt;height:51pt }</style></head><body><p class="Normal_p1"><span>23/12/14</span></p><p class="Normal_p1">Dear <span style="color:#BE0E00">Shaun Stoltz</span><span style="color:#800062"> <span style="color:#000000">from
```

HTML debug option an:

```
<html xmlns="http://www.w3.org/1999/xhtml"><head><title>New 4D Write Pro Document</title><style type="text/css">body { background-color:#FFFFFF }ul, ol { margin:0;padding:0 }
```

- oder ein *String*. Es wird folgende Eigenschaft unterstützt (nur bei Verwenden des Formats [wp_mime html](#)):
 - "CID host domain name: Host Domain, das mit '@' als Trenner an generierte CID URLs angefügt wird. Übergeben Sie zum Beispiel gmail.com, wird '123@gmail.com' angefügt, wenn die einmalige ID von CID 123 ist. Dies wird standardmäßig weggelassen, d.h. nur die einmalige ID von CID erscheint (wird von den meisten Mail Servern akzeptiert).

Beispiel

Ein E-Mail mit formatiertem Text, 4D Referenzen und Bildern senden. Dazu verwenden Sie einen 4D Write Pro Bereich, exportiert im Format MIME HTML, und senden über einen Befehl der 4D Internet Commands:

```
C_LONGINT($smtpid_l;$err_l;$smtpOption_l;$smtpPort_l)
C_TEXT($str;$emailBody_t;$smtpHost_t;$emailTo_t;$emailFrom_t;$smtpPass_t)

//Bereich in passendem Format exportieren
WP_EXPORT_VARIABLE(myWPArea;$str;wk mime html)
$emailTo_t:="johnsmith@4d.com"
$emailFrom_t:"testWritePro@gmail.com"
$emailBody_t:=$str
$smtpHost_t:"smtp.gmail.com"
$smtpOption_l:=9
$smtpPort_l:=465
$smtpPass_t:"QRN_on_bretzelburg"

$err_l:=SMTP_QuickSend($smtpHost_t;$emailFrom_t;$emailTo_t;$emailTitle_t;\
$emailBody_t;$smtpOption_l;$smtpPort_l;$smtpUser_t;$smtpPass_t)

if(($err_l=0))
    ALERT("Email gesendet an "+emailTo_t)
Else
    ALERT("Fehler in Parametern, bitte erneut versuchen.")
End if
```


⚙ WP GET ATTRIBUTES

WP GET ATTRIBUTES (ZielObj ; attribName ; attribWert { ; attribName2 ; attribWert2 ; ... ; attribNameN ; attribWertN })

Parameter	Typ	Beschreibung
ZielObj	Objekt	➔ Ausschnitt, Referenz auf Element oder 4D Write Pro Dokument
attribName	Text	➔ Name des zu erhaltenden Attributs
attribWert	Text, Zahl, Array, Lange, Ganzzahl, Array, Zahl, Bild, Datum	➔ Aktueller Wert des Attributs für das Zielobjekt

Beschreibung

Der Befehl **WP GET ATTRIBUTES** gibt den Wert jedes Attributs in einem 4D Write Pro Ausschnitt, Kopf-, Haupt-, Fußteil, Tabellenreferenz oder einem Dokument zurück. Damit können Sie auf jede Art von internen Attributen in 4D Write Pro zugreifen: Zeichen, Absatz, Dokument, Tabelle oder Bild.

In *ZielObj* übergeben Sie:

- Ausschnitt oder
- Referenz auf ein Element (Tabelle, Absatz, verankertes Bild, Bild im Textfluss, Kopf-, Haupt-, Fußteil) oder
- Ein 4D Write Pro Dokument

Die Liste der Attribute für *attribName* mit Reichweite und Werten finden Sie im Abschnitt **4D Write Pro Attribute**.

Gibt es verschiedene Werte für das gleiche Attribut im als Parameter übergebenen Element, gibt der Befehl jeweils folgendes zurück:

- Bei numerischen Werten wk mixed
- Bei einem Array, ein leeres Array (TabStops, Farbe wenn *attribWert* als Array definiert ist), mit Ausnahme von wk text shadow offset. Hierfür enthält der Wert des Array immer 2 Eingaben, die getrennt in wk mixed gesetzt sind, wenn entweder horizontaler Versatz oder vertikaler Versatz (oder beide) gemischt sind.
- Bei String-Werten einen leeren String
- Bei Bildwerten ein leeres Bild

Beispiel

Die Hintergrundfarbe für den gewählten Bereich erhalten:

```
$range:=WP Get selection(*;"WParea")
WP GET ATTRIBUTES($range;wk background color;$bcol)
```

⚙ WP Get body

WP Get body (wpDok) -> Funktionsergebnis

Parameter	Typ		Beschreibung
wpDok	Objekt	→	4D Write Pro Dokument
Funktionsergebnis	Objekt	↩	Referenz auf 4D Write Pro Hauptteil

Beschreibung

Die Funktion **WP Get body** gibt eine Referenz auf den Hauptteil des 4D Write Pro Dokuments *wpDok* zurück.
Diese Referenz übergeben Sie in:

- **WP SELECT**, um den Hauptteil des Dokuments auszuwählen
- **WP Create range**, um den Hauptteil als einen Ausschnitt zu definieren
- **WP SET ATTRIBUTES** und **WP GET ATTRIBUTES**, um die Attribute des Hauptteils zu verwalten

Beispiel

Den Hauptteil eines 4D Write Pro Dokuments in ein anderes 4D Write Pro Dokument kopieren:

```
C_OBJECT($bodySource;$rangeSource;$tempoc;$bodyTarget;$rangeTarget)
```

```
$bodySource:=WP Get body([TEMPLATES]WPtemplate)
```

```
$rangeSource:=WP Create range($bodySource;wk start text;wk end text)
```

```
$tempoc:=WP New($rangeSource)
```

```
$bodyTarget:=WP Get body([Docs]MyWPDoc)
```

```
$rangeTarget:=WP Create range($bodyTarget;wk start text;wk end text)
```

```
WP INSERT DOCUMENT($rangeTarget;$tempoc;wk replace)
```

⚙ WP Get bookmark range

WP Get bookmark range (wpDok ; bkName) -> Funktionsergebnis

Parameter	Typ	Beschreibung
wpDok	Objekt	→ 4D Write Pro Dokument
bkName	Text	→ Name des Lesezeichens, dessen Ausschnitt Sie erhalten wollen
Funktionsergebnis	Objekt	↩ Ausschnitt des Lesezeichens

Beschreibung

Die Funktion **WP Get bookmark range** gibt ein Objekt Textbereich (*AusschnittObj*) mit dem Ausschnitt des Lesezeichens zurück, angegeben im Parameter *bkName* in *wpDok*.

Existiert das Lesezeichen *bkName* nicht in *wpDok*, wird ein leeres Objekt *AusschnittObj* zurückgegeben.

Beispiel

Den Bereich von "MyBookmark" in Ihrem Dokument anzeigen:

```
C_OBJECT($wpRange)
$wpRange:=WP Get bookmark range(WParea;"MyBookmark")
WP SELECT(WParea;$wpRange)
```

WP GET BOOKMARKS

WP GET BOOKMARKS (wpDok ; arrBK Namen)

Parameter	Typ		Beschreibung
wpDok	Objekt	→	4D Write Pro Dokument
arrBK Namen	Array Text	←	Array mit Namen der Lesezeichen

Beschreibung

Der Befehl **WP GET BOOKMARKS** gibt ein Array mit den Namen aller Lesezeichen an, definiert in *wpDok*.

Nach Ausführen des Befehls wird das Array *arrBK Namen* mit allen Lesezeichennamen im Dokument gefüllt. Im Array werden die Namen nach der Position des Lesezeichens innerhalb des Dokuments sortiert. Gibt es mehrere Lesezeichen an der gleichen Position, werden sie in alphabetischer Reihenfolge sortiert.

Beispiel

Sie wollen die Anzahl der im Dokument definierten Lesezeichen wissen:

```
ARRAY TEXT($_bookmarks;0)
WP GET BOOKMARKS(WParea;$_bookmarks)
ALERT("Das Dokument enthält "+Size of array($_bookmarks)+" Lesezeichen.")
```

⚙ WP Get element by ID

WP Get element by ID (wpDok ; ID) -> Funktionsergebnis

Parameter	Typ		Beschreibung
wpDok	Objekt	→	4D Write Pro Dokument
ID	String	→	Kennung des gesuchten Elements
Funktionsergebnis	Objekt	↩	Referenz des 4D Write Pro Elements

Beschreibung

Die Funktion **WP Get element by ID** gibt ein Objekt mit der Referenz auf das im Parameter *ID* angegebene Element von *wpDok* zurück.

Im Parameter *wpDok* übergeben Sie eine Referenz auf das 4D Write Pro Dokument.

Im Parameter *ID* übergeben Sie das ID Attribut des gesuchten Elements. Es kann innerhalb des Dokuments liegen, wie Bild, Absatz, Tabelle, Zeile, etc., oder das Dokument selbst sein. Hat kein Element im Dokument die angegebene Kennung, wird ein Nullwert zurückgegeben.

Beispiel 1

Referenz auf ein Bild mit der Kennung "img1" erhalten:

```
C_OBJECT(oblImage)
oblImage:=WP Get element by ID(myDoc;"img1")
```

Beispiel 2

Ein Tabellenelement finden und auf einen breiteren Rand mit der Farbe Purpur ändern:

```
C_OBJECT($element)
$element:=WP Get element by ID(myDoc;"Table1")
WP SET ATTRIBUTES($element;wk border style;wk solid)
WP SET ATTRIBUTES($element;wk border width;"4px")
WP SET ATTRIBUTES($element;wk border color;"purple")
```

Beispiel 3

Ein Element wiederfinden und seine Kennung ändern:

```
$colTable:=WP Create table range([INFO]Sample)
$elements:=WP Get elements($colTable)
$elem:=WP Get element by ID([INFO]Sample;"Paris")
If($elem#command_5Null)
  $elem.ID:="Lyon"
End if
```

WP Get elements

WP Get elements (ZielObj {; ElementTyp}) -> Funktionsergebnis

Parameter	Typ	Beschreibung
ZielObj	Objekt	→ Ausschnitt, Referenz auf Element oder 4D Write Pro Dokument
ElementTyp	Lange Ganzzahl	→ Gewünschter Typ des gesuchten Elements
Funktionsergebnis	Collection	↪ Collection mit Referenzen auf Elemente

Beschreibung

Die Funktion **WP Get elements** gibt eine Collection von Objekten mit Referenzen auf Absätze, Bilder, Tabellen bzw. Zeilen zurück.

Im Parameter *ZielObj* übergeben Sie:

- Ausschnitt oder
- Referenz auf ein Element (Tabelle, Zeile, Absatz, Kopf-, Haupt-, Fußteil) oder
- 4D Write Pro Dokument

Wenn ein typisierter Bereich oder eine Referenz übergeben wird, gibt die Funktion eine Collection zurück, die nur Elemente des entsprechenden Typs enthält (es sei denn, Sie geben mit dem Parameter *ElementTyp* einen Typ an, siehe unten). Sonst gibt sie eine Collection mit allen in *ZielObj* verfügbaren Elementen an.

Optional können Sie im Parameter *ElementTyp* die Elementtypen angeben, die in der Collection zurückgegeben werden sollen. Es gibt folgende Typen:

Konstante	Wert	Kommentar
wk type default	0	Standardtyp Ausschnitt (Wert für <u>wk type</u>)
wk type image	2	Verankertes Bild (Wert für <u>wk type</u>)
wk type image anchored	200	Referenz verankertes Bild für Dokument Objekte
wk type image inline	100	Referenz Bild im Textfluss für Dokument Objekte
wk type paragraph	1	Ausschnitttyp Absatz (Wert für <u>wk type</u>)
wk type table	4	Referenz Tabelle (Wert für <u>wk type</u>)
wk type table row	5	Referenz Tabellenzeile (Wert für <u>wk type</u>)

Hinweise:

- Mit dem Selektor wk type default gibt die Funktion eine ungefilterte Collection zurück.
- Mit den Selektoren wk type paragraph, wk type image oder wk type table enthält die zurückgegebene Collection nur Elemente des angegebenen Typs, selbst wenn er sich vom Typ unterscheidet, der im Parameter *ElementTyp* angegeben ist.
- Die Elemente in der zurückgegebenen Collection werden in derselben Reihenfolge wie in *ZielObj* sortiert, außer wenn eine Referenz auf ein 4D Write Pro Dokument übergeben ist. Dann werden die Elemente nach ihrer Kennung sortiert.
- Ist ein Element im Ausschnitt, übergeben in *ZielObj*, nur teilweise enthalten, ist das ganze Element im Ergebnis übernommen. Enthält der Ausschnitt beispielsweise einen Teil und nicht die komplette Tabelle, wird die Referenz auf die komplette Tabelle zurückgegeben, jedoch sind nur die Zellenreferenzen (Absatz) innerhalb des Ausschnitts enthalten.

Beispiel 1

Eine Collection mit allen Absätzen in einem Dokument zurückgeben:

```
C_COLLECTION($docElements)
$docElements:=WP Get elements($myDoc;wk type paragraph)
//gibt eine Collection zurück, die nur die Elemente vom Typ Absatz enthält
```

Beispiel 2

Ausrichtung und Farbe der Absätze in einem Dokument abwechselnd verwenden:

```
C_COLLECTION($col)
C_OBJECT($obj)
C_OBJECT($body)
C_LONGINT($i;$n)
$col:=New collection
$body:=WP Get body(myDoc)
$col:=WP Get elements($body;wk type paragraph)
$i:=0
$n:=$col.length
For($i;0;$n-1)
  $obj:=$col[$i]
  WP RESET ATTRIBUTES($obj)
  WP SET ATTRIBUTES($obj;wk font bold;wk false)
```

WP SET ATTRIBUTES(\$obj;wk font italic;wk false)

If(\$i%2=0)

WP SET ATTRIBUTES(\$obj;wk text align;wk left)

WP SET ATTRIBUTES(\$obj;wk font bold;wk true)

WP SET ATTRIBUTES(\$obj;wk text color;"#command_5804040")

WP SET ATTRIBUTES(\$obj;wk margin right;"5cm")

WP SET ATTRIBUTES(\$obj;wk margin left;"1cm")

Else

WP SET ATTRIBUTES(\$obj;wk text align;wk right)

WP SET ATTRIBUTES(\$obj;wk font italic;wk true)

WP SET ATTRIBUTES(\$obj;wk text color;"#command_5404020")

WP SET ATTRIBUTES(\$obj;wk margin left;"5cm")

WP SET ATTRIBUTES(\$obj;wk margin right;"1cm")

End if

End for

WP Get footer

WP Get footer (AusschnittObj) | (wpDok ; AbschnittIndex {; UnterAbschnittTyp}) -> Funktionsergebnis

Parameter	Typ	Beschreibung
AusschnittObj	Objekt	→ 4D Write Pro Ausschnitt
wpDok	Objekt	→ 4D Write Pro Dokument
AbschnittIndex	Lange Ganzzahl	→ Index des Abschnitts
UnterAbschnittTyp	Lange Ganzzahl	→ Typ des Unterabschnitts (wk first page, wk left page oder wk right page)
Funktionsergebnis	Objekt	→ 4D Write Pro Fußteilverferenz

Beschreibung

Die Funktion **WP Get footer** gibt eine Referenz auf den Fußteil zurück, der dem Ausschnitt *AusschnittObj* zugewiesen oder in *AbschnittIndex* des Dokuments *wpDok* angegeben ist.

- Mit der ersten Syntax (Parameter *AusschnittObj*) gibt die Funktion den Fußteil der ersten Seite im Ausschnitt zurück. Ist auf der Seite ein Unterabschnitt aktiv, wird der Fußteil des aktiven Unterabschnitts zurückgegeben, so dass immer der sichtbare Fußteil zurückgegeben wird.
- Mit der zweiten Syntax (Parameter *wpDok*) gibt die Funktion den in *AbschnittIndex* angegebenen Fußteil zurück (Wert von *AbschnittIndex* muss ≥ 1 sein).
Ist der Parameter *UnterAbschnittTyp* übergeben, gibt die Funktion den Fußteil des hier angegebenen Typs zurück. Sie können eine der folgenden Konstanten übergeben:

Konstante	Typ	Wert
wk first page	Lange Ganzzahl	1
wk left page	Lange Ganzzahl	2
wk right page	Lange Ganzzahl	3

Ist *UnterAbschnittTyp* nicht angegeben, wird der Fußteil des Hauptabschnitts zurückgegeben. Ist in diesem Fall ein Unterabschnitt aktiv, ist der Fußteil des Hauptabschnitts *undefiniert*, d.h. der Fußteil des Hauptabschnitts ist nur verfügbar, wenn es keinen aktiven Unterabschnitt wk first page oder wk left page gibt.

Beispiel

Den Fußteil der ersten Seite eines 4D Write Pro Dokuments in die Fußteile aller Seiten eines anderen 4D Write Pro Dokuments kopieren:

```
C_OBJECT($footerSource,$rangeSource;$tempoc;$footerTarget;$rangeTarget)
```

```
$footerSource:=WP Get footer([TEMPLATES]WPtemplate;1;wk first page) //den Fußteil von Abschnitt 1 auf der ersten Seite erhalten
```

```
$rangeSource:=WP Create range($footerSource;wk start text;wk end text)
```

```
$tempoc:=WP New($rangeSource)
```

```
$footerTarget:=WP Get footer([Docs]MyWPDoc;1) //alle Seiten von Abschnitt 1
```

```
$rangeTarget:=WP Create range($footerTarget;wk start text;wk end text)
```

```
WP INSERT DOCUMENT($rangeTarget;$tempoc;wk replace)
```


WP Get frame

WP Get frame ({ * ; } wpArea) -> Funktionsergebnis

Parameter	Typ	Beschreibung
*	Operator	➔ Mit Stern: wpArea ist Name des Formularobjekts (String) ➔ Ohne Stern: wpArea ist ein Objektfeld oder Variable
wpArea	String	➔ Name des Formularobjekts (mit *) oder 4D Write Pro Objektvariable oder Feld (ohne *)
Funktionsergebnis	Lange Ganzzahl	➔ Rahmen, wo der Cursor aktuell steht

Beschreibung

Die Funktion **WP Get frame** gibt den Rahmen innerhalb des Bereichs 4D Write Pro *wpArea* zurück, wo der Cursor aktuell steht. Mit dem optionalen Parameter * geben Sie an, dass *wpArea* ein Formularobjekt Name (String) ist. Ohne diesen Parameter geben Sie an, dass *wpArea* eine 4D Write Pro Objektvariable oder Feld ist.

Der zurückgegebene Wert lässt sich mit einer der folgenden Konstanten unter dem Thema **4D Write Pro Konstanten** vergleichen:

Konstante	Typ	Wert	Kommentar
wk body	Lange Ganzzahl	0	Hauptteilrahmen des Bereichs
wk current section default footer	Lange Ganzzahl	10	Hauptfußteil des Bereichs (nur verfügbar, wenn kein Fußteil für Unterbereich aktiv ist)
wk current section default header	Lange Ganzzahl	9	Hauptkopfteil des Bereichs (nur verfügbar, wenn kein Kopfteil für Unterbereich aktiv ist)
wk current section first footer	Lange Ganzzahl	4	Fußteil der ersten Seite des Bereichs
wk current section first header	Lange Ganzzahl	3	Kopfteil der ersten Seite des Bereichs
wk current section left footer	Lange Ganzzahl	6	Fußteil der linken Seite(n) des Bereichs
wk current section left header	Lange Ganzzahl	5	Kopfteil der linken Seite(n) des Bereichs
wk current section right footer	Lange Ganzzahl	8	Fußteil der rechten Seite(n) des Bereichs
wk current section right header	Lange Ganzzahl	7	Kopfteil der rechten Seite(n) des Bereichs

Beispiel

Sie wollen sicherstellen, dass der Benutzer den Cursor in einen Kopfteil- bzw. Fußteilrahmen gesetzt hat:

```
C_LONGINT($frameGet)
$frameGet:=WP Get frame(*;"WParea")
If($frameGet=wk body)
  ALERT("Bitte wählen Sie ein Fuß- oder Kopfteil aus.")
End if
```

WP Get header

WP Get header (AusschnittObj) | (wpDok ; AusschnittIndex {; UnterAbschnittTyp}) -> Funktionsergebnis

Parameter	Typ	Beschreibung
AusschnittObj	Objekt	→ 4D Write Pro Ausschnitt
wpDok	Objekt	→ 4D Write Dokument
AusschnittIndex	Lange Ganzzahl	→ Index des Ausschnitts
UnterAbschnittTyp	Lange Ganzzahl	→ Typ des Unterabschnitts (wk first page, wk left page oder wk right page)
Funktionsergebnis	Objekt	→ 4D Write Pro Referenz auf Kopfteil

Beschreibung

Die Funktion **WP Get header** gibt eine Referenz auf den Kopfteil zurück, der dem Ausschnitt *AusschnittObj* zugewiesen oder in *AbschnittIndex* des Dokuments *wpDok* angegeben ist.

- Mit der ersten Syntax (Parameter *AusschnittObj*) gibt die Funktion den Kopfteil der ersten Seite im Ausschnitt zurück. Ist auf der Seite ein Unterabschnitt aktiv, wird der Kopfteil des aktiven Unterabschnitts zurückgegeben, so dass immer der sichtbare Kopfteil zurückgegeben wird.
- Mit der zweiten Syntax (Parameter *wpDok*) gibt die Funktion den in *AbschnittIndex* angegebenen Kopfteil zurück (Wert von *AbschnittIndex* muss ≥ 1 sein). Ist der Parameter *UnterAbschnittTyp* übergeben, gibt die Funktion den Kopfteil des hier angegebenen Typs zurück. Sie können eine der folgenden Konstanten übergeben:

Konstante	Typ	Wert
wk first page	Lange Ganzzahl	1
wk left page	Lange Ganzzahl	2
wk right page	Lange Ganzzahl	3

Ist *UnterAbschnittTyp* nicht angegeben, wird der Kopfteil des Hauptabschnitts zurückgegeben. Ist in diesem Fall ein Unterabschnitt aktiv, ist der Kopfteil des Hauptabschnitts *undefiniert*, d.h. der Kopfteil des Hauptabschnitts ist nur verfügbar, wenn es keinen aktiven Unterabschnitt [#command_5cst id="3479137"/] oder wk left page gibt.

Beispiel

Den Kopfteil der ersten Seite eines 4D Write Pro Dokuments in die Kopfteile aller Seiten eines anderen 4D Write Pro Dokuments kopieren:

```
C_OBJECT($headerSource;$rangeSource;$tempoc;$headerTarget;$rangeTarget)
```

```
$headerSource:=WP Get header([TEMPLATES]WPtemplate;1;wk first page) //den Kopfteil von Abschnitt 1 auf der ersten Seite erhalten
```

```
$rangeSource:=WP Create range($headerSource;wk start text;wk end text)
```

```
$tempoc:=WP New($rangeSource)
```

```
$headerTarget:=WP Get header([Docs]MyWPDoc;1) //alle Seiten von Abschnitt 1
```

```
$rangeTarget:=WP Create range($headerTarget;wk start text;wk end text)
```

```
WP INSERT DOCUMENT($rangeTarget;$tempoc;wk replace)
```

⚙️ WP Get page count

WP Get page count (wpDok) -> Funktionsergebnis

Parameter	Typ	Beschreibung
wpDok	Objekt	→ 4D Write Pro Dokument
Funktionsergebnis	Lange Ganzzahl	↩️ Anzahl Seiten im Dokument

Beschreibung

Die Funktion **WP Get page count** gibt die Gesamtanzahl der Seiten zurück, definiert im 4D Write Pro Dokument *wpDok*.

WP Get page count verwendet die gleichen Ansichtseinstellungen wie der Befehl **WP PRINT**:

- Kopfteile: sichtbar
- Fußteile: sichtbar
- Ausdrücke: berechnet und angezeigt
- Seitenmodus: Seite oder Draft

Wird *wpDoc* in einem 4D Write Pro Formularbereich verwendet, achten Sie darauf, dass die Ansichtseinstellungen des Bereichs zu den Ansichtseinstellungen der Funktion passen, damit Sie konsistente Angaben erhalten. 4D Write Pro muss das Dokument zum Berechnen des Layouts bei jedem Aufruf des Befehls "clonen", was zeitaufwändig sein kann. Deshalb empfiehlt es sich für bessere Performance, das Dokument offline zu erstellen und erst in den Formularbereich zu kopieren, wenn es komplett erstellt ist.

Beispiel

Die gesamte Seitenanzahl der 4D Write Pro Dokumente herausfinden, gespeichert im Feld "Handbuch" in Ihrer aktuellen Auswahl. Sie schreiben folgenden Code:

```
C_LONGINT($pageCount)
C_LONGINT($totalCount)
FIRST RECORD([Items])
While(Not(End selection([Items]))
    $pageCount:=WP Get page count([Items]Handbuch)
    $totalCount:=$totalCount+$pageCount
NEXT RECORD([Items])
End while
ALERT("Gesamte Seitenanzahl des Handbuchs: "+String($totalCount))
```

WP Get position

WP Get position (targetObj {; layout}) -> Funktionsergebnis

Parameter	Typ	Beschreibung
targetObj	Objekt	➔ Ausschnitt, Referenz auf Element oder 4D Write Pro Dokument
layout	Lange Ganzzahl	➔ Layout des 4D Write Pro Dokuments für die Bewertung: 0 (Standard)= 4D Write Pro Layout, 1 = HTML WYSIWYG
Funktionsergebnis	Objekt	➔ Angabe der Position

Beschreibung

Der Befehl **WP Get position** gibt ein Objekt mit der aktuellen Position von *targetObj* zurück.

Im Parameter *targetObj* übergeben Sie:

- Ausschnitt oder
- Referenz auf ein Element (Tabelle, Zeile, Absatz, Kopf-, Haupt-, Fußteil, Bild) oder
- 4D Write Pro Dokument

Der Befehl gibt die Positionsangaben in einem Objekt mit den folgenden Eigenschaften zurück:

Name der Eigenschaft	Typ	Beschreibung
Abschnitt	Zahl	Nummer des ersten Abschnitts, der <i>targetObj</i> überschneidet.
Seite	Zahl	Nummer der ersten Seite, die <i>targetObj</i> überschneidet.
Spalte	Zahl	Nummer der ersten Spalte, die <i>targetObj</i> überschneidet. (auf 1 basierend)
Zeile	Zahl	Index der ersten Zeile, die <i>targetObj</i> überschneidet. (relativ zur Spalte - oder Seite, falls es keine Spalte gibt - auf 1 basierend)
Position	Zahl	Position in der Zeile mit dem ersten Zeichen <i>targetObj</i> (auf 1 basierend)

Wenn *targetObj* ein leerer Bereich ist, gibt **WP Get position** die Position am Anfang des Ausschnitts zurück.

Wenn *targetObj* auf einen Text im Kopf- oder Fußteil verweist, gibt **WP Get position** die Position zurück, wo Kopf- und Fußzeile im ersten Abschnitt/Seite angezeigt werden.

Wenn *targetObj* auf ein verankertes Bild verweist (das auf mehreren Seiten wiederholt werden kann), gibt **WP Get position** die Position zurück, wo das Bild im ersten Abschnitt/Seite verankert ist.

Mit dem optionalen Parameter *Layout* kann die HTML wysiwyg Ansicht für die *targetObj* Bewertung gesetzt werden. Sie können eine der folgenden Konstanten aus dem Thema "4D Write Pro" übergeben:

Konstante	Typ	Wert	Kommentar
wk 4D Write Pro layout	Lange Ganzzahl	0	Standardmäßige 4D Write Pro Ausgabe mit ein paar spezifischen Stilattributen
wk html wysiwyg	Lange Ganzzahl	1	In dieser Ausgabe werden die erweiterten 4D Write Pro Attribute entfernt, die nicht mit allen Browsern kompatibel sind.

Ohne den Parameter *layout* wird standardmäßig das 4D Write Pro Layout verwendet.

WP Get position verwendet die gleichen Einstellungen zur Anzeige wie der Befehl **WP PRINT**:

- Kopfteile: sichtbar
- Fußteile: sichtbar
- Ausdrücke: berechnet und angezeigt
- Seitenmodus: Seite oder Draft (HTML wysiwyg Ansicht, wie vom Befehl definiert)

Wenn *targetObj* in einem 4D Write Pro Formularbereich angezeigt wird, stellen Sie sicher, dass die Anzeigeeinstellungen des Bereichs mit den Einstellungen für den Befehl übereinstimmen, um konsistente Informationen zu erhalten. Beachten Sie auch, dass 4D Write Pro das Dokument "klonen" muss, um das Layout bei jedem Aufruf des Befehls zu berechnen, was zeitaufwändig sein kann. Aus Performance-Gründen empfiehlt es sich in diesem Fall, das Dokument erst offline zu erstellen und anschließend in den Formularobjektbereich zu kopieren.

Beispiel 1

Sie möchten die Seitennummer eines Ausschnitts wissen:

```
C_OBJECT($range;$rangePosition)
$range:=WP Get selection(WParea) //Ausschnitt aus einer Benutzerauswahl
$rangePosition:=WP Get position($range;wk 4D Write Pro layout)
ALERT("First page number in selection is "+String($rangePosition.page))
```

Beispiel 2

Sie möchten den Text auf der ersten und letzten Seite des Dokuments blau einfärben:

```
C_LONGINT($nbPages)
C_OBJECT($body)
C_COLLECTION($_paragraphs)
```

```

$nbPages:=WP Get page count([DOC]Sample)
$body:=WP Get body([DOC]Sample)
$_paragraphs:=WP Get elements($body;wk_type_paragraph)
For each($paragraph;$_paragraphs)
  $info:=WP Get position($paragraph)
  If($info.page=1)|($info.page=$nbPages) //Abschnitte der ersten und letzten Seite in blau
    WP SET ATTRIBUTES($paragraph;wk_text_color;"blue")
  Else //Abschnitte anderer Seiten in schwarz
    WP SET ATTRIBUTES($paragraph;wk_text_color;"black")
  End if
End for each

```

Beispiel 3

Beim Erstellen eines Dokuments möchten Sie Aufteilungen in Äbsätze vermeiden:

```

C_OBJECT(WParea;$body)
C_COLLECTION($_paragraphs)
WParea:=WP New

$body:=WP Get body([DOC]Sample)
$_paragraphs:=WP Get elements($body;wk_type_paragraph)

For each($paragraph;$_paragraphs)
  $insert:=WP New($paragraph)

  //aktuellen Bereich und Seitenzahl speichern
  $rangeBefore:=WP Create range(WParea;wk_end_text;wk_end_text)
  $info:=WP Get position($rangeBefore)
  $memoPage:=$info.page

  //den Inhalt einfügen, den Bereich unverändert lassen
  WP INSERT DOCUMENT($rangeBefore;$insert;wk_append;wk_exclude_from_range)

  //Position nach dem Einsetzen prüfen
  $rangeAfter:=WP Create range(WParea;wk_end_text;wk_end_text)
  $info:=WP Get position($rangeAfter)

  If($memoPage#command_5$info.page) // Wenn sich die Seite geändert hat, einen Seitenumbruch einfügen.
    WP INSERT BREAK($rangeBefore;wk_page_break;wk_replace)
  End if
End for each

```

⚙ WP Get selection

WP Get selection ({ * ; } wpArea) -> Funktionsergebnis

Parameter	Typ	Beschreibung
*	Operator	➔ Mit Stern: wpArea ist Name des Formularobjekts (String) Ohne Stern: wpArea ist ein Objektfeld oder Variable
wpArea	String	➔ Name des Formularobjekts (mit *) oder 4D Write Pro Objektvariable oder Feld (ohne *)
Funktionsergebnis	Objekt	➔ Objekt Ausschnitt oder Bild

Beschreibung

Die Funktion **WP Get selection** gibt ein Objekt neuer Ausschnitt oder Bild zurück, gemäß dem aktuell ausgewählten Inhalt im 4D Write Pro Bereich *wpArea*.

Mit dem optionalen Parameter *** geben Sie an, dass *wpArea* ein Formularobjekt Name (String) ist. Ohne diesen Parameter geben Sie an, dass *wpArea* eine 4D Write Pro Objektvariable oder Feld ist. Ist in *wpArea* kein gültiger 4D Write Pro Bereich übergeben, wird ein leeres Objekt zurückgegeben.

Hinweis: Diese Funktion lässt sich nur verwenden, wenn *wpArea* einem Formularobjekt zugeordnet ist, z.B. im Formular angezeigt wird.

Ist in *wpArea* ein verankertes Bild ausgewählt, d.h. ein Bild an einer fixen Position, gibt **WP Get selection** ein Objekt Bildreferenz zurück. Weitere Informationen dazu finden Sie unter der Funktion **WP Add picture**. Sonst gibt **WP Get selection** einen neuen Objektausschnitt zurück. Weitere Informationen dazu finden Sie im Abschnitt **Befehle zum Verwalten des Ausschnitts**.

Beispiel

Den ausgewählten Text von einem 4D Write Pro Bereich erhalten:

```
$range:=WP Get selection(*;"WParea")
```

WP Get text

WP Get text (targetObj {; expressions}) -> Funktionsergebnis

Parameter	Typ	Beschreibung
targetObj	Objekt	→ Ausschnitt, Referenz auf Element oder 4D Write Pro Dokument
expressions	Zahl	→ Anweisung zum Bewerten der Ausdrücke
Funktionsergebnis	Text	↪ Text des Zielobjekts

Beschreibung

Der Befehl **WP Get text** ruft Textinhalte aus einem 4D Write Pro Dokument im angegebenen Parameter *targetObj* ab.

WP Get text gibt nur reinen Text zurück. Alle angewandten Stile (fett, kursiv, etc.) werden ignoriert.

Im Parameter *targetObj* übergeben Sie:

- Ausschnitt oder
- Referenz auf ein Element (Tabelle, Zeile, Absatz, Kopf-, Haupt-, Fußteil) oder
- 4D Write Pro Dokument

Der optionale Parameter *expressions* liefert Anweisungen für die Auswertung von 4D-Ausdrücken in *targetObj*. Die Optionen sind:

Konstante	Kommentar
wk expressions as source	Der ursprüngliche String der Referenz auf 4D Ausdrücke wird zurückgegeben.
wk expressions as space	Jede Referenz wird als nicht-umbrechendes Leerzeichen zurückgegeben.
wk expressions as value	Referenzen auf 4D Ausdrücke werden in ihrer ausgewerteten Form zurückgegeben. (Standard, wenn die Option nicht übergeben ist).

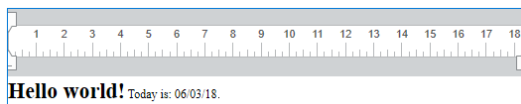
Wichtig: Nur reiner Text mit wk expressions as space ist konsistent mit Indices zu Textausschnitten, die WP oder ST Befehle verwenden, da WP und ST Befehle immer annehmen, dass die Größe des Ausdrucks = 1 Zeichen ist.

Hinweise:

- Wenn *targetObj* Bilder enthält, werden die Bilder ignoriert und als Leerzeichen (" ") zurückgegeben.
- Wenn *targetObj* Tabellen enthält, wird der Inhalt jeder Zelle als einzelne Absätze behandelt und als Text zurückgegeben, der durch Tabulatoren getrennt ist. Zeilen werden durch Zeilenumbrüche getrennt.

Beispiel

Um den Text in diesem Dokument abzurufen:



können Sie schreiben:

```
$range:=WP Create range(myDoc2;wk start text;wk end text)
vText:=WP Get text($range;wk expressions as value) //ergibt "Hello world! Today is: 06/03/18."
vText2:=WP Get text($range;wk expressions as source) //ergibt "Hello world! Today is: Current date."
vText3:=WP Get text($range;wk expressions as space) //ergibt "Hello world! Today is: ."
```

WP Import document

WP Import document (DateiPfad) -> Funktionsergebnis

Parameter	Typ	Beschreibung
DateiPfad	String →	Pfad zum 4D Write Dokument (.4w7 oder .4wt) oder 4D Write Pro Dokument (.4wp)
Funktionsergebnis	Objekt ↪	Objekt 4D Write Pro

Beschreibung

Die Funktion **WP Import document** konvertiert ein vorhandenes 4D Write Pro oder 4D Write Dokument (4wp., .4w7 oder .4wt) in ein neues Objekt 4D Write Pro.

Im Parameter *DateiPfad* übergeben Sie den Pfad zu einem auf der Festplatte gespeicherten Dokument. Folgende Dokumenttypen werden unterstützt:

- Bisherige 4D Write Dokumente (.4w7 oder .4wt). Die ausführliche Liste der 4D Write Features, die derzeit in 4D Write Pro Objekten unterstützt werden, finden Sie im Abschnitt **4D Write Dokumente importieren**.
- Dokumente im Format 4D Write Pro (.4wp). Weitere Informationen dazu finden Sie im Abschnitt **Dokumentformat .4wp**.

Sie müssen den vollständigen Pfad angeben, außer das Dokument liegt auf der gleichen Ebene wie die Strukturdatei. Dann genügt es, nur seinen Namen zu übergeben.

Die Funktion gibt das Objekt 4D Write Pro zurück, das sich aus dieser Konvertierung ergibt.

Hinweis: Um ein Dokument zu importieren, das in einem 4D BLOB Feld gespeichert ist, können Sie auch die Funktion **WP New** verwenden.

Ist der Parameter *DateiPfad* ungültig, fehlt die Datei oder wird das Dateiformat nicht unterstützt, wird ein Fehler zurückgegeben.

Beispiel

```
C_OBJECT(WPDoc)
WPDoc:=WP Import document("C:\\documents\\4DWriteDocs\\Letter.4w7")
```


WP INSERT BREAK

WP INSERT BREAK (AusschnittObj ; UmbruchTyp ; Modus {; UpdateAusschnitt})

Parameter	Typ		Beschreibung
AusschnittObj	Objekt	⇒	Ausschnitt
UmbruchTyp	Lange Ganzzahl	⇒	Typ des einzufügenden Umbruchs
Modus	Lange Ganzzahl	⇒	Einfügemodus
UpdateAusschnitt	Lange Ganzzahl	⇒	Modus für UpdateAusschnitt

Beschreibung

Der Befehl **WP INSERT BREAK** fügt einen neuen Umbruch vom Typ *UmbruchTyp* im Bereich *AusschnittObj* ein gemäß den Parametern *Modus* zum Einfügen und *UpdateAusschnitt*.

In *AusschnittObj* übergeben Sie einen Ausschnitt.

In *UmbruchTyp* können Sie eine der folgenden Konstanten unter dem Thema **4D Write Pro Konstanten** übergeben:

Konstante	Typ	Wert	Kommentar
wk column break	Lange Ganzzahl	3	Spaltenumbruch: Fügt einen Spaltenumbruch ein
wk line break	Lange Ganzzahl	0	Zeilenumbruch (im gleichen Absatz)
wk page break	Lange Ganzzahl	2	Seitenumbruch: definiert eine neue Seite
wk section break	Lange Ganzzahl	1	Abschnittsumbruch: definiert einen neuen Abschnitt

In *Modus* übergeben Sie eine Konstante für den Einfügemodus des Umbruchs im Zielbereich *AusschnittObj*:

Konstante	Typ	Wert	Kommentar
wk append	Lange Ganzzahl	2	Fügt Inhalt am Ende des Ausschnitts ein
wk prepend	Lange Ganzzahl	1	Fügt Inhalt am Anfang des Ausschnitts ein
wk replace	Lange Ganzzahl	0	Ersetzt Inhalt des Ausschnitts

Im optionalen Parameter *UpdateAusschnitt* können Sie über folgende Konstanten festlegen, ob der eingefügte Inhalt in den aktualisierten Ausschnitt übernommen wird oder nicht:

Konstante	Typ	Wert	Kommentar
wk exclude from range	Lange Ganzzahl	1	Eingefügter Inhalt ist nicht in aktualisierten Ausschnitt enthalten
wk include in range	Lange Ganzzahl	0	Eingefügter Inhalt ist im aktualisierten Ausschnitt enthalten (Standard)

Standardmäßig, d.h. ohne den Parameter *UpdateAusschnitt* wird der eingefügte Inhalt in den aktualisierten Ausschnitt übernommen.

Beispiel

Beim Einrichten von Rechnungen sollen Seitenumbrüche eingefügt werden mit Ausnahme der letzten Seite:

```
$nblInvoices:=Records in selection([INVOICE])
For($j;1;$nblInvoices)
... //Rechnungen bearbeiten
If($j#command_5$nblInvoices) //Fügt Seitenumbruch ein mit Ausnahme der letzten Seite
    WP INSERT BREAK($buildRange;wk page break;wk append;wk exclude from range)
End if
End for
```

WP INSERT DOCUMENT

WP INSERT DOCUMENT (AusschnittObj ; wpDok ; Modus {; UpdateAusschnitt})

Parameter	Typ		Beschreibung
AusschnittObj	Objekt	⇒	Ausschnitt
wpDok	Objekt	⇒	4D Write Pro Dokument
Modus	Lange Ganzzahl	⇒	Einfügemodus
UpdateAusschnitt	Lange Ganzzahl	⇒	Modus für Update Ausschnitt

Beschreibung

Der Befehl **WP INSERT DOCUMENT** fügt das Dokument *wpDok* im Bereich *AusschnittObj* ein, gemäß den Parametern *Modus* zum Einfügen und *UpdateAusschnitt*.

In *AusschnittObj* übergeben Sie einen Ausschnitt.

Das eingefügte Objekt *wpDok* kann ein beliebiges Objekt 4D Write Pro Dokument sein, erstellt mit dem Befehl **WP New** oder **WP Import document**. Nur die Unterelemente des Hauptteils werden eingefügt (Abschnitte und Bookmarks im Zielbereich werden beibehalten). Außerdem werden die Elemente kopiert, so dass sich *wpDok* mehrere Male verwenden lässt.

Im Parameter *Modus* übergeben Sie eine oder mehrere miteinander kombinierte Konstanten unter dem Thema **4D Write Pro Konstanten** für den Einfügemodus im Zielbereich *AusschnittObj*:

Konstante	Typ	Wert	Kommentar
wk append	Lange Ganzzahl	2	Fügt Inhalt am Ende des Ausschnitts ein
wk prepend	Lange Ganzzahl	1	Fügt Inhalt am Anfang des Ausschnitts ein
wk replace	Lange Ganzzahl	0	Ersetzt Inhalt des Ausschnitts

Sie können eine der vorigen Konstanten mit den folgenden Einfügeooptionen kombinieren:

Konstante	Typ	Wert	Kommentar
wk freeze expressions	Lange Ganzzahl	64	Friert Ausdrücke beim Einfügen ein
wk keep paragraph styles	Lange Ganzzahl	128	Stilarten des Zielabsatzes beibehalten

Im optionalen Parameter *UpdateAusschnitt* können Sie eine der folgenden Konstanten zum Einfügen in den aktualisierten Ausschnitt übergeben:

Konstante	Typ	Wert	Kommentar
wk exclude from range	Lange Ganzzahl	1	Eingefügter Inhalt ist nicht in aktualisierten Ausschnitt enthalten
wk include in range	Lange Ganzzahl	0	Eingefügter Inhalt ist im aktualisierten Ausschnitt enthalten (Standard)

Standardmäßig, d.h. ohne den Parameter *UpdateAusschnitt* wird der eingefügte Inhalt in den Zielbereich übernommen.

Beispiel 1

Den Inhalt eines Dokuments mit einen anderen ersetzen:

```
$tempRange:=WP Get selection(WPTemplate) //Auswahl des Benutzers im Dokument WPTemplate finden
$doctoCopy:=WP New($tempRange) //Ein neues Dokument aus der Vorlage WPTemplate erstellen
WP INSERT DOCUMENT(WPDoc;$doctoCopy;wk replace) //Inhalt von WPDoc mit Inhalt des neuen Dokuments ersetzen
```

Beispiel 2

Sie haben eine Vorlage mit verschiedenen vorformatierten Teilen definiert, jeder Teil wird als Lesezeichen gespeichert. Beim Erstellen des endgültigen Dokuments aus der Vorlage können Sie jedes Lesezeichen als ein neues Dokument entnehmen und in das endgültige Dokument einfügen.

```
ARRAY TEXT($_BookmarkNames;0)
WP GET BOOKMARKS([TEMPLATES]WP;$_BookmarkNames) //Die Lesezeichen von der Vorlage erhalten
$targetRange:=WP New //Ein leeres Dokument erstellen (wird das endgültige Dokument)

$p:=Find in array($_BookmarkNames;"Main_Header") //Den Hauptkopfteil verwalten
If($p>0)
  $Range:=WP Get bookmark range(WParea;$_BookmarkNames{$p}) //Den Ausschnitt auswählen
  $RangeDoc:=WP New($Range) //Ein neues Dokument aus dem Ausschnitt erstellen
  WP INSERT DOCUMENT($targetRange;$RangeDoc;wk append+wk freeze expressions) //wk append=nach Ersetzen ist $targetRange
  gleich dem Ende des ersetzten Textes
End if
```

WP INSERT PICTURE

WP INSERT PICTURE (AusschnittObj ; Bild ; Modus {; UpdateAusschnitt})

Parameter	Typ	Beschreibung
AusschnittObj	Objekt	→ Objektausschnitt
Bild	Bild	→ Bildfeld oder -variable, oder Pfad zu Bildfeld auf der Festplatte
Modus	Lange Ganzzahl	→ Einfügemodus
UpdateAusschnitt	Lange Ganzzahl	→ Updatemodus für Ausschnitt

Beschreibung

Der Befehl **WP INSERT PICTURE** fügt das *Bild* in *AusschnittObj* mit dem in *Modus* und *UpdateAusschnitt* angegebenen Einfügemodus ein. Das Bild wird als Zeichen in *AusschnittObj* eingefügt.

In *AusschnittObj* übergeben Sie einen Ausschnitt.

In *Bild* können Sie folgendes übergeben:

- Feld oder Variable vom Typ Bild oder
- Dateipfad (String) zu einer Bilddatei auf der Festplatte, in der Syntax des jeweiligen Betriebssystems. Sie können den kompletten Pfadnamen, einen Pfadnamen in Bezug zur Strukturdatei der Anwendung übergeben. Sie können auch nur die Dateiname angeben, dann muss die Datei auf derselben Ebene wie die Strukturdatei der Datenbank liegen und es ist eine Dateierweiterung erforderlich.

Jedes von 4D unterstützte Bildformat ist verwendbar (siehe Abschnitt **Überblick über Bilder**). Die Liste der verfügbaren Bildformate erhalten Sie über den Befehl **PICTURE CODEC LIST**. Enthält *Bild* verschiedene Formate (Codecs), behält 4D Write Pro im Dokument nur ein Format für die Anzeige und ein Format zum Drucken bei (falls unterschiedlich). Es werden automatisch die am besten passenden Formate ausgewählt.

Im Parameter *Modus* übergeben Sie eine der folgenden Konstanten für die Position zum Einfügen des Bildes im Dokument:

Konstante	Typ	Wert	Kommentar
wk append	Lange Ganzzahl	2	Fügt Inhalt am Ende des Ausschnitts ein
wk prepend	Lange Ganzzahl	1	Fügt Inhalt am Anfang des Ausschnitts ein
wk replace	Lange Ganzzahl	0	Ersetzt Inhalt des Ausschnitts

Im optionalen Parameter *UpdateAusschnitt* übergeben Sie eine der folgenden Konstanten, ob das eingefügte Bild im Ausschnitt enthalten ist oder nicht:

Konstante	Typ	Wert	Kommentar
wk exclude from range	Lange Ganzzahl	1	Eingefügter Inhalt ist nicht in aktualisierten Ausschnitt enthalten
wk include in range	Lange Ganzzahl	0	Eingefügter Inhalt ist im aktualisierten Ausschnitt enthalten (Standard)

Ohne den Parameter *UpdateAusschnitt* ist das eingefügte Bild standardmäßig im Ausschnitt enthalten.

Beispiel

In diesem Beispiel kann der Benutzer das gewünschte Bild zum Einfügen in den Ausschnitt wählen und erhält eine Meldung, wenn das Bild nicht eingefügt werden konnte:

```
C_OBJECT($wpRange)
$wpRange:=WP Get selection([EXAMPLES]wpDoc)

C_BOOLEAN($fail)
$fail:=False

//Benutzer fragen, welches Bild von der Festplatte eingefügt werden soll
$imgRef:=Open document("")
//Bricht der Benutzer nicht ab
if(OK=1)
//Ist die Datei ein unterstütztes Bildformat
if(Is picture file(document))
// das vom Benutzer gewählte Bild einfügen
WP INSERT PICTURE($wpRange;document;wk replace)
Else
$fail:=True
End if
Else
$fail:=True
End if
//schlägt das Einfügen fehl, Meldung an Benutzer senden
if($fail)
ALERT("Einfügen des Bildes ist fehlgeschlagen")
End if
```

WP Insert table

WP Insert table (objAusschnitt ; Modus {; UpdateAusschnitt {; AnzSpalten {; AnzZeilen}} }) -> Funktionsergebnis

Parameter	Typ		Beschreibung
objAusschnitt	Objekt	→	Objekt Ausschnitt
Modus	Lange Ganzzahl	→	Einfügemodus
UpdateAusschnitt	Lange Ganzzahl	→	Modus Update Ausschnitt
AnzSpalten	Lange Ganzzahl	→	Anzahl Spalten für die Tabelle
AnzZeilen	Lange Ganzzahl	→	Anzahl Zeilen für die Tabelle
Funktionsergebnis	Objekt	↻	Neues Objekt Tabelle

Beschreibung

Die Funktion **WP Insert table** fügt eine neue Tabelle in *ObjAusschnitt* gemäß der Angabe in *Modus* ein und gibt das entsprechende neue Objekt Tabelle zurück. Optional können Sie einen Parameter *UpdateAusschnitt* sowie eine vordefinierte Anzahl Spalten in *AnzSpalten* und Zeilen in *AnzZeilen* zurückgeben.

In *ObjAusschnitt* übergeben Sie einen Ausschnitt.

Hinweis: Eine Tabelle lässt sich nicht innerhalb einer anderen Tabelle einfügen (Es wird ein Fehler zurückgegeben).

Im Parameter *Modus* übergeben Sie eine der folgenden Konstanten, um den Einfügemodus für die Tabelle im Dokument zu definieren:

Konstante	Typ	Wert	Kommentar
wk append	Lange Ganzzahl	2	Fügt Inhalt am Ende des Ausschnitts ein
wk prepend	Lange Ganzzahl	1	Fügt Inhalt am Anfang des Ausschnitts ein
wk replace	Lange Ganzzahl	0	Ersetzt Inhalt des Ausschnitts

Im optionalen Parameter *UpdateAusschnitt* übergeben Sie eine der folgenden Konstanten, um festzulegen, ob die eingefügte Tabelle im Ausschnitt enthalten ist:

Konstante	Typ	Wert	Kommentar
wk exclude from range	Lange Ganzzahl	1	Eingefügter Inhalt ist nicht in aktualisierten Ausschnitt enthalten
wk include in range	Lange Ganzzahl	0	Eingefügter Inhalt ist im aktualisierten Ausschnitt enthalten (Standard)

Übergeben Sie keinen Parameter *UpdateAusschnitt*, ist die eingefügte Tabelle standardmäßig im Ausschnitt enthalten.

Über die optionalen Parameter *AnzSpalten* und *AnzZeilen* können Sie die Anzahl Spalten und Zeilen in der Tabelle definieren. Standardmäßig, d.h. ohne diese Parameter, wird eine neue Tabelle mit 0 Spalten und 0 Zeilen angelegt. Sie können diese Elemente dann mit der Funktion **WP Table append row** hinzufügen.

Die Funktion gibt die neue Tabelle als Tabellenobjekt zurück, das Sie dann in anderen Befehlen übergeben können, deren Name mit "WP Table" beginnt. Die Tabelle wird mit Standardschrift, -rahmen und Farbattributen erstellt. Mit dem Befehl **WP SET ATTRIBUTES** können Sie die Standardeinstellungen verändern.

Beispiel

In einem leeren Dokument eine neue Tabelle mit 3 Spalten und 2 Zeilen hinzufügen:

```
C_OBJECT($WPRange;$WPTable)
$WPRange:=WP Create range(WParea;wk start text;wk end text)
$WPTable:=WP Insert table($WPRange;wk append;wk exclude from range;3;2)
```

Ergebnis:

WP Is font style supported

WP Is font style supported (ZielObj ; wpSchriftStil) -> Funktionsergebnis

Parameter	Typ	Beschreibung
ZielObj	Objekt	→ Ausschnitt oder Referenz auf ein Element
wpSchriftStil	Lange Ganzzahl	→ Konstante für Schriftstil
Funktionsergebnis	Boolean	→ Wahr, wenn in einem Teil von Zielobj wpSchriftstil vorkommt, sonst Falsch

Beschreibung

Die Funktion **WP Is font style supported** gibt wahr zurück, wenn ein Teil des Textes in *Zielobj* den Stil *wpSchriftStil* unterstützt.

In *Zielobj* übergeben Sie

- Ausschnitt oder
- Referenz auf ein Element (Tabelle, Zeile, Absatz, Bild im Textfluss, Kopf-, Haupt-, Fußteil)

In *wpSchriftStil* übergeben Sie eine der folgenden Konstanten unter dem Thema "4D Write Pro":

Konstante	Typ	Wert	Kommentar
wk font bold	Zeichenkette	fontBold	<p>Gibt die Textstärke an (richtet sich nach verfügbaren Schriftstilen). Mögliche Werte:</p> <ul style="list-style-type: none">• <u>wk true</u> um ausgewählte Zeichen auf den Schriftstil Fett zu setzen; mit dem Befehl WP GET ATTRIBUTES wird <u>wk true</u> zurückgegeben, wenn mindestens ein ausgewähltes Zeichen in Fettschrift ist.• <u>wk false</u> (Standard) um den Schriftstil Fett - falls vorhanden - von den ausgewählten Zeichen zu entfernen; mit dem Befehl WP GET ATTRIBUTES wird <u>wk false</u> zurückgegeben, wenn keins der ausgewählten Zeichen in Fettschrift ist.
wk font italic	Zeichenkette	fontItalic	<p>Gibt den Stil Kursiv für Text an (richtet sich nach den Schriftstilen). Mögliche Werte:</p> <ul style="list-style-type: none">• <u>wk true</u>, um ausgewählte Zeichen auf Kursiv- oder Schrägschrift zu setzen; der Befehl WP GET ATTRIBUTES gibt <u>wk true</u> zurück, wenn mindestens ein ausgewähltes Zeichen Kursiv- oder Schrägschrift unterstützt.• <u>wk false</u> (Standard), um in gewählten Zeichen Kursiv- oder Schrägschrift - wenn vorhanden - zu entfernen; der Befehl gibt <u>wk false</u> zurück, wenn keines der gewählten Zeichen in Kursiv- oder Schrägschrift ist.
wk text linethrough style	Zeichenkette	textLinethroughStyle	<p>Gibt Stil für durchgestrichenen Text an (wenn vorhanden). Mögliche Werte:</p> <ul style="list-style-type: none">• <u>wk none</u> (Standard): Keine Auswirkung• <u>wk solid</u>: Zeichnet durchgezogene Linie im ausgewählten Text• <u>wk dotted</u>: Zeichnet gepunktete Linie im ausgewählten Text• <u>wk dashed</u>: Zeichnet gestrichelte Linie im ausgewählten Text• <u>wk double</u>: Zeichnet doppelte Linie im ausgewählten Text• <u>wk semi transparent</u>: gedimmte Linie im ausgewählten Text. Ist mit anderem Linienstil kombinierbar.• <u>wk word</u>: Zeichnet die Linie nur bei Wörtern (Leerzeichen werden ausgelassen). Ist mit anderem Linienstil kombinierbar.
wk text underline style	Zeichenkette	textUnderlineStyle	<p>Gibt Stil für unterstrichenen Text an (wenn vorhanden). Mögliche Werte:</p> <ul style="list-style-type: none">• <u>wk none</u> (Standard): Keine Unterstreichung• <u>wk solid</u>: Zeichnet durchgezogene Linie im ausgewählten Text• <u>wk dotted</u>: Zeichnet gepunktete Linie im ausgewählten Text• <u>wk dashed</u>: Zeichnet gestrichelte Linie im ausgewählten Text• <u>wk double</u>: Zeichnet doppelte Linie im ausgewählten Text• <u>wk semi transparent</u>: gedimmte Linie im ausgewählten Text. Ist mit anderem Linienstil kombinierbar.• <u>wk word</u>: Zeichnet die Linie nur bei Wörtern (Leerzeichen werden ausgelassen). Ist mit anderem Linienstil kombinierbar.

Diese Funktion ermöglicht Entwicklern, eigene Oberflächenobjekte einzufügen, wie z.B. ein PopUp-Menü mit verschiedenen Stiloptionen für den ausgewählten Text.

WP New

WP New {{ Quelle }} -> Funktionsergebnis

Parameter	Typ	Beschreibung
Quelle	String, BLOB, Objekt	→ HTML Quelle oder 4D Write Blob Dokument oder 4D Write Pro Dokument oder Ausschnitt oder Referenz auf Element
Funktionsergebnis	Objekt	↪ 4D Write Pro Objekt

Beschreibung

Die Funktion **WP New** erstellt bzw. gibt ein Objekt 4D Write Pro zurück.

Lassen Sie den Parameter *Quelle* weg, gibt die Funktion standardmäßig ein leeres Objekt 4D Write Pro zurück.

Übergeben Sie den Parameter *Quelle*, wird das neue Objekt 4D Write Pro mit dem Inhalt von *Quelle* gefüllt. Sie können folgendes übergeben:

- **String**: Damit übergeben Sie eine 4D HTML Quelle, z.B. einen Text, der mit **WP EXPORT VARIABLE** mit der Option `wk_web_page_html 4D` exportiert wurde. Dieser Text kann Referenzen (4D Tags und Ausdrücke) und eingebundene Bilder enthalten.
- **BLOB**: Damit übergeben Sie entweder:
 - ein Dokument im 4D Write Pro Format (.4wp), das in einem BLOB gespeichert ist. Weitere Informationen dazu finden Sie im Abschnitt **Dokumentformat .4wp**.
 - oder einen bisher üblichen 4D Write Bereich, der in ein BLOB geladen ist (unterstützt werden Blobs mit .4w7 oder .4wt Dokumenten).
Eine ausführliche Liste von 4D Write Features, die derzeit in 4D Write Pro Objekten unterstützt werden, finden Sie im Abschnitt **4D Write Dokumente importieren**. Um ein 4D Write Dokument (.4w7 oder .4wt) von der Festplatte zu importieren, können Sie auch die Funktion **WP Import document** verwenden.
- **Objekt**: Damit übergeben Sie entweder:
 - einen Ausschnitt.
Dann gibt **WP New** ein neues Dokument mit dem spezifischen Ausschnitt zurück. Ist der Ausschnitt nicht gleich mit dem ganzen Dokumentbereich, wird nur der erste Abschnitt exportiert, evtl. vorhandene Lesezeichen werden nicht exportiert.
 - oder eine Referenz auf ein Element (Tabelle, Zeile, Absatz, Bild im Textfluss, Kopf- oder Fußteil).
Dann gibt **WP New** ein neues Dokument zurück, das nur den Inhalt eines Elements im Hauptteil enthält. Für eine Zeile enthält das neue Dokument ein Tabellenelement mit der Zeile.

Das zurückgegebene Objekt ist ein komplettes Dokument, das z.B. dem Befehl **WP INSERT DOCUMENT** übergeben werden kann.

Beispiel 1

Ein leeres Objekt 4D Write Pro erstellen:

```
myWPObject:=WP New
```

Beispiel 2

Ein Objekt 4D Write Pro mit Referenz auf einen 4D Ausdruck erstellen:

```
C_TEXT(myText)
myText:="Heute ist "
ST INSERT EXPRESSION(myText;"string(current date;System date long)";ST End text)
myWPA:=WP New(myText)
```

Beispiel 3

Ihren 4D Write Pro Bereich mit einer zuvor erstellten Vorlage initialisieren:

```
//Vorlage aus einem vorhandenen Bereich exportieren
C_TEXT(wpTemplate)
WP EXPORT VARIABLE(myWPAArea;wpTemplate;wk_web_page_html 4D)

// Die Vorlage für einen neuen Bereich verwenden
C_OBJECT(myNewWPA)
myNewWPA:=WP New(wpTemplate)
```

Beispiel 4

Ein 4D Write Dokument, das in einem 4D Feld des aktuellen Datensatzes gespeichert ist, in einen neuen 4D Write Pro Bereich importieren:

```
C_OBJECT(wpArea)
wpArea=WP New([Templates]Reference_)
```

Beispiel 5

Sie haben eine Vorlage mit verschiedenen vorformatierten Teilen definiert, jeder Teil wird als Lesezeichen gespeichert. Beim Erstellen des endgültigen Dokuments aus der Vorlage können Sie jedes Lesezeichen als ein neues Dokument entnehmen und in das endgültige Dokument einfügen.

```
ARRAY TEXT($_BookmarkNames;0)
WP GET BOOKMARKS([TEMPLATES]WP;$_BookmarkNames) //Die Lesezeichen von der Vorlage erhalten
$targetRange:=WP New //Ein leeres Dokument erstellen (wird das endgültige Dokument)

$p:=Find in array($_BookmarkNames;"Main_Header") //Den Hauptkopfteil verwalten
if($p>0)
  $Range:=WP Get bookmark range(WParea;$_BookmarkNames{$p}) //Den Ausschnitt auswählen
  $RangeDoc:=WP New($Range) //Ein neues Dokument aus dem Ausschnitt erstellen
  WP INSERT DOCUMENT($targetRange;$RangeDoc;wk append+wk freeze expressions) //wk append=nach Ersetzen ist $targetRange
  gleich dem Ende des ersetzten Textes
End if
```

WP PRINT

WP PRINT (wpDok {; Druckausgabe})

Parameter	Typ	Beschreibung
wpDok	Objekt	→ Name des 4D Write Pro Dokuments
Druckausgabe	Lange Ganzzahl	→ Druckausgabe für das 4D Write Pro Dokument: 0 (Standard)=4D Write Pro Ausgabe, 1=HTML WYSIWYG

Beschreibung

Der Befehl **WP PRINT** startet einen Druckauftrag für das 4D Write Pro Dokument, definiert in *wpDok*, oder fügt das Dokument zum aktuellen Druckauftrag hinzu, wenn es zwischen den Befehlen **OPEN PRINTING JOB** und **CLOSE PRINTING JOB** aufgerufen wird (nur 64-bit Versionen von 4D, siehe unten). **WP PRINT** verwendet die Druckeinstellungen, die über die regulären 4D Befehle **PRINT SETTINGS** oder **SET PRINT OPTION** definiert wurden, außer für Druckränder, die immer von den Seiteneinstellungen der 4D Write Pro Dokumente übernommen werden. **WP PRINT** verwendet die aktuellen Seitenoptionen (wie Seitengröße und Ausrichtung) oder die Optionen des Dokuments, wenn zuvor **WP USE PAGE SETUP** aufgerufen wurde.

Mit dem optionalen Parameter *Druckausgabe* lässt sich die HTML wysiwyg Ansicht für die Druckausgabe verwenden. Sie können eine der folgenden Konstanten unter dem Thema "4D Write Pro" übergeben:

Konstante	Typ	Wert	Kommentar
wk 4D Write Pro layout	Lange Ganzzahl	0	Standardmäßige 4D Write Pro Ausgabe mit ein paar spezifischen Stilattributen
wk html wysiwyg	Lange Ganzzahl	1	In dieser Ausgabe werden die erweiterten 4D Write Pro Attribute entfernt, die nicht mit allen Browsern kompatibel sind.

Ohne den Parameter *Druckausgabe* wird standardmäßig das 4D Write Pro Layout (0) verwendet.

Hinweis: Beim Drucken mit **WP PRINT** werden 4D Write Pro Dokumente immer im Modus Seite gedruckt, unabhängig welcher Seitenmodus für den Bereich gewählt wurde. Weitere Informationen dazu finden Sie im Abschnitt **Darstellungseigenschaften konfigurieren**.

Hinweis für 32-bit Versionen

Der Befehl **WP PRINT** wird in 4D 32-bit Versionen unterstützt, jedoch mit folgender Einschränkung: Er lässt sich nicht in einem mit **OPEN PRINTING JOB** gestarteten 4D Druckauftrag aufrufen. In diesem Fall wird ein Fehler zurückgegeben.

Beispiel

Ein 4D Write Pro Dokument je nach dem Wert der Variablen im Standard Layout oder im HTML wysiwyg Layout drucken:

```
// Mit einem spezifischen HTML wysiwyg Layout oder dem 4D Write Pro Layout drucken
if(rb_htmlwysiwyg=1)
  WP PRINT(writeProDoc;wk html wysiwyg)
Else
  WP PRINT(writeProDoc;wk 4D Write Pro layout)
End if
```


WP RESET ATTRIBUTES

WP RESET ATTRIBUTES (ZielObj ; attribName { ; attribName2 ; ... ; attribNameN })

Parameter	Typ	Beschreibung
ZielObj	Objekt	→ Ausschnitt, Referenz auf Element oder 4D Write Pro Dokument
attribName	String	→ Name des Attributs zum Entfernen

Beschreibung

Der Befehl **WP RESET ATTRIBUTES** setzt den Wert eines oder mehrerer Attribute in einem in *Zielobj* übergebenen Ausschnitt, Referenz auf ein Element oder Dokument zurück. Es lassen sich alle Arten interner Attribute in 4D Write Pro entfernen: Zeichen, Absatz, Dokument, Tabelle oder Bild.

Im Parameter *Zielobj* übergeben Sie:

- Ausschnitt oder
- Referenz auf ein Element (Tabelle, Absatz, verankertes Bild, Bild im Textfluss, Kopf-, Haupt-, Fußteil) oder
- Ein 4D Write Pro Dokument

Wird der Wert eines Attributs über **WP RESET ATTRIBUTES** entfernt, wird es auf den Standardwert zurückgesetzt. Weitere Informationen dazu finden Sie im Absatz **4D Write Pro Attribute**.

War das zurückzusetzende Attribut nicht im als Parameter übergebenen Element definiert, führt der Befehl nichts aus.

Beispiel

In folgender Auswahl mehrere Attribute entfernen:

Dazu setzen Sie folgenden Code:

```
$range:=WP Get selection(*;"WParea")
WP RESET ATTRIBUTES($range;wk.padding)
WP RESET ATTRIBUTES($range;wk.background_color)
WP RESET ATTRIBUTES($range;wk.text_underline_style)
WP RESET ATTRIBUTES($range;wk.margin)
WP RESET ATTRIBUTES($range;wk.border_style)
```

Das ergibt folgendes Dokument:

WP SELECT ({ * ; } wpArea { ; ZielObj } | { ; StartAusschnitt ; EndeAusschnitt })

Parameter	Typ	Beschreibung
*	Operator	➔ Mit Stern: wpArea ist Formularobjekt Name (String) ➔ Ohne Stern: wpArea ist ein Objektfeld oder Variable
wpArea	String, Objekt	➔ Formularobjekt Name (mit *) oder 4D Write Pro Objektvariable oder Feld (ohne *)
ZielObj	Objekt	➔ Ausschnitt oder Referenz auf Element
StartAusschnitt	Lange Ganzzahl	➔ Anfang des neuen Ausschnitts
EndeAusschnitt	Lange Ganzzahl	➔ Ende des neuen Ausschnitts

Beschreibung

Der Befehl **WP SELECT** erstellt eine neue Auswahl im 4D Write Pro Bereich *wpArea*, gemäß dem Parameter *Zielobj* oder einen neuen Ausschnitt, definiert durch *StartAusschnitt* und *EndeAusschnitt*.

Mit dem optionalen Parameter *** geben Sie an, dass *wpArea* ein Formularobjekt Name (String) ist. Ohne diesen Parameter geben Sie an, dass *wpArea* eine 4D Write Pro Objekt Variable oder Feld ist. Ist in *wpArea* kein gültiger 4D Write Pro Bereich übergeben, führt der Befehl nichts aus.

Hinweis: Dieser Befehl lässt sich nur verwenden, wenn *wpArea* einem Formularobjekt zugeordnet ist, z.B. im Formular angezeigt wird.

Zum Definieren der Auswahl können Sie entweder ein vorhandenes Objekt in *Zielobj* übergeben oder den Ausschnitt mit dem Wertepaar *StartAusschnitt* / *EndeAusschnitt* festlegen:

- **Erste Syntax:** `WP SELECT({ * ; } wpArea ; Zielobj)`
Zielobj kann folgendes sein:
 - Ausschnitt, oder
 - Referenz auf ein Element (Tabelle, Zeile, Absatz, verankertes Bild, Bild im Textfluss, Kopfteil, Fußteil)
- **Zweite Syntax:** `WP SELECT({ * ; } wpArea ; StartAusschnitt ; EndeAusschnitt)`
 In diesem Fall übergeben Sie in *StartAusschnitt* und *EndeAusschnitt* Werte für die Position des ersten und letzten Zeichens der Auswahl im Bereich. Sie können wk start text in *StartAusschnitt* übergeben, um den Anfang des Ausschnitts zu definieren und wk end text in *EndeAusschnitt*, um das Ende des Ausschnitts zu definieren. Beachten Sie, dass ein 4D Write Pro Dokument nicht nur sichtbaren Text, sondern innerhalb des Ausschnitts auch Tags zur Formatierung enthält.

Beispiel

Nachfolgender Code:

```
$range2:=WP Create range([SAMPLE]WP;wk start text;12)
WP SELECT(*;"WParea";$range2)
```

... ergibt dasselbe Ergebnis wie:

```
WP SELECT(*;"WParea";wk start text;12)
```

WP SET ATTRIBUTES

WP SET ATTRIBUTES (ZielObj ; attribName ; attribWert { ; attribName2 ; attribWert2 ; ... ; attribNameN ; attribWertN })

Parameter	Typ	Beschreibung
ZielObj	Objekt	→ Ausschnitt, Referenz auf Element oder 4D Write Pro Dokument
attribName	Text	→ Name des zu setzenden Attributs
attribWert	Text, Zahl, Array Lange Ganzzahl, Array Zahl, Bild, Datum	→ Neuer Attributwert

Beschreibung

Der Befehl **WP SET ATTRIBUTES** setzt den Wert jedes Attributs in einem Ausschnitt, Referenz auf ein Element oder 4D Write Pro Dokument. Dieser Befehl gibt Ihnen Zugriff auf die verschiedenen internen Attribute von 4D Write Pro: Zeichen, Absatz, Dokument, Tabelle oder Bild.

Im Parameter *Zielobj* übergeben Sie:

- Ausschnitt oder
- Referenz auf ein Element (Tabelle, Absatz, verankertes Bild, Bild im Textfluss, Kopf-, Haupt-, Fußteil) oder
- Ein 4D Write Pro Dokument

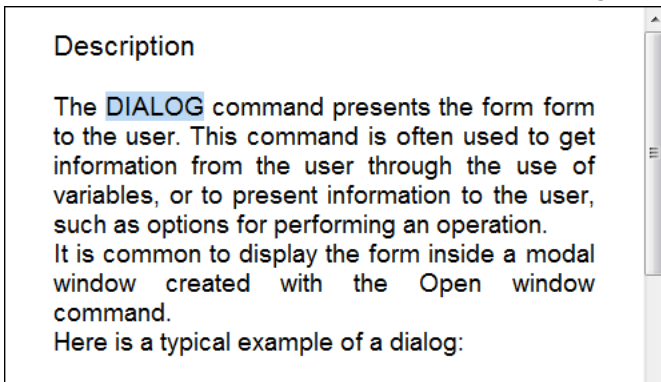
In *attribName* übergeben Sie den Namen des Attributs für das Ziel, in *Attribwert* den neu zu setzenden Wert. Weitere Informationen dazu finden Sie im Abschnitt **4D Write Pro Attribute**.

Sie können beliebig viele Paare *attribName* / *attribWert* übergeben.

Hinweis: Müssen Sie mehrere Attribute für dasselbe Ziel setzen, ist es optimaler, einen einzelnen Aufruf von **WP SET ATTRIBUTES** mit allen Attribut/Werte Paaren zu setzen als **WP SET ATTRIBUTES** mehrere Male aufzurufen.

Beispiel 1

In diesem 4D Write Pro Bereich haben Sie ein Wort ausgewählt:



Führen Sie folgenden Code aus:

```
$range:=WP Get selection(*;"WParea") //ausgewählten Bereich erhalten

// Schattenversatz für das gewählte Wort in pt setzen
WP SET ATTRIBUTES($range;wk text shadow offset;1)
//Füllen für dieses Wort setzen
WP SET ATTRIBUTES($range;wk padding;1)
//einen Rahmen mit 10 pt setzen
WP SET ATTRIBUTES($range;wk border style;wk solid;wk border width;10)
//Rahmenfarbe definieren
WP SET ATTRIBUTES($range;wk border color;"blau";wk border color bottom;"#command_500FA9A";wk border color
right;"#command_500FA9A")
```

erhalten Sie dieses Ergebnis:

Description

The **DIALOG** command presents the form form to the user. This command is often used to get information from the user through the use of variables, or to present information to the user, such as options for performing an operation.

It is common to display the form inside a modal window created with the Open window command.

Here is a typical example of a dialog:

Beispiel 2

Dieses Beispiel zeigt die Verwendung der Konstanten `wk inside` und `wk outside`:

```
$wpRange:=WP Get selection(writeProdoc)
WP SET ATTRIBUTES($wpRange;wk border style+wk inside;wk dotted)
WP SET ATTRIBUTES($wpRange;wk border style+wk outside;wk solid)
WP SET ATTRIBUTES($wpRange;wk border color+wk outside;"#command_500FA9A")
```

Wurde der ganze Inhalt ausgewählt, erhalten Sie dieses Ergebnis:

```
.....
Lorem ipsum dolor sit amet, consectetur adipiscing elit. Sed non risus.
Suspendisse lectus tortor, dignissim sit amet, adipiscing nec, ultricies sed,
dolor.
.....
Cras elementum ultrices diam. Maecenas ligula massa, varius a, semper
congue, euismod non, mi. Proin porttitor, orci nec nonummy molestie, enim
est eleifend mi, non fermentum diam nisl sit amet erat. Duis semper.
.....
Duis arcu massa, scelerisque vitae, consequat in, pretium a, enim.
Pellentesque congue. Ut in risus volutpat libero pharetra tempor. Cras
vestibulum bibendum augue. Praesent egestas leo in pede. Praesent blandit
odio eu enim.
.....
Pellentesque sed dui ut augue blandit sodales. Vestibulum ante ipsum primis
in faucibus orci luctus et ultrices posuere cubilia Curae; Aliquam nibh. Mauris
ac mauris sed pede pellentesque fermentum. Maecenas adipiscing ante non
diam sodales hendrerit.
```

Beispiel 3

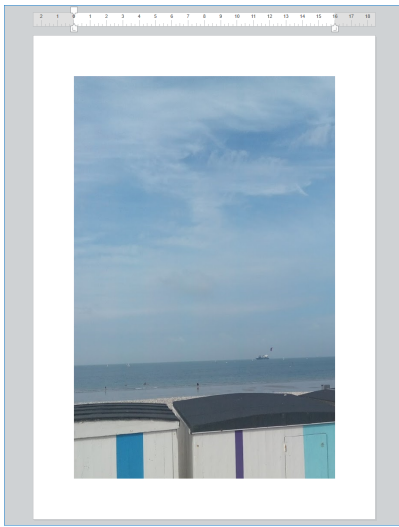
Sie wollen ein Hintergrundbild für das Dokument setzen:

```
C_OBJECT(WParea)
WParea:=WP New

READ PICTURE FILE("C:\\Pictures\\boats.jpg";$picture)

WP SET ATTRIBUTES(WParea;wk background image;$picture)
```

Das Ergebnis ist:



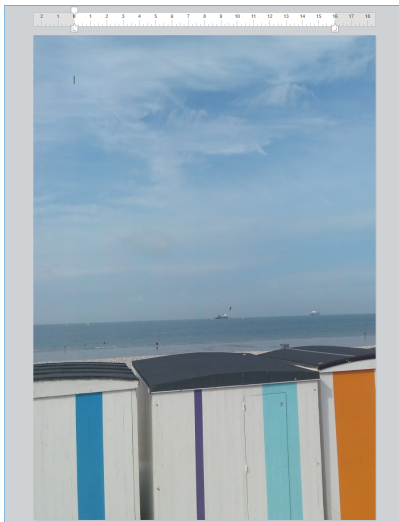
Sie wollen ein Hintergrundbild setzen, das den gesamten druckbaren Bereich ausfüllt:

```
C_OBJECT(WParea)
WParea:=WP New

READ PICTURE FILE("C:\\Pictures\\boats.jpg";$picture)

WP SET ATTRIBUTES(WParea;wk background image;$picture)
WP SET ATTRIBUTES(WParea;wk background clip;wk paper box)
WP SET ATTRIBUTES(WParea;wk background origin;wk paper box)
```

Das Ergebnis ist:



Hinweis: Die Konstante wk_paper_box ist nur für Dokumente und Abschnitte verwendbar.

Beispiel 4

Tabulatoren mit verschiedenen Abständen und verschiedenen führenden Zeichen setzen:

```
C_OBJECT(WParea)

ARRAY TEXT(_position;0)
ARRAY LONGINT(_type;0)
ARRAY TEXT(_leadings;0)

APPEND TO ARRAY(_position;"3cm")
APPEND TO ARRAY(_type;wk left)
APPEND TO ARRAY(_leadings;".")

APPEND TO ARRAY(_position;"7.5cm")
APPEND TO ARRAY(_type;wk right)
APPEND TO ARRAY(_leadings;".")

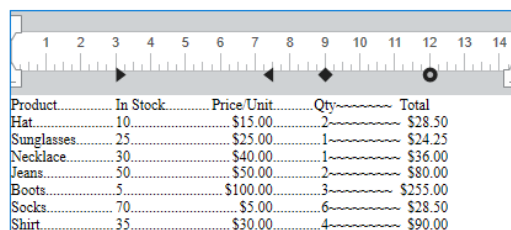
APPEND TO ARRAY(_position;"9cm")
APPEND TO ARRAY(_type;wk center)
APPEND TO ARRAY(_leadings;".")
```

APPEND TO ARRAY(_position;"1 2cm")
APPEND TO ARRAY(_type;wk decimal)
APPEND TO ARRAY(_leadings;"~")

APPEND TO ARRAY(_position;"2cm")
APPEND TO ARRAY(_type;wk left)
APPEND TO ARRAY(_leadings;"")

\$range:=WP Get selection(WParea)
WP SET ATTRIBUTES(\$range;wk tab stop offsets;_position)
WP SET ATTRIBUTES(\$range;wk tab stop types;_type)
WP SET ATTRIBUTES(\$range;wk tab stop leadings;_leadings)

Das Ergebnis ist:



Product	In Stock	Price Unit	Qty	Total
Hat	10	\$15.00	2	\$28.50
Sunglasses	25	\$25.00	1	\$24.25
Necklace	30	\$40.00	1	\$36.00
Jeans	50	\$50.00	2	\$80.00
Boots	5	\$100.00	3	\$255.00
Socks	70	\$5.00	6	\$28.50
Shirt	35	\$30.00	4	\$90.00

WP SET FRAME

WP SET FRAME ({ * ; } wpArea ; RahmenWahl)

Parameter	Typ	Beschreibung
*	Operator	➔ Mit Stern: wpArea ist Name des Formularobjekts (String) ➔ Ohne Stern: wpArea ist ein Objektfeld oder Variable
wpArea	String	➔ Name des Formularobjekts (mit *) oder 4D Write Pro Objektvariable oder Feld (ohne *)
RahmenWahl	Lange Ganzzahl	➔ Rahmen zum Setzen des Cursors

Beschreibung

Der Befehl **WP SET FRAME** setzt den Cursor in den Rahmen, der innerhalb des 4D Write Pro Bereichs *wpArea* im Parameter *RahmenWahl* angegeben ist..

Mit dem optionalen Parameter *** geben Sie an, dass *wpArea* ein Formularobjekt Name (String) ist. Ohne diesen Parameter geben Sie an, dass *wpArea* eine 4D Write Pro Objektvariable oder Feld ist.

In *RahmenWahl* können Sie eine der folgenden Konstanten unter dem Thema **4D Write Pro Konstanten** übergeben:

Konstante	Typ	Wert	Kommentar
wk body	Lange Ganzzahl	0	Hauptteilrahmen des Bereichs
wk current page footer	Lange Ganzzahl	2	
wk current page header	Lange Ganzzahl	1	
wk current section default footer	Lange Ganzzahl	10	Hauptfußteil des Bereichs (nur verfügbar, wenn kein Fußteil für Unterbereich aktiv ist)
wk current section default header	Lange Ganzzahl	9	Hauptkopfteil des Bereichs (nur verfügbar, wenn kein Kopfteil für Unterbereich aktiv ist)
wk current section first footer	Lange Ganzzahl	4	Fußteil der ersten Seite des Bereichs
wk current section first header	Lange Ganzzahl	3	Kopfteil der ersten Seite des Bereichs
wk current section left footer	Lange Ganzzahl	6	Fußteil der linken Seite(n) des Bereichs
wk current section left header	Lange Ganzzahl	5	Kopfteil der linken Seite(n) des Bereichs
wk current section right footer	Lange Ganzzahl	8	Fußteil der rechten Seite(n) des Bereichs
wk current section right header	Lange Ganzzahl	7	Kopfteil der rechten Seite(n) des Bereichs

- Ist wk current page header oder wk current page footer in *RahmenWahl* übergeben, setzt **WP SET FRAME** den Fokus auf den ersten gewählten Kopf- oder Fußteil der Seite, unabhängig vom Typ des Unterabschnitts (enthält er keinen Kopf- oder Fußteil, führt der Befehl nichts aus).
- Ist eine der Konstanten wk current section[...] in *RahmenWahl* übergeben, setzt **WP SET FRAME** den Fokus auf den angegebenen Kopf- oder Fußteil des Unterabschnitts des ersten gewählten Bereichs (enthält er keinen Kopf- oder Fußteil, führt der Befehl nichts aus).
- **WP SET FRAME** führt nichts aus, wenn der Rahmen in *RahmenWahl* für den Auswahlbereich nicht existiert (z.B. wk current section left header, wenn es in *wpArea* keine Unterscheidung zwischen linken und rechten Seiten gibt).

Beispiel

Den Cursor auf den linken Fußteil der aktuellen Auswahl setzen:

```
WP SET FRAME(*;"WPArea";wk current section left footer)
```

WP SET TEXT

WP SET TEXT (wpRange ; newText ; textPosition {; RangeUpdate})

Parameter	Typ		Beschreibung
wpRange	Objekt	→	Objekt Ausschnitt
newText	Text	→	Text zum Einfügen
textPosition	Zahl	→	Einfügestelle (ersetzen, davor, danach)
RangeUpdate	Zahl	→	Setzt den Inhalt innerhalb des Ausschnitts ein oder nicht

Beschreibung

Der Befehl **WP SET TEXT** fügt den im Parameter *newText* übergebenen Text als reinen Text in ein 4D Write Pro Dokument an einer bestimmten Position ein, die durch die Parameter *wpRange* und *textPosition* festgelegt ist.

Im Parameter *wpRange* übergeben Sie ein Objekt Bereich, um die Position zu bestimmen, an der Text eingefügt werden soll. Der Bereich kann eine explizite Position im 4D Write Pro Dokument sein oder auf der Auswahl eines Benutzers basieren.

Der Parameter *newText* enthält den Text zum Einfügen in *wpRange*. **WP SET TEXT** fügt nur reinen Text ein und zwar genau so, wie er geschrieben wurde, aber ohne Style-Tags.

Im Parameter *textPosition* lässt sich die Position des eingefügten Textes über einen der folgenden Selektoren festlegen:

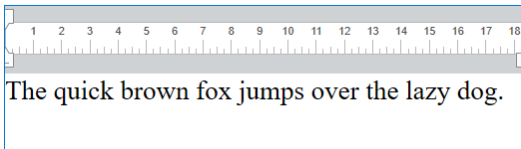
Konstante	Kommentar
wk append	Fügt Inhalt am Ende des Ausschnitts ein
wk prepend	Fügt Inhalt am Anfang des Ausschnitts ein
wk replace	Ersetzt Inhalt des Ausschnitts

Im optionalen Parameter *UpdateAusschnitt* können Sie über folgende Konstanten festlegen, ob der eingefügte Inhalt in den aktualisierten Ausschnitt übernommen wird oder nicht:

Konstante	Kommentar
wk exclude from range	Eingefügter Inhalt ist nicht in aktualisierten Ausschnitt enthalten
wk include in range	Eingefügter Inhalt ist im aktualisierten Ausschnitt enthalten (Standard)

Beispiel

Sie möchten in diesem Dokument Text hinzufügen:



Vor dem vorhandenen Text:

```
$range:=WP Create range(myDoc;wk start text;wk end text)
WP SET TEXT($range;" Hello world!";wk prepend)
//schreibt " Hello world! The quick brown fox jumps over the lazy dog."
```

Nach dem vorhandenen Text:

```
$range:=WP Create range(myDoc;wk start text;wk end text)
WP SET TEXT($range;" Hello world!";wk append)
//schreibt "The quick brown fox jumps over the lazy dog. Hello world!"
```

Den vorhandenen Text ersetzen:

```
$range:=WP Create range(myDoc;wk start text;wk end text)
WP SET TEXT($range;" Hello world!";wk replace)
//schreibt " Hello world!"
```


WP Table append row

WP Table append row (TabelleRef ; Wert {; Wert2 ; ... ; WertN}) -> Funktionsergebnis

Parameter	Typ	Beschreibung
TabelleRef	Objekt	→ Objekt Tabellenreferenz
Wert	Text, Zahl, Zeit, Datum, Bild	→ Zu setzender Wert/Werte in der Zeile
Funktionsergebnis	Objekt	→ Objekt Zeilenausschnitt

Beschreibung

Die Funktion **WP Table append row** fügt an die Tabelle *TabelleRef* eine Zeile an, füllt sie mit *Wert* und gibt das neue Objekt Zeilenausschnitt zurück.

Diese Funktion fügt die Anzahl Zellen in der Zeile hinzu, die im Parameter *Wert* definiert wurde. Sie können eine beliebige Anzahl Werte unterschiedlichen Typs übergeben; die standardmäßige Zellenausrichtung richtet sich nach dem jeweiligen Typ:

- Text ist links ausgerichtet
- Bilder sind zentriert
- Andere Typen (Zahlen, Datum und Zeit) sind rechts ausgerichtet

Hinweis: Werte vom Typ Array werden nicht unterstützt.

Die Funktion gibt die neue Zeile als Objekt Zeilenausschnitt zurück.

Beispiel 1

Eine leere Tabelle erstellen und mehrere Zeilen mit unterschiedlicher Größe anhängen:

```
C_OBJECT($wpTable;$wpRange;$wpRow1;$wpRow2;$wpRow3)
$wpRange:=WP Create range(WParea;wk start text;wk end text)
$wpTable:=WP Insert table($wpRange;wk append)
$wpRow1:=WP Table append row($wpTable;"Paul";"Smith";25)
$wpRow2:=WP Table append row($wpTable;"John";"Richmond";40)
$wpRow3:=WP Table append row($wpTable;"Mary";"Trenton";18;"New!")
```

Paul	Smith	25	
John	Richmond	40	
Mary	Trenton	18	New!

Beispiel 2

In einer Anwendung Rechnungen eine Tabelle anlegen, die automatisch mit verknüpften Rechnungszeilen gefüllt wird:

```
C_OBJECT($wpTable;$wpRange)

$wpRange:=WP Get range(4DWPArea;wk start text;wk end text)

$wpTable:=WP Insert table($wpRange;wk append) //Tabelle anlegen

// Eine Kopfzeile hinzufügen
$row:=WP Table append row($wpTable;"Name";"Quantity";"Unit Price";"Discount Rate";"Total")
WP SET ATTRIBUTES($row;wk font bold;wk true;wk text align;wk center)

//Einfach auf die Auswahl anwenden
APPLY TO SELECTION([INVOICE_LINES];WP Table append row($wpTable;[INVOICE_LINES]ProductName;[INVOICE_LINES]Quantity;
[INVOICE_LINES]ProductUnitPrice;[INVOICE_LINES]DiscountRate;[INVOICE_LINES]Total))

//Eine Fußzeile hinzufügen
$row:=WP Table append row($wpTable;"Total:";Sum([INVOICE_LINES]Quantity);"";"";Sum([INVOICE_LINES]Total))

//Tabelle formatieren
$range:=WP Table get columns($wpTable;1;5)
WP SET ATTRIBUTES($range;wk width;"80pt")
WP SET ATTRIBUTES($wpTable;wk font size;10)
```

Name	Quantity	Unit Price	Discount Rate	Total
Stylo Bille Granger Bright Yellow	7	20	10	126
Stylo Multifonction Voldermont Green	4	5	4	19,2
Stylo Bille Dumbledore Bright Red	4	5	7	18,6
Stylo Bille Rogue Green	3	35	7	97,65
Stylo Plume Granger Yellow	1	30	6	28,2
Stylo Multifonction Voldermont Red	3	15	0	45
Total:	22			334,65

WP Table get cells

WP Table get cells (TabelleRef ; StartSpalte ; StartZeile {; AnzSpalten {; AnzZeilen}}) -> Funktionsergebnis

Parameter	Typ		Beschreibung
TabelleRef	Objekt	→	Referenz auf Tabelle
StartSpalte	Lange Ganzzahl	→	Position der ersten Spaltenzeile
StartZeile	Lange Ganzzahl	→	Position der ersten Spaltenzeile
AnzSpalten	Lange Ganzzahl	→	Anzahl der zu erhaltenden Spalten
AnzZeilen	Lange Ganzzahl	→	Anzahl der zu erhaltenden Zeilen
Funktionsergebnis	Objekt	↪	Neuer Zellenausschnitt

Beschreibung

Die Funktion **WP Table get cells** gibt ein neues Objekt Zellenausschnitt mit der Auswahl der Zellen von *TabelleRef* zurück, definiert durch *StartSpalte*, *StartZeile*, *AnzSpalten* und *AnzZeilen*.

Die Parameter *StartSpalte* und *StartZeile* definieren die Koordinaten der linken oberen Ecke der ersten zu erhaltenden Zelle. Optional können Sie die Parameter *AnzSpalten* und *AnzZeilen* übergeben, um jeweils die Anzahl Spalten und Zeilen für den Zellenausschnitt anzugeben. Ohne den Parameter *AnzZeilen* wird eine einzelne Zeile zurückgegeben. Werden *AnzSpalten* und *AnzZeilen* weggelassen, enthält der zurückgegebene Ausschnitt eine einzelne Zelle.

Übersteigt die Summe von *StartSpalte* plus *AnzSpalte* bzw. *StartZeile* plus *AnzZeilen* die Anzahl von Spalten/Zeilen in *TabelleRef*, oder ist *StartSpalte* bzw. *StartZeile* größer als die Anzahl Spalten/Zeilen in *TabelleRef*, enthält der zurückgegebene Ausschnitt die max. möglichen Spalten/Zeilen.

Beispiel 1

Die Attribute eines Zellenausschnitts ändern:

```
C_OBJECT($wpTable;$wpRange;$wpRow1;$wpRow2;$wpRow3;$wpRow4;$wpRow5;$cells)
$wpRange:=WP Create range(WParea;wk start text;wk end text)
```

```
$wpTable:=WP Insert table($wpRange;wk append)
$wpRow1:=WP Table append row($wpTable;"Paul";"Smith";25)
$wpRow2:=WP Table append row($wpTable;"John";"Richmond";40)
$wpRow3:=WP Table append row($wpTable;"Mary";"Trenton";18)
$wpRow4:=WP Table append row($wpTable;"Christopher";"Lee";53)
$wpRow5:=WP Table append row($wpTable;"Henry";"Cartier";42)
```

```
$cells:=WP Table get cells($wpTable;2;2;2;2)
WP SET ATTRIBUTES($cells;wk background color;0x00E0E0E0)
WP SET ATTRIBUTES($cells;wk text color;"blue")
WP SET ATTRIBUTES($cells;wk border color;0x00E000E0)
WP SET ATTRIBUTES($cells;wk text align;wk center)
```

Ergebnis:

Paul	Smith	25
John	Richmond	40
Mary	Trenton	18
Christopher	Lee	53
Henry	Cartier	42

Beispiel 2

Einen Zellenausschnitt ab der Zelle "5x7" bis zum Ende der Zeile erhalten:

```
$cellRange:=WP Table get cells(tableRef;5;7;MAXLONG;1)
```

Einen Zellenausschnitt ab der Zelle "5x7" bis zum Ende der Spalte erhalten:

```
$cellRange:=WP Table get cells(tableRef;5;7;1;MAXLONG)
```

Einen Zellenausschnitt ab der Zelle "5x7" bis zum Ende der Tabelle erhalten:

```
$cellRange:=WP Table get cells(tableRef;5;7;MAXLONG;MAXLONG)
```

WP Table get columns

WP Table get columns (TabelleRef ; StartSpalte { ; AnzSpalten }) -> Funktionsergebnis

Parameter	Typ	Beschreibung
TabelleRef	Objekt	➔ Referenz auf Tabelle
StartSpalte	Lange Ganzzahl	➔ Position der ersten Spalte
AnzSpalten	Lange Ganzzahl	➔ Anzahl der zu erhaltenden Spalten
Funktionsergebnis	Objekt	➔ Neuer Tabellenausschnitt mit den gewählten Spalten

Beschreibung

Die Funktion **WP Table get columns** gibt ein neues Objekt Zellausschnitt mit der Auswahl der Spalten von *TabelleRef* zurück, definiert durch *StartSpalte* und *AnzSpalten* (optional).

Der Parameter *StartSpalte* gibt die erste zurückzugebende Tabellenspalte an. Der optionale Parameter *AnzSpalten* enthält die Anzahl der zurückzugebenden Spalten. Ohne *AnzSpalten* wird die einzelne Spalte *StartSpalte* zurückgegeben.

Übersteigt *StartSpalte* plus *AnzSpalten* die Anzahl Spalten in *TabelleRef*, oder ist *StartSpalte* größer als die Anzahl Spalten in *TabelleRef*, enthält der zurückgegebene Bereich die maximal möglichen Spalten.

Beispiel 1

Die Attribute der letzten beiden Spalten einer Tabelle ändern:

```
C_OBJECT($wpTable;$wpRange;$wpRow1;$wpRow2;$wpRow3;$wpRow4;$wpRow5;$cols)
$wpRange:=WP Create range(WParea;wk start text;wk end text)
$wpTable:=WP Insert table($wpRange;wk append)
$wpRow1:=WP Table append row($wpTable;"Paul";"Smith";25)
$wpRow2:=WP Table append row($wpTable;"John";"Richmond";40)
$wpRow3:=WP Table append row($wpTable;"Mary";"Trenton";18)
$wpRow4:=WP Table append row($wpTable;"Christopher";"Lee";53)
$wpRow5:=WP Table append row($wpTable;"Henry";"Cartier";42)

$cols:=WP Table get columns($wpTable;2;2)
WP SET ATTRIBUTES($cols;wk background color;0x00E0E0E0;wk text color;"red";wk height;"36pt";wk text align;wk center)
```

Ergebnis:

Paul	Smith	25
John	Richmond	40
Mary	Trenton	18
Christopher	Lee	53
Henry	Cartier	42

Beispiel 2

Einen Ausschnitt ab der 10. Spalte bis zum Ende erhalten:

```
WP Table get columns(tableRef;10;MAXLONG)
```

WP Table get rows

WP Table get rows (TabelleRef ; StartZeile {; AnzZeilen}) -> Funktionsergebnis

Parameter	Typ		Beschreibung
TabelleRef	Objekt	→	Referenz auf Tabelle
StartZeile	Lange Ganzzahl	→	Position der ersten Zeile
AnzZeilen	Lange Ganzzahl	→	Anzahl der zu erhaltenden Zeilen
Funktionsergebnis	Objekt	→	Neuer Ausschnitt mit den gewählten Zeilen

Beschreibung

Die Funktion **WP Table get rows** gibt ein neues Objekt Zellausschnitt mit der Auswahl der Zeilen von *TabelleRef* zurück, definiert durch *StartZeile* und *AnzZeilen* (optional).

Der Parameter *StartZeile* gibt die erste zu erhaltende Tabellenzeile an. Der optionale Parameter *AnzZeilen* gibt an, wieviel Zeilen zurückgegeben werden sollen. Ohne *AnzZeilen* wird nur die Zeile *StartZeile* zurückgegeben.

Übersteigt *StartZeile* plus *AnzZeilen* die Anzahl Zeilen in *TabelleRef*, oder ist *StartZeile* größer als die Anzahl Zeilen in *TabelleRef*, enthält der zurückgegebene Bereich die maximal möglichen Zeilen.

Beispiel 1

Für die beiden ersten Zeilen einer Tabelle eine Hintergrundfarbe setzen und den Rahmen der dritten Zeile ändern:

```
C_OBJECT($wpTable;$wpRange;$wpRow1;$wpRow2;$wpRow3;$wpRow4;$wpRow5;$rows;$rows2)
$wpRange:=WP Create range(WParea;wk start text;wk end text)
```

```
$wpTable:=WP Insert table($wpRange;wk append)
$wpRow1:=WP Table append row($wpTable;"Paul";"Smith";25)
$wpRow2:=WP Table append row($wpTable;"John";"Richmond";40)
$wpRow3:=WP Table append row($wpTable;"Mary";"Trenton";18)
$wpRow4:=WP Table append row($wpTable;"Christopher";"Lee";53)
$wpRow5:=WP Table append row($wpTable;"Henry";"Cartier";42)
```

```
$rows:=WP Table get rows($wpTable;1;2)
WP SET ATTRIBUTES($rows;wk background color;0x00E0F0FF)
$rows2:=WP Table get rows($wpTable;3)
WP SET ATTRIBUTES($rows2;wk border style;wk solid)
WP SET ATTRIBUTES($rows2;wk border width;4)
```

Paul	Smith	25
John	Richmond	40
Mary	Trenton	18
Christopher	Lee	53
Henry	Cartier	42

Beispiel 2

Einen Ausschnitt ab der 10. Zeile bis zum Ende erhalten:

```
WP Table get rows(tableRef;10;MAXLONG)
```

WP USE PAGE SETUP

WP USE PAGE SETUP (wpDok)

Parameter	Typ		Beschreibung
wpDok	Objekt	→	Name des 4d Write Pro Dokuments

Beschreibung

Der Befehl **WP USE PAGE SETUP** ändert die aktuellen Seiteneinstellungen des Druckers gemäß den 4D Write Pro Attributen für Seitengröße und Ausrichtung. Dieser Befehl sollte direkt vor **WP PRINT** aufgerufen werden, um die aktuellen Seiteneinstellungen des Druckers mit den Seiteneinstellungen des 4D Write Pro Dokuments zu synchronisieren.

Andere Einstellungen werden über den 4D Befehl **PRINT SETTINGS** definiert. Die aktuellen Druckeinstellungen gelten für die gesamte 4D Sitzung.

Beispiel

Vor dem Drucken des Dokuments Größe und Ausrichtung auf die im Dokument gesetzten Werte setzen:

```
WP USE PAGE SETUP(writeProDoc)
```


Konstante	Typ	Wert	Kommentar
wk 4D Write Pro layout	Lange Ganzzahl	0	Standardmäßige 4D Write Pro Ausgabe mit ein paar spezifischen Stilattributen
wk 4wp	Lange Ganzzahl	4	Das 4D Write Pro Dokument wird in einem nativen Archivformat gesichert (gezipptes HTML und Bilder in einem eigenen Ordner abgelegt). 4D spezifische Tags sind enthalten, 4D Ausdrücke werden nicht berechnet. Dieses Format ist besonders geeignet, um 4D Write Pro Dokumente ohne Verluste auf der Festplatte zu sichern und archivieren.
wk anchor all	Lange Ganzzahl	-1	Wert Bildanker für wk anchor section und wk anchor page
wk anchor embedded	Lange Ganzzahl	-2	Wert eingebundener Bildanker für wk anchor section und wk anchor page
wk anchor first page	Lange Ganzzahl	-2	Wert Bildanker für wk anchor page
wk anchor horizontal align	Zeichenkette	anchorHorizontalAlign	<p>Definiert die horizontale Ausrichtung eines Bildes in Bezug zum Ursprung (siehe wk anchor origin). Mögliche Werte:</p> <ul style="list-style-type: none"> wk left - Ausrichtung links wk center - Ausrichtung mittig (<i>nicht kompatibel mit HTML, Bilder erscheinen nicht im Web</i>) wk right - Ausrichtung rechts
wk anchor horizontal offset	Zeichenkette	anchorHorizontalOffset	<p>Definiert den horizontalen Versatz eines Bildes als CSS Positionsangabe (Text wie z.B. "19px") oder Lange Ganzzahl (z.B. 19) von wk layout unit. Wert kann in cm, pt oder pixel sein. Mögliche Werte:</p> <ul style="list-style-type: none"> Linker oder rechter Rand der Seite in Bezug auf wk anchor horizontal align Linker oder rechter Rand des Hauptteils im eingebundenen Modus (wenn wk anchor section = wk anchor embedded) <p>Standardwert = 0</p> <p>Definiert die Layoutposition eines Bildes in Bezug auf den Text auf der Seite. Mögliche Werte:</p>
wk anchor layout	Zeichenkette	anchorLayout	<ul style="list-style-type: none"> wk behind text - Bild ist verankert und hinter dem Text wk in front of text - Bild ist verankert und vor dem Text wk inline with text - Bild ist im Textfluss (Standard für Bilder, eingefügt mit WP INSERT PICTURE). Nur-Lesen Attribute (Bilder im Textfluss sind nicht per Programmierung in verankerte Bilder konvertierbar)
wk anchor left page	Lange Ganzzahl	-3	Wert Bildanker für wk anchor page
wk anchor origin	Zeichenkette	anchorOrigin	<p>Definiert, ob das Bild mit der Seite, dem Kopfteil oder Fußteil verankert ist. Mögliche Werte:</p> <ul style="list-style-type: none"> wk paper box (Standard) - Bild wird mit dem Seitenrand verankert wk header box - Bild wird mit dem Kopfteil des Dokuments verankert. Ist der Kopfteil ausgeblendet, erscheint kein Bild. wk footer box - Bild wird mit dem Fußteil des Dokuments verankert. Ist der Fußteil ausgeblendet, erscheint kein Bild. <p>Dieser Selektor wird im eingebundenen Modus ignoriert.</p> <p>Definiert den Seitenindex oder Seitentyp, mit dem ein Bild verankert ist. Mögliche Werte:</p>
wk anchor page	Zeichenkette	anchorPage	<ul style="list-style-type: none"> wk anchor all - verankert ein Bild mit allen Seiten des Bereichs, definiert durch wk anchor section wk anchor embedded - verankert ein Bild mit dem Hauptteil des Dokuments im eingebundenen Modus (Bild ist nur im eingebundenen Modus sichtbar) wk anchor first page - verankert ein Bild mit der ersten Seite des Bereichs, definiert durch wk anchor section wk anchor left page - verankert ein Bild mit den linken Seiten des Bereichs, definiert durch wk anchor section wk anchor right page - verankert ein Bild mit den rechten Seiten des Bereichs, definiert durch wk anchor section die Zahl (Longint >= 0) gibt die Seite an, mit der das Bild verankert werden soll. In diesem Fall gilt wk anchor section = wk anchor embedded. Bei Verankerung mit einer einzelnen Seite wird die Verankerung mit Bereich ignoriert.
wk anchor right page	Lange Ganzzahl	-4	Wert Bildanker für wk anchor page

Hinweis: Bilder im Seitenmodus werden in Browsern nicht angezeigt.

Konstante	Typ	Wert	Kommentar
wk anchor section	Zeichenkette	anchorSection	<p>Definiert den Bereichsindex oder Bereichstyp, mit dem das Bild verankert wird. Mögliche Werte:</p> <ul style="list-style-type: none"> • wk_anchor_all (Standard) - verankert ein Bild mit allen Bereichen in einem Dokument (Bild ist nur im Seitenmodus sichtbar) • wk_anchor_embedded - verankert ein Bild mit dem Hauptteil des Dokuments im eingebundenen Modus (Bild ist nur im eingebundenen Modus sichtbar) • Zahl (Longint >= 1) gibt den Bereich an, mit dem das Bild verankert werden soll. (Bild ist nur im Seitenmodus sichtbar) <p>Hinweis: Bilder im Seitenmodus erscheinen nicht in Browsern.</p> <p>Definiert die vertikale Ausrichtung eines Bildes relativ zum Ursprung (siehe wk_anchor_origin). Mögliche Werte:</p> <ul style="list-style-type: none"> • wk_top - Ausrichtung oben • wk_center - Ausrichtung mittig (<i>nicht kompatibel mit HTML, Bilder werden nicht in Browsern angezeigt</i>) • wk_bottom - Ausrichtung unten <p>Definiert die vertikale Position eines Bildes als CSS Positionsangabe (Text wie z.B. "19px") oder Lange Ganzzahl (z.B. 19) von wk_layout_unit. Wert kann in cm, pt oder pixel sein. Mögliche Werte:</p> <ul style="list-style-type: none"> • Oberer, unterer Rand oder Mitte der Seite (siehe wk_anchor_origin) oder • Oberer, unterer Rand oder Mitte des Hauptteils im eingebundenen Modus (wenn wk_anchor_section = wk_anchor_embedded). <p>Standardwert = 0.</p>
wk append	Lange Ganzzahl	2	Fügt Inhalt am Ende des Ausschnitts ein
wk armenian	Lange Ganzzahl	19	Traditionell armenischer Stil für Nummerierung (Wert für wk_list_style_type)
wk author	Zeichenkette	author	<p>Verwendung: Dokumente Standardwert: keiner Mögliche Werte: Eigene Zeichenkette</p>
wk auto	Lange Ganzzahl	0	Der Wert der Eigenschaft (Konstante), auf die sie angewendet wird, wird automatisch an Inhalt oder Kontext des Elements angepasst. Gibt den Zeichenbereich des Hintergrunds an. Mögliche Werte:
wk background clip	Zeichenkette	backgroundClip	<ul style="list-style-type: none"> • wk_border_box (Standard): Der Hintergrund wird bis zum äußeren Rand des Rahmens gezeichnet • wk_content_box: Der Hintergrund wird innerhalb des Inhaltsteils gezeichnet • wk_padding_box: Der Hintergrund wird bis zum äußeren Rand der Einrückung gezeichnet (oder bis zum inneren Rand des Rahmens, falls vorhanden) • wk_paper_box: Hintergrund wird bis zum Papierrand gezeichnet (nur für Dokument oder Abschnitte) <p>Gibt die Hintergrundfarbe eines Elements an. Mögliche Werte:</p> <ul style="list-style-type: none"> • CSS Farbe ("command_5010101" oder "command_5FFFFFF" oder "rot"). • 4D Farbwert als lange Ganzzahl (siehe) • Array Lange Ganzzahl mit einem Element pro R, G, B Komponente (0-255) <p>Standard für Dokumente ist "command_5FFFFFF" und wk_transparent, oder "transparent" für Absätze und Bilder.</p>
wk background color	Zeichenkette	backgroundColor	

Konstante	Typ	Wert	Kommentar
wk background height	Zeichenkette	backgroundHeight	<p>Gibt die vertikale Größe des Hintergrundbildes an. Mögliche Werte:</p> <ul style="list-style-type: none"> • wk_auto (Standard): Das Hintergrundbild richtet sich nach seiner Höhe • wk_contain: Skaliert das Hintergrundbild mit Beibehalten der Proportionen so klein wie möglich, so dass es komplett in den Inhaltsteil passt. Diese Option ändert auch den Wert der anderen Größenattribute. • wk_cover: Skaliert das Hintergrundbild mit Beibehalten der Proportionen so groß wie möglich, so dass der Hintergrundbereich komplett ausgefüllt ist. Evtl. sind Teile des Hintergrundbildes abgeschnitten. Diese Option ändert auch den Wert der anderen Größenattribute. • Definierte Größe: Vertikale Größe des Hintergrundbildes als Zahl oder String: <ul style="list-style-type: none"> ◦ Zahl: Größe in wk_layout_unit ◦ String: CSS String mit zusammengezogenem Wert und Einheit, z.B.: 12pt für 12 Punkt oder 1.5cm für 1,5 Zentimeter. Mindestwert: Opt, Maximumwert: 10000pt. Relativer Wert (in %) wird unterstützt. <p>Gibt Referenz für Hintergrundbild an. Kann jedes gültige Bild wie 4D Bildvariable oder Ausdruck sein:</p>
wk background image	Zeichenkette	backgroundImage	<ul style="list-style-type: none"> • Zurückgegebener Wert (WP GET ATTRIBUTES): Wurde das Bild über eine Netzwerk URL definiert, wird das Zielbild zurückgegeben, wenn es bereits geladen ist, sonst ein leeres Bild. <p>Verwenden Sie wk_background_image_url, wenn Sie Bilder über URLs oder lokale URIs verwalten wollen.</p> <p>Hintergrundbild, definiert über eine URL (String). Kann Netzwerk URL oder Daten URI und absolut oder relativ zur Strukturdatei sein.</p>
wk background image url	Zeichenkette	backgroundImageURL	<ul style="list-style-type: none"> • Zurückgegebener Wert (WP GET ATTRIBUTES): Netzwerk URL oder Daten URI. Kann sich von der ursprünglichen URL unterscheiden, wenn das Bild keine Referenz über die Netzwerk URL hat (nur Netzwerk URLs werden beibehalten). Für lokale Datei URLs wird der Bild-Stream selbst im Dokument beibehalten und so ist die zurückgegebene URL eine Daten URI mit dem Bild-Stream codiert in base64. <p>Verwenden Sie wk_background_image, wenn Sie Hintergrundbilder als Bildausdrücke verwalten wollen.</p>
wk background origin	Zeichenkette	backgroundOrigin	<p>Gibt die Positionierung des Hintergrundbildes an. Mögliche Werte:</p> <ul style="list-style-type: none"> • wk_padding_box (Standard): Das Hintergrundbild startet im Viereck der Einrückung (oder innerer Rahmenrand) • wk_border_box: Das Hintergrundbild startet im Viereck des Rahmens (äußerer Rand) • wk_content_box: Das Hintergrundbild startet im Viereck des Inhalts • wk_paper_box: Hintergrundbild startet am Papierrand (nur für Dokument oder Abschnitte)
wk background position horizontal	Zeichenkette	backgroundPositionHorizontal	<p>Gibt die horizontale Startposition des Hintergrundbildes an. Mögliche Werte:</p> <ul style="list-style-type: none"> • wk_left (Standard): Das Hintergrundbild beginnt horizontal auf der linken Seite des Elements • wk_center: Das Hintergrundbild beginnt horizontal in der Mitte des Elements • wk_right: Das Hintergrundbild beginnt horizontal auf der rechten Seite des Elements
wk background position vertical	Zeichenkette	backgroundPositionVertical	<p>Gibt die vertikale Startposition des Hintergrundbildes an. Mögliche Werte:</p> <ul style="list-style-type: none"> • wk_top (Standard): Das Hintergrundbild beginnt vertikal oben im Element • wk_middle: Das Hintergrundbild beginnt vertikal in der Mitte des Elements • wk_bottom: Das Hintergrundbild beginnt vertikal unten im Element

Konstante	Typ	Wert	Kommentar
wk background repeat	Zeichenkette	backgroundRepeat	<p>Gibt an, ob und wie das Hintergrundbild wiederholt wird. Mögliche Werte:</p> <ul style="list-style-type: none"> wk_repeat (Standard): Das Hintergrundbild wird vertikal und horizontal wiederholt wk_no_repeat: Das Hintergrundbild wird nicht wiederholt wk_repeat x: Das Hintergrundbild wird nur horizontal wiederholt wk_repeat y: Das Hintergrundbild wird nur vertikal wiederholt <p>Gibt die horizontale Größe des Hintergrundbildes an. Mögliche Werte:</p> <ul style="list-style-type: none"> wk_auto (Standard): Das Hintergrundbild richtet sich nach seiner Breite wk_contain: Skaliert das Hintergrundbild mit Beibehalten der Proportionen so groß wie möglich, so dass es komplett in den Inhaltsbereich passt. Diese Option ändert auch den Wert der anderen Größenattribute. wk_cover: Skaliert das Hintergrundbild mit Beibehalten der Proportionen so groß wie möglich, so dass der Hintergrundbereich komplett ausgefüllt ist. Evtl. sind Teile des Hintergrundbildes abgeschnitten. Diese Option ändert auch den Wert der anderen Größenattribute. Definierte Größe: Horizontale Größe des Hintergrundbildes als Zahl oder String: <ul style="list-style-type: none"> Zahl: Größe in wk layout unit String: CSS String mit zusammengesetztem Wert und Einheit, z.B.: 12pt für 12 Punkt, oder 1,5cm für 1,5 Zentimeter. Mindestwert: 0pt, Maximumwert: 10000pt. Relativer Wert (in %) wird unterstützt.
wk background width	Zeichenkette	backgroundWidth	
wk bar	Lange Ganzzahl	4	Fügt eine vertikale Leiste an der Tabulatorposition ein (Wert für wk tab stop types)
wk baseline	Lange Ganzzahl	4	Richtet die Grundlinie des Elements an der Grundlinie des übergeordneten Elements aus (Wert für wk vertical align)
wk behind text	Lange Ganzzahl	0	Wert für wk anchor layout
wk body	Lange Ganzzahl	0	Hauptteilrahmen des Bereichs
wk border box	Lange Ganzzahl	0	Hintergrund wird bis zur Rahmenbox gezeichnet (Wert für wk background clip)
wk border color	Zeichenkette	borderColor	<p>Setzt Farbe für alle vier Ränder. Mögliche Werte:</p> <ul style="list-style-type: none"> CSS Farbe ("#command_5010101" oder "#command_5FFFFFFF" oder "rot"). 4D Farbwert als Lange Ganzzahl (siehe Befehl OBJECT SET COLOR) Array Lange Ganzzahl mit einem Element pro R, G, B Komponente (0-255) <p>Standard ist "#command_5000000" (bei Wert als String). Bei mehreren Farben gibt WP GET ATTRIBUTES einen leeren String zurück.</p> <p>Setzt Farbe des unteren Rands. Mögliche Werte:</p> <ul style="list-style-type: none"> CSS Farbe ("#command_5010101" oder "#command_5FFFFFFF" oder "rot"). 4D Farbwert als Lange Ganzzahl (siehe Befehl OBJECT SET COLOR) Array Lange Ganzzahl mit einem Element pro R, G, B Komponente (0-255) <p>Setzt Farbe des linken Rands. Mögliche Werte:</p> <ul style="list-style-type: none"> CSS Farbe ("#command_5010101" oder "#command_5FFFFFFF" oder "rot"). 4D Farbwert als Lange Ganzzahl (siehe Befehl OBJECT SET COLOR) Array Lange Ganzzahl mit einem Element pro R, G, B Komponente (0-255) <p>Setzt Farbe des rechten Rands. Mögliche Werte:</p> <ul style="list-style-type: none"> CSS Farbe ("#command_5010101" oder "#command_5FFFFFFF" oder "rot"). 4D Farbwert als Lange Ganzzahl (siehe Befehl OBJECT SET COLOR) Array Lange Ganzzahl mit einem Element pro R, G, B Komponente (0-255)
wk border color bottom	Zeichenkette	borderColorBottom	
wk border color left	Zeichenkette	borderColorLeft	
wk border color right	Zeichenkette	borderColorRight	

Konstante	Typ	Wert	Kommentar
wk border color top	Zeichenkette	borderColorTop	<p>Setzt Farbe des oberen Rands. Mögliche Werte:</p> <ul style="list-style-type: none"> • CSS Farbe ("command_5010101" oder "command_5FFFFFF" oder "rot"). • 4D Farbwert als Lange Ganzzahl (siehe Befehl OBJECT SET COLOR) • Array Lange Ganzzahl mit einem Element pro R, G, B Komponente (0-255) <p>Gibt abgerundeten Rand an. Mögliche Werte:</p> <ul style="list-style-type: none"> • wk_none (Standard): Rand hat keine abgerundeten Ecken • Radiuswert als Zahl oder String: <ul style="list-style-type: none"> ◦ Zahl: Radius in wk_layout unit. ◦ String: CSS String mit zusammengesetztem Wert und Einheit. z.B: 12pt für 12 Punkt oder 1.5cm für 1.5 Zentimeter. <p>Gibt Stil für alle vier Ränder an. Mögliche Werte:</p> <ul style="list-style-type: none"> • wk_none (Standard): Kein Rand • wk_hidden: genauso wie wk_none, außer bei Auflösung Randkonflikt • wk_solid: durchgezogener Rand • wk_dotted: gepunkteter Rand • wk_dashed: gestrichelter Rand • wk_double: doppelter Rand • wk_groove: 3D Rahmen vertieft (aktuelle Auswirkung richtet sich nach der Rahmenfarbe) • wk_ridge: 3D Rahmen erhoben (aktuelle Auswirkung richtet sich nach der Rahmenfarbe) • wk_inset: 3D Rahmen nach innen (aktuelle Auswirkung richtet sich nach der Rahmenfarbe)
wk border radius	Zeichenkette	borderRadius	<p>Gibt Stil für unteren Rand an. Mögliche Werte:</p> <ul style="list-style-type: none"> • wk_none (Standard): Kein Rand • wk_hidden: genauso wie wk_none, außer bei Auflösung Randkonflikt • wk_solid: durchgezogener Rand • wk_dotted: gepunkteter Rand • wk_dashed: gestrichelter Rand • wk_double: doppelter Rand • wk_groove: 3D Rand vertieft (aktuelle Auswirkung richtet sich nach der Randfarbe) • wk_ridge: 3D Rand erhoben (aktuelle Auswirkung richtet sich nach der Randfarbe) • wk_inset: 3D Rand nach innen (aktuelle Auswirkung richtet sich nach der Randfarbe) <p>Gibt Stil für linken Rand an:</p> <ul style="list-style-type: none"> • wk_none (Standard): Kein Rand • wk_hidden: genauso wie wk_none, außer bei Auflösung Randkonflikt • wk_solid: durchgezogener Rand • wk_dotted: gepunkteter Rand • wk_dashed: gestrichelter Rand • wk_double: doppelter Rand • wk_groove: 3D Rand vertieft (aktuelle Auswirkung richtet sich nach der Randfarbe) • wk_ridge: 3D Rand erhoben (aktuelle Auswirkung richtet sich nach der Randfarbe) • wk_inset: 3D Rand nach innen (aktuelle Auswirkung richtet sich nach der Randfarbe)
wk border style	Zeichenkette	borderStyle	<p>Gibt Stil für rechten Rand an:</p> <ul style="list-style-type: none"> • wk_none (Standard): Kein Rand • wk_hidden: genauso wie wk_none, außer bei Auflösung Randkonflikt • wk_solid: durchgezogener Rand • wk_dotted: gepunkteter Rand • wk_dashed: gestrichelter Rand • wk_double: doppelter Rand • wk_groove: 3D Rand vertieft (aktuelle Auswirkung richtet sich nach der Randfarbe) • wk_ridge: 3D Rand erhoben (aktuelle Auswirkung richtet sich nach der Randfarbe) • wk_inset: 3D Rand nach innen (aktuelle Auswirkung richtet sich nach der Randfarbe)
wk border style bottom	Zeichenkette	borderStyleBottom	
wk border style left	Zeichenkette	borderStyleLeft	
wk border style right	Zeichenkette	borderStyleRight	

Konstante	Typ	Wert	Kommentar
			Gibt Randstil oben an. Mögliche Werte: <ul style="list-style-type: none"> wk_none (Standard): Kein Rand wk_hidden: genauso wie wk_none, außer bei Auflösung Randkonflikt wk_solid: durchgezogener Rand wk_dotted: gepunkteter Rand wk_dashed: gestrichelter Rand wk_double: doppelter Rand wk_groove: 3D Rand vertieft (aktuelle Auswirkung richtet sich nach der Randfarbe) wk_ridge: 3D Rand erhoben (aktuelle Auswirkung richtet sich nach der Randfarbe) wk_inset: 3D Rand nach innen (aktuelle Auswirkung richtet sich nach der Randfarbe)
wk border style top	Zeichenkette	borderStyleTop	
			Gibt Breite für alle vier Ränder an. Sie müssen zuvor den Randstil festlegen. Mögliche Werte: <ul style="list-style-type: none"> Breite als Zahl oder String: <ul style="list-style-type: none"> Zahl: Breite in wk_layout unit. String: CSS String mit zusammengezogenem Wert und Einheit, z.B: 12pt für 12 Punkt oder 1.5cm für 1,5 Zentimeter Standardwert: 2pt
wk border width	Zeichenkette	borderWidth	
			Gibt Breite für den unteren Rand an. Mögliche Werte: <ul style="list-style-type: none"> Breite als Zahl oder String: <ul style="list-style-type: none"> Zahl: Breite in wk_layout unit. String: CSS String mit zusammengezogenem Wert und Einheit, z.B: 12pt für 12 Punkt oder 1.5cm für 1.5 Zentimeter Standardwert: 2pt
wk border width bottom	Zeichenkette	borderWidthBottom	
			Gibt Breite für linken Rand an. Mögliche Werte: <ul style="list-style-type: none"> Breite als Zahl oder String: <ul style="list-style-type: none"> Zahl: Breite in wk_layout unit. String: CSS String mit zusammengezogenem Wert und Einheit, z.B.: 12pt für 12 Punkt oder 1.5cm für 1.5 Zentimeter Standardwert: 2pt
wk border width left	Zeichenkette	borderWidthLeft	
			Gibt Breite für rechten Rand an. Mögliche Werte: <ul style="list-style-type: none"> Breite als Zahl oder String: <ul style="list-style-type: none"> Zahl: Breite in wk_layout unit. String: CSS String mit zusammengezogenem Wert und Einheit, z.B: 12pt für 12 Punkt oder 1.5cm für 1.5 Zentimeter Standardwert: 2pt
wk border width right	Zeichenkette	borderWidthRight	
			Gibt Breite für oberen Rand an. Mögliche Werte: <ul style="list-style-type: none"> Breite als Zahl oder String: <ul style="list-style-type: none"> Zahl: Breite in wk_layout unit. String: CSS String mit zusammengezogenem Wert und Einheit, z.B: 12pt für 12 Punkt oder 1.5cm für 1.5 Zentimeter Standardwert: 2pt
wk border width top	Zeichenkette	borderWidthTop	
wk bottom	Lange Ganzzahl	1	Setzt Position des Hintergrundbildes (Wert für wk_background position v) oder Unterkante des Elements, das nach dem untersten Element der Ausrichtung ausgerichtet wird (Wert für wk_vertical align).
wk capitalize	Lange Ganzzahl	1	Wandelt das erste Zeichen jedes Worts in Großbuchstaben um (Wert für wk_text transform)
wk cell count	Zeichenkette	cellCount	Gesamtanzahl der Zellen in der Reihe. Wertetyp: Lange Ganzzahl (Wert für wk_type table row)
wk center	Lange Ganzzahl	2	Zentriert Text oder Bild (Wert für wk_background position h , wk_text align bzw. wk_tab stop types)
wk circle	Lange Ganzzahl	11	Verwendet Symbol Kreis (Wert für wk_list style type)
wk cjk ideographic	Lange Ganzzahl	24	Verwendet volle ideographische Zahlen (Wert für wk_list style type) (Def: Ideographie = Ideenschrift)
wk club	Lange Ganzzahl	27	Verwendet Symbol Clubs (Wert für wk_list style type)
wk column break	Lange Ganzzahl	3	Spaltenumbruch: Fügt einen Spaltenumbruch ein

Konstante	Typ	Wert	Kommentar
wk column count	Zeichenkette	columnCount	(verfügbar für Tabellen, Dokumente und Ausschnitte) Anzahl der Spalten. Wert: Lange Ganzzahl <ul style="list-style-type: none"> Für eine Tabelle: Attribut nur lesbar Für ein Dokument oder einen Ausschnitt: Attribut im Lese-/Schreibmodus. Standardwert=1 (einzelne Spalte). Maximumwert=20
wk column rule color	Zeichenkette	columnRuleColor	Farbe der vertikalen Spaltenregel. Mögliche Werte: <ul style="list-style-type: none"> eine CSS-Farbe ("#command_5010101" oder "#command_5FFFFFFF" oder "rot"). Standard ist "#command_5000000" (schwarz) ein 4D-Farbwert vom Typ Lange Ganzzahl (siehe Befehl OBJECT SET COLOR) ein Longint-Array, das ein Element für jede R-, G-, B-Komponente (0-255) enthält
wk column rule style	Zeichenkette	columnRuleStyle	Stil der vertikalen Spaltenregel. Mögliche Werte: <ul style="list-style-type: none"> wk_none (Standard): keine Regel wk_hidden: identisch mit wk_none, außer beim Auflösen von Regelkonflikten wk_solid: Regel durchgezogen wk_dotted: Regel gepunktet wk_dashed: Regel gestrichelt wk_double: Regel doppelt wk_groove: Regel 3D vertieft (der tatsächliche Effekt hängt von der Regelfarbe ab) wk_ridge: Regel 3D erhoben (der tatsächliche Effekt hängt von der Regelfarbe ab) wk_inset: Regel 3D nach innen (der tatsächliche Effekt hängt von der Regelfarbe ab)
wk column rule width	Zeichenkette	columnRuleWidth	Breite der vertikalen Spaltenregel. Mögliche Werte: <ul style="list-style-type: none"> Zahl: Breite in wk layout unit. String: Wert für Breite und Einheit zusammengezogen. (<i>Beispiel</i>: "12pt" für 12 Punkt oder "1.5cm" für 1,5 Zentimeter) Standardwert="2.5pt"
wk column spacing	Zeichenkette	columnSpacing	(nur für Dokumente oder Abschnitte) Abstand zwischen zwei Spalten. Mögliche Werte: <ul style="list-style-type: none"> Zahl: Breite in wk layout unit String: Wert und Einheit der Breite zusammengezogen. (<i>e.g.</i>: "12pt" für 12 Punkt oder "1.5cm" für 1,5 Zentimeter). Standardwert="12pt"
wk column width	Zeichenkette	columnWidth	(Nur für Dokumente oder Abschnitte) Nur lesbares Attribut. Aktuelle Breite für jede Spalte, d.h. berechnete Breite basierend auf tatsächlicher Seitenbreite, Seitenrändern, Spaltenanzahl und Spaltenabstand. Für das Dokument wird die standardmäßige Spaltenbreite des Abschnitts verwendet, kann also von der tatsächlichen Spaltenbreite des Abschnitts/der Abschnitte abweichen, wenn einige Attribute in einem Abschnitt überschrieben werden. Mögliche Werte: <ul style="list-style-type: none"> Zahl: Breite in wk layout unit. String: Wert für Breite und Einheit zusammengezogen. (<i>Beispiel</i>: "12pt" für 12 Punkt oder "1.5cm" für 1,5 Zentimeter)
wk company	Zeichenkette	company	Gibt eine dem Dokument zugeordnete Firma an (String)
wk contain	Lange Ganzzahl	-1	Skaliert Bild auf die größten Ausmaße, so dass Breite und Höhe in den Inhaltsteil passen (Wert für wk background size h bzw. wk background size v)
wk content box	Lange Ganzzahl	2	Hintergrund an Inhaltsbox abgeschnitten (Wert für wk background clip) oder Hintergrundbild startet an der oberen linken Ecke des Inhalts (Wert für wk background origin)
wk cover	Lange Ganzzahl	-2	Skaliert Hintergrundbild auf die kleinsten Ausmaße, so dass der Hintergrundteil komplett damit ausgefüllt ist (Wert für wk background size h bzw. wk background size v)
wk current page footer	Lange Ganzzahl	2	
wk current page header	Lange Ganzzahl	1	

Konstante	Typ	Wert	Kommentar
wk current section default footer	Lange Ganzzahl	10	Hauptfußteil des Bereichs (nur verfügbar, wenn kein Fußteil für Unterbereich aktiv ist)
wk current section default header	Lange Ganzzahl	9	Hauptkopfteil des Bereichs (nur verfügbar, wenn kein Kopfteil für Unterbereich aktiv ist)
wk current section first footer	Lange Ganzzahl	4	Fußteil der ersten Seite des Bereichs
wk current section first header	Lange Ganzzahl	3	Kopfteil der ersten Seite des Bereichs
wk current section left footer	Lange Ganzzahl	6	Fußteil der linken Seite(n) des Bereichs
wk current section left header	Lange Ganzzahl	5	Kopfteil der linken Seite(n) des Bereichs
wk current section right footer	Lange Ganzzahl	8	Fußteil der rechten Seite(n) des Bereichs
wk current section right header	Lange Ganzzahl	7	Kopfteil der rechten Seite(n) des Bereichs
wk custom	Lange Ganzzahl	29	Verwendet eigene Marker (Wert für wk list style type)
wk dashed	Lange Ganzzahl	3	Verwendet gestrichelte Linie (Wert für wk text linethrough style bzw. wk text underline style)
wk date creation	Zeichenkette	dateCreation	Gibt Erstellungsdatum des Dokuments zurück. Dieser Wert ist im Lesemodus und lässt sich nicht setzen.
wk date modified	Zeichenkette	dateModified	Gibt letztes Änderungsdatum des Dokuments zurück. Dieser Wert ist im Lesemodus und lässt sich nicht setzen.
wk decimal	Lange Ganzzahl	3	Ausrichtung nach Dezimalzeichen (Wert für wk tab stop types) oder verwendete Zahlen (Wert für wk list style type)
wk decimal greek	Lange Ganzzahl	28	Griechische Nummerierung verwenden (Wert für wk list style type)
wk decimal leading zero	Lange Ganzzahl	13	Dezimalzahlen werden mit führenden Nullen aufgefüllt (Wert für wk list style type)
wk default	Lange Ganzzahl	-1	Standardwert der Eigenschaft (Konstante) wird verwendet.
wk diamond	Lange Ganzzahl	26	Verwendet Symbol Raute (Wert für wk list style type)
wk direction	Zeichenkette	direction	Gibt Textausrichtung für Absatz an. Mögliche Werte: <ul style="list-style-type: none"> wk left to right (Standard) wk right to left
wk disc	Lange Ganzzahl	10	Verwendet Symbol gefüllter Kreis (Wert für wk list style type)
wk dotted	Lange Ganzzahl	2	Gepunktete Linie oder Rand (Wert für wk border style , wk text linethrough style bzw. wk text underline style)
wk double	Lange Ganzzahl	4	Doppelte Linie oder Rand (Wert für wk border style , wk text linethrough style bzw. wk text underline style)
wk dpi	Zeichenkette	dpi	Für interne Pixel verwendete DPI <->Punktkonvertierung (Ganzzahl). Immer 96 (Nur-Lesen)
wk end	Zeichenkette	end	(Nur lesbares Attribut für Ausschnitt). Ausschnitt Ende Versatz. Wert: Lange Ganzzahl
wk end text	Lange Ganzzahl	0	Setzt Dokumentende als Ende des Textbereichs
wk exclude from range	Lange Ganzzahl	1	Eingefügter Inhalt wird nicht in aktualisierten Ausschnitt eingefügt
wk expressions as source	Lange Ganzzahl	1	Der ursprüngliche String der Referenz auf 4D Ausdrücke wird zurückgegeben.
wk expressions as space	Lange Ganzzahl	2	Jede Referenz wird als nicht-umbrechendes Leerzeichen zurückgegeben.
wk expressions as value	Lange Ganzzahl	0	Referenzen auf 4D Ausdrücke werden in ihrer ausgewerteten Form zurückgegeben. (Standard, wenn die Option nicht übergeben ist).
wk false	Lange Ganzzahl	0	

Konstante	Typ	Wert	Kommentar
wk first column	Zeichenkette	firstColumn	(Nur lesbares Attribut für Ausschnitt) Nummer der ersten Spalte im Ausschnitt. Wert: Lange Ganzzahl
wk first page	Lange Ganzzahl	1	
wk first row	Zeichenkette	firstRow	(Nur lesbares Attribut für Ausschnitt) Nummer der ersten Zeile im Ausschnitt. Wert: Lange Ganzzahl
wk font	Zeichenkette	font	Gibt kompletten Schriftnamen mit Stilarten an, wie vom Befehl FONT STYLE LIST zurückgegeben. Bei einem ungültigen Schriftnamen führt der Befehl nichts aus. Standardwert: "Times New Roman". Gibt die Textstärke an (richtet sich nach verfügbaren Schriftstilen). Mögliche Werte: <ul style="list-style-type: none"> wk true um ausgewählte Zeichen auf den Schriftstil Fett zu setzen; mit dem Befehl WP GET ATTRIBUTES wird wk true zurückgegeben, wenn mindestens ein ausgewähltes Zeichen in Fettschrift ist. wk false (Standard) um den Schriftstil Fett - falls vorhanden - von den ausgewählten Zeichen zu entfernen; mit dem Befehl WP GET ATTRIBUTES wird wk false zurückgegeben, wenn keins der ausgewählten Zeichen in Fettschrift ist.
wk font bold	Zeichenkette	fontBold	
wk font family	Zeichenkette	fontFamily	Gibt den Namen der Schriftfamilie an wie von wk font definiert. Standardwert: "Times New Roman". Der Befehl WP GET ATTRIBUTES gibt einen leeren String zurück, wenn die ausgewählten Zeichen unterschiedliche Schriftfamilien-Eigenschaften enthalten. Gibt den Stil Kursiv für Text an (richtet sich nach den Schriftstilen). Mögliche Werte: <ul style="list-style-type: none"> wk true, um ausgewählte Zeichen auf Kursiv- oder Schrägschrift zu setzen; der Befehl WP GET ATTRIBUTES gibt wk true zurück, wenn mindestens ein ausgewähltes Zeichen Kursiv- oder Schrägschrift unterstützt. wk false (Standard), um in gewählten Zeichen Kursiv- oder Schrägschrift - wenn vorhanden - zu entfernen; der Befehl gibt wk false zurück, wenn keines der gewählten Zeichen in Kursiv- oder Schrägschrift ist.
wk font italic	Zeichenkette	fontItalic	
wk font size	Zeichenkette	fontSize	Gibt die Schriftgröße für Text an. Mögliche Werte (nur in Punkt): <ul style="list-style-type: none"> Wert Zahl (Standard = 12) CSS String mit zusammengesetztem Wert und Einheit, z.B.: 12pt für 12 Punkt.
wk footer box	Lange Ganzzahl	1	Wert Bildanker für wk anchor origin
wk freeze expressions	Lange Ganzzahl	64	Friert Ausdrücke beim Einfügen ein
wk georgian	Lange Ganzzahl	20	Verwendet traditional Georgian Nummerierung (Wert für wk list style type)
wk groove	Lange Ganzzahl	6	Vertiefen 3D Rahmen verwenden (Wert für wk border style)
wk header box	Lange Ganzzahl	0	Wert Bildanker für wk anchor origin
wk hebrew	Lange Ganzzahl	21	Verwendet traditionnel hebräische Nummerierung (Wert für wk list style type) Setzt Höhe des Elements, dazu gehören nicht Einrückung, Ränder oder Rahmen; d.h. die Höhe des Bereichs liegt innerhalb von Einrückung, Rändern und Rahmen des Elements. Mögliche Werte: <ul style="list-style-type: none"> wk auto (Standard): Höhe basiert auf dem Inhalt des Elements Größe wird als Zahl oder Stringwert ausgedrückt: <ul style="list-style-type: none"> Zahl: Größe in wk layout unit. String: CSS String mit zusammengesetztem Wert und Einheit, z.B.: 12pt für 12 Punkt oder 1.5cm für 1,5 Zentimeter. Mindestwert: Opt, Maximumwert: 10.000pt.
wk height	Zeichenkette	height	Das Attribut wk height wird von wk min height überschrieben (wenn definiert).
wk hidden	Lange Ganzzahl	5	Kein Rahmenstil. Gleich mit Keiner (kein Rahmen). Hat jedoch Vorrang vor allen anderen konkurrierenden Rahmen. (Wert für wk border style)
wk hiragana	Lange Ganzzahl	22	Traditionelle Hiragana Nummerierung (Wert für wk list style type)

Konstante	Typ	Wert	Kommentar
wk hollow square	Lange Ganzzahl	25	Verwendet Symbol leeres Quadrat (Wert für wk list style type)
wk html debug	Lange Ganzzahl	1	Formatierter HTML Code ("pretty print"), leichter zu debuggen
wk html wysiwyg	Lange Ganzzahl	1	In dieser Ausgabe werden die erweiterten 4D Write Pro Attribute entfernt, die nicht mit allen Browsern kompatibel sind.
wk id	Zeichenkette	id	(Nur lesbares Attribut) ID des Elements. Wert: String Gibt ein Bild an. Mögliche Werte: <ul style="list-style-type: none"> Bild URL (String). Kann absolut oder relativ zur Strukturdatei sein. Bildvariable oder Feld.
wk image	Zeichenkette	image	Zurückgegebener Wert (WP GET ATTRIBUTES): URI (Netzwerk URL oder Daten URI). Kann sich von der ursprünglichen URL unterscheiden, wenn das Bild keine Referenz über die Netzwerk URL hat (nur Netzwerk URLs werden beibehalten). Für lokale Datei URLs wird der Bild-Stream selbst im Dokument beibehalten und so ist die zurückgegebene URL eine Daten URI mit dem Bild-Stream codiert in base64.
wk image alternate text	Zeichenkette	imageAlternateText	Gibt Ersatztext für Bild an, wenn das Bild sich nicht anzeigen lässt. Gibt ein Bild, definiert über eine URL (String), an. Kann Netzwerk URL oder Daten URI und absolut oder relativ zur Strukturdatei sein. <ul style="list-style-type: none"> Zurückgegebener Wert (WP GET ATTRIBUTES): Netzwerk URL oder Daten URI. Kann sich von der ursprünglichen URL unterscheiden, wenn das Bild keine Referenz über die Netzwerk URL hat (nur Netzwerk URLs werden beibehalten). Für lokale Datei URLs wird der Bild-Stream selbst im Dokument beibehalten und so ist die zurückgegebene URL eine Daten URI mit dem Bild-Stream codiert in base64.
wk image url	Zeichenkette	imageURL	Verwenden Sie wk image , wenn Sie Bilder als Bildausdrücke verwalten wollen.
wk in front of text	Lange Ganzzahl	1	Wert Bildanzeige für wk anchor layout
wk include in range	Lange Ganzzahl	0	Eingefügter Inhalt wird in aktualisierten Ausschnitt eingefügt (Standard)
wk inherit style from paragraph	Lange Ganzzahl	32	Eingefügter Inhalt erbt Schriftstil vom Standard Schriftstil des Absatzes
wk inline with text	Lange Ganzzahl	-1	Wert Bildanzeige für wk anchor layout
wk inset	Lange Ganzzahl	8	Verwendet 3D Rahmen nach innen (Wert für wk border style)
wk inside	Zeichenkette	Inside	Gibt bei mehreren Absätzen im ausgewählten Bereich an, dass das Attribut nur für die entsprechende innere Eigenschaft des Absatzes gilt (nicht außerhalb). Gilt nur für Attribute für Außenabstand, Innenabstand und Ränder und muss zum angegebenen Attribut hinzugefügt werden. Siehe Beispiel 2 des Befehls WP SET ATTRIBUTES .
wk justify	Lange Ganzzahl	5	Nur für 4D Write Pro Bereiche verfügbar
wk katakana	Lange Ganzzahl	23	Verwendet traditional Katakana Nummerierung (Wert für wk list style type)
wk keep paragraph styles	Lange Ganzzahl	128	Stilarten des Zielabsatzes beibehalten

Konstante	Typ	Wert	Kommentar
			<p>Gibt die standardmäßige Maßeinheit für das Dokument an, wenn ein Wert als Zahl gesetzt oder erhalten wird. Gibt die Einheit für das Lineal und Attribute für Ausmaße an, wie z.B. wk width - mit Ausnahme von wk font size, wk border width und seine Variationen, wk border radius und wk text shadow offset, für die als Einheit immer Punkt gilt. Mögliche Werte:</p> <ul style="list-style-type: none"> wk unit cm (Standard): Zentimeter wk unit pt: Punkt wk unit px: Pixel wk unit percent (nur für wk line height und wk background size h / wk background size v) wk unit mm: Millimeter wk unit inch: Zoll <p>Hinweis: Wird eine vom Lineal nicht unterstützte Einheit ausgewählt, wie z.B. wk unit px oder wk unit percent, verwendet das Lineal die Einheit cm.</p>
wk layout unit	Zeichenkette	userUnit	
wk left	Lange Ganzzahl	0	Richtet Text oder Tabulator linksbündig aus (Wert für wk text align oder wk tab stop types) oder setzt Startposition des Hintergrundbildes (Wert für wk background position h)
wk left page	Lange Ganzzahl	2	
wk left to right	Lange Ganzzahl	0	Verwendet Text/Schreibrichtung links-nach-rechts (Wert für wk direction)
wk line break	Lange Ganzzahl	0	<p>Zeilenumbruch (im gleichen Absatz)</p> <p>Gibt Zeilenabstand an. Mögliche Werte:</p> <ul style="list-style-type: none"> wk normal (Standard): Wert richtet sich nach der Textgröße Höhe wird als Zahl oder Stringwert ausgedrückt: <ul style="list-style-type: none"> Zahl: Höhe in wk layout unit. String: CSS String mit zusammengesetztem Wert und Einheit, z.B.: 12pt für 12 Punkt oder 1.5cm für 1.5 Zentimeter. Auch relativer Wert (Prozent %) wird unterstützt.
wk line height	Zeichenkette	lineHeight	<p>Dem Ausschnitt zugewiesener Hyperlink. Mögliche Werte:</p> <ul style="list-style-type: none"> Absolute Url, z.B. "http://www.4d.com/" Relativer Link, z.B. "/test/page.html" (Link in Bezug auf die Strukturdatei der Anwendung) Leerer String = kein Link
wk link url	Zeichenkette	linkUrl	
wk list auto	Lange Ganzzahl	2147483647	<p>Wendet automatische Stilwerte für Liste an oder stellt sie wieder her.</p> <p>Gibt den kompletten Schriftnamen an, wie vom Befehl FONT STYLE LIST zurückgegeben, um Marker für Listeneintrag anzuzeigen (nicht den Absatztext). Erkennt das System den Schriftnamen nicht, sorgt es für Ersatz. Ist der Schriftname ungültig, führt der Befehl nichts aus. Standardwert: "Times".</p>
wk list font	Zeichenkette	listFont	Gibt den Namen der Schriftfamilie an, definiert über wk list font , um Marker für Listeneintrag anzuzeigen (nicht Absatztext). Standardwert ist "Times New Roman".
wk list font family	Zeichenkette	listFontFamily	Setzt Startwert für geordnete Liste. Mögliche Werte:
wk list start number	Zeichenkette	listStartNumber	<ul style="list-style-type: none"> wk auto (Standard): Startwert richtet sich nach vorigem Listeneintrag, wenn vorhanden Zahlenwert: Startwert <p>Stringformat für Marker des Listeneintrags in Absatzrichtung links-nach-rechts. Wenn definiert, überschreibt er das standardmäßige Stringformat des Markers.</p>
wk list string format LTR	Zeichenkette	listStringFormatLtr	<ul style="list-style-type: none"> Für ungeordnete Liste: String wird als Marker für Listeneintrag verwendet (in der Regel ein einzelnes Zeichen, z.B. "-") Für geordnete Liste: String mit dem Zeichen "#command_5", das als Platzhalter für berechnete Zahlen oder Buchstaben dient. Standard ist "#command_5.", ist z.B. die aktuelle Zahl des Listeneintrags 15 und der Stiltyp für Liste Dezimal, lautet der String des Markers "15."

Konstante	Typ	Wert	Kommentar
wk list string format RTL	Zeichenkette	listStringFormatRtl	<p>Stringformat für Marker des Listeneintrags in Absatzrichtung rechts-nach-links. Wenn definiert, überschreibt er das standardmäßige Stringformat des Markers.</p> <ul style="list-style-type: none"> Für ungeordnete Liste: String wird als Marker für Listeneintrag verwendet (in der Regel ein einzelnes Zeichen, z.B. "-") Für geordnete Liste: String mit dem Zeichen "#command_5", das als Platzhalter für berechnete Zahlen oder Buchstaben dient. Standard ist "#command_5.", ist z.B. die aktuelle Zahl des Listeneintrags 15 und der Stiltyp für Liste Dezimal, lautet der String des Markers "15."
wk list style image	Zeichenkette	listStyleImage	<p>Gibt in einer ungeordneten Liste ein Bild als Marker des Listeneintrags zurück. Mögliche Werte:</p> <ul style="list-style-type: none"> wk none (Standard): Marker des Listeneintrags wird nicht über ein Bild definiert Bild URL für lokale Datei (String). Kann absolut oder relativ zum Ressourcen-Verzeichnis der Anwendung sein. <p>Zurückgegebener Wert (WP GET ATTRIBUTES): URI (Netzwerk URL oder Daten URI). Kann sich von der ursprünglichen URL unterscheiden, wenn das Bild keine Referenz über die Netzwerk URL hat (nur Netzwerk URLs werden beibehalten). Für lokale Datei URLs wird der Bild-Stream selbst im Dokument beibehalten und so ist die zurückgegebene URL eine Daten URI mit dem Bild-Stream codiert in base64.</p> <p>Setzt Bildhöhe für Marker des Listeneintrags. Mögliche Werte:</p> <ul style="list-style-type: none"> wk auto (Standard): Höhe richtet sich nach Bildgröße Definierte Größe: Größe wird als Zahl oder Stringwert ausgedrückt: <ul style="list-style-type: none"> Zahl: Größe in wk layout unit. String: CSS String mit zusammengesetztem Wert und Einheit, z.B.: 12pt für 12 Punkt oder 1.5cm für 1.5 Zentimeter. Mindestwert: 0pt, Maximumwert: 10000pt. <p>Bild als Marker des Listeneintrags in einer ungeordneten Liste, definiert über eine URL (String). Mögliche Werte:</p> <ul style="list-style-type: none"> wk none (Standard): Marker des Listeneintrags wird nicht über ein Bild definiert Netzwerk URL oder Daten URI, absolut oder relativ zur Strukturdatei
wk list style image height	Zeichenkette	listStyleImageHeight	<p>Setzt Bildhöhe für Marker des Listeneintrags. Mögliche Werte:</p> <ul style="list-style-type: none"> wk auto (Standard): Höhe richtet sich nach Bildgröße Definierte Größe: Größe wird als Zahl oder Stringwert ausgedrückt: <ul style="list-style-type: none"> Zahl: Größe in wk layout unit. String: CSS String mit zusammengesetztem Wert und Einheit, z.B.: 12pt für 12 Punkt oder 1.5cm für 1.5 Zentimeter. Mindestwert: 0pt, Maximumwert: 10000pt. <p>Bild als Marker des Listeneintrags in einer ungeordneten Liste, definiert über eine URL (String). Mögliche Werte:</p> <ul style="list-style-type: none"> wk none (Standard): Marker des Listeneintrags wird nicht über ein Bild definiert Netzwerk URL oder Daten URI, absolut oder relativ zur Strukturdatei
wk list style image url	Zeichenkette	imageURL	<p>Zurückgegebener Wert (WP GET ATTRIBUTES): Netzwerk URL oder Daten URI. Kann sich von der ursprünglichen URL unterscheiden, wenn das Bild keine Referenz über die Netzwerk URL hat (nur Netzwerk URLs werden beibehalten). Für lokale Datei URLs wird der Bild-Stream selbst im Dokument beibehalten und so ist die zurückgegebene URL eine Daten URI mit dem Bild-Stream codiert in base64.</p> <p>Verwenden Sie wk list style image, wenn Sie Bilder als Marker des Listeneintrags als Bildausdrücke verwalten wollen.</p>

Konstante Typ Wert

Kommentar

Gibt den Typ für Marker des Listeneintrags an. Mögliche Werte:

Konstante	Typ	Wert	Kommentar
wk list style type	Zeichenkette	listStyleType	<ul style="list-style-type: none"> • wk_disc (Standard) • wk_circle • wk_square • wk_decimal: 1 2 3 • wk_decimal_leading_zero: 01 02 03 • wk_lower_latin: a b c • wk_lower_roman: i ii iii iv • wk_upper_latin: A B C • wk_upper_roman: I II III IV • wk_lower_greek: alpha, beta, gamma, etc. • wk_armenian • wk_georgian • wk_hebrew • wk_hiragana • wk_katakana • wk_cjk_ideographic • wk_hollow_square • wk_diamond • wk_club • wk_decimal_greek • wk_custom: ungeordnete Liste mit "-" als standardmäßigem Marker für Listeneintrag; dieser Stil eignet sich, um einen Marker für Listeneintrag mit wk_list_string_format_LTR oder wk_list_string_format RTL anzupassen, ohne die standardmäßigen Marker für Listeneinträge zu verändern. • wk_none
wk lower greek	Lange Ganzzahl	18	Verwendet klassische griechische Zeichen kleingeschrieben (Wert für wk_list_style_type)
wk lower latin	Lange Ganzzahl	14	Verwendet ASCII Buchstaben kleingeschrieben (Wert für wk_list_style_type)
wk lower roman	Lange Ganzzahl	15	Verwendet römische Ziffern kleingeschrieben (Wert für wk_list_style_type)
wk lowercase	Lange Ganzzahl	2	Wandelt alle Zeichen in Kleinschreibung um (Wert für wk_text_transform)
wk margin	Zeichenkette	margin	<p>Gibt Größe für alle Ränder des Elements an. Mögliche Werte:</p> <ul style="list-style-type: none"> • Größe wird als Zahl oder Stringwert ausgedrückt: <ul style="list-style-type: none"> ◦ Zahl: Größe in wk_layout_unit. ◦ String: CSS String mit zusammengesetztem Wert und Einheit, z.B.: 12pt für 12 Punkt oder 1.5cm für 1,5 Zentimeter. • wk_none (Standard): Kein spezifischer Rand
wk margin bottom	Zeichenkette	marginBottom	<p>Gibt Größe des unteren Rands eines Elements an. Mögliche Werte:</p> <ul style="list-style-type: none"> • Größe wird als Zahl oder Stringwert ausgedrückt: <ul style="list-style-type: none"> ◦ Zahl: Größe in wk_layout_unit. ◦ String: CSS String mit zusammengesetztem Wert und Einheit, z.B.: 12pt für 12 Punkt oder 1.5cm für 1,5 Zentimeter. • wk_none (Standard): Kein spezifischer Rand
wk margin left	Zeichenkette	marginLeft	<p>Gibt Größe des linken Rands eines Elements an. Mögliche Werte:</p> <ul style="list-style-type: none"> • Größe wird als Zahl oder Stringwert ausgedrückt: <ul style="list-style-type: none"> ◦ Zahl: Größe in wk_layout_unit. ◦ String: CSS String mit zusammengesetztem Wert und Einheit, z.B.: 12pt für 12 Punkt oder 1.5cm für 1,5 Zentimeter. • wk_none (Standard): Kein spezifischer Rand
wk margin right	Zeichenkette	marginRight	<p>Gibt Größe des rechten Rands eines Elements an. Mögliche Werte:</p> <ul style="list-style-type: none"> • Größe wird als Zahl oder Stringwert ausgedrückt: <ul style="list-style-type: none"> ◦ Zahl: Größe in wk_layout_unit. ◦ String: CSS String mit zusammengesetztem Wert und Einheit, z.B.: 12pt für 12 Punkt oder 1.5cm für 1,5 Zentimeter. • wk_none (Standard): Kein spezifischer Rand

Konstante	Typ	Wert	Kommentar
wk margin top	Zeichenkette	marginTop	<p>Gibt Größe des oberen Rands eines Elements an. Mögliche Werte:</p> <ul style="list-style-type: none"> Größe wird als Zahl oder Stringwert ausgedrückt: <ul style="list-style-type: none"> Zahl: Größe in wk layout unit. String: CSS String mit zusammengesetztem Wert und Einheit, z.B.: 12pt für 12 Punkt oder 1.5cm für 1,5 Zentimeter. wk none (Standard): Kein spezifischer Rand
wk middle	Lange Ganzzahl	2	<p>Setzt Position des Hintergrundbildes (Wert für wk background position v) oder zentriert Element im übergeordneten Element (Wert für wk vertical align)</p>
wk mime html	Lange Ganzzahl	1	<p>Das 4D Write Pro Dokument wird als standard MIME HTML mit html Dokumenten und Bildern, eingebunden als MIME Teile (codiert in base64), gesichert. Ausdrücke werden berechnet und 4D spezifische Tags werden entfernt. Dieses Format eignet sich besonders, um E-Mails in HTML mit dem Befehl zu senden.</p>
wk min height	Zeichenkette	minHeight	<p>Setzt Mindesthöhe des Elements. Verhindert, dass der Wert von wk height kleiner als wk min height wird. Mögliche Werte:</p> <ul style="list-style-type: none"> wk auto (Standard): Höhe basiert auf dem Inhalt des Elements Größe wird als Zahl oder Stringwert ausgedrückt: <ul style="list-style-type: none"> Zahl: Größe in wk layout unit. String: CSS String mit zusammengesetztem Wert und Einheit, z.B.: 12pt für 12 Punkt oder 1.5cm für 1,5 Zentimeter. Mindestwert: 0pt, Maximumwert: 10.000pt. <p>Der Wert von wk min height überschreibt den Wert von wk height.</p>
wk min width	Zeichenkette	minWidth	<p>Setzt Mindestbreite des Elements. Verhindert, dass der Wert von wk width kleiner als wk min width wird. Mögliche Werte:</p> <ul style="list-style-type: none"> wk auto (Standard): Höhe basiert auf dem Inhalt des Elements Größe wird als Zahl oder Stringwert ausgedrückt: <ul style="list-style-type: none"> Zahl: Größe in wk layout unit. String: CSS String mit zusammengesetztem Wert und Einheit, z.B.: 12pt für 12 Punkt oder 1.5cm für 1,5 Zentimeter. Mindestwert: 0pt, Maximumwert: 10.000pt. <p>Der Wert von wk min width überschreibt den Wert von wk width.</p>
wk mixed	Lange Ganzzahl	-2147483648	<p>Wird bei unterschiedlichen Werten für ein Attribut im Ausschnitt oder Dokument zurückgegeben</p>
wk new line style sheet	Zeichenkette	newLineStyleSheet	<p>Gibt zu verwendende Stilvorlage an, wenn im Absatz eine neue Zeile hinzugefügt wird. Mögliche Werte:</p> <ul style="list-style-type: none"> Name von vorhandener Stilvorlage wk none (Standard)
wk no repeat	Lange Ganzzahl	3	Hintergrundbild wird nicht wiederholt (Wert für wk background repeat)
wk none	Lange Ganzzahl	0	
wk normal	Lange Ganzzahl	0	Standard HTML Code
wk notes	Zeichenkette	notes	Gibt Kommentare über das Dokument an (String).
wk outset	Lange Ganzzahl	9	Verwendet 3D Rahmen nach außen (Wert für wk border style)
wk outside	Zeichenkette	Outside	Gibt bei mehreren Absätzen im ausgewählten Bereich an, dass das Attribut nur für die entsprechende äußere Eigenschaft des Absatzes gilt (nicht innerhalb). Gilt nur für Attribute für Außenabstand, Innenabstand und Ränder und muss zum angegebenen Attribut hinzugefügt werden. Siehe Beispiel 2 des Befehls WP SET ATTRIBUTES .
wk owner	Zeichenkette	owner	(Nur lesbares Attribut) Eigentümer des Ausschnitts/Objekts. Wert: Objekt
wk padding	Zeichenkette	padding	<p>Gibt Einrückung eines Elements an. Mögliche Werte:</p> <ul style="list-style-type: none"> Einrückung wird als Zahl oder Stringwert ausgedrückt: <ul style="list-style-type: none"> Zahl: Größe in wk layout unit. String: CSS String mit zusammengesetztem Wert und Einheit, z.B.: 12pt für 12 Punkt oder 1.5cm für 1,5 Zentimeter. wk none (Standard): Keine spezifische Einrückung

Konstante	Typ	Wert	Kommentar
wk padding bottom	Zeichenkette	paddingBottom	Gibt Einrückung des Elements unten an. Mögliche Werte: <ul style="list-style-type: none"> Einrückung wird als Zahl oder Stringwert ausgedrückt: <ul style="list-style-type: none"> Zahl: Größe in wk layout unit. String: CSS String mit zusammengesetztem Wert und Einheit, z.B.: 12pt für 12 Punkt oder 1.5cm für 1,5 Zentimeter. wk none (Standard): Keine spezifische Einrückung
wk padding box	Lange Ganzzahl	1	Hintergrund wird an Einrückung abgeschnitten (Wert für wk background clip) oder Hintergrundbild startet an der oberen linken Ecke der Einrückung (Wert für wk background origin) Gibt Einrückung linke Seite des Elements an. Mögliche Werte: <ul style="list-style-type: none"> Einrückung wird als Zahl oder Stringwert ausgedrückt: <ul style="list-style-type: none"> Zahl: Größe in wk layout unit. String: CSS String mit zusammengesetztem Wert und Einheit, z.B.: 12pt für 12 Punkt oder 1.5cm für 1,5 Zentimeter. wk none (Standard): Keine spezifische Einrückung
wk padding left	Zeichenkette	paddingLeft	Gibt Einrückung rechte Seite des Elements an. Mögliche Werte: <ul style="list-style-type: none"> Einrückung wird als Zahl oder Stringwert ausgedrückt: <ul style="list-style-type: none"> Zahl: Größe in wk layout unit. String: CSS String mit zusammengesetztem Wert und Einheit, z.B.: 12pt für 12 Punkt oder 1.5cm für 1,5 Zentimeter. wk none (Standard): Keine spezifische Einrückung
wk padding right	Zeichenkette	paddingRight	Gibt Einrückung des Elements oben an. Mögliche Werte: <ul style="list-style-type: none"> Einrückung wird als Zahl oder Stringwert ausgedrückt: <ul style="list-style-type: none"> Zahl: Größe in wk layout unit. String: CSS String mit zusammengesetztem Wert und Einheit, z.B.: 12pt für 12 Punkt oder 1.5cm für 1,5 Zentimeter. wk none (Standard): Keine spezifische Einrückung
wk padding top	Zeichenkette	paddingTop	Gibt Einrückung des Elements oben an. Mögliche Werte: <ul style="list-style-type: none"> Größe wird als Zahl oder Stringwert ausgedrückt: <ul style="list-style-type: none"> Zahl: Größe in wk layout unit. String: CSS String mit zusammengesetztem Wert und Einheit, z.B.: 12pt für 12 Punkt oder 1.5cm für 1,5 Zentimeter. wk none (Standard): Keine spezifische Einrückung
wk page break	Lange Ganzzahl	2	Seitenumbruch: definiert eine neue Seite Page height (in page mode) expressed using a real or string value (Read-only with section or sub-section). Possible values:
wk page height	Zeichenkette	pageHeight	<ul style="list-style-type: none"> Real: Height in wk layout unit. String: CSS string with value and unit concatenated (e.g.: "12pt" for 12 points, or "1.5cm" for 1.5 centimeters). Supported units: pt,cm,mm, inches.
wk paper box	Lange Ganzzahl	3	Hintergrund wird bis zum Rand gezeichnet (Wert für wk background clip) oder Hintergrundbild startet am Rand (Wert für wk background origin). Gilt nur für Dokument und Abschnitte.
wk prepend	Lange Ganzzahl	1	Fügt Inhalt am Anfang des Ausschnitts ein
wk repeat	Lange Ganzzahl	0	Hintergrundbild wird vertikal und horizontal wiederholt (Wert für wk background repeat)
wk repeat x	Lange Ganzzahl	1	Hintergrundbild wird nur horizontal wiederholt (Wert für wk background repeat)
wk repeat y	Lange Ganzzahl	2	Hintergrundbild wird nur vertikal wiederholt (Wert für wk background repeat)
wk replace	Lange Ganzzahl	0	Ersetzt Inhalt des Ausschnitts
wk ridge	Lange Ganzzahl	7	Verwendet erhobenen 3D Rahmen (Wert für wk border style)
wk right	Lange Ganzzahl	1	Richtet Text oder Tabulator rechtsbündig aus (Wert für wk text align oder wk tab stop types) oder setzt Startposition des Hintergrundbildes (Wert für wk background position h)
wk right page	Lange Ganzzahl	3	
wk right to left	Lange Ganzzahl	1	Verwendet rechts-nach-links Ausrichtung (Wert für wk direction)
wk row count	Zeichenkette	rowCount	(Nur lesbares Attribut für Ausschnitt) Gesamtanzahl der Zeilen im Ausschnitt. Wert: Lange Ganzzahl
wk section break	Lange Ganzzahl	1	Abschnittsumbruch: definiert einen neuen Abschnitt
wk semi transparent	Lange Ganzzahl	5	Setzt gedimmte Linie (Wert für wk text linethrough style bzw. wk text underline style)

Konstante	Typ	Wert	Kommentar
wk small uppercase	Lange Ganzzahl	4	Wandelt alle Zeichen in Kapitälchen um (small caps) (Wert für wk text transform)
wk solid	Lange Ganzzahl	1	Verwendet kompakte Linie oder Rahmen (Wert für wk border style , wk text linethrough style bzw. wk text underline style)
wk square	Lange Ganzzahl	12	Verwendet Marker als Quadrat (Wert für wk list style type)
wk start	Zeichenkette	start	(nur lesbares Attribut für Ausschnitt). Ausschnitt Start Versatz. Wert: Lange Ganzzahl
wk start text	Lange Ganzzahl	1	Setzt Dokumentanfang als Start des Textausschnitts
wk style sheet	Zeichenkette	styleSheet	Gibt aktuelle Stilvorlage für das/die ausgewählten Elemente an. Mögliche Werte: <ul style="list-style-type: none"> wk none (Standard) Name von vorhandener Stilvorlage
wk subject	Zeichenkette	subject	Gibt Betreff des Dokuments an (String)
wk subscript	Lange Ganzzahl	6	Richtet Element tiefgestellt aus (Wert für wk vertical align)
wk superscript	Lange Ganzzahl	5	Richtet Element hochgestellt aus (Wert für wk vertical align)
wk tab stop leadings	Zeichenkette	tabStopLeadings	Gibt die führenden Zeichen für Tabulatoren innerhalb Ausschnitten von Absätzen an. Mögliche Werte: <ul style="list-style-type: none"> Text Array: Sortierte Liste mit führenden Zeichen für jeden Tabulator-Versatz, ab dem linken Rand. Das durch den letzten Wert definierte Zeichen wird für alle weiteren im Absatz gesetzten Tabulatoren wiederholt. Text: Ein einzelnes Zeichen wird als führendes Zeichen verwendet. Sind mehrere Zeichen angegeben, wird nur das erste verwendet. Synchronisiert mit den Arrays wk tab stop offsets und wk tab stop types . Passen die Arrays nicht in der Größe, wird das standardmäßige führende Zeichen verwendet (Nichts). Gibt Tabulator-Stops für den Absatz an. Mögliche Werte: <ul style="list-style-type: none"> Array mit Tabulatorwerten: Sortierte Liste mit absoluten Werten vom linken Rand aus. Der vom letzten Wert definierte Versatz wird für jeden weiteren im Absatz eingegebenen Tabulator-Stopp wiederholt. Ist der Versatz größer als die Absatzbreite, springt der Text in die nächste Zeile und startet mit dem ersten Tabulatorwert. Ist ein Wert im Array kleiner als der vorige, wird er ignoriert (außer für den letzten Wert im Array, der ein Standardversatz ist). Skalarer Wert (Standard 35.45pt): Standardversatz für den ganzen Absatz. Der Befehl WP GET ATTRIBUTES gibt den letzten Versatz zurück (Standardversatz in Bezug auf den letzten absoluten Versatz). Hinweis: Sie können Arrays und Skalare nicht im gleichen Aufruf für verschiedene Attribute verwenden. Werte werden über CSS Strings (Standard) oder Zahlen in wk layout unit ausgedrückt. Maximumwert ist 10.000pt. Synchronisiert mit den Arrays wk tab stop leadings und wk tab stop types . Gibt Typ des Tabulator-Stops für den Absatz an. Mögliche Werte: <ul style="list-style-type: none"> Array mit Werten vom Typ Tabulator-Stopp (wenn Tabulator-Stops über ein Array definiert wurden). wk left (Standard): Setzt Text rechts vom Tabulator Stopp wk right: Setzt Text links vom Tabulator Stopp, bis der Abstand zum Tabulator gefüllt ist wk center: Zentriert Text am Tabulator-Stopp wk decimal: Richtet Text nach dem Dezimalzeichen aus wk bar: Vertikale Linie an der angegebenen Position Synchronisiert mit den Arrays wk tab stop offsets und wk tab stop leadings .
wk tab stop offsets	Zeichenkette	tabStopOffsets	
wk tab stop types	Zeichenkette	tabStopTypes	

Konstante	Typ	Wert	Kommentar
wk table align	Zeichenkette	tableAlign	<p>Setzt die horizontale Ausrichtung für ein Element der Tabelle. Mögliche Werte:</p> <ul style="list-style-type: none"> wk left (Standard) wk right wk center
wk table ID	Zeichenkette	tableID	<p>(Nur lesbares Attribut für Ausschnitt) ID der übergeordneten Tabelle des Ausschnitts. Wert: String Gibt die horizontale Textausrichtung im Absatz an. Mögliche Werte:</p> <ul style="list-style-type: none"> wk left (Standard) wk right wk justify wk center
wk text align	Zeichenkette	textAlign	<p>Gibt die Textfarbe an. Mögliche Werte:</p> <ul style="list-style-type: none"> Eine CSS Farbe("#command_5010101" oder "#command_5FFFFFF" oder "rot"). Bei String ist "#command_5000000" Standard Ein 4D Farbwert als Lange Ganzzahl (siehe Befehl OBJECT SET COLOR) Ein Array Lange Ganzzahl mit je einem Element pro R, G, B Komponente (0-255)
wk text color	Zeichenkette	color	<p>Gibt die Einrückung der ersten Zeile im Absatz an. Mögliche Werte:</p> <ul style="list-style-type: none"> Zahl: Größe in wk layout unit. Standard ist 0. String: CSS String mit zusammengezogenem Wert und Einheit, z.B.: 12pt für 12 Punkt oder 1.5cm für 1,5 Zentimeter. Mindestwert: 0pt, Maximumwert: 10000pt.
wk text indent	Zeichenkette	textIndent	<p>Gibt Farbe des durchgestrichenen Textes an. Mögliche Werte:</p> <ul style="list-style-type: none"> CSS Farbe ("#command_5010101" oder "#command_5FFFFFF" oder "rot"). 4D Farbwert als Lange Ganzzahl (siehe Befehl OBJECT SET COLOR) Array Lange Ganzzahl mit einem Element pro R, G, B Komponente (0-255) <p>Standard ist "currentColor" wenn String, oder wk default wenn Lange Ganzzahl.</p>
wk text linethrough color	Zeichenkette	textLinethroughColor	<p>Gibt Stil für durchgestrichenen Text an (wenn vorhanden). Mögliche Werte:</p> <ul style="list-style-type: none"> wk none (Standard): Keine Auswirkung wk solid: Zeichnet durchgezogene Linie im ausgewählten Text wk dotted: Zeichnet gepunktete Linie im ausgewählten Text wk dashed: Zeichnet gestrichelte Linie im ausgewählten Text wk double: Zeichnet doppelte Linie im ausgewählten Text wk semi transparent: gedimmte Linie im ausgewählten Text. Ist mit anderem Linienstil kombinierbar. wk word: Zeichnet die Linie nur bei Wörtern (Leerzeichen werden ausgelassen). Ist mit anderem Linienstil kombinierbar.
wk text linethrough style	Zeichenkette	textLinethroughStyle	<p>Gibt Schattenfarbe des gewählten Textes an. Mögliche Werte:</p> <ul style="list-style-type: none"> CSS Farbe ("#command_5010101" oder "#command_5FFFFFF" oder "rot"). 4D Farbwert als Lange Ganzzahl (siehe Befehl OBJECT SET COLOR) Array Lange Ganzzahl mit einem Element pro R, G, B Komponente (0-255) wk transparent (Standard)
wk text shadow color	Zeichenkette	textShadowColor	<p>Gibt Versatz (offset) des Schatteneffekts an. Mögliche Werte:</p> <ul style="list-style-type: none"> Größenangabe in Punkt, Standardwert: 1pt
wk text shadow offset	Zeichenkette	textShadowOffset	<p>Setzt Buchstaben im Text in Groß- oder Kleinschreibung. Mögliche Werte:</p> <ul style="list-style-type: none"> wk capitalize: Schreibt die Anfangsbuchstaben groß wk lowercase: Schreibt die Buchstaben klein wk uppercase: Schreibt die Buchstaben groß wk small uppercase: Setzt Buchstaben in Kapitälchen wk none (Standard): Keine Umwandlung
wk text transform	Zeichenkette	textTransform	

Konstante	Typ	Wert	Kommentar
wk text underline color	Zeichenkette	textUnderlineColor	<p>Gibt Farbe des unterstrichenen Textes an. Mögliche Werte:</p> <ul style="list-style-type: none"> • CSS Farbe ("command_5010101" oder "command_5FFFFFF" oder "rot"). • 4D Farbwert als Lange Ganzzahl (siehe Befehl OBJECT SET COLOR) • Array Lange Ganzzahl mit einem Element pro R, G, B Komponente (0-255) <p>Standard ist "currentColor" wenn String, oder <u>wk default</u> wenn Lange Ganzzahl.</p> <p>Gibt Stil für unterstrichenen Text an (wenn vorhanden). Mögliche Werte:</p> <ul style="list-style-type: none"> • <u>wk none</u> (Standard): Keine Unterstreichung • <u>wk solid</u>: Zeichnet durchgezogene Linie im ausgewählten Text • <u>wk dotted</u>: Zeichnet gepunktete Linie im ausgewählten Text • <u>wk dashed</u>: Zeichnet gestrichelte Linie im ausgewählten Text • <u>wk double</u>: Zeichnet doppelte Linie im ausgewählten Text • <u>wk semi transparent</u>: gedimmte Linie im ausgewählten Text. Ist mit anderem Linienstil kombinierbar. • <u>wk word</u>: Zeichnet die Linie nur bei Wörtern (Leerzeichen werden ausgelassen). Ist mit anderem Linienstil kombinierbar.
wk text underline style	Zeichenkette	textUnderlineStyle	<p>Gibt Dokumenttitel (String) an. Standard ist "Neues 4D Write Pro Dokument"</p>
wk top	Lange Ganzzahl	0	Setzt Position des Hintergrundbildes (Wert für <u>wk background position v</u>) oder richtet Element an Oberkante des größten Elements aus (Wert für <u>wk vertical align</u>)
wk transparent	Lange Ganzzahl	-1	Setzt Farbe auf transparent (Wert für <u>wk background color</u> oder <u>wk text shadow color</u>)
wk true	Lange Ganzzahl	1	(Nur lesbares Attribut für Ausschnitt) Typ des 4D Write Pro Objekts. Mögliche Werte:
wk type	Zeichenkette	type	<ul style="list-style-type: none"> • <u>wk type default</u>: Typ Standardausschnitt • <u>wk type paragraph</u>: Typ Absatzausschnitt • <u>wk type image</u>: Verankertes Bild • <u>wk type container</u>: z.B. Kopf- oder Fußteil • <u>wk type table</u>: Referenz Tabelle • <u>wk type table row</u>: Referenz Tabellenzeile • <u>wk type table cell</u>: Referenz Tabellenzelle • <u>wk type table column</u>: Referenz Tabellenspalte (nur für Spaltenausschnitt) • <u>wk type body</u>: Referenz Hauptteil
wk type body	Lange Ganzzahl	8	Referenz Hauptteil (Wert für <u>wk type</u>)
wk type container	Lange Ganzzahl	3	z.B. Kopf- oder Fußteil (Wert <u>wk type</u>)
wk type default	Lange Ganzzahl	0	Standardtyp Ausschnitt (Wert für <u>wk type</u>)
wk type image	Lange Ganzzahl	2	Verankertes Bild (Wert für <u>wk type</u>)
wk type image anchored	Lange Ganzzahl	200	Referenz verankertes Bild für Dokument Objekte
wk type image inline	Lange Ganzzahl	100	Referenz Bild im Textfluss für Dokument Objekte
wk type paragraph	Lange Ganzzahl	1	Ausschnitttyp Absatz (Wert für <u>wk type</u>)
wk type table	Lange Ganzzahl	4	Referenz Tabelle (Wert für <u>wk type</u>)
wk type table cell	Lange Ganzzahl	6	Referenz Tabellenzelle (Wert für <u>wk type</u>)
wk type table column	Lange Ganzzahl	7	Referenz Spalte - nur für Spaltenausschnitt (Wert für <u>wk type</u>)
wk type table row	Lange Ganzzahl	5	Referenz Tabellenzeile (Wert für <u>wk type</u>)
wk unit cm	Zeichenkette	cm	Verwendet Zentimeter als Einheit (Wert für <u>wk layout unit</u>)
wk unit inch	Zeichenkette	in	Verwendet Zoll als Einheit (Wert für <u>wk layout unit</u>)

Konstante	Typ	Wert	Kommentar
wk unit mm	Zeichenkette	mm	Verwendet Millimeter als Einheit (Wert für wk layout unit)
wk unit percent	Zeichenkette	%	Verwendete Einheit ist Prozent (Wert für wk layout unit)
wk unit pt	Zeichenkette	pt	Verwendete Einheit ist Punkt (Wert für wk layout unit)
wk unit px	Zeichenkette	px	Verwendete Einheit ist Pixel (Wert für wk layout unit)
wk upper latin	Lange Ganzzahl	16	Verwendet großgeschriebene ASCII Buchstaben (Wert für wk list style type)
wk upper roman	Lange Ganzzahl	17	Verwendet großgeschriebene römische Zahlen (Wert für wk list style type)
wk uppercase	Lange Ganzzahl	3	Setzt alle Zeichen in Großschreibung (Wert für wk text transform)
wk value unit not percentage	Lange Ganzzahl	-100000	Wird an den Befehl WP GET ATTRIBUTES zurückgegeben, wenn der aktuelle Wert der Einheit nicht Prozent ist und eine String Variable übergeben wurde (Ergebnis ist ungültig).
wk value unit percentage	Lange Ganzzahl	-100001	Wird an den Befehl WP GET ATTRIBUTES zurückgegeben, wenn der aktuelle Wert der Einheit Prozent ist und eine numerische Variable übergeben wurde (Ergebnis ist gültig).
wk version	Zeichenkette	version	Gibt die interne 4DWP Version des Dokuments zurück (Zahl). Diese Zahl lässt sich mit dem Befehl WP GET ATTRIBUTES nur lesen; sie kann nicht gesetzt werden. Setzt die vertikale Ausrichtung eines Elements. Verwendbar für Zeichen, Absätze, Bilder, Tabellen, Tabellenzeilen sowie Tabellenspalten/zellen. Mögliche Werte: <ul style="list-style-type: none"> wk baseline (Standard): Richtet die Grundlinie eines Elements nach der Grundlinie des übergeordneten Elements aus wk top: Richtet die Oberkante des Elements nach der Oberkante des größten Elements in der Reihe aus wk bottom: Richtet die Unterkante des Elements nach dem untersten Element in der Reihe aus wk middle: Element wird in die Mitte des übergeordneten Elements gesetzt wk superscript: Richtet das Element aus, als ob es hochgestellt wäre wk subscript: Richtet das Element aus, als ob es tiefgestellt wäre Für Zeichen haben wk top und wk bottom denselben Effekt wie wk baseline . Für Absätze haben wk baseline , wk superscript und wk subscript denselben Effekt wie wk top . Für Tabellen, Tabellenzeilen und Tabellenspalten/zellen werden nur die Werte wk top , wk bottom und wk middle unterstützt.
wk vertical align	Zeichenkette	verticalAlign	
wk web page complete	Lange Ganzzahl	2	Endung .htm oder .html. Das Dokument wird als Standard HTML gesichert, seine Ressourcen werden getrennt gesichert. 4D Tags werden entfernt und Ausdrücke berechnet. Dieses Format ist besonders geeignet, um ein 4D Write Pro Dokument in einem Web Browser anzuzeigen.
wk web page html 4D	Lange Ganzzahl	3	Das 4D Write Pro Dokument wird als HTML mit spezifischen 4D Tags gesichert; jeder Ausdruck wird als nicht-umgebrochener Bereich eingefügt. Da dieses Format verlustlos ist, eignet es sich zum Speichern in einem Textfeld. Setzt Breite des Elements. Mögliche Werte: <ul style="list-style-type: none"> wk auto (Standard): Breite basiert auf dem Inhalt des Elements Größe wird als Zahl oder Stringwert ausgedrückt: <ul style="list-style-type: none"> Zahl: Größe in wk layout unit. String: CSS String mit zusammengesetztem Wert und Einheit, z.B.: 12pt für 12 Punkt oder 1.5cm für 1,5 Zentimeter. Mindestwert: 0pt, Maximumwert: 10.000pt. Der Wert von wk min width überschreibt den Wert von wk width .
wk width	Zeichenkette	width	
wk word	Lange Ganzzahl	6	Nur Wörter unterstreichen (ohne Leerzeichen) (Wert für wk text linethrough style bzw. wk text underline style)

☰ 4D Write Pro Attribute

Über 4D Write Pro Attribute können Sie alle grafischen Aspekte von Text und Bildern in Ihren Dokumenten steuern. Dafür gibt es folgende Befehle:

- **WP SET ATTRIBUTES**
- **WP GET ATTRIBUTES**
- **WP RESET ATTRIBUTES**

Hinweis: Sie können Attribute für 4D Write Pro Bereiche auch über 4D Objektnotation oder generische Befehle verwalten:

- Objektnotation (*) - z.B. sind folgende Anweisungen ähnlich:

```
$bcol:=$range[wk background color]
$bcol:=$range.backgroundColor //gleiches Ergebnis
```

- Befehle **OB SET** und **OB Get**, z.B.:

```
$bcol:=OB Get($range;wk background color)
```

(*) Die Objektnotation muss in der Anwendung aktiviert sein. Weitere Informationen dazu finden Sie im Abschnitt **Objektnotation verwenden**.

Hintergrund

Über die Attribute für Hintergrund können Sie Hintergrundeffekte in Ihren Dokumenten definieren. Sie gelten für:

Dokumente	Absätze	Zeichen	Bilder	Tabellen	Zeilen	Spalte/Zellen	Kopf-/Fuß-/Hauptteile
X	X		X	X	X	X	X

Konstante Kommentar

Gibt den Zeichenbereich des Hintergrunds an. Mögliche Werte:

- [wk border box](#) (Standard): Der Hintergrund wird bis zum äußeren Rand des Rahmens gezeichnet
- [wk content box](#): Der Hintergrund wird innerhalb des Inhaltsteils gezeichnet
- [wk padding box](#): Der Hintergrund wird bis zum äußeren Rand der Einrückung gezeichnet (oder bis zum inneren Rand des Rahmens, falls vorhanden)
- [wk paper box](#): Hintergrund wird bis zum Papierrand gezeichnet (nur für Dokument oder Abschnitte)

Gibt die Hintergrundfarbe eines Elements an. Mögliche Werte:

- CSS Farbe ("[#command_5010101](#)" oder "[#command_5FFFFFFF](#)" oder "rot").
- 4D Farbwert als lange Ganzzahl (siehe **OBJECT SET COLOR**)
- Array Lange Ganzzahl mit einem Element pro R, G, B Komponente (0-255)

Standard für Dokumente ist "[#command_5FFFFFFF](#)" und [wk transparent](#), oder "transparent" für Absätze und Bilder.

Gibt die vertikale Größe des Hintergrundbildes an. Mögliche Werte:

- [wk auto](#) (Standard): Das Hintergrundbild richtet sich nach seiner Höhe
- [wk contain](#): Skaliert das Hintergrundbild mit Beibehalten der Proportionen so groß wie möglich, so dass es komplett in den Inhaltsteil passt. Diese Option ändert auch den Wert der anderen Größenattribute.
- [wk cover](#): Skaliert das Hintergrundbild mit Beibehalten der Proportionen so klein wie möglich, so dass der Hintergrundbereich komplett ausgefüllt ist. Evtl. sind Teile des Hintergrundbildes abgeschnitten. Diese Option ändert auch den Wert der anderen Größenattribute.
- Definierte Größe: Vertikale Größe des Hintergrundbildes als Zahl oder String:
 - Zahl: Größe in [wk layout unit](#)
 - String: CSS String mit zusammengezogenem Wert und Einheit, z.B.: 12pt für 12 Punkt oder 1.5cm für 1,5 Zentimeter. Mindestwert: 0pt, Maximumwert: 10000pt. Relativer Wert (in %) wird unterstützt.

Gibt Referenz für Hintergrundbild an. Kann jedes gültige Bild wie 4D Bildvariable oder Ausdruck sein:

- Zurückgegebener Wert (**WP GET ATTRIBUTES**): Wurde das Bild über eine Netzwerk URL definiert, wird das Zielbild zurückgegeben, wenn es bereits geladen ist, sonst ein leeres Bild.

Verwenden Sie [wk background image url](#), wenn Sie Bilder über URLs oder lokale URIs verwalten wollen.

Hintergrundbild, definiert über eine URL (String). Kann Netzwerk URL oder Daten URI und absolut oder relativ zur Strukturdatei sein.

- Zurückgegebener Wert (**WP GET ATTRIBUTES**): Netzwerk URL oder Daten URI. Kann sich von der ursprünglichen URL unterscheiden, wenn das Bild keine Referenz über die Netzwerk URL hat (nur Netzwerk URLs werden beibehalten). Für lokale Datei URLs wird der Bild-Stream selbst im Dokument beibehalten und so ist die zurückgegebene URL eine Daten URI mit dem Bild-Stream codiert in base64.

Verwenden Sie [wk background image](#), wenn Sie Hintergrundbilder als Bildausdrücke verwalten wollen.

Gibt die Positionierung des Hintergrundbildes an. Mögliche Werte:

- [wk padding box](#) (Standard): Das Hintergrundbild startet im Viereck der Einrückung (oder innerer Rahmenrand)
- [wk border box](#): Das Hintergrundbild startet im Viereck des Rahmens (äußerer Rand)
- [wk content box](#): Das Hintergrundbild startet im Viereck des Inhalts
- [wk paper box](#): Hintergrundbild startet am Papierrand (nur für Dokument oder Abschnitte)

Gibt die horizontale Startposition des Hintergrundbildes an. Mögliche Werte:

- [wk left](#) (Standard): Das Hintergrundbild beginnt horizontal auf der linken Seite des Elements
- [wk center](#): Das Hintergrundbild beginnt horizontal in der Mitte des Elements
- [wk right](#): Das Hintergrundbild beginnt horizontal auf der rechten Seite des Elements

Gibt die vertikale Startposition des Hintergrundbildes an. Mögliche Werte:

- [wk top](#) (Standard): Das Hintergrundbild beginnt vertikal oben im Element
- [wk middle](#): Das Hintergrundbild beginnt vertikal in der Mitte des Elements
- [wk bottom](#): Das Hintergrundbild beginnt vertikal unten im Element

Gibt an, ob und wie das Hintergrundbild wiederholt wird. Mögliche Werte:

- [wk repeat](#) (Standard): Das Hintergrundbild wird vertikal und horizontal wiederholt
- [wk no repeat](#): Das Hintergrundbild wird nicht wiederholt
- [wk repeat x](#): Das Hintergrundbild wird nur horizontal wiederholt
- [wk repeat y](#): Das Hintergrundbild wird nur vertikal wiederholt

Konstante Kommentar

Gibt die horizontale Größe des Hintergrundbildes an. Mögliche Werte:

wk
background
width

- [wk auto](#) (Standard): Das Hintergrundbild richtet sich nach seiner Breite
- [wk contain](#): Skaliert das Hintergrundbild mit Beibehalten der Proportionen so groß wie möglich, so dass es komplett in den Inhaltsbereich passt. Diese Option ändert auch den Wert der anderen Größenattribute.
- [wk cover](#): Skaliert das Hintergrundbild mit Beibehalten der Proportionen so groß wie möglich, so dass der Hintergrundbereich komplett ausgefüllt ist. Evtl. sind Teile des Hintergrundbildes abgeschnitten. Diese Option ändert auch den Wert der anderen Größenattribute.
- Definierte Größe: Horizontale Größe des Hintergrundbildes als Zahl oder String:
 - Zahl: Größe in [wk layout unit](#)
 - String: CSS String mit zusammengezogenem Wert und Einheit, z.B.: 12pt für 12 Punkt, oder 1,5cm für 1,5 Zentimeter. Mindestwert: 0pt, Maximumwert: 10000pt. Relativer Wert (in %) wird unterstützt.

Rahmen

Über die Attribute für Rahmen können Sie Stil, Breite und Farbe eines Rahmenelements definieren. Sie gelten für:

Dokumente	Absätze	Zeichen	Bilder	Tabellen	Zeilen	Spalte/Zellen	Kopf-/Fuß-/Hauptteile
X	X		X	X	X	X	X

Konstante Kommentar

Setzt Farbe für alle vier Ränder. Mögliche Werte:

- CSS Farbe ("[#command_5010101](#)" oder "[#command_5FFFFFFF](#)" oder "rot").
- 4D Farbwert als Lange Ganzzahl (siehe Befehl [OBJECT SET COLOR](#))
- Array Lange Ganzzahl mit einem Element pro R, G, B Komponente (0-255)

Standard ist "[#command_5000000](#)" (bei Wert als String). Bei mehreren Farben gibt [WP GET ATTRIBUTES](#) einen leeren String zurück.

Setzt Farbe des unteren Rands. Mögliche Werte:

- CSS Farbe ("[#command_5010101](#)" oder "[#command_5FFFFFFF](#)" oder "rot").
- 4D Farbwert als Lange Ganzzahl (siehe Befehl [OBJECT SET COLOR](#))
- Array Lange Ganzzahl mit einem Element pro R, G, B Komponente (0-255)

Setzt Farbe des linken Rands. Mögliche Werte:

- CSS Farbe ("[#command_5010101](#)" oder "[#command_5FFFFFFF](#)" oder "rot").
- 4D Farbwert als Lange Ganzzahl (siehe Befehl [OBJECT SET COLOR](#))
- Array Lange Ganzzahl mit einem Element pro R, G, B Komponente (0-255)

Setzt Farbe des rechten Rands. Mögliche Werte:

- CSS Farbe ("[#command_5010101](#)" oder "[#command_5FFFFFFF](#)" oder "rot").
- 4D Farbwert als Lange Ganzzahl (siehe Befehl [OBJECT SET COLOR](#))
- Array Lange Ganzzahl mit einem Element pro R, G, B Komponente (0-255)

Setzt Farbe des oberen Rands. Mögliche Werte:

- CSS Farbe ("[#command_5010101](#)" oder "[#command_5FFFFFFF](#)" oder "rot").
- 4D Farbwert als Lange Ganzzahl (siehe Befehl [OBJECT SET COLOR](#))
- Array Lange Ganzzahl mit einem Element pro R, G, B Komponente (0-255)

Gibt abgerundeten Rand an. Mögliche Werte:

- [wk none](#) (Standard): Rand hat keine abgerundeten Ecken
- Radiuswert als Zahl oder String:
 - Zahl: Radius in [wk layout unit](#).
 - String: CSS String mit zusammengesetztem Wert und Einheit. z.B: 12pt für 12 Punkt oder 1.5cm für 1.5 Zentimeter.

Gibt Stil für alle vier Ränder an. Mögliche Werte:

- [wk none](#) (Standard): Kein Rand
- [wk hidden](#): genauso wie [wk none](#), außer bei Auflösung Randkonflikt
- [wk solid](#): durchgezogener Rand
- [wk dotted](#): gepunkteter Rand
- [wk dashed](#): gestrichelter Rand
- [wk double](#): doppelter Rand
- [wk groove](#): 3D Rahmen vertieft (aktuelle Auswirkung richtet sich nach der Rahmenfarbe)
- [wk ridge](#): 3D Rahmen erhoben (aktuelle Auswirkung richtet sich nach der Rahmenfarbe)
- [wk inset](#): 3D Rahmen nach innen (aktuelle Auswirkung richtet sich nach der Rahmenfarbe)

Gibt Stil für unteren Rand an. Mögliche Werte:

- [wk none](#) (Standard): Kein Rand
- [wk hidden](#): genauso wie [wk none](#), außer bei Auflösung Randkonflikt
- [wk solid](#): durchgezogener Rand
- [wk dotted](#): gepunkteter Rand
- [wk dashed](#): gestrichelter Rand
- [wk double](#): doppelter Rand
- [wk groove](#): 3D Rand vertieft (aktuelle Auswirkung richtet sich nach der Randfarbe)
- [wk ridge](#): 3D Rand erhoben (aktuelle Auswirkung richtet sich nach der Randfarbe)
- [wk inset](#): 3D Rand nach innen (aktuelle Auswirkung richtet sich nach der Randfarbe)

Gibt Stil für linken Rand an:

- [wk none](#) (Standard): Kein Rand
- [wk hidden](#): genauso wie [wk none](#), außer bei Auflösung Randkonflikt
- [wk solid](#): durchgezogener Rand
- [wk dotted](#): gepunkteter Rand
- [wk dashed](#): gestrichelter Rand
- [wk double](#): doppelter Rand
- [wk groove](#): 3D Rand vertieft (aktuelle Auswirkung richtet sich nach der Randfarbe)
- [wk ridge](#): 3D Rand erhoben (aktuelle Auswirkung richtet sich nach der Randfarbe)
- [wk inset](#): 3D Rand nach innen (aktuelle Auswirkung richtet sich nach der Randfarbe)

Konstante Kommentar

Gibt Stil für rechten Rand an:

- [wk none](#) (Standard): Kein Rand
- [wk hidden](#): genauso wie [wk none](#), außer bei Auflösung Randkonflikt
- [wk solid](#): durchgezogener Rand
- [wk dotted](#): gepunkteter Rand
- [wk dashed](#): gestrichelter Rand
- [wk double](#): doppelter Rand
- [wk groove](#): 3D Rand vertieft (aktuelle Auswirkung richtet sich nach der Randfarbe)
- [wk ridge](#): 3D Rand erhoben (aktuelle Auswirkung richtet sich nach der Randfarbe)
- [wk inset](#): 3D Rand nach innen (aktuelle Auswirkung richtet sich nach der Randfarbe)

wk border style right

Gibt Randstil oben an. Mögliche Werte:

- [wk none](#) (Standard): Kein Rand
- [wk hidden](#): genauso wie [wk none](#), außer bei Auflösung Randkonflikt
- [wk solid](#): durchgezogener Rand
- [wk dotted](#): gepunkteter Rand
- [wk dashed](#): gestrichelter Rand
- [wk double](#): doppelter Rand
- [wk groove](#): 3D Rand vertieft (aktuelle Auswirkung richtet sich nach der Randfarbe)
- [wk ridge](#): 3D Rand erhoben (aktuelle Auswirkung richtet sich nach der Randfarbe)
- [wk inset](#): 3D Rand nach innen (aktuelle Auswirkung richtet sich nach der Randfarbe)

wk border style top

Gibt Breite für alle vier Ränder an. Sie müssen zuvor den Randstil festlegen. Mögliche Werte:

- Breite als Zahl oder String:
 - Zahl: Breite in [wk layout unit](#).
 - String: CSS String mit zusammengezogenem Wert und Einheit, z.B: 12pt für 12 Punkt oder 1.5cm für 1,5 Zentimeter
- Standardwert: 2pt

wk border width

Gibt Breite für den unteren Rand an. Mögliche Werte:

- Breite als Zahl oder String:
 - Zahl: Breite in [wk layout unit](#).
 - String: CSS String mit zusammengezogenem Wert und Einheit, z.B: 12pt für 12 Punkt oder 1.5cm für 1.5 Zentimeter
- Standardwert: 2pt

wk border width bottom

Gibt Breite für linken Rand an. Mögliche Werte:

- Breite als Zahl oder String:
 - Zahl: Breite in [wk layout unit](#).
 - String: CSS String mit zusammengezogenem Wert und Einheit, z.B.: 12pt für 12 Punkt oder 1.5cm für 1.5 Zentimeter
- Standardwert: 2pt

wk border width left

Gibt Breite für rechten Rand an. Mögliche Werte:

- Breite als Zahl oder String:
 - Zahl: Breite in [wk layout unit](#).
 - String: CSS String mit zusammengezogenem Wert und Einheit, z.B: 12pt für 12 Punkt oder 1.5cm für 1.5 Zentimeter
- Standardwert: 2pt

wk border width right

Gibt Breite für oberen Rand an. Mögliche Werte:

- Breite als Zahl oder String:
 - Zahl: Breite in [wk layout unit](#).
 - String: CSS String mit zusammengezogenem Wert und Einheit, z.B: 12pt für 12 Punkt oder 1.5cm für 1.5 Zentimeter
- Standardwert: 2pt

wk border width top

wk inside Gibt bei mehreren Absätzen im ausgewählten Bereich an, dass das Attribut nur für die entsprechende innere Eigenschaft des Absatzes gilt (nicht außerhalb). Gilt nur für Attribute für Außenabstand, Innenabstand und Ränder und muss zum angegebenen Attribut hinzugefügt werden. Siehe Beispiel 2 des Befehls **WP SET ATTRIBUTES**.

wk outside Gibt bei mehreren Absätzen im ausgewählten Bereich an, dass das Attribut nur für die entsprechende äußere Eigenschaft des Absatzes gilt (nicht innerhalb). Gilt nur für Attribute für Außenabstand, Innenabstand und Ränder und muss zum angegebenen Attribut hinzugefügt werden. Siehe Beispiel 2 des Befehls **WP SET ATTRIBUTES**.

Dokument

Über Attribute für Dokumente können Sie Standardinformationen zum Dokument wie Thema, Name des Autors, der Firma und Notizen setzen oder erhalten. Sie gelten für:

Dokumente	Absätze	Zeichen	Bilder	Tabellen	Zeilen	Spalte/Zellen	Kopf-/Fuß-/Hauptteile
X							X

Konstante	Kommentar
wk author	<p>Verwendung: Dokumente Standardwert: keiner Mögliche Werte: Eigene Zeichenkette</p>
wk column count	<p>(Nur lesbares Attribut für Ausschnitt) Gesamtanzahl der Spalten im Ausschnitt. Wert: Lange Ganzzahl Farbe der vertikalen Spaltenregel. Mögliche Werte:</p>
wk column rule color	<ul style="list-style-type: none"> • eine CSS-Farbe ("#command_5010101" oder "#command_5FFFFFF" oder "rot"). Standard ist "#command_5000000" (schwarz) • ein 4D-Farbwert vom Typ Lange Ganzzahl (siehe Befehl OBJECT SET COLOR) • ein Longint-Array, das ein Element für jede R-, G-, B-Komponente (0-255) enthält
wk column rule style	<p>Stil der vertikalen Spaltenregel. Mögliche Werte:</p> <ul style="list-style-type: none"> • wk none (Standard): keine Regel • wk hidden: identisch mit wk none, außer beim Auflösen von Regelkonflikten • wk solid: Regel durchgezogen • wk dotted: Regel gepunktet • wk dashed: Regel gestrichelt • wk double: Regel doppelt • wk groove: Regel 3D vertieft (der tatsächliche Effekt hängt von der Regelfarbe ab) • wk ridge: Regel 3D erhoben (der tatsächliche Effekt hängt von der Regelfarbe ab) • wk inset: Regel 3D nach innen (der tatsächliche Effekt hängt von der Regelfarbe ab)
wk column rule width	<p>Breite der vertikalen Spaltenregel. Mögliche Werte:</p> <ul style="list-style-type: none"> • Zahl: Breite in wk layout unit. • String: Wert für Breite und Einheit zusammengezogen. (<i>Beispiel:</i> "12pt" für 12 Punkt oder "1.5cm" für 1,5 Zentimeter) Standardwert="2.5pt"
wk column spacing	<p>(nur für Dokumente oder Abschnitte) Abstand zwischen zwei Spalten. Mögliche Werte:</p> <ul style="list-style-type: none"> • Zahl: Breite in wk layout unit • String: Wert und Einheit der Breite zusammengezogen. (<i>e.g.:</i> "12pt" für 12 Punkt oder "1.5cm" für 1,5 Zentimeter). Standardwert="12pt"
wk column width	<p>(Nur für Dokumente oder Abschnitte) Nur lesbares Attribut. Aktuelle Breite für jede Spalte, d.h. berechnete Breite basierend auf tatsächlicher Seitenbreite, Seitenrändern, Spaltenanzahl und Spaltenabstand. Für das Dokument wird die standardmäßige Spaltenbreite des Abschnitts verwendet, kann also von der tatsächlichen Spaltenbreite des Abschnitts/der Abschnitte abweichen, wenn einige Attribute in einem Abschnitt überschrieben werden. Mögliche Werte:</p> <ul style="list-style-type: none"> • Zahl: Breite in wk layout unit. • String: Wert für Breite und Einheit zusammengezogen. (<i>Beispiel:</i> "12pt" für 12 Punkt oder "1.5cm" für 1,5 Zentimeter)
wk company	Gibt eine dem Dokument zugeordnete Firma an (String)
wk date creation	Gibt Erstellungsdatum des Dokuments zurück. Dieser Wert ist im Lesemodus und lässt sich nicht setzen.
wk date modified	Gibt letztes Änderungsdatum des Dokuments zurück. Dieser Wert ist im Lesemodus und lässt sich nicht setzen.
wk dpi	Für interne Pixel verwendete DPI <->Punktkonvertierung (Ganzzahl). Immer 96 (Nur-Lesen) Gibt die standardmäßige Maßeinheit für das Dokument an, wenn ein Wert als Zahl gesetzt oder erhalten wird. Gibt die Einheit für das Lineal und Attribute für Ausmaße an, wie z.B. wk width - mit Ausnahme von wk font size , wk border width und seine Variationen, wk border radius und wk text shadow offset , für die als Einheit immer Punkt gilt. Mögliche Werte:
wk layout unit	<ul style="list-style-type: none"> • wk unit cm (Standard): Zentimeter • wk unit pt: Punkt • wk unit px: Pixel • wk unit percent (nur für wk line height und wk background size h / wk background size v) • wk unit mm: Millimeter • wk unit inch: Zoll <p>Hinweis: Wird eine vom Lineal nicht unterstützte Einheit ausgewählt, wie z.B. wk unit px oder wk unit percent, verwendet das Lineal die Einheit cm.</p>
wk notes	Gibt Kommentare über das Dokument an (String).
wk subject	Gibt Betreff des Dokuments an (String)
wk title	Gibt Dokumenttitel (String) an. Standard ist "Neues 4D Write Pro Dokument"
wk version	Gibt die interne 4DWP Version des Dokuments zurück (Zahl). Diese Zahl lässt sich mit dem Befehl WP GET ATTRIBUTES nur lesen; sie kann nicht gesetzt werden.

Schrift und Text

Über diese Attribute können Sie Schriftfamilie, Schriftgröße und Schriftstil von Text definieren. Sie gelten für:

Dokumente	Absätze	Zeichen	Bilder	Tabellen	Zeilen	Spalte/Zellen	Kopf-/Fuß-/Hauptteile
	X			X*	X*	X*	X*

* gilt für Absatzzeichen in Zellen

Konstante Kommentar

wk font	<p>Gibt kompletten Schriftnamen mit Stilarten an, wie vom Befehl FONT STYLE LIST zurückgegeben. Bei einem ungültigen Schriftnamen führt der Befehl nichts aus. Standardwert: "Times New Roman".</p> <p>Gibt die Textstärke an (richtet sich nach verfügbaren Schriftstilen). Mögliche Werte:</p> <ul style="list-style-type: none">• <u>wk true</u> um ausgewählte Zeichen auf den Schriftstil Fett zu setzen; mit dem Befehl WP GET ATTRIBUTES wird <u>wk true</u> zurückgegeben, wenn mindestens ein ausgewähltes Zeichen in Fettschrift ist.• <u>wk false</u> (Standard) um den Schriftstil Fett - falls vorhanden - von den ausgewählten Zeichen zu entfernen; mit dem Befehl WP GET ATTRIBUTES wird <u>wk false</u> zurückgegeben, wenn keins der ausgewählten Zeichen in Fettschrift ist.
wk font bold	
wk font family	<p>Gibt den Namen der Schriftfamilie an wie von <u>wk font</u> definiert. Standardwert: "Times New Roman". Der Befehl WP GET ATTRIBUTES gibt einen leeren String zurück, wenn die ausgewählten Zeichen unterschiedliche Schriftfamilien-Eigenschaften enthalten.</p> <p>Gibt den Stil Kursiv für Text an (richtet sich nach den Schriftstilen). Mögliche Werte:</p> <ul style="list-style-type: none">• <u>wk true</u>, um ausgewählte Zeichen auf Kursiv- oder Schrägschrift zu setzen; der Befehl WP GET ATTRIBUTES gibt <u>wk true</u> zurück, wenn mindestens ein ausgewähltes Zeichen Kursiv- oder Schrägschrift unterstützt.• <u>wk false</u> (Standard), um in gewählten Zeichen Kursiv- oder Schrägschrift - wenn vorhanden - zu entfernen; der Befehl gibt <u>wk false</u> zurück, wenn keines der gewählten Zeichen in Kursiv- oder Schrägschrift ist.
wk font italic	
wk font size	<p>Gibt die Schriftgröße für Text an. Mögliche Werte (nur in Punkt):</p> <ul style="list-style-type: none">• Wert Zahl (Standard = 12)• CSS String mit zusammengesetztem Wert und Einheit, z.B.: 12pt für 12 Punkt.
wk text color	<p>Gibt die Textfarbe an. Mögliche Werte:</p> <ul style="list-style-type: none">• Eine CSS Farbe("#command_5010101" oder "#command_5FFFFFFF" oder "rot"). Bei String ist "#command_5000000" Standard• Ein 4D Farbwert als Lange Ganzzahl (siehe Befehl OBJECT SET COLOR)• Ein Array Lange Ganzzahl mit je einem Element pro R, G, B Komponente (0-255)
wk text linethrough color	<p>Gibt Farbe des durchgestrichenen Textes an. Mögliche Werte:</p> <ul style="list-style-type: none">• CSS Farbe ("#command_5010101" oder "#command_5FFFFFFF" oder "rot").• 4D Farbwert als Lange Ganzzahl (siehe Befehl OBJECT SET COLOR)• Array Lange Ganzzahl mit einem Element pro R, G, B Komponente (0-255) <p>Standard ist "currentColor" wenn String, oder <u>wk default</u> wenn Lange Ganzzahl.</p> <p>Gibt Stil für durchgestrichenen Text an (wenn vorhanden). Mögliche Werte:</p> <ul style="list-style-type: none">• <u>wk none</u> (Standard): Keine Auswirkung• <u>wk solid</u>: Zeichnet durchgezogene Linie im ausgewählten Text• <u>wk dotted</u>: Zeichnet gepunktete Linie im ausgewählten Text• <u>wk dashed</u>: Zeichnet gestrichelte Linie im ausgewählten Text• <u>wk double</u>: Zeichnet doppelte Linie im ausgewählten Text• <u>wk semi transparent</u>: gedimmte Linie im ausgewählten Text. Ist mit anderem Linienstil kombinierbar.• <u>wk word</u>: Zeichnet die Linie nur bei Wörtern (Leerzeichen werden ausgelassen). Ist mit anderem Linienstil kombinierbar.
wk text shadow color	<p>Gibt Schattenfarbe des gewählten Textes an. Mögliche Werte:</p> <ul style="list-style-type: none">• CSS Farbe ("#command_5010101" oder "#command_5FFFFFFF" oder "rot").• 4D Farbwert als Lange Ganzzahl (siehe Befehl OBJECT SET COLOR)• Array Lange Ganzzahl mit einem Element pro R, G, B Komponente (0-255)• <u>wk transparent</u> (Standard)
wk text shadow offset	<p>Gibt Versatz (offset) des Schatteneffekts an. Mögliche Werte:</p> <ul style="list-style-type: none">• Größenangabe in Punkt, Standardwert: 1pt
wk text transform	<p>Setzt Buchstaben im Text in Groß- oder Kleinschreibung. Mögliche Werte:</p> <ul style="list-style-type: none">• <u>wk capitalize</u>: Schreibt die Anfangsbuchstaben groß• <u>wk lowercase</u>: Schreibt die Buchstaben klein• <u>wk uppercase</u>: Schreibt die Buchstaben groß• <u>wk small uppercase</u>: Setzt Buchstaben in Kapitälchen• <u>wk none</u> (Standard): Keine Umwandlung
wk text underline color	<p>Gibt Farbe des unterstrichenen Textes an. Mögliche Werte:</p> <ul style="list-style-type: none">• CSS Farbe ("#command_5010101" oder "#command_5FFFFFFF" oder "rot").• 4D Farbwert als Lange Ganzzahl (siehe Befehl OBJECT SET COLOR)• Array Lange Ganzzahl mit einem Element pro R, G, B Komponente (0-255) <p>Standard ist "currentColor" wenn String, oder <u>wk default</u> wenn Lange Ganzzahl.</p>

Konstante Kommentar

Gibt Stil für unterstrichenen Text an (wenn vorhanden). Mögliche Werte:

- [wk none](#) (Standard): Keine Unterstreichung
- [wk solid](#): Zeichnet durchgezogene Linie im ausgewählten Text
- [wk dotted](#): Zeichnet gepunktete Linie im ausgewählten Text
- [wk dashed](#): Zeichnet gestrichelte Linie im ausgewählten Text
- [wk double](#): Zeichnet doppelte Linie im ausgewählten Text
- [wk semi transparent](#): gedimmte Linie im ausgewählten Text. Ist mit anderem Linienstil kombinierbar.
- [wk word](#): Zeichnet die Linie nur bei Wörtern (Leerzeichen werden ausgelassen). Ist mit anderem Linienstil kombinierbar.

wk text
underline
style

Setzt die vertikale Ausrichtung eines Elements. Verwendbar für Zeichen, Absätze und Bilder. Mögliche Werte:

- [wk baseline](#) (Standard): Richtet die Grundlinie eines Elements nach der Grundlinie des übergeordneten Elements aus
- [wk top](#): Richtet die Oberkante des Elements nach der Oberkante des größten Elements in der Reihe aus
- [wk bottom](#): Richtet die Unterkante des Elements nach dem untersten Element in der Reihe aus
- [wk middle](#): Element wird in die Mitte des übergeordneten Elements gesetzt
- [wk superscript](#): Richtet das Element aus, als ob es hochgestellt wäre
- [wk subscript](#): Richtet das Element aus, als ob es tiefgestellt wäre

wk vertical
align

Für Zeichen haben [wk top](#) und [wk bottom](#) denselben Effekt wie [wk baseline](#).

Für Absätze haben [wk baseline](#), [wk superscript](#) und [wk subscript](#) denselben Effekt wie [wk top](#).

Höhe/Breite

Über diese Attribute können Sie die Höhe und Breite von Absätzen und Bildern setzen. Sie gelten für:

Dokumente	Absätze	Zeichen	Bilder	Tabellen	Zeilen	Spalte/Zellen	Kopf-/Fuß-/Hauptteile
	X		X		X	X*	X**

*gilt für Zellen

**Nur-Lesen für Kopf- und Fußteile (automatisch gesetzt).

Konstante Kommentar

Setzt Höhe des Elements, dazu gehören nicht Einrückung, Ränder oder Rahmen; d.h. die Höhe des Bereichs liegt innerhalb von Einrückung, Rändern und Rahmen des Elements. Mögliche Werte:

- [wk auto](#) (Standard): Höhe basiert auf dem Inhalt des Elements
- Größe wird als Zahl oder Stringwert ausgedrückt:
 - Zahl: Größe in [wk layout unit](#).
 - String: CSS String mit zusammengesetztem Wert und Einheit, z.B.: 12pt für 12 Punkt oder 1.5cm für 1,5 Zentimeter. Mindestwert: 0pt, Maximumwert: 10.000pt.

wk height

Das Attribut [wk height](#) wird von [wk min height](#) überschrieben (wenn definiert).

Setzt Mindesthöhe des Elements. Verhindert, dass der Wert von [wk height](#) kleiner als [wk min height](#) wird. Mögliche Werte:

- [wk auto](#) (Standard): Höhe basiert auf dem Inhalt des Elements
- Größe wird als Zahl oder Stringwert ausgedrückt:
 - Zahl: Größe in [wk layout unit](#).
 - String: CSS String mit zusammengesetztem Wert und Einheit, z.B.: 12pt für 12 Punkt oder 1.5cm für 1,5 Zentimeter. Mindestwert: 0pt, Maximumwert: 10.000pt.

wk min
height

Der Wert von [wk min height](#) überschreibt den Wert von [wk height](#).

Setzt Mindestbreite des Elements. Verhindert, dass der Wert von [wk width](#) kleiner als [wk min width](#) wird. Mögliche Werte:

- [wk auto](#) (Standard): Höhe basiert auf dem Inhalt des Elements
- Größe wird als Zahl oder Stringwert ausgedrückt:
 - Zahl: Größe in [wk layout unit](#).
 - String: CSS String mit zusammengesetztem Wert und Einheit, z.B.: 12pt für 12 Punkt oder 1.5cm für 1,5 Zentimeter. Mindestwert: 0pt, Maximumwert: 10.000pt.

wk min
width

Der Wert von [wk min width](#) überschreibt den Wert von [wk width](#).

Setzt Breite des Elements. Mögliche Werte:

- [wk auto](#) (Standard): Breite basiert auf dem Inhalt des Elements
- Größe wird als Zahl oder Stringwert ausgedrückt:
 - Zahl: Größe in [wk layout unit](#).
 - String: CSS String mit zusammengesetztem Wert und Einheit, z.B.: 12pt für 12 Punkt oder 1.5cm für 1,5 Zentimeter. Mindestwert: 0pt, Maximumwert: 10.000pt.

wk width

Der Wert von [wk min width](#) überschreibt den Wert von [wk width](#).

Bilder

Über die Attribute für Bilder können Sie Bilder innerhalb des Bereichs verwalten. Sie gelten für:

Dokumente	Absätze	Zeichen	Bilder	Tabellen	Zeilen	Spalte/Zellen	Kopf-/Fuß-/Hauptteile
			X	X*	X*	X*	

*gilt für Bilder in Zellen

Zur Erinnerung: 4D Write Pro unterstützt, wie im Abschnitt **Bilder verwalten** erläutert, zwei Arten von Bildern:

- Ein im Textfluss eingefügtes Bild über die Befehle **WP INSERT PICTURE** oder **ST INSERT EXPRESSION**
- Ein in der Seite verankertes Bild über den Befehl **WP Add picture**

Folgende Attribute sind für beide Arten von Bildern verfügbar:

Konstante Kommentar

Gibt ein Bild an. Mögliche Werte:

- Bild URL (String). Kann absolut oder relativ zur Strukturdatei sein.
- Bildvariable oder Feld.

wk image

Zurückgegebener Wert (**WP GET ATTRIBUTES**): URI (Netzwerk URL oder Daten URI). Kann sich von der ursprünglichen URL unterscheiden, wenn das Bild keine Referenz über die Netzwerk URL hat (nur Netzwerk URLs werden beibehalten). Für lokale Datei URLs wird der Bild-Stream selbst im Dokument beibehalten und so ist die zurückgegebene URL eine Daten URI mit dem Bild-Stream codiert in base64.

wk image
alternate
text

Gibt Ersatztext für Bild an, wenn das Bild sich nicht anzeigen lässt.

Gibt ein Bild, definiert über eine URL (String), an. Kann Netzwerk URL oder Daten URI und absolut oder relativ zur Strukturdatei sein.

wk image
url

- Zurückgegebener Wert (**WP GET ATTRIBUTES**): Netzwerk URL oder Daten URI. Kann sich von der ursprünglichen URL unterscheiden, wenn das Bild keine Referenz über die Netzwerk URL hat (nur Netzwerk URLs werden beibehalten). Für lokale Datei URLs wird der Bild-Stream selbst im Dokument beibehalten und so ist die zurückgegebene URL eine Daten URI mit dem Bild-Stream codiert in base64.

wk owner

Verwenden Sie wk image, wenn Sie Bilder als Bildausdrücke verwalten wollen.
(Nur lesbares Attribut) Eigentümer des Ausschnitts/Objekts. Wert: Objekt
(Nur lesbares Attribut für Ausschnitt) Typ des 4D Write Pro Objekts. Mögliche Werte:

wk type

- wk type default: Typ Standardausschnitt
- wk type paragraph: Typ Absatzausschnitt
- wk type image: Verankertes Bild
- wk type container: z.B. Kopf- oder Fußteil
- wk type table: Referenz Tabelle
- wk type table row: Referenz Tabellenzeile
- wk type table cell: Referenz Tabellenzelle
- wk type table column: Referenz Tabellenspalte (nur für Spaltenausschnitt)
- wk type body: Referenz Hauptteil

Folgende Attribute sind nur für Bilder im Textfluss verfügbar:

Konstante Kommentar

wk end

(Nur lesbares Attribut für Ausschnitt). Ausschnitt Ende Versatz. Wert: Lange Ganzzahl

wk start

(nur lesbares Attribut für Ausschnitt). Ausschnitt Start Versatz. Wert: Lange Ganzzahl

wk vertical
align

Setzt die vertikale Ausrichtung eines Elements. Verwendbar für Zeichen, Absätze und Bilder. Mögliche Werte:

- wk baseline (Standard): Richtet die Grundlinie eines Elements nach der Grundlinie des übergeordneten Elements aus
- wk top: Richtet die Oberkante des Elements nach der Oberkante des größten Elements in der Reihe aus
- wk bottom: Richtet die Unterkante des Elements nach dem untersten Element in der Reihe aus
- wk middle: Element wird in die Mitte des übergeordneten Elements gesetzt
- wk superscript: Richtet das Element aus, als ob es hochgestellt wäre
- wk subscript: Richtet das Element aus, als ob es tiefgestellt wäre

Für Zeichen haben wk top und wk bottom denselben Effekt wie wk baseline.

Für Absätze haben wk baseline, wk superscript und wk subscript denselben Effekt wie wk top.

Folgende Attribute sind nur für verankerte Bilder verfügbar:

Konstante Kommentar

wk anchor horizontal align	<p>Definiert die horizontale Ausrichtung eines Bildes in Bezug zum Ursprung (siehe wk anchor origin). Mögliche Werte:</p> <ul style="list-style-type: none">• wk left - Ausrichtung links• wk center - Ausrichtung mittig (<i>nicht kompatibel mit HTML, Bilder erscheinen nicht im Web</i>)• wk right - Ausrichtung rechts <p>Definiert den horizontalen Versatz eines Bildes als CSS Positionsangabe (Text wie z.B. "19px") oder Lange Ganzzahl (z.B. 19) von wk layout unit. Wert kann in cm, pt oder pixel sein. Mögliche Werte:</p>
wk anchor horizontal offset	<ul style="list-style-type: none">• Linker oder rechter Rand der Seite in Bezug auf wk anchor horizontal align• Linker oder rechter Rand des Hauptteils im eingebundenen Modus (wenn wk anchor section = wk anchor embedded) <p>Standardwert = 0</p> <p>Definiert die Layoutposition eines Bildes in Bezug auf den Text auf der Seite. Mögliche Werte:</p>
wk anchor layout	<ul style="list-style-type: none">• wk behind text - Bild ist verankert und hinter dem Text• wk in front of text - Bild ist verankert und vor dem Text• wk inline with text - Bild ist im Textfluss (Standard für Bilder, eingefügt mit WP INSERT PICTURE). Nur-Lesen Attribute (Bilder im Textfluss sind nicht per Programmierung in verankerte Bilder konvertierbar) <p>Definiert, ob das Bild mit der Seite, dem Kopfteil oder Fußteil verankert ist. Mögliche Werte:</p>
wk anchor origin	<ul style="list-style-type: none">• wk paper box (Standard) - Bild wird mit dem Seitenrand verankert• wk header box - Bild wird mit dem Kopfteil des Dokuments verankert. Ist der Kopfteil ausgeblendet, erscheint kein Bild.• wk footer box - Bild wird mit dem Fußteil des Dokuments verankert. Ist der Fußteil ausgeblendet, erscheint kein Bild. <p>Dieser Selektor wird im eingebundenen Modus ignoriert.</p> <p>Definiert den Seitenindex oder Seitentyp, mit dem ein Bild verankert ist. Mögliche Werte:</p>
wk anchor page	<ul style="list-style-type: none">• wk anchor all - verankert ein Bild mit allen Seiten des Bereichs, definiert durch wk anchor section• wk anchor embedded - verankert ein Bild mit dem Hauptteil des Dokuments im eingebundenen Modus (Bild ist nur im eingebundenen Modus sichtbar)• wk anchor first page - verankert ein Bild mit der ersten Seite des Bereichs, definiert durch wk anchor section• wk anchor left page - verankert ein Bild mit den linken Seiten des Bereichs, definiert durch wk anchor section• wk anchor right page - verankert ein Bild mit den rechten Seiten des Bereichs, definiert durch wk anchor section• die Zahl (Longint >= 0) gibt die Seite an, mit der das Bild verankert werden soll. In diesem Fall gilt wk anchor section = wk anchor embedded. Bei Verankerung mit einer einzelnen Seite wird die Verankerung mit Bereich ignoriert. <p>Hinweis: Bilder im Seitenmodus werden in Browsern nicht angezeigt.</p> <p>Definiert den Bereichsindex oder Bereichstyp, mit dem das Bild verankert wird. Mögliche Werte:</p>
wk anchor section	<ul style="list-style-type: none">• wk anchor all (Standard) - verankert ein Bild mit allen Bereichen in einem Dokument (Bild ist nur im Seitenmodus sichtbar)• wk anchor embedded - verankert ein Bild mit dem Hauptteil des Dokuments im eingebundenen Modus (Bild ist nur im eingebundenen Modus sichtbar)• Zahl (Longint >= 1) gibt den Bereich an, mit dem das Bild verankert werden soll. (Bild ist nur im Seitenmodus sichtbar) <p>Hinweis: Bilder im Seitenmodus erscheinen nicht in Browsern.</p> <p>Definiert die vertikale Ausrichtung eines Bildes relativ zum Ursprung (siehe wk anchor origin). Mögliche Werte:</p>
wk anchor vertical align	<ul style="list-style-type: none">• wk top - Ausrichtung oben• wk center - Ausrichtung mittig (<i>nicht kompatibel mit HTML, Bilder werden nicht in Browsern angezeigt</i>)• wk bottom - Ausrichtung unten <p>Definiert die vertikale Position eines Bildes als CSS Positionsangabe (Text wie z.B. "19px") oder Lange Ganzzahl (z.B. 19) von wk layout unit. Wert kann in cm, pt oder pixel sein. Mögliche Werte:</p>
wk anchor vertical offset	<ul style="list-style-type: none">• Oberer, unterer Rand oder Mitte der Seite (siehe wk anchor origin) oder• Oberer, unterer Rand oder Mitte des Hauptteils im eingebundenen Modus (wenn wk anchor section = wk anchor embedded). <p>Standardwert = 0.</p>
wk id	(Nur lesbares Attribut) ID des Elements. Wert: String

Links

Über Attribute für Links können Sie in Ausschnitten eingefügte URLs setzen oder erhalten. Sie gelten für:

Dokumente	Abschnitte	Zeichen	Bilder	Tabellen	Zeilen	Spalte/Zellen	Kopf-/Fuß-/Hauptteile
X	X	X	X*	X	X	X	

Konstante **Kommentar**

Dem Ausschnitt zugewiesener Hyperlink. Mögliche Werte:

- wk link url
- Absolute Url, z.B. "http://www.4d.com/"
 - Relativer Link, z.B. "/test/page.html" (Link in Bezug auf die Strukturdatei der Anwendung)
 - Leerer String = kein Link

* nur im Textfluss eingefügte Bilder

Listen

4D Write Pro unterstützt zwei Haupttypen von Listen:

- Ungeordnete Listen: Die Einträge werden mit "Bullets" markiert
- Geordnete Listen: Die Einträge werden mit fortlaufenden Nummern oder Buchstaben markiert

Über Attribute für Listen können Sie Ihre Listen konfigurieren und unterschiedliche Schriften oder Marker für Listeneinträge setzen. Sie gelten für:

Dokumente	Absätze	Zeichen	Bilder	Tabellen	Zeilen	Spalte/Zellen	Kopf-/Fuß-/Hauptteile
	X			X*	X*	X*	

* gilt für Absätze in Zellen

Konstante Kommentar

wk list font	Gibt den kompletten Schriftnamen an, wie vom Befehl FONT STYLE LIST zurückgegeben, um Marker für Listeneintrag anzuzeigen (nicht den Absatztext). Erkennt das System den Schriftnamen nicht, sorgt es für Ersatz. Ist der Schriftname ungültig, führt der Befehl nichts aus. Standardwert: "Times".
wk list font family	Gibt den Namen der Schriftfamilie an, definiert über wk list font , um Marker für Listeneintrag anzuzeigen (nicht Absatztext). Standardwert ist "Times New Roman".
wk list start number	Setzt Startwert für geordnete Liste. Mögliche Werte: <ul style="list-style-type: none">• wk auto (Standard): Startwert richtet sich nach vorigem Listeneintrag, wenn vorhanden• Zahlenwert: Startwert
	Stringformat für Marker des Listeneintrags in Absatzrichtung links-nach-rechts. Wenn definiert, überschreibt er das standardmäßige Stringformat des Markers.
wk list string format LTR	<ul style="list-style-type: none">• Für ungeordnete Liste: String wird als Marker für Listeneintrag verwendet (in der Regel ein einzelnes Zeichen, z.B. "-")• Für geordnete Liste: String mit dem Zeichen "#command_5", das als Platzhalter für berechnete Zahlen oder Buchstaben dient. Standard ist "#command_5.", ist z.B. die aktuelle Zahl des Listeneintrags 15 und der Stiltyp für Liste Dezimal, lautet der String des Markers "15."
	Stringformat für Marker des Listeneintrags in Absatzrichtung rechts-nach-links. Wenn definiert, überschreibt er das standardmäßige Stringformat des Markers.
wk list string format RTL	<ul style="list-style-type: none">• Für ungeordnete Liste: String wird als Marker für Listeneintrag verwendet (in der Regel ein einzelnes Zeichen, z.B. "-")• Für geordnete Liste: String mit dem Zeichen "#command_5", das als Platzhalter für berechnete Zahlen oder Buchstaben dient. Standard ist "#command_5.", ist z.B. die aktuelle Zahl des Listeneintrags 15 und der Stiltyp für Liste Dezimal, lautet der String des Markers "15."
	Gibt in einer ungeordneten Liste ein Bild als Marker des Listeneintrags zurück. Mögliche Werte: <ul style="list-style-type: none">• wk none (Standard): Marker des Listeneintrags wird nicht über ein Bild definiert• Bild URL für lokale Datei (String). Kann absolut oder relativ zum Ressourcen-Verzeichnis der Anwendung sein.
wk list style image	Zurückgegebener Wert (WP GET ATTRIBUTES): URI (Netzwerk URL oder Daten URI). Kann sich von der ursprünglichen URL unterscheiden, wenn das Bild keine Referenz über die Netzwerk URL hat (nur Netzwerk URLs werden beibehalten). Für lokale Datei URLs wird der Bild-Stream selbst im Dokument beibehalten und so ist die zurückgegebene URL eine Daten URI mit dem Bild-Stream codiert in base64. Setzt Bildhöhe für Marker des Listeneintrags. Mögliche Werte: <ul style="list-style-type: none">• wk auto (Standard): Höhe richtet sich nach Bildgröße• Definierte Größe: Größe wird als Zahl oder Stringwert ausgedrückt:<ul style="list-style-type: none">◦ Zahl: Größe in wk layout unit.◦ String: CSS String mit zusammengesetztem Wert und Einheit, z.B.: 12pt für 12 Punkt oder 1.5cm für 1.5 Zentimeter. Mindestwert: 0pt, Maximumwert: 10000pt.
	Bild als Marker des Listeneintrags in einer ungeordneten Liste, definiert über eine URL (String). Mögliche Werte: <ul style="list-style-type: none">• wk none (Standard): Marker des Listeneintrags wird nicht über ein Bild definiert• Netzwerk URL oder Daten URI, absolut oder relativ zur Strukturdatei
wk list style image url	<ul style="list-style-type: none">• Zurückgegebener Wert (WP GET ATTRIBUTES): Netzwerk URL oder Daten URI. Kann sich von der ursprünglichen URL unterscheiden, wenn das Bild keine Referenz über die Netzwerk URL hat (nur Netzwerk URLs werden beibehalten). Für lokale Datei URLs wird der Bild-Stream selbst im Dokument beibehalten und so ist die zurückgegebene URL eine Daten URI mit dem Bild-Stream codiert in base64.
	Verwenden Sie wk list style image , wenn Sie Bilder als Marker des Listeneintrags als Bildausdrücke verwalten wollen.

Konstante Kommentar

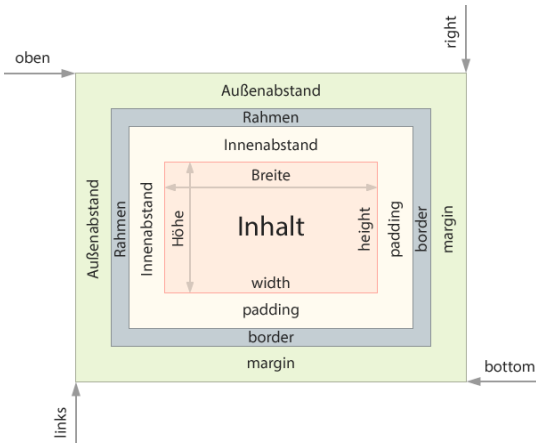
Gibt den Typ für Marker des Listeneintrags an. Mögliche Werte:

- [wk_disc](#) (Standard)
- [wk_circle](#)
- [wk_square](#)
- [wk_decimal](#): 1 2 3
- [wk_decimal_leading_zero](#): 01 02 03
- [wk_lower_latin](#): a b c
- [wk_lower_roman](#): i ii iii iv
- [wk_upper_latin](#): A B C
- [wk_upper_roman](#): I II III IV
- [wk_lower_greek](#): alpha, beta, gamma, etc.
- [wk_armenian](#)
- [wk_georgian](#)
- [wk_hebrew](#)
- [wk_hiragana](#)
- [wk_katakana](#)
- [wk_cjk_ideographic](#)
- [wk_hollow_square](#)
- [wk_diamond](#)
- [wk_club](#)
- [wk_decimal_greek](#)
- [wk_custom](#): ungeordnete Liste mit "-" als standardmäßigem Marker für Listeneintrag; dieser Stil eignet sich, um einen Marker für Listeneintrag mit [wk_list_string_format_LTR](#) oder [wk_list_string_format_RTL](#) anzupassen, ohne die standardmäßigen Marker für Listeneinträge zu verändern.
- [wk_none](#)

wk list
style type

Außenabstand

Außenabstände sind der Bereich außerhalb des Rahmens für ein Element. Sie sind transparent. Nachfolgendes Bild zeigt die verschiedenen Teile, die für eine Elementbox möglich sind:



Attribute für Außenabstand gelten für:

Dokumente	Absätze	Zeichen	Bilder	Tabellen	Zeilen	Spalte/Zellen	Kopf-/Fuß-/Hauptteile
X	X		X	X			X

Konstante Kommentar

wk inside Gibt bei mehreren Absätzen im ausgewählten Bereich an, dass das Attribut nur für die entsprechende innere Eigenschaft des Absatzes gilt (nicht außerhalb). Gilt nur für Attribute für Außenabstand, Innenabstand und Ränder und muss zum angegebenen Attribut hinzugefügt werden. Siehe Beispiel 2 des Befehls **WP SET ATTRIBUTES**.

Gibt Größe für alle Ränder des Elements an. Mögliche Werte:

- Größe wird als Zahl oder Stringwert ausgedrückt:
 - Zahl: Größe in [wk layout unit](#).
 - String: CSS String mit zusammengesetztem Wert und Einheit, z.B.: 12pt für 12 Punkt oder 1.5cm für 1,5 Zentimeter.
- [wk none](#) (Standard): Kein spezifischer Rand

Gibt Größe des unteren Rands eines Elements an. Mögliche Werte:

- Größe wird als Zahl oder Stringwert ausgedrückt:
 - Zahl: Größe in [wk layout unit](#).
 - String: CSS String mit zusammengesetztem Wert und Einheit, z.B.: 12pt für 12 Punkt oder 1.5cm für 1,5 Zentimeter.
- [wk none](#) (Standard): Kein spezifischer Rand

Gibt Größe des linken Rands eines Elements an. Mögliche Werte:

- Größe wird als Zahl oder Stringwert ausgedrückt:
 - Zahl: Größe in [wk layout unit](#).
 - String: CSS String mit zusammengesetztem Wert und Einheit, z.B.: 12pt für 12 Punkt oder 1.5cm für 1,5 Zentimeter.
- [wk none](#) (Standard): Kein spezifischer Rand

Gibt Größe des rechten Rands eines Elements an. Mögliche Werte:

- Größe wird als Zahl oder Stringwert ausgedrückt:
 - Zahl: Größe in [wk layout unit](#).
 - String: CSS String mit zusammengesetztem Wert und Einheit, z.B.: 12pt für 12 Punkt oder 1.5cm für 1,5 Zentimeter.
- [wk none](#) (Standard): Kein spezifischer Rand

Gibt Größe des oberen Rands eines Elements an. Mögliche Werte:

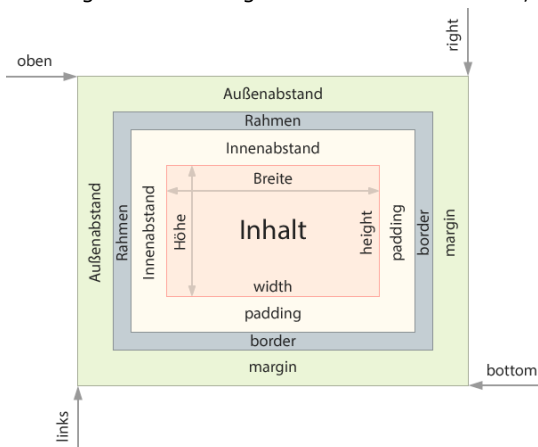
- Größe wird als Zahl oder Stringwert ausgedrückt:
 - Zahl: Größe in [wk layout unit](#).
 - String: CSS String mit zusammengesetztem Wert und Einheit, z.B.: 12pt für 12 Punkt oder 1.5cm für 1,5 Zentimeter.
- [wk none](#) (Standard): Kein spezifischer Rand

wk outside Gibt bei mehreren Absätzen im ausgewählten Bereich an, dass das Attribut nur für die entsprechende äußere Eigenschaft des Absatzes gilt (nicht innerhalb). Gilt nur für Attribute für Außenabstand, Innenabstand und Ränder und muss zum angegebenen Attribut hinzugefügt werden. Siehe Beispiel 2 des Befehls **WP SET ATTRIBUTES**.

Innenabstand

Innenabstände oder Einrückung sind der weiße Bereich zwischen Inhalt und Rahmen des Elements. Die Hintergrundfarbe des Elements wirkt sich auf den Innenabstand aus.

Nachfolgendes Bild zeigt die verschiedenen Teile, die sich für eine Elementbox konfigurieren lassen:



Attribute für Innenabstand/Einrückung gelten für:

Dokumente	Absätze	Zeichen	Bilder	Tabellen	Zeilen	Spalte/Zellen	Kopf-/Fuß-/Hauptteile
X	X		X	X*	X*	X	X

*gilt für Zellen

Konstante Kommentar

wk inside Gibt bei mehreren Absätzen im ausgewählten Bereich an, dass das Attribut nur für die entsprechende innere Eigenschaft des Absatzes gilt (nicht außerhalb). Gilt nur für Attribute für Außenabstand, Innenabstand und Ränder und muss zum angegebenen Attribut hinzugefügt werden. Siehe Beispiel 2 des Befehls **WP SET ATTRIBUTES**.

wk outside Gibt bei mehreren Absätzen im ausgewählten Bereich an, dass das Attribut nur für die entsprechende äußere Eigenschaft des Absatzes gilt (nicht innerhalb). Gilt nur für Attribute für Außenabstand, Innenabstand und Ränder und muss zum angegebenen Attribut hinzugefügt werden. Siehe Beispiel 2 des Befehls **WP SET ATTRIBUTES**.

Gibt Einrückung eines Elements an. Mögliche Werte:

- Einrückung wird als Zahl oder Stringwert ausgedrückt:
 - Zahl: Größe in wk layout unit.
 - String: CSS String mit zusammengesetztem Wert und Einheit, z.B.: 12pt für 12 Punkt oder 1.5cm für 1,5 Zentimeter.
- wk none (Standard): Keine spezifische Einrückung

Gibt Einrückung des Elements unten an. Mögliche Werte:

- Einrückung wird als Zahl oder Stringwert ausgedrückt:
 - Zahl: Größe in wk layout unit.
 - String: CSS String mit zusammengesetztem Wert und Einheit, z.B.: 12pt für 12 Punkt oder 1.5cm für 1,5 Zentimeter.
- wk none (Standard): Keine spezifische Einrückung

Gibt Einrückung linke Seite des Elements an. Mögliche Werte:

- Einrückung wird als Zahl oder Stringwert ausgedrückt:
 - Zahl: Größe in wk layout unit.
 - String: CSS String mit zusammengesetztem Wert und Einheit, z.B.: 12pt für 12 Punkt oder 1.5cm für 1,5 Zentimeter.
- wk none (Standard): Keine spezifische Einrückung

Gibt Einrückung rechte Seite des Elements an. Mögliche Werte:

- Einrückung wird als Zahl oder Stringwert ausgedrückt:
 - Zahl: Größe in wk layout unit.
 - String: CSS String mit zusammengesetztem Wert und Einheit, z.B.: 12pt für 12 Punkt oder 1.5cm für 1,5 Zentimeter.
- wk none (Standard): Keine spezifische Einrückung

Gibt Einrückung des Elements oben an. Mögliche Werte:

- Größe wird als Zahl oder Stringwert ausgedrückt:
 - Zahl: Größe in wk layout unit.
 - String: CSS String mit zusammengesetztem Wert und Einheit, z.B.: 12pt für 12 Punkt oder 1.5cm für 1,5 Zentimeter.
- wk none (Standard): Keine spezifische Einrückung

Absätze

Über Attribute für Absätze können Sie Eigenschaften für die Textgestaltung innerhalb eines Absatzes definieren. Sie gelten für:

Dokumente	Absätze	Zeichen	Bilder	Tabellen	Zeilen	Spalte/Zellen	Kopf-/Fuß-/Hauptteile
	X			X*	X*	X*	X*

*gilt für Absätze in Zellen

Konstante Kommentar

Gibt Textausrichtung für Absatz an. Mögliche Werte:

- wk direction
- [wk left to right](#) (Standard)
 - [wk right to left](#)

Gibt Zeilenabstand an. Mögliche Werte:

- wk line height
- [wk normal](#) (Standard): Wert richtet sich nach der Textgröße
 - Höhe wird als Zahl oder Stringwert ausgedrückt:
 - Zahl: Höhe in [wk layout unit](#).
 - String: CSS String mit zusammengesetztem Wert und Einheit, z.B.: 12pt für 12 Punkt oder 1.5cm für 1.5 Zentimeter. Auch relativer Wert (Prozent %) wird unterstützt.

Gibt die führenden Zeichen für Tabulatoren innerhalb Ausschnitten von Absätzen an.
Mögliche Werte:

- wk tab stop leadings
- Text Array: Sortierte Liste mit führenden Zeichen für jeden Tabulator-Versatz, ab dem linken Rand. Das durch den letzten Wert definierte Zeichen wird für alle weiteren im Absatz gesetzten Tabulatoren wiederholt.
 - Text: Ein einzelnes Zeichen wird als führendes Zeichen verwendet. Sind mehrere Zeichen angegeben, wird nur das erste verwendet.

Synchronisiert mit den Arrays [wk tab stop offsets](#) und [wk tab stop types](#). Passen die Arrays nicht in der Größe, wird das standardmäßige führende Zeichen verwendet (Nichts).

Gibt Tabulator-Stoppes für den Absatz an. Mögliche Werte:

- wk tab stop offsets
- Array mit Tabulatorwerten: Sortierte Liste mit absoluten Werten vom linken Rand aus. Der vom letzten Wert definierte Versatz wird für jeden weiteren im Absatz eingegebenen Tabulator-Stopp wiederholt. Ist der Versatz größer als die Absatzbreite, springt der Text in die nächste Zeile und startet mit dem ersten Tabulatorwert. Ist ein Wert im Array kleiner als der vorige, wird er ignoriert (außer für den letzten Wert im Array, der ein Standardversatz ist).
 - Skalärer Wert (Standard 35.45pt): Standardversatz für den ganzen Absatz. Der Befehl **WP GET ATTRIBUTES** gibt den letzten Versatz zurück (Standardversatz in Bezug auf den letzten absoluten Versatz).

Hinweis: Sie können Arrays und Skalare nicht im gleichen Aufruf für verschiedene Attribute verwenden. Werte werden über CSS Strings (Standard) oder Zahlen in [wk layout unit](#) ausgedrückt. Maximumwert ist 10.000pt.

Synchronisiert mit den Arrays [wk tab stop leadings](#) und [wk tab stop types](#).

Gibt Typ des Tabulator-Stoppes für den Absatz an. Mögliche Werte:

- wk tab stop types
- Array mit Werten vom Typ Tabulator-Stopp (wenn Tabulator-Stoppes über ein Array definiert wurden).
 - [wk left](#) (Standard): Setzt Text rechts vom Tabulator Stopp
 - [wk right](#): Setzt Text links vom Tabulator Stopp, bis der Abstand zum Tabulator gefüllt ist
 - [wk center](#): Zentriert Text am Tabulator-Stopp
 - [wk decimal](#): Richtet Text nach dem Dezimalzeichen aus
 - [wk bar](#): Vertikale Linie an der angegebenen Position

Synchronisiert mit den Arrays [wk tab stop offsets](#) und [wk tab stop leadings](#).

Gibt die horizontale Textausrichtung im Absatz an. Mögliche Werte:

- wk text align
- [wk left](#) (Standard)
 - [wk right](#)
 - [wk justify](#)
 - [wk center](#)

Gibt die Einrückung der ersten Zeile im Absatz an. Mögliche Werte:

- wk text indent
- Zahl: Größe in [wk layout unit](#). Standard ist 0.
 - String: CSS String mit zusammengezogenem Wert und Einheit, z.B.: 12pt für 12 Punkt oder 1.5cm für 1,5 Zentimeter. Mindestwert: 0pt, Maximumwert: 10000pt.

Setzt die vertikale Ausrichtung eines Elements. Verwendbar für Zeichen, Absätze und Bilder. Mögliche Werte:

- wk vertical align
- [wk baseline](#) (Standard): Richtet die Grundlinie eines Elements nach der Grundlinie des übergeordneten Elements aus
 - [wk top](#): Richtet die Oberkante des Elements nach der Oberkante des größten Elements in der Reihe aus
 - [wk bottom](#): Richtet die Unterkante des Elements nach dem untersten Element in der Reihe aus
 - [wk middle](#): Element wird in die Mitte des übergeordneten Elements gesetzt
 - [wk superscript](#): Richtet das Element aus, als ob es hochgestellt wäre
 - [wk subscript](#): Richtet das Element aus, als ob es tiefgestellt wäre

Für Zeichen haben [wk top](#) und [wk bottom](#) denselben Effekt wie [wk baseline](#).

Für Absätze haben [wk baseline](#), [wk superscript](#) und [wk subscript](#) denselben Effekt wie [wk top](#).

Stilvorlagen

Über Attribute für Stilvorlagen können Sie vordefinierte Stilvorlagen verwalten. Sie gelten für:

Dokumente	Absätze	Zeichen	Bilder	Tabellen	Zeilen	Spalte/Zellen	Kopf-/Fuß-/Hauptteile
	X		X**				X*

* Nur Lesen für Kopf- und Fußteile, sie werden von 4D Write Pro verwaltet (es gibt eine Stilvorlage pro Kopf- oder Fußteilbereich).

** Nur im Textfluss eingefügte Bilder.

Konstante	Kommentar
wk new line style sheet	Gibt zu verwendende Stilvorlage an, wenn im Absatz eine neue Zeile hinzugefügt wird. Mögliche Werte: <ul style="list-style-type: none">Name von vorhandener Stilvorlagewk none (Standard)
wk style sheet	Gibt aktuelle Stilvorlage für das/die ausgewählten Elemente an. Mögliche Werte: <ul style="list-style-type: none">wk none (Standard)Name von vorhandener Stilvorlage

Ausschnitte

Attribute für Ausschnitte sind nur lesbar (außer [wk id](#)). Sie gelten nur für Ausschnittobjekte.

Konstante	Kommentar
wk end	(Nur lesbares Attribut für Ausschnitt). Ausschnitt Ende Versatz. Wert: Lange Ganzzahl
wk id	(Nur lesbares Attribut) ID des Elements. Wert: String
wk owner	(Nur lesbares Attribut) Eigentümer des Ausschnitts/Objekts. Wert: Objekt
wk start	(nur lesbares Attribut für Ausschnitt). Ausschnitt Start Versatz. Wert: Lange Ganzzahl
wk type	(Nur lesbares Attribut für Ausschnitt) Typ des 4D Write Pro Objekts. Mögliche Werte: <ul style="list-style-type: none">wk type default: Typ Standardausschnittwk type paragraph: Typ Absatzausschnittwk type image: Verankertes Bildwk type container: z.B. Kopf- oder Fußteilwk type table: Referenz Tabellewk type table row: Referenz Tabellenzeilewk type table cell: Referenz Tabellenzellewk type table column: Referenz Tabellenspalte (nur für Spaltenausschnitt)wk type body: Referenz Hauptteil

Tabellen

Über Attribute für Tabellen können Sie Tabellenbereiche definieren. Sie gelten für:

Dokumente Absätze Zeichen Bilder Tabellen Zeilen Spalte/Zellen Kopf-/Fuß-/Hauptteile
X

Konstante	Kommentar
wk cell count	Gesamtanzahl der Zellen in der Reihe. Wertetyp: Lange Ganzzahl (Wert für wk type table row)
wk column count	(Nur lesbares Attribut für Ausschnitt) Gesamtanzahl der Spalten im Ausschnitt. Wert: Lange Ganzzahl
wk first column	(Nur lesbares Attribut für Ausschnitt) Nummer der ersten Spalte im Ausschnitt. Wert: Lange Ganzzahl
wk first row	(Nur lesbares Attribut für Ausschnitt) Nummer der ersten Zeile im Ausschnitt. Wert: Lange Ganzzahl
wk row count	(Nur lesbares Attribut für Ausschnitt) Gesamtanzahl der Zeilen im Ausschnitt. Wert: Lange Ganzzahl
wk table align	Setzt die horizontale Ausrichtung für ein Element der Tabelle. Mögliche Werte: <ul style="list-style-type: none">wk left (Standard)wk rightwk center
wk table ID	(Nur lesbares Attribut für Ausschnitt) ID der übergeordneten Tabelle des Ausschnitts. Wert: String

Werden unterschiedliche Attribute auf konkurrierende Elemente einer Tabelle angewandt, gilt folgende Priorität für das Rendern:

1. Die Tabelle wird zuerst gerendert
2. Zeilen werden gerendert (Überschreiben von Tabellenattributen)
3. Zellen/Spalten werden gerendert (Überschreiben von Zeilenattributen).