

4D Draw 6.0

ランゲージリファレンス
Windows® and Mac™ OS



4D Draw 6.0 ランゲージリファレンス Windows® and Mac™ OS

Copyright©1985 - 2000 4D SA

All rights reserved.

このマニュアルに記載されている事項は、将来予告なしに変更されることがあり、いかなる変更に関しても 4D SA は一切の責任を負いかねます。このマニュアルで説明されるソフトウェアは、本製品に同梱の License Agreement (使用許諾契約書) のもとでのみ使用することができます。

ソフトウェアおよびマニュアルの一部または全部を、ライセンス保持者がこの契約条件を許諾した上での個人使用目的以外に、いかなる目的であれ、電子的、機械的、またどのような形であっても、無断で複製、配布することはできません。

4th Dimension、4D Server、4D、4D ロゴ、およびその他の 4D 製品の名称は、4D SA の商標または登録商標です。

Microsoft と Windows は Microsoft Corporation 社の登録商標です。

Apple, Macintosh, Mac, Power Macintosh, Laser Writer, Image Writer, ResEdit, QuickTime は Apple Computer Inc. の登録商標または商標です。

その他、記載されている会社名、製品名は、各社の登録商標または商標です。

注意

このソフトウェアの使用に際し、本製品に同梱の License Agreement (使用許諾契約書) に同意する必要があります。ソフトウェアを使用する前に、License Agreement を注意深くお読みください。

第1章	4D Drawランゲージの概要	9
	はじめに.....	9
	4D Drawのマニュアル.....	9
	ランゲージの表記法について.....	9
	マルチプラットフォームでのドキュメントの管理.....	10
	ファイル管理の原則.....	10
	MacintoshとWindowsで作成されたドキュメントの対応.....	11
	マルチプラットフォームでのテンプレート.....	11
	フォントの管理.....	12
	アプリケーション/システムのフォント.....	12
	メソッドエディタ内のコマンド.....	13
	4D Drawエリア内のドキュメント.....	15
	エリアID番号とエリア変数.....	15
	レイアウト上の4D Drawエリア.....	16
	プラグインウインドウ上の4D Drawエリア.....	17
	Open external window.....	17
	オフスクリーンエリアの4D Draw.....	18
	デフォルトエリアの使用.....	20
	図形の参照.....	21
	図形ID番号.....	21
	文字図形へのアクセス不可.....	21
	図形座標の指定.....	22
	スコープの指定.....	23
	隠れている図形とバックグラウンド図形の修正.....	23
第2章	エリア制御 (Area Control)	25
	DR ADD TO BACKGROUND.....	25
	DR DO COMMAND.....	26
	DR Error.....	27
	DR EVENT FILTER.....	28
	DR EXPERT COMMAND.....	29
	DR EXPERT MODE.....	30
	DR GET AREA BOUNDARY.....	31
	DR Get update mode.....	32
	DR Get zoom.....	33

	DR LAST CLICK	34
	DR Last event	35
	DR MENU STATUS	36
	DR ON ERROR	37
	DR ON EVENT	38
	DR ON MENU	40
	DR REDRAW	41
	DR RELEASE BACKGROUND	42
	DR REMOVE FROM BACKGROUND	43
	DR SCROLL DOCUMENT	44
	DR SET ENTERABLE	45
	DR SET UPDATE MODE	46
	DR ZOOM	47
第3章	エリアオプション (Area Options)	4 9
	DR COORDINATES	49
	DR DISPLAY OPTIONS	50
	DR Get display	52
	DR GET DOCUMENT SIZE	53
	DR Get draw mode	54
	DR GET GLOBAL PREFERENCES	55
	DR GET PREFERENCES	56
	DR SET DISPLAY	58
	DR SET DOCUMENT SIZE	60
	DR SET DRAW MODE	61
	DR SET GLOBAL PREFERENCES	62
	DR SET PREFERENCES	63
第4章	エリア (Areas)	6 5
	DR AREA TO AREA	65
	DR AREA TO FIELD	66
	DR Area to picture	68
	DR DELETE OFFSCREEN AREA	69
	DR FIELD TO AREA	70
	DR NEW DRAWING	71
	DR New offscreen area	72
	DR OPEN DOCUMENT	73
	DR PICTURE TO AREA	75
	DR SAVE DOCUMENT	76
第5章	バインド (Binding)	7 9
	DR ACTIVATE BIND	79
	DR ADD TO BIND	81
	DR DEACTIVATE BIND	82
	DR DELETE BIND	83

	DR New bind	84
	DR REMOVE FROM BIND	85
第6章	属性を求める (Get Attributes)	87
	DR GET ARC SPECS	87
	DR Get attribute lock	89
	DR GET BOUNDARY	90
	DR Get corner rounding	91
	DR GET ENDMARKS	92
	DR GET FILL ATTRIBUTES	94
	DR Get handle state	96
	DR GET HIGHLIGHT	97
	DR Get ID	98
	DR GET LINE ATTRIBUTES	100
	DR GET LINE SPECS	102
	DR Get name	104
	DR Get object type	105
	DR GET POLYGON VERTEX	107
	DR Get refnum	108
	DR Get rotation	109
	DR Get text	110
	DR GET TEXT ATTRIBUTES	111
	DR Get text width	113
第7章	ホットリンク (Hot Links)	115
	DR ADD TO HOT LINK	115
	DR PUBLISH	117
	DR Subscribe	118
	DR UNPUBLISH	119
	DR UNSUBSCRIBE	120
第8章	取り込み / 書き出し (Import and Export)	121
	DR ARRAY TO ATTRIBUTE	121
	DR Array to polygon	123
	DR ATTRIBUTE TO ARRAY	125
	DR POLYGON TO ARRAY	127
第9章	図形作成 (Object Creation)	129
	DR Draw arc	129
	DR Draw line	131
	DR Draw oval	132
	DR Draw rectangle	133
	DR Draw text	135
	DR End polygon	137
	DR Objects to bitmap	138

	DR PLACE PICTURE	139
	DR POLYGON CURVE	140
	DR POLYGON LINE	142
	DR START POLYGON	143
第10章	図形操作 (Object Manipulation)	145
	DR ADD TO BITMAP	145
	DR ALIGN	146
	DR Count	148
	DR DELETE	149
	DR GROUP	150
	DR HIDE	151
	DR LOCK	152
	DR MOVE	154
	DR ROTATE	155
	DR SCALE	156
	DR SIZE	157
	DR UNGROUP	158
第11章	図形選択 (Object Selection)	159
	DR SELECT	159
	DR SELECT BY ATTRIBUTE	160
	DR SELECT BY REGION	161
第12章	プリント (Printing)	163
	DR PRINT	163
	DR PRINT BACKGROUND	164
	DR PRINT FOREGROUND	165
	DR PRINT MERGE	166
第13章	4D環境コマンド	167
	DR INSERT EXPRESSION	167
	DR INSERT FIELD	169
	DR Place field	171
	DR SET FORMAT	172
第14章	定規 (Rulers)	175
	DR ARRAY BASE TO SCALE	175
	DR ARRAY SCALE TO BASE	176
	DR Base to scale	177
	DR GET ORIGIN	178
	DR GET RULER	179
	DR GET RULER OPTIONS	181
	DR Scale to base	182
	DR SET ORIGIN	183
	DR SET RULER	184

	DR SET RULER OPTIONS	185
第15章	属性設定 (Set Attributes)	187
	DR SET ARC SPECS	187
	DR SET ATTRIBUTE LOCK	188
	DR SET CORNER ROUNDING	189
	DR SET ENDMARKS	190
	DR SET FILL ATTRIBUTES	191
	DR SET HANDLE STATE	192
	DR SET HIGHLIGHT	192
	DR SET LINE ATTRIBUTES	194
	DR SET LINE SPECS	195
	DR SET NAME	196
	DR SET POLYGON VERTEX	197
	DR SET REFNUM	198
	DR SET TEXT	199
	DR SET TEXT ATTRIBUTES	200
第16章	ユーティリティ (Utilities)	203
	DR Calculate area	203
	DR Calculate perimeter	204
	DR Clipboard to picture	205
	DR Color to index	206
	DR COLOR TO RGB	207
	DR Font name	208
	DR Font number	209
	DR Index to color	210
	DR PICTURE TO CLIPBOARD	211
	DR RGB to color	212
付録A	属性コード	213
付録B	エラーコード	215
付録C	イベントコード	219
付録D	図形コード	221
付録E	ユニットコード	223
付録F	コマンドコード	225
	コマンド索引	229

はじめに

4D Drawは、4th Dimensionのプログラム言語に約150のコマンドを追加するプラグインモジュールです。4D Drawの各コマンドを使用して、手動で行う操作を制御することができます。

例えば、4D Drawコマンドの実行、ドキュメントを開く、保存する、図形の描画、選択、修正、ホットリンクの作成と使用を行うことができます。

4D Drawのマニュアル

4D Drawは、ユーザリファレンスとランゲージリファレンスの2つのマニュアルで構成されています。このランゲージリファレンスでは、4D Drawで使用するプログラム言語の使用方法について説明します。4D Drawの使用方法については、『4D Drawユーザリファレンス』を参照してください。

ランゲージの表記法について

すべての4D Drawコマンドには“DR”という頭文字が付き、4th Dimensionの標準コマンドや他のモジュールによって追加されたコマンドと区別します。

このマニュアルでは、4D Drawの各コマンドを大文字の特殊フォント（例：**DR LOCK**）、4D Drawの関数は頭文字のみ大文字の特殊フォント（例：**DR Count**）で表記します。

4D Drawのコマンドや関数がメソッドやスクリプトの中に出てきた場合、それらは4th Dimensionにあるコマンドや関数と区別するために、太字の斜体で表現されます。太字のみの場合は、4th Dimensionの言語であることを示します。

このマニュアルの例題の中には、マニュアルスペースの関係上、本来、メソッド内では1行で作成されるコードが2、3行にまたがって表記されているものがあります。しかし、その例題を使用する場合は、enterキーを押したりせず、必ず1行でそのコードを入力してください。そうしないと、メソッドは正常に動作しません。

マルチプラットフォームでのドキュメントの管理

4D Drawは、4th Dimension、4D Serverのように、マルチプラットフォーム用に設計されたプログラムです。そのため、4D Drawで使用するデータベースは、Macintoshで作成されても、そのままWindowsで開く、または使用することができます（WindowsからMacintoshでも同様）。ただし、適切なバージョンのソフトウェアが必要です。互換性の詳細は、『4D Drawユーザリファレンス』第2章の「4th Dimensionとの互換性」の節を参照してください。

ファイル管理の原則

4D Drawドキュメントを両方のプラットフォーム下で管理するためには、以下に示す原則に特に注意してください。

Macintoshの場合、4D Drawはファイルの識別のために、そのドキュメントファイルのファイルタイプを使用します。例えば、ファイルタイプが"4DRW"のとき、それは4D Drawのファイルであることを示します。

絶対パスによる参照では、ディスク名、フォルダ名、ドキュメントファイル名を、それぞれコロン(:)で区切ります。次に例を示します。

Mydisk;Folder1:Folder2:Mydatabase

Windowsの場合、4D Drawはファイルの識別のために、ファイルの拡張子を使用します。".4DW"という拡張子が、4D Drawドキュメントであることを示します。また、Windows3.1.1以下のバージョンでは、ファイル名は拡張子を除いて半角8文字までです。絶対パスによる参照では、ディスク名、フォルダ名、ドキュメントファイル名を、それぞれ円記号(¥)で区切ります。次に例を示します。

D:¥Folder1¥Folder2¥Mydatabase

Macintoshで作成し、Windows上にコピーした4D Drawドキュメントは、拡張子を付けることによって直接開くことができます。例えば、"MyDoc"というドキュメントファイルを"MyDoc.4DW"というファイル名で保存し、それをWindowsマシン上にコピーすれば、Windowsのデータベースから特別な修正なしに直接開くことができます。

Windowsで作成し、Macintosh上にコピーした4D Drawドキュメントは、Macintoshでファイル識別をするための特殊なデータを含んでいないため、直接開くことができません。Macintosh上でこのドキュメントファイルを使用するためには、4th Dimensionや4D Serverで提供されている4D Transporterを使用しなければなりません。

注：4D Transporterに関する詳細は、『4D Transporterリファレンス』を参照してください。

MacintoshとWindowsで作成されたドキュメントの対応

次の表は、MacintoshとWindowsでの4D Drawドキュメントの対応表です。詳細については、**DR OPEN DOCUMENT**コマンドと**DR SAVE DOCUMENT**コマンドの説明を参照してください。

ドキュメント	Macintosh 種類	Windows 拡張子
4D Draw	4DRW	4DW
MacPaint	PNTG	.PNT
EPSF	EPSF	.EPS
PICT	PICT	.PCT

マルチプラットフォームでのテンプレート

サーバでのプラットフォームによらずに、テンプレートをMacintoshとWindowsクライアントで共有するには、以下のようにしてください。

テンプレートの読み込み時には、まずAreaName_4DWファイルを探します。見つからない場合には、拡張子を除いたAreaName_ファイルを探します。

このことを利用して、AreaName_4DWという名前でテンプレートを保存しておけば、プラットフォームによらずに、テンプレートを読むことができます。

フォントの管理

MacintoshとWindowsでは、フォントの管理方法が違います。Macintoshでは、インストールされた各フォントはそれぞれ個別の名前と番号を持っています。したがって、フォントを適用する時には、フォント名とフォント番号という、個別の識別子2つを参照することになります。

Windowsでもフォントはそれぞれ名前と番号を持っていますが、番号はフォントの書体によって変わっていて、太字、斜体等の各書体は異なる番号を持ちます。したがって、同じフォント名のフォントが異なるフォント番号を返すこともあります。

しかし、標準のフォントはいつも同じフォント番号を返します。コンピュータ上のASIFONT.MAPファイルは、Macintoshのフォントに対応するWindowsのフォントをリスト表示しているものです。テキストエディタでこのファイルを編集することができます。このファイルによって、4D Draw内では、**DR Font name**と**DR Font number**が影響を受けます。

注：Windowsでは、「アウトライン」と「シャドウ」の2つの書体は機能しません。

アプリケーション/システムのフォント

DR Font nameと**DR Font number**の2つの関数の戻り値は、システムにインストールされているフォントによって変わります。以下の2つのフォントは、システムフォントとアプリケーションフォントであり、それぞれ個別のフォント番号を持っています。

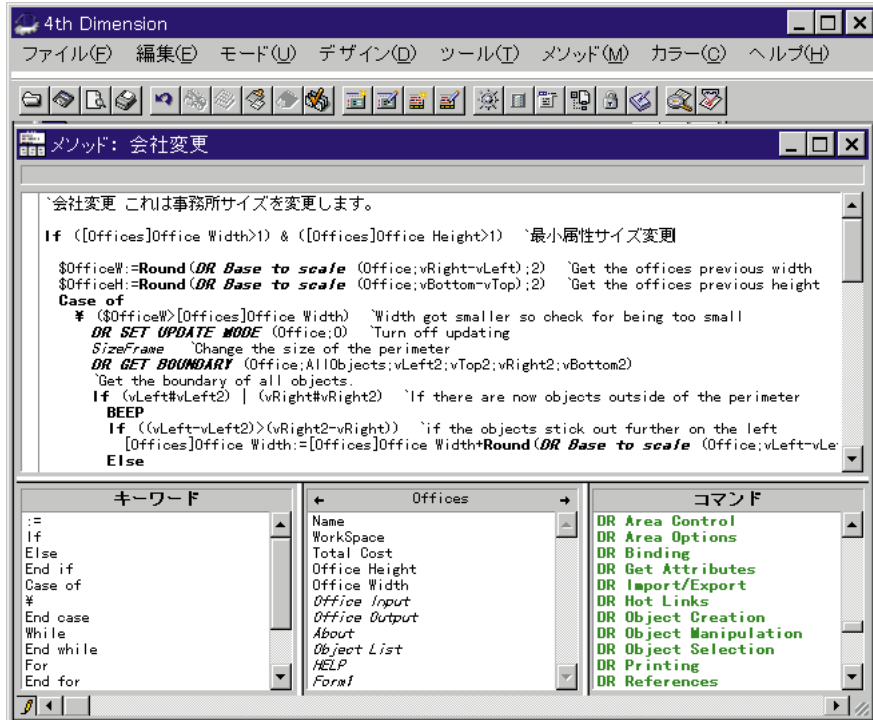
アプリケーションフォントは常にフォント番号1です。しかし、以下の注意が必要です。

Windowsでは、4D DrawのアプリケーションフォントはMSゴシックです。普通、MSゴシックフォントのフォント番号は21です。したがって、**DR Font name (1)**を実行すると"MSゴシック"を返し、**DR Font number ("MSゴシック")**を実行すると21を返します。

Macintoshでは、アプリケーションフォントはOsakaです。普通、Osakaフォントのフォント番号は3なので、**DR Font name (1)**を実行すると"Osaka"を返し、**DR Font number ("Osaka")**を実行すると3を返します。

メソッドエディタ内のコマンド

4D Drawの各コマンドは、「メソッド」エディタ内で項目別にグループ化され、「ルーチン」リストの最後にこれらの項目が追加されます。

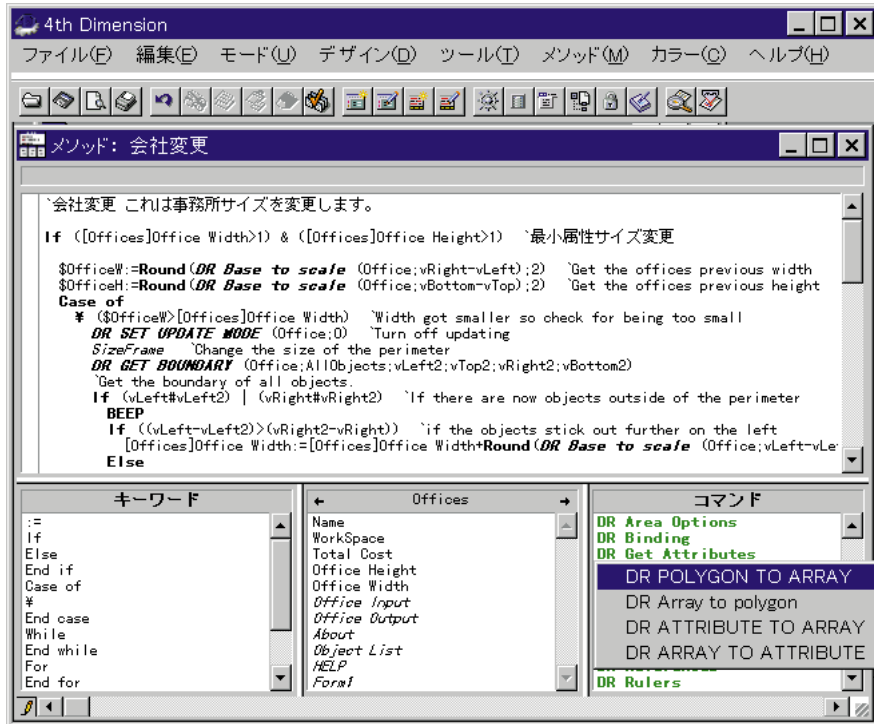


2つ以上のモジュールがインストールされている場合は、そのモジュールのインストール順にリスト表示されます。

4th Dimensionのコマンドが、アルファベット順に並んでいても、または項目別にグループ化されていても、4D Drawの各コマンドは常に項目別にグループ化されます。

4D Drawコマンドは、4th Dimensionコマンド同様、メソッドにキーボードから入力するか、または「コマンド」リストのポップアップメニューから選択して、プロシージャ内に配置します。

4D Drawコマンドは、プロジェクト、テーブル、レイアウト、オブジェクトの各メソッドで使用することができます。これらのメソッドは、4D Drawエリアを含んだレイアウトのオブジェクトで特に有効です。



4D Drawエリア内のドキュメント

4D Drawドキュメントは、次の3つのエリアのいずれかに存在します。

レイアウト上のプラグインエリア

プラグインウインドウ

オフスクリーンエリア

4D Drawドキュメントは、レイアウト上にプラグインエリアを作成するか、プラグインウインドウを開くことによって使用することができます。デザインモードでレイアウト上に描画することによってプラグインエリアを作ったり、ユーザモードで「ウインドウ」メニューから4D Drawを選択するか、**Open external window**関数を実行することができます。目に見えるエリアだけでなく、目に見えないエリアも作成できます。

エリアID番号とエリア変数

コマンドを使って4D Drawドキュメントを操作する場合は、4D DrawのエリアID番号でドキュメントを特定する必要があります。ドキュメントがどこにあると、4D Drawはドキュメントの格納とその操作のためにエリアID番号を必要とします。このエリアID番号は4D Drawに組み込まれた番号で、通常、変数で保存されています。

4D Drawは、4D Drawエリアやプラグインウインドウ、オフスクリーンエリアの場所を格納するために変数を使用します。この変数はコマンドや関数の引数（パラメータ）としてのエリアID番号を含んでおり、この変数を受け渡すことにより、操作を行いたいエリアを参照することができます。

次章のコマンドの説明において、<エリア>という引数（パラメータ）がドキュメントエリアを識別する変数を参照します。

エリア変数は、次の2つに分類されます。

プラグインエリアの名前

プラグインウインドウやオフスクリーンエリア作成用の変数

この2つの変数タイプに関しては、次節で説明します。

プラグインエリア名

4D Drawエリアを作成し、そのエリアに名前を付けると、4th Dimensionは自動的にそのエリアを参照する変数として、4D Drawエリアの名前を認識します。例えば、引数<エリア>に“間取図”を指定することにより、4D Drawエリア「間取図」を参照することができます。

プラグインウィンドウとオフスクリーンエリアの変数

Open external window関数や**DR New offscreen area**関数を使用して、プラグインウィンドウやオフスクリーンエリアを作成すると、これらの関数によって返されるエリアID番号は倍長整数型の変数に格納されます。そして、その変数を別のコマンドや関数のプラグインウィンドウやオフスクリーンエリアを参照するために使用することができます。変数内の値を格納するには、プロシージャにあるこれらの関数の左辺にその変数名と代入演算子(:=)を配置します。

ほとんどの4D Drawコマンドは、実行する際に対象となるエリアを指定する必要があります。

次の例は、4D Drawのプラグインウィンドウを作成し、「MyArea」変数にエリアID番号を格納します。

```
MyArea:=Open external window (30;30;350;450;8;"デザイン";"_4D Draw")
```

レイアウト上の4D Drawエリア

4D Drawドキュメントを4th Dimensionレイアウト上に作成したい場合、レイアウト上にプラグインエリアを作成し、それに固有の名前をつけて、4D Drawのプラグインエリアとして指定します。

4D Drawは、任意のレイアウト上に配置することができます。通常は、ドキュメントが操作できるように入力レイアウト上に配置します。また、出力レイアウト上に配置すれば、情報の表示とプリントを行うことができます。

4D Drawは、レイアウト上にフィールド等のレイアウトオブジェクトと共存することができます。

4D Drawは、プラグインエリアで使用します。このプラグインエリアは4th Dimensionの各アクティブオブジェクト（ボタン、入力可エリア、スクロールエリア等）のひとつで、4th Dimensionの各モジュールで使用されます。プラグインエリアに関する詳細は、『4th Dimension デザインリファレンス』と『4th Dimension ランゲージリファレンス』を参照してください。

レイアウト上でドキュメントを参照する場合は、4D Drawエリアを組み込んだプラグインエリアに設定した名前を使用します。

プラグインウィンドウ上の4D Drawエリア

4th Dimensionでは、4D Drawドキュメントをプラグインウィンドウという、4th Dimensionとは独立したエリアに作成できます。いつでも描画機能を使えるようにしたい場合にプラグインウィンドウは便利です。

4th Dimensionの**Open external window**関数は、メソッドの中から指定されたウィンドウを開き、エリアID番号を倍長整数型の変数として返します。プラグインウィンドウに対して操作を行う場合に、この変数を使用します。

Open external window

Open external window (左;上;右;下;タイプ;タイトル;プラグインエリア) 倍長整数

引数	タイプ	説明
左	数値	画面の左端からウィンドウの左端までのピクセル
上	数値	画面の上端からウィンドウの上端までのピクセル
右	数値	画面の左端からウィンドウの右端までのピクセル
下	数値	画面の上端からウィンドウの下端までのピクセル
タイプ	数値	ウィンドウのタイプ
タイトル	文字列	ウィンドウのタイトル
プラグインエリア	文字列	プラグインエリアのコマンド名
戻り値	数値	ウィンドウの参照番号

説明

Open external window関数は、新しいウィンドウを開き、4Dのプラグインによって提供されるプラグインエリアを表示します。

Open External window関数は、プラグインウィンドウやプラグインエリアを参照するための倍長整数型の変数を返します。

1~6番目の引数は**Open window**コマンドと同じですが、オプションにできるパラメータはなく、すべてのパラメータを指定する必要があります。

Open external window関数はウィンドウを作成します。ユーザの入力を待つことがないので、いくつものウィンドウを同時に開くこともできます。作成された各ウィンドウの中から、編集したいウィンドウをクリックします。ウィンドウがタイトルバーを持つ場合、ウィンドウを閉じるために、コントロールメニューボックス（Windowsの場合）またはクローズボックス（Macintoshの場合）が付きます。

例題

次の例では、プラグインウィンドウを開いて4D Writeのプラグインエリアを表示します。

```
wrWind:=Open external window (50; 50; 350; 450; 8; "タイトル"; "_4D WRITE")
```

次の例題では、上の例題で開いたプラグインウィンドウを閉じます。

```
CLOSE WINDOW (wrWind)
```

参照

なし

オフスクリーンエリアの4D Draw

オフスクリーンエリアはメモリ上に格納されるエリアで、プログラマーやユーザには見えません。オフスクリーンエリアを使用することにより、4D Drawドキュメントを表示する、あるいは保存する前にそのドキュメントへの修正を行うことができます。そして、必要なら、ユーザはそのドキュメントを元に戻すことができます。

オフスクリーンエリアにおける4D Draw操作は、画面をリドロウ（再描画）する必要がないため、とても高速に実行されます。

DR New offscreen area関数を使用すると、オフスクリーンエリアを作成することができます。**DR PICTURE TO AREA**コマンドを使用すると、オフスクリーンエリアに4D Drawエリアを含むことができる4Dのピクチャフィールドを配置することができます。

オフスクリーンエリアを使い終わったら、それをメモリからクリアし、メモリを解放してください。すべてのオフスクリーンエリアがクリアされる前にデータベースを閉じてしまうと、4th Dimensionはエラーメッセージを表示します。

次のようなメソッドをプロジェクトメソッドに作成すると、4D Drawドキュメント保存用のオフスクリーンエリアを作成します。レイアウト上のボタンを使用することにより、ユーザはオリジナルドキュメントに戻すことができます。

```
Area:=DR New offscreen area
```

```
QUERY ([Designs];[Designs]CustID=vCustID)
```

```
If (Records in selection ([Designs])=1)
```

```
    DR PICTURE TO AREA (Area;[Designs]FloorPlan_)
```

```
        `オフスクリーンエリアに間取図を格納する
```

```
    MODIFY RECORD ([Designs])
```

```
        `デザインレコードを修正する
```

DR DELETE OFFSCREEN AREA (Area)

　`オフスクリーンエリアで使用されているメモリを解放する

End if

入力レイアウト上にボタンを作成し、そのボタンに次のようなスクリプトを割り当てます。

Review:=**DR Area to picture** (Area;-2)

　`間取図レイアウト上のプラグインエリア内にオリジナルドキュメントを含んだオフ

　`スクリーンエリアを配置する。

デフォルトエリアの使用

デフォルトエリアはRAM上にあるテンプレートのことで、4D Drawエリアとプラグインウインドウのデフォルト属性を設定するために使用されます。4D Drawエリアに関するコマンドはすべて、引数<エリア>に-1を指定することができます。デフォルトエリアについても、他のエリアと同じように操作することができます。

デフォルトエリアを使うと、4D Drawエリアに関する無駄な操作を省くことができます。例えば、4D Drawエリアとプラグインウインドウをスクロールバーなしで表示したい場合、各フォームのOn Loadイベントでスクロールバーを表示しないように操作する必要はありません。

4D Drawエリアとプラグインウインドウの両方の属性を設定することができます。デフォルトエリアは、レイアウトやプラグインウインドウが開かれる場合に必ず、自動的にテンプレートとして使われます。コードを実行する必要がないので、デフォルトエリアにより、描画エリアを素早くカスタマイズすることができます。

必ずしもすべての4D Drawエリアにデフォルトエリアを適用する必要がない場合、ディスク上に4D Drawエリアのテンプレートを作るか、またはフォームのOn Loadイベントにコードを作ってデフォルトエリアを無効にすることもできます。ディスク上のテンプレートやフォームのOn Loadイベントのコードは、デフォルトエリアに優先します。

図形の参照

図形ID番号

4D Drawドキュメント内の各図形は、図形IDという固有の番号を持っています。図形IDは、図形の作成時に4D Drawによって自動的に割り当てられます。

つまり、パレットで図形を描いたり、クリップボードから図形をペーストする、または複数の図形をグループ化する、既存の図形を複製する、ホットリンクを使用する、フィールド参照をペーストするたびに、新しいIDが割り当てられます。図形IDは固有の番号なので、図形を参照するのに便利です。また、図形IDが、ドキュメント内で再利用されることはありません。たとえその図形が削除されても、ドキュメントが存在する間は“未使用”のままになります。

図形IDが譲渡されることはありません。1つの4D Drawドキュメント内のIDが5の図形を他のドキュメントにペーストしても、同じIDを持つとは限りません。

図形作成用の標準コマンドは、すべて関数で、作成した図形IDを返します。パレットで作成した図形と違い、コマンドで作成した図形は自動的に選択されません。

DR Get ID関数を使用して、図形IDを獲得することができます。

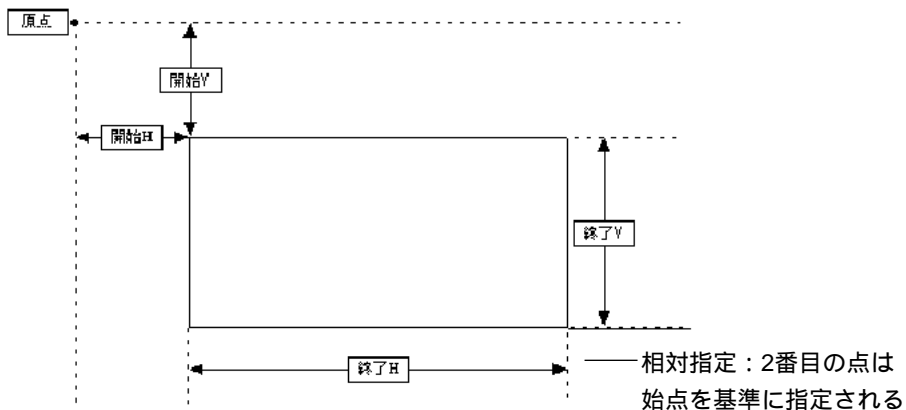
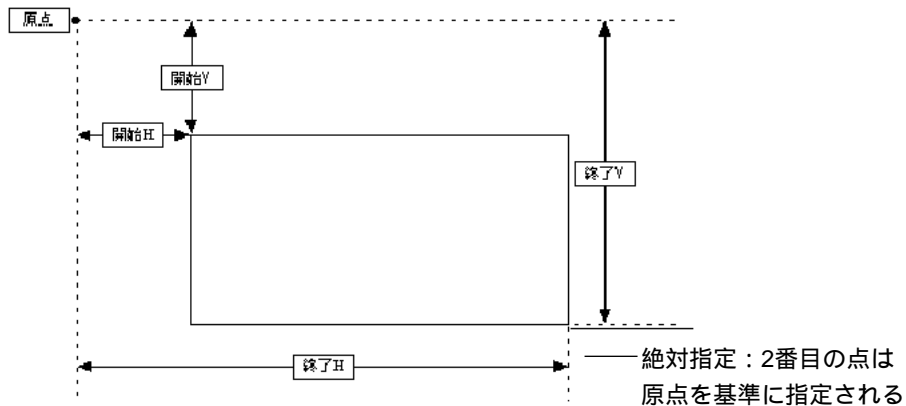
文字図形へのアクセス不可

文字図形の属性をアクセス不可にしたい場合は、「図形」メニューの「文字」と「属性」メニューアイテムの両方を選択不可にする必要があります。それは、文字図形はどちらの方法でもアクセスすることができるからです。

文字図形以外の図形に対する「属性」メニューアイテムへのアクセスを使用不可にしない場合は、そのメニューアイテムが選択されたかどうかを検査する**DR ON MENU**コマンドを使用するメソッドをインストールします。すると、「属性」メニューアイテムの選択前にユーザが選択したタイプを調べるだけで済みます。もし、選択範囲に文字タイプの図形を含んでいると、警告ボックスを表示します。含んでいない場合は、**DR DOCOMMAND**コマンドを使用して、そのメニューアイテムを実行することができます。

図形座標の指定

図形の位置とサイズのことを“座標”といいます。座標を設定または指定するコマンドは、すべて「ベースの単位」で処理を行います。「ベースの単位」は「定規設定」ダイアログボックスの最初のポップアップメニューで指定する単位です。「ベースの単位」と「目盛の単位」を変換する場合は、**DR Scale to base**関数と**DR Base to scale**関数を使います。



座標が正の場合は、位置が右または下、座標が負の場合は、位置が左または上であることを表します。

スコープの指定

4D Drawのほとんどのコマンドは、<スコープ>という引数を持っています。4D Drawドキュメント内に図形や文字列を指定する<スコープ>はコマンドによって影響を受けません。

次の表は、スコープの一般的な規則です。所定のコマンドにスコープがどのように反映されるかは、第16章の各コマンドの説明を参照してください。

スコープ	対象
>0	図形ID
0	選択図形
-1	全図形
-2	デフォルト
-3	選択文字列
-4	フォアグラウンドの図形
-5	バックグラウンドの図形

隠れている図形とバックグラウンド図形の修正

4D Drawコマンドを使用して、隠れている（目に見えない）図形やバックグラウンドの図形を選択することができます。これらの図形を選択する際、4D Draw上の操作を戻す前に必ず次のような処理を行なってください。

1. 図形を選択する。
2. 選択した図形上で修正を行う。
3. 図形を選択解除する。

図形が選択されたままだと、その図形上で誤って修正等が行われるかもしれないので、特に3番目の処理は大切です。

エリア制御 (Area Control)

DR ADD TO BACKGROUND

DR ADD TO BACKGROUND (エリア;スコープ)

引数	タイプ	説明
エリア	倍長整数型の変数	4D Drawエリア
スコープ	倍長整数	-1=全図形、0=選択図形、>0=図形ID

説明

DR ADD TO BACKGROUNDコマンドは、<エリア>のバックグラウンドに<スコープ>で指定された図形を追加します。このコマンドは、「バックグラウンドに追加」メニューアイテムと同じ機能を持ちます。

<スコープ>で指定された図形はエリアのバックグラウンドの一部になり、選択できなくなります。図形を修正するには、「バックグラウンド解除」メニューアイテムまたは**DR RELEASE BACKGROUND**コマンドを使用してバックグラウンドを解除する必要があります。バックグラウンド上の図形は、メソッドを通してしか修正することはできません。

参照

DR REMOVE FROM BACKGROUND

DR DO COMMAND

DR DO COMMAND (エリア;コマンド)

引数	タイプ	説明
エリア	倍長整数型の変数	4D Drawエリア
コマンド	倍長整数	コマンドコード

説明

DR DO COMMANDコマンドは、<コマンド>に指定したメニューアイテムを、あたかもユーザが4D Drawメニューからそれを選択したかのように実行します。このコマンドは、メソッドに存在しない動作を実行するのに使用します。

<コマンド>に指定する値は、「付録F コマンドコード」を参照してください。これらの数値は、4D Drawがバージョンアップされ、メニューアイテムが変更または移動しても変わりません。

例題

次の例は、エリアの選択図形を最背面へ移動します。

DR DO COMMAND (Area;5002)

参照

DR MENU STATUS

DR Error

DR Error ({メッセージ}) 整数

引数	タイプ	説明
メッセージ	テキスト	エラーメッセージを受け取るテキスト変数
戻り値	整数	4D Drawで実行された最終操作の状態

説明

DR Error関数は、4D Drawが実行した最終操作の状態を表す数値を返します。

DR Error関数が0の場合は、最終操作にはエラーが生じていませんが、0でない場合は、最終操作でエラーが生じています。また、同じレイアウト上で複数のエリアがアクティブな場合は、**DR Error**関数はそれらのエリアを区別することなく最終エラーを返します。エラーコードリストは「付録C イベントコード」を参照してください。

オプション引数<メッセージ>を指定する場合、<メッセージ>は変数でなければなりません。呼び出し後にエラーメッセージが設定されます。

エラーが一般的なデータベース管理エラーの場合、**DR Error**関数は4Dのエラーコードを返します。

参照

DR ON ERROR

DR EVENT FILTER

DR EVENT FILTER (エリア;マスク)

引数	タイプ	説明
エリア	倍長整数型の変数	4D Drawエリア
マスク	倍長整数	処理するイベント

説明

DR EVENT FILTERコマンドは、<エリア>のオブジェクトメソッドまたはイベントメソッドを実行するイベントを指定します。

デフォルトでは、ユーザがエリア外のオブジェクトを選択した時に、4D Drawエリアに設定したスクリプトが実行されます。**DR EVENT FILTER**コマンドを使用してオブジェクトメソッドを実行する他のイベントを指定することができます。さらに、**DR ON EVENT**コマンドでインストールしたメソッドを実行することもできます。

<マスク>は、使用するイベントで、イベントコードの和として表します。イベントコードは、「付録C イベントコード」を参照してください。

<エリア>に-1を指定した場合は、レイアウト上とプラグインウィンドウ上に新たに作成されたすべての4D Drawエリアにイベントフィルタがデフォルトフィルタとして設定されます。これにより、「ユーザ」モードの「ウィンドウ」メニューから作成されたエリアをトラップすることができます。また、各エリアを処理する一貫性のあるイベント処理が可能になります。

参照

DR Last event、**DR ON EVENT**、**DR LAST CLICK**

DR EXPERT COMMAND

DR EXPERT COMMAND (エリア;コマンド;状態)

引数	タイプ	説明
エリア	倍長整数型の変数	4D Drawエリア
コマンド	倍長整数	コマンドコード
状態	整数	0=使用可、1=使用不可

説明

DR EXPERT COMMANDコマンドは、4D Drawのエキスパートモードのメニューコマンドを使用可または使用不可にします。

<状態> が0の場合は、<コマンド> で指定したメニューコマンドがエキスパートモードで使用可になり、<状態> が1の場合は、使用不可になります。<コマンド> に設定する値は「付録B エラーコード」を参照してください。

メニューアイテムが**DR EXPERT COMMAND**コマンドで使用不可の場合でも、**DR DOCOMMAND**コマンドを使用すれば、メニューアイテムを実行することができます。

特定のメニューアイテムを使用不可にすると、4D Drawの他の操作にも反映されます。例えば、「フルウインドウ」メニューアイテムを使用不可にすると、4D Drawメニューバー上のズームボックスも削除されます。また、「属性設定...」メニューアイテムを使用不可にすると、図形をダブルクリックしても「属性設定」ダイアログボックスは表示されません。「ロック」メニューアイテムを使用不可にすると、「属性設定」ダイアログボックス内の「ロック」ボタンも使用不可になります。

例題

次の例は、「属性設定...」メニューアイテムを使用不可にして、「属性設定」ダイアログボックスが開けないようにします。

DR EXPERT COMMAND (Area;4018;1)

DR EXPERT MODE (Area;1)

参照

DR EXPERT MODE

DR EXPERT MODE

DR EXPERT MODE (エリア;モード)

引数	タイプ	説明
エリア	倍長整数型の変数	4D Drawエリア
モード	整数	0=オフ、1=オン

説明

DR EXPERT MODEコマンドは、エキスパートモードをオンまたはオフにします。エキスパートモードをオンにすると、4D Drawメニューの特定のメニューアイテムが使用不可になります。

<モード> が1の場合は、エキスパートモードがオンになり、<モード> が0の場合は、オフになります。エキスパートモードにすると、**DR EXPERT COMMAND**コマンドで指定されたメニューアイテムが使用不可になります。

例題

DR EXPERT COMMANDコマンドの例を参照してください。

参照

DR EXPERT COMMAND

DR GET AREA BOUNDARY

DR GET AREA BOUNDARY (エリア;左;上;右;下)

引数	タイプ	説明
エリア	倍長整数型の変数	4D Drawエリア
左	実数型の変数	エリアの左の境界
上	実数型の変数	エリアの上の境界
右	実数型の変数	エリアの右の境界
下	実数型の変数	エリアの下の境界

説明

DR GET AREA BOUNDARY コマンドは、<エリア>の表示範囲の座標を、<左>、<上>、<右>、<下>の各変数に返します。座標は、原点の位置を基準に“ベースの単位”で表します。“ベースの単位”から“目盛の単位”に変換する場合は、**DR Base to scale**関数を使用します。

例題

次の例は、選択図形を表示エリアの左上端に移動します。

DR GET AREA BOUNDARY (Area;\$Left;\$Top;\$Right;\$Low)

DR MOVE (Area;0;\$Left;\$Top;0)

参照

DR GET BOUNDARY

DR Get update mode

DR Get update mode (エリア) 整数

引数	タイプ	説明
エリア	倍長整数型の変数	4D Drawエリア
戻り値	整数	スクリーンの更新状態

説明

DR Get update mode関数は、<エリア>の更新状態を示す整数を返します。

0が返された場合、画面更新はオフ、1が返された場合、画面更新はオンです。**DR Get update mode**関数は、構造化プログラム設計に便利です。

例題

次の例は、画面更新をオフにし、複数の修正を実行するプロジェクトメソッドを呼び出した後、画面更新の状態をリセットします。このコマンドは、ネストしたメソッドを作成するときに便利です。

```
$Mode:=DR Get update mode (Area)      `更新モードを求める  
DR SET UPDATE MODE (Area;0)          `画面更新をオフにする  
REFORMAT(Area)                       `エリアは倍長整数  
DR SET UPDATE MODE (Area;$Mode)     `更新モードをリセットする
```

参照

DR SET UPDATE MODE、**DR REDRAW**

DR Get zoom

DR Get zoom (エリア) 数値

引数	タイプ	説明
エリア	倍長整数型の変数	4D Drawエリア
戻り値	数値	4D Drawエリアの表示倍率

説明

DR Get zoom関数は、<エリア>のズーム倍率を返します。この関数から返される数値は、4D Drawエリアの左下隅にある「ズーム倍率」インジケータに表示されている数値と同じものです。

参照

なし

DR LAST CLICK

DR LAST CLICK (エリア;横;縦)

引数	タイプ	説明
エリア	倍長整数型の変数	4D Drawエリア
横	実数型の変数	最後にクリックした横の位置
縦	実数型の変数	最後にクリックした縦の位置

説明

DR LAST CLICKコマンドは、<エリア>でマウスを最後にクリックした位置を原点の位置を基準に<横>と<縦>の各変数に返します。

<横>と<縦>はどちらも“ベースの単位”で表します。“ベースの単位”から“目盛の単位”に変換する場合は、**DR Base to scale**関数を使用します。

DR LAST CLICKコマンドは、マウスボタンから指を離れた位置ではなく、マウスでクリックした位置を返します。ドラッグやサイズ変更のような操作の場合は、マウスボタンを押した時にポインタがあった位置を返します。エリアのオブジェクトメソッド、またはイベントメソッドで**DR LAST CLICK**コマンドを使用すれば、図形のクリックされた位置を求めることができます。

例題

次の例は、レイアウト上の4D Drawエリアのスクリプトです。このスクリプトが実行されるたびに、**DR LAST CLICK**コマンドはマウスの位置を“vHorizontal”と“vVertical”の各変数に格納します。これらの変数をレイアウト上に表示すれば、マウスがクリックされた位置を迅速かつ正確にフィードバックすることができます。

DR LAST CLICK (Area;vHorizontal;vVertical)

参照

DR EVENT FILTER、**DR Last event**

DR Last event

DR Last event (エリア) 倍長整数

引数	タイプ	説明
エリア	倍長整数型の変数	4D Drawエリア
戻り値	倍長整数	エリア内で最後に起こったイベント

説明

DR Last event関数は、<エリア>に生じた最終イベントのコードを返します。

DR Last event関数は4D Drawエリアのオブジェクトメソッド、または**DR ON EVENT**コマンドを使用してインストールしたイベントメソッドで実行したイベントを認識するのに使用します。**DR EVENT FILTER**コマンドと共に使用すると、ユーザの操作をもとにメソッドを実行することができます。イベントコードについては、「付録C イベントコード」を参照してください。

例題

1.次の例は、4D Drawエリアのオブジェクトメソッドです。このメソッドは、Ctrlキーを押しながらクリック（Windowsの場合）またはcommandキーを押しながらクリック（Macintoshの場合）によりスクリプトが実行されたかどうかを確認し、選択図形が1つだけかどうかを判定します。これらの条件が共に満たされた場合は、その図形の参照番号をプロジェクトメソッドに受け渡します。

```

If (DR Last event (Area)=64           `Ctrlキーを押しながらクリック(Windowsの場合)または
                                     `commandキーを押しながらクリック(Macintoshの場合)
                                     `を確認する
      If (DR Count (Area;0)=1)         `選択図形の数を判定する
          PART INFO (DR Get return (Area;0)) `参照番号をPART INFOに受け渡す
      End if
End if

```

2.次のメソッドは“PART INFO”です。このメソッドは部品を検索し、中央にウインドウを開くプロジェクトメソッドを呼び出した後、修正するレコードを表示します。

```

QUERY ([Parts];[Parts]ID=$1)       `図形に関連するレコードを検索する
CENTERWINDOW (400;300)            `中央のウインドウを開く
MODIFY RECORD ([Parts];*)         `修正を要求にする
CLOSE WINDOW                      `ウインドウを閉じる

```

参照

DR EVENT FILTER、**DR ON ERROR**

DR MENU STATUS

DR MENU STATUS (エリア;コマンド;チェック;アクティブ;名前)

引数	タイプ	説明
エリア	倍長整数型の変数	4D Drawエリア
コマンド	倍長整数	コマンドコード
チェック	整数型の変数	0=チェックマークなし、 1=チェックマークあり
アクティブ	整数型の変数	0=使用不可、1=使用可
名前	文字列型の変数	メニューアイテムの名前

説明

DR MENU STATUSコマンドは、<エリア>のメニューコマンドに関する情報を、<チェック>、<アクティブ>、<名前>の各変数に返します。

引数<コマンド>で使用できる値は、「付録F」に記述しています。

<アクティブ>が0の場合、メニューアイテムは使用不可、1の場合は、使用可です。

「チェック」が0の場合は、メニューアイテムにチェックマークが付いていません。1の場合は、チェックマークが付いています。チェックが意味を持つメニューアイテムは、「変形」、「回転」、「実サイズ」、「縮小」、「グリッドに吸着」、および「表示」サブメニュー内の全項目です。「チェック」は、チェックマークの有無を認識する以外に、「値を表示」と「参照名を表示」を認識するのにも使用できます。「値を表示」がメニュー上に表示されている場合は、0になります。「参照名を表示」がメニュー上に表示されている場合は、1になります。

「名前」は、メニューアイテムの文字です。

例題

次の例は、**DR MENU STATUS**コマンドを使用して「グリッドに吸着」が現在アクティブ状態かどうかを確認します。

```
DR MENU STATUS (Area;6008;vCheck;vActive;vName)    `6008は「グリッドに吸着」  
If (vCheck=1)                                       `グリッドがオンの場合  
    vMessage:="「グリッドに吸着」がアクティブ状態です。" `ユーザに知らせる  
End if
```

参照

DR DO COMMAND

DR ON ERROR

DR ON ERROR (メソッド)

引数	タイプ	説明
メソッド	文字列	実行するメソッドの名前

説明

DR ON ERRORコマンドは、4D Drawのエラー発生時の割り込みメソッドとして<メソッド>をインストールします。

<メソッド>に空の文字列を指定するとメソッドが呼び出されなくなります。エラー処理メソッドをインストールすると、4D Drawのエラーが生じた時に<メソッド>が呼び出されます。

<メソッド>が呼び出されると、4D Drawはエラーを管理する3つの引数(\$1、\$2、\$3)を受け渡します。\$1はエラーが生じた4D Drawエリアを表す倍長整形の変数です。エラーが4D Drawエリアに限定されない場合は0になります。\$2はエラーコードを表す整数です。\$3はテキストタイプで、内容は、エラーメッセージです。\$2と\$3は、**DR Error**関数を参照するのと同じです。

データベースをコンパイルする予定がある場合は、これらの引数のタイプを下記のように事前に宣言しておく必要があります。

```
C_LONGINT ($1;$2)
C_TEXT ($3)
```

例題

1 次の例は、エラー処理メソッドのインストール方法を示しています。

DR ON ERROR ("描画エラ - ")

2 次のメソッドは“描画エラ - ”です。“描画エラ - ”は、\$1を判定して特定のエリアでエラーが発生したかどうかを調べた後、エラーコードとエラーメッセージを警告ボックスに表示します。

```
C_LONGINT ($1;$2)
C_TEXT ($3)
```

```
If ($1=Area)
    ALERT ("4D Drawエリア “ Area ” でエラーが発生しました")
End if
ALERT ("エラーコード"+String ($2)+Char (13)+$3)
```

参照

DR Error

DR ON EVENT

DR ON EVENT (メソッド)

引数	タイプ	説明
メソッド	文字列	実行するメソッドの名前

説明

DR ON EVENTコマンドは、「指定したイベントが生じた時に実行するイベントメソッドをインストールします。

メソッドを実行するイベントは、**DR EVENT FILTER**コマンドで設定します。 <メソッド> に空の文字列を指定した場合、メソッドは実行されません。エリアにスクリプトとイベントメソッドの両方がある場合は、スクリプトが最後に実行されます。**DR ON EVENT**コマンドはプラグインウインドウ上の4D Drawエリアの場合、特に便利です。なぜなら、プラグインウインドウ上のエリアにはスクリプトが設定できないからです。

<メソッド> を呼び出すと、4D Drawはイベントを管理するための4つの引数 (\$1、\$2、\$3、\$4) を受け渡します。\$1はイベントが生じた4D Drawエリアを表す倍長整数型の変数です。\$2はイベントコードを表す整数で、**DR Last event**関数を呼び出すのと同じです。イベントコードについては、「付録C イベントコード」を参照してください。\$3はエリアが属するレイアウトのファイル番号で、-1の場合は、エリアがプラグインウインドウ上にあります。

\$4はエリアが自動保存されるフィールドの番号で、0の場合は、エリアは自動保存されません。

データベースをコンパイルする予定がある場合は、これらの引数のタイプを下記のように事前に宣言しておく必要があります。

C_LONGINT (\$1;\$2;\$3;\$4)

例題

1.次の例は、イベントメソッドのインストール方法を示しています。プラグインウインドウを開いて、イベントメソッド “ EventProc ” をインストールし、イベントとしてCtrlキーを押しながらクリック (Windowsの場合) またはcommandキーを押しながらクリックを指定します。

`プラグインウインドウを開く

```
vArea:=Open externall window (20;50;400;350;0;"描画";"_4D Draw")
```

```
DR ON EVENT ("EventProc")
```

```
DR EVENT FILTER (vArea;64)
```

```
`メソッド “ EventProc ” をインストールする
```

```
` 「Ctrl+クリック」 ( Windowsの場合 ) または
```

```
` 「command+クリック」 ( Macintoshの場合 )
```

```
`でメソッドを呼び出す
```

2.次のメソッドは“EventProc”です。このメソッドは選択図形の数を確認し、図形がひとつだけ選択された場合は、その図形の名前を「リクエスト」ダイアログボックスに表示して変更できるようにします。

C_LONGINT (\$1;\$2;\$3;\$4)

If (*DR Count*(vArea)#1) `1つ以上の図形を選択または未選択の場合

ALERT ("1つの図形だけを選択してください") `ユーザに警告する

Else

 \$Name:=*DR Get name* (vArea;0) `選択図形の名前を求める

 \$Name:=**Request** ("図形の名前...";\$Name) `新しい名前の入力要求

If (OK=1) `ユーザが入力した場合

DR SET NAME (vArea;0;\$Name) `新しい名前を設定する

End if

End if

参照

DR EVENT FILTER、**DR Last event**

DR ON MENU

DR ON MENU (エリア;メソッド)

引数	タイプ	説明
エリア	倍長整数型の変数	4D Drawエリア (-1=すべてのエリア)
メソッド	文字列	呼び出すメソッドの名前

説明

DR ON MENUコマンドは、「ユーザ」モードや「ランタイム」モード、あるいは**DR DO COMMAND**コマンドによってメニューアイテムが動作するたびに、<メソッド>を実行します。

メソッドには、3つの引数が渡されます。

\$1：倍長整数。4D DrawエリアのIDが入っています。

\$2：倍長整数。メニューアイテム番号が入っています。

\$3：倍長整数。キー入力されたモディファイアキー番号が入っています。

引数\$3には、次に示すModifierキー（またはその組み合わせ）がセットされます。

	Macintosh	Windows
0	モディファイアキーなし	モディファイアキーなし
1	commandキー	Ctrlキー
2	shiftキー	shiftキー
4	optionキー	Altキー
8	Ctrlキー	

モディファイアキーの組み合わせがキーインされた場合、渡される値はキーに対応する値の和になります。例えば、値“10”は、ユーザがメニューを選択した時に、shiftキーとCtrlキーを一緒に押していたことを示します。

注：データベースをコンパイルする場合は、**DR ON MENU**コマンドから実行するメソッド内で変数\$1、\$2、\$3を宣言しておく必要があります。

参照

なし

DR REDRAW

DR REDRAW (エリア)

引数	タイプ	説明
エリア	倍長整数型の変数	4D Drawエリア

説明

DR REDRAWコマンドは、<エリア>を再描画します。このコマンドは、**DR SETUPDATE MODE**コマンドで画面更新をオフにして、4D Drawエリアを再描画する場合に便利です。

例題

次の例は、画面更新をオフにし、複数の修正を実行するプロジェクトメソッドを呼び出した後、画面更新をオンに戻さずに<エリア>を再描画します。

DR SET UPDATE MODE (Area;0)	`画面更新をオフにする
REFORMAT (Area)	`エリアは倍長整数
DR REDRAW (Area)	`再描画して変更内容を表示する

参照

DR SET UPDATE MODE、**DR Get update mode**

DR RELEASE BACKGROUND

DR RELEASE BACKGROUND (エリア)

引数	タイプ	説明
エリア	倍長整数型の変数	4D Drawエリア

説明

DR RELEASE BACKGROUNDコマンドは、4D Drawエリアのバックグラウンド上の図形を解除し、本来の位置に置きます。このコマンドは、「バックグラウンド解除」メニューアイテムと同じ機能を持ちます。

参照

なし

DR REMOVE FROM BACKGROUND

DR REMOVE FROM BACKGROUND (エリア;図形ID)

引数	タイプ	説明
エリア	倍長整数型の変数	4D Drawエリア
図形ID	整数	図形のID番号

説明

DR REMOVE FROM BACKGROUNDコマンドは、指定された図形を<エリア>のバックグラウンドから解除して元の場所に配置します。

参照

DR ADD TO BACKGROUND

DR SCROLL DOCUMENT

DR SCROLL DOCUMENT (エリア;横;縦;モード)

引数	タイプ	説明
エリア	倍長整数型の変数	4D Drawエリア
横	実数	横の位置
縦	実数	縦の位置
モード	整数	1=相対、0=絶対

説明

DR SCROLL DOCUMENT コマンドは、<横>と<縦>の各引数に従って<エリア>のドキュメントをスクロールします。<横>と<縦>はともに“ベースの単位”で指定します。“目盛の単位”から“ベースの単位”に変換する場合は、**DR Scale to base**関数を使用します。

<モード>が1の場合、<エリア>は現在位置から<横>と<縦>で指定した“ベースの単位”の位置までスクロールします。<横>と<縦>が正の数値の場合は、<エリア>は右と下の方向にそれぞれスクロールします。負の数値の場合は、<エリア>は左上の方向にそれぞれスクロールします。

また、<モード>が0の場合は、<エリア>は原点から<横>と<縦>で指定した“ベースの単位”の絶対位置までスクロールします。

例題

次の例は、2つのボタン用のスクリプトです。1つのボタンは、ドキュメントをベースの1単位だけ右に、もう1つのボタンはドキュメントをベースの1単位だけ左にスクロールします。

```
`b右ボタン用のオブジェクトメソッド  
DR SCROLL DOCUMENT (Area;1;0;1)  
`b左ボタン用のオブジェクトメソッド  
DR SCROLL DOCUMENT (Area;-1;0;1)
```

参照

DR SET ORIGIN

DR SET ENTERABLE

DR SET ENTERABLE (エリア;モード)

引数	タイプ	説明
エリア	倍長整数型の変数	4D Drawエリア
モード	整数	0=入力不可、1=入力可

説明

DR SET ENTERABLEコマンドは、「エリア」へのアクセスを制御します。

<モード> が1の場合、<エリア> は入力可になり、通常通りに機能します。<モード> が0の場合、<エリア> は入力不可になり、メソッドによる修正はできても、ユーザによる修正はできません。また、エリアをスクロールする、あるいは選択図形をクリップボードにコピーすることはできても、選択図形を変更する、または4D Drawメニューやパレットを使用することはできません。

例題

次の例は、フォームメソッドです。エリアのすべての表示オプションをオフにした後、エリアを入力不可にします。

```

If (Form event=On load)
    DR DISPLAY OPTIONS (Area;-1;0)
    DR SET ENTERABLE (Area;0)
End if

```

`On Loadイベントの場合
`すべての表示オプションをオフにする
`エリアを入力不可にする

参照

DR DISPLAY OPTIONS

DR SET UPDATE MODE

DR SET UPDATE MODE (エリア;モード)

引数	タイプ	説明
エリア	倍長整数型の変数	4D Drawエリア
モード	整数	0=オフ、1=オン

説明

DR SET UPDATE MODEコマンドは、<エリア>の画面更新をオンまたはオフにします。

<モード>が0の場合は、画面更新がオフになります。<モード>が1の場合は、画面更新がオンになります。このコマンドは、4D Drawコマンドによる操作とユーザによる操作の双方に反映します。

画面更新をオフにすると、4D Drawの各コマンドが速やかに実行されます。例えば、4D Drawエリアに多くの修正を行う場合は、開始前に画面更新をオフにします。修正終了時に画面更新をオンに戻せば、各コマンドが速やかに実行されるだけでなく、画面の再描画もスムーズになります。

画面更新は、ユーザがエリアにアクセスする前に必ずオンにしてください。オンにしないと、エリア内をクリックまたはドラッグしても再描画されません。

例題

次の例は、画面更新をオフにして、複数の修正を実行するプロジェクトメソッドを呼び出した後、画面更新をオンに戻します。

```
DR SET UPDATE MODE (Area;0)      `画面更新をオフにする  
REFORMAT (Area)                  `エリアは倍長整数  
DR SET UPDATE MODE (Area;1)      `画面更新をオンにする
```

参照

DR Get update mode、**DR REDRAW**

DR ZOOM

DR ZOOM (エリア;ズーム;水平;垂直;移動)

引数	タイプ	説明
エリア	倍長整数型の変数	4D Drawエリア
ズーム	数値	ズーム倍率
水平	数値	水平座標
垂直	数値	垂直座標
移動	整数	0=中揃え、1=上揃え、下揃え

説明

DR ZOOMコマンドは、4D Drawエリアを <ズーム> 倍率に拡大または縮小します。

<移動> が0の場合は、点（水平,垂直）が4D Drawエリアの表示領域の中央になるように表示されます。

<移動> が1の場合は、点（水平,垂直）が4D Drawエリアの表示領域の左上隅になるように表示されます。

参照

なし

エリアオプション (Area Options)

DR COORDINATES

DR COORDINATES (エリア; コーディネイト; モード)

引数	タイプ	説明
エリア	倍長整数型の変数	4D Drawエリア
コーディネイト	倍長整数	表示する属性
モード	整数	0=表示しない、1=表示する、2=切り替える

説明

DR COORDINATESコマンドは、<エリア>に座標パネルを表示した場合に表示するコーディネイト情報を制御します。「環境設定」ダイアログボックスの「コーディネイト表示」の各チェックボックスと同じように、プロシージャから、反映すべきコーディネイト情報を<コーディネイト>に指定します。

また、<コーディネイト>は、コーディネイトコードの和として表します。1つのコードが、1つのコーディネイト情報（高さや幅等）を表します。<コーディネイト>に設定する値は次の通りです。

コード	コーディネイト情報
1	幅
2	高さ
4	水平スケーリング
8	垂直スケーリング
16	横の変動値
32	縦の変動値
64	線の長さ
128	回転角度

<モード>は、コーディネイトの表示方法を制御します。各表示オプションは、<モード>が0の場合は表示されず、1の場合は表示され、2の場合は切り替わります。

参照
なし

DR DISPLAY OPTIONS

DR DISPLAY OPTIONS (エリア;オプション;モード)

引数	タイプ	説明
エリア	倍長整数型の変数	4D Drawエリア
オプション	整数	表示オプションのコード
モード	整数	0=表示しない、1=表示する、2=切り替える

説明

DR DISPLAY OPTIONSコマンドは、<エリア>の表示オプションを制御します。表示オプションとは、グリッドやページブレイク等の描画補助機能のことです。

<オプション>は、反映するオプションを指定します。<オプション>は、表示オプションのコードの和として表します。1つのコードが、1つの表示オプションを表します。

表示オプションのコードは次の通りです。

コード	表示オプション
-1	すべてのオプション
1	定規
2	グリッド
4	ページブレイク
8	定規
16	メニューバー
32	ツールパレット
64	スクロールバー
128	エリア境界
256	プリント領域
512	用紙の領域
1024	ペーストボード
2048	ズームボックス

<モード>は、表示オプションの変更方法を制御します。各表示オプションは、「モード」が0の場合は表示されず、1の場合は表示され、2の場合は切り替わります。

例題

次の例は、フォームメソッドのOn Loadイベントでエリアのすべての表示オプションをオフにします。

```
If (Form event=On Load)  
  DR DISPLAY OPTIONS (Area;-1;0)  
End if
```

参照

なし

DR Get display

DR Get display (エリア;オプション) 整数

引数	タイプ	説明
エリア	倍長整数型の変数	4D Drawエリア
オプション番号	整数	表示オプションの番号

説明

DR Get display関数は<エリア>内で<オプション番号>で指定された表示オプションが表示されているかどうかを返します。

表示されている場合、**DR Get display**関数は1、表示されていない場合は0を返します。

参照

なし

DR GET DOCUMENT SIZE

DR GET DOCUMENT SIZE (エリア;幅;高さ)

引数	タイプ	説明
エリア	倍長整数型の変数	4D Drawエリア
幅	実数	ドキュメントの幅
高さ	実数	ドキュメントの高さ

説明

DR GET DOCUMENT SIZE コマンドは、“ベースの単位” で表された <エリア> のドキュメントのサイズを <幅> と <高さ> に返します。

“ベースの単位” から “目盛の単位” に変換する場合は、**DR Base to scale** 関数を使用します。

例題

DR SET DOCUMENT SIZE コマンドの例を参照してください。

参照

DR SET DOCUMENT SIZE

DR Get draw mode

DR Get draw mode (エリア) 整数

引数	タイプ	説明
エリア	倍長整数型の変数	4D Drawエリア
戻り値	整数	エリアに対する現在の描画モード

説明

DR Get draw mode関数は、<エリア>の描画モードを値として返します。

0の場合は、図形をコーナーから描画するように描画モードが設定されています。1の場合は、図形が中心から描画するように描画モードが設定されています。

参照

DR SET DRAW MODE

DR GET GLOBAL PREFERENCES

DR GET GLOBAL PREFERENCES (書き出し;取り込み)

引数	タイプ	説明
書き出し	整数型の変数	書き出す場所 0=クライアント、1=サーバー
取り込み	整数型の変数	取り込む場所 0=クライアント、1=サーバー

説明

4D Serverでは、テンプレートはクライアントかサーバーのどちらかで保存されます。**DR GET GLOBAL PREFERENCES**コマンドは、テンプレートを取り込む場所と書き出す場所の環境設定を返します。

<取り込み>は、テンプレートを取り込む場所です。4D Drawは、レイアウト上の4D Drawエリアにテンプレートを取り込む際にこの環境設定を使用します。

メニューアイテムが**DR EXPERT COMMAND**コマンドで使用不可の場合でも、**DR DOCOMMAND**コマンドを使用すれば、メニューアイテムを実行することができます。

<書き出し>は、テンプレートを保存する場所です。4D Drawは、4D Drawの「ファイル」メニューから「テンプレートとして保存」を選択する、または**DR DO COMMAND**コマンドを使用して、このメニューアイテム(コマンドコード=1006)を呼び出した場合に、この環境設定を使用します。

これらの環境設定は、**DR SET GLOBAL PREFERENCES**コマンドを使用して行うことができます。また、Customizer Plusを使用しても設定できます。

参照

DR SET GLOBAL PREFERENCES

DR GET PREFERENCES

DR GET PREFERENCES (エリア;プリント順序;PICT;ロック時の警告;自動スクロール;可変プリント領域;確認{;保存方法})

引数	タイプ	説明
エリア	倍長整数型の変数	4D Drawエリア
プリント順序	整数	型の変数プリント順序
PICT	整数型の変数	0=ピクチャ、1=図形
ロック時の警告	整数型の変数	0=オフ、1=オン
自動スクロール	整数型の変数	0=オフ、1=オン
可変プリント領域	整数型の変数	0=固定、1=可変
確認	整数	0=確認なし、1=確認あり、-1=変更しない
保存方法	整数	0=ピクチャとデータ、1=ピクチャのみ 2=データのみ、-1=変更しない

説明

DR GET PREFERENCES コマンドは、<エリア>の環境設定情報を、<プリント順序>、<PICT>、<ロック時の警告>、<自動スクロール>、<可変プリント領域>の各変数に返します。これらの引数は、「環境設定」ダイアログボックスの環境設定情報に対応します。

<プリント順序>は、<エリア>のプリント時のページのプリント順序を表す属性です。<プリント順序>が0の場合は、上から下へ、さらに左から右へページをプリントします。1の場合は、左から右へ、さらに上から下へページをプリントします。デフォルトは、左から右にページをプリントします。

<PICT>は、<エリア>における、オープン時、取り込み時、ペースト時のPICT図形とPICTドキュメントの解釈方法を表す属性です。<PICT>が0の場合は、PICTドキュメントまたはペーストしたPICT図形を1つの図形と見なします。1の場合は、PICT図形を構成要素(4D Drawの図形)に分解します。デフォルトでは、1つの図形としてPICTドキュメントをペーストします。

<ロック時の警告>は、ロックされた図形の属性を変更しようとした場合に警告ボックスを表示するかどうかを表す属性です。<ロック時の警告>が0の場合は、警告ボックスが表示されません。1の場合は、アラートボックスが表示されます。デフォルトでは、警告ボックスを表示します。

<自動スクロール>は、<エリア>の境界を越えてドラッグした場合に<エリア>のドキュメントを自動的にスクロールするかどうかを表す属性です。<自動スクロール>が0の場合は、自動スクロール機能がオフになります。1の場合はオンになります。デフォルトでは、自動スクロールになっています。

<可変プリント領域>は、プリント時に<可変プリント領域>オプションを使用するかどうかを表す属性です。<可変プリント領域>が0の場合は、<エリア>を定義したサイズでエリアをプリントします。1の場合は、ドキュメント内の全図形がプリントできるようにエリアを縦方向に拡張します。デフォルトでは、可変プリント領域でプリントします。

<確認>は、自動保存でない4D Drawエリアを含んだレコードを受け入れる場合に、確認のダイアログボックスを表示するかどうかを指定します。デフォルトでは、4D Drawエリア内のドキュメントを保存するかどうかを確認するダイアログボックスを表示します。<確認>が0の場合は、ダイアログボックスは表示されません。1の場合は表示されます。

オプション引数の<保存方法>は、4D Drawエリア内のドキュメントの保存方法を指定します。<保存方法>が0の場合は、イメージを再構築するために使用されるピクチャとデータの両方を保存します。そして、これがデフォルトになります。<保存方法>が1の場合は、ピクチャ (PICT) のみ保存します。そのため、保存した図形を個々に操作することができなくなります。2の場合は、4D Drawエリアの図形に関するデータのみを保存します。イメージは、後で保存されたデータ内の情報を使用することにより再構築されます。この保存オプションは、保存を最も高速に行うことができ、使用するメモリ量も最小で済みます。メモリ量が足りない場合は、別の方法を選択するためのダイアログボックスが表示されます。

参照

DR SET PREFERENCES

DR SET DISPLAY

DR SET DISPLAY (エリア;オプション;表示)

引数	タイプ	説明
エリア	倍長整数型の変数	4D Drawエリア
オプション	整数	表示オプションの番号
表示 える	整数	0=表示しない、1=表示する、2=切り替 える

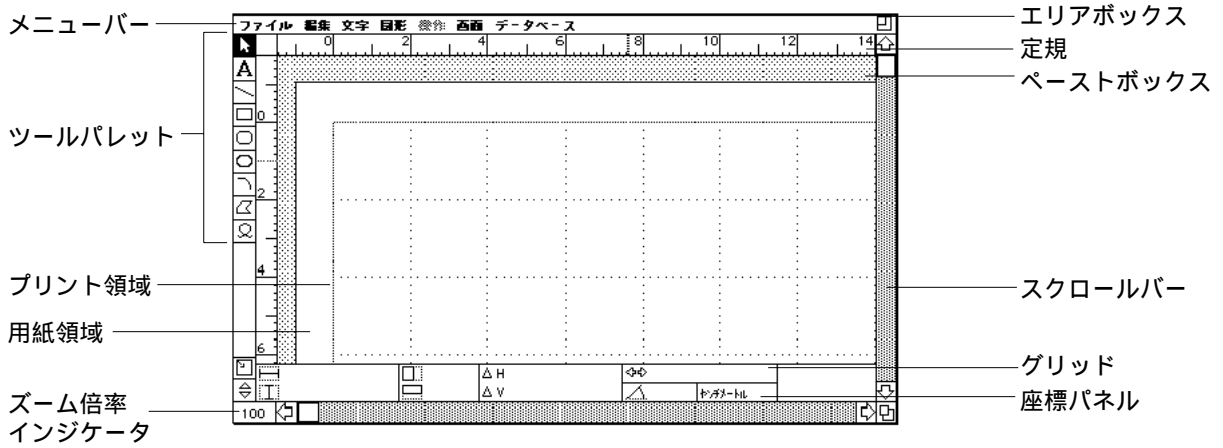
説明

DR SET DISPLAYコマンドは、4D Drawエリアの定規やツールパレット等が表示されるかどうかを管理します。表示オプションのいくつかは、バージョン1.0でも**DR DISPLAY OPTIONS**コマンドを使って設定することができます。

DR SET DISPLAYコマンドと**DR Get display**関数の表示オプションには、次のようなものがあります。

番号	表示オプション
1	定規
2	グリッド
3	ページブレイク
4	座標
5	メニューバー
6	ツールパレット
7	スクロールバー
8	エリア境界
9	プリント領域
10	用紙領域
11	ペーストボード
12	ズーム倍率インジケータ

次の図は、前ページの表で記述されている表示オプションを示したものです。



参照
なし

DR SET DOCUMENT SIZE

DR SET DOCUMENT SIZE (エリア;幅;高さ)

引数	タイプ	説明
エリア	倍長整数型の変数	4D Drawエリア
幅	実数	ドキュメントの幅
高さ	実数	ドキュメントの高さ

説明

DR SET DOCUMENT SIZEコマンドは、<エリア>のドキュメントサイズを設定します。

DR SET DOCUMENT SIZEコマンドを使用すると、<幅>をドキュメントの幅に、<高さ>をドキュメントの高さに、それぞれ設定します。<幅>と<高さ>は、“ベースの単位”で設定します。“目盛の単位”から“ベースの単位”に変換する場合は、**DR Scale to base**関数を使用します。

例題

次の例は、エリアを含むレイアウト上の2つの入力エリア「v幅」と「v高さ」のスク립トから呼び出されるオブジェクトメソッドです。このメソッドは、On Loadイベントでドキュメントサイズを2つの変数に格納し、On Date Changeイベントで2つの変数に入力された値でドキュメントサイズを設定します。

Case of

¥(Form event=On Load)

DR SET DOCUMENT SIZE (Area;vWide;vHigh)

¥(Form event=On Date Change)

DR GET DOCUMENT SIZE (Area;vWide;vHigh)

End case

参照

DR GET DOCUMENT SIZE

DR SET DRAW MODE

DR SET DRAW MODE (エリア;モード)

引数	タイプ	説明
エリア	倍長整数型の変数	4D Drawエリア
モード	整数	0=コーナーから、1=中心から

説明

DR SET DRAW MODEコマンドは、<エリア>の図形を中心から描画するか、コーナーから描画するかを指定します。

<モード>が0の場合は、コーナーから図形を描画します。1の場合は、中心から図形を描画します。デフォルトは、コーナーから描画です。

例題

次の例は、“bモード”という名前のチェックボックスのスクリプトです。On Loadイベントでエリアの描画モードを求め、その値をチェックボックスに設定します。このチェックボックスをクリックすることで描画モードを設定します。チェックボックスがオンの場合は、図形を中心から描画します。チェックマークがオフの場合は、コーナーから描画します。

Case of

```

¥(Form event=On Load)
    bFormat:=DR Get draw mode (Area) ` “b形式” に描画形式を代入する
¥(Form event=On Date Change)
    DR SET DRAW MODE (Area;bFormat) ` 描画形式を設定する
    
```

End case

参照

DR Get draw mode

DR SET GLOBAL PREFERENCES

DR SET GLOBAL PREFERENCES (書き出し;取り込み)

引数	タイプ	説明
書き出し	整数型の変数	書き出す場所 0=クライアント、1=サーバ
取り込み	整数型の変数	取り込む場所 0=クライアント、1=サーバ

説明

4D Serverでは、テンプレートはクライアントかサーバのどちらかで保存されます。**DR SET GLOBAL PREFERENCES**コマンドは、テンプレートを取り込む場所と書き出す場所の環境設定を返します。

<取り込み>は、テンプレートを取り込む場所です。4D Drawは、レイアウト上の4D Drawエリアにテンプレートを取り込む際にこの環境設定を使用します。4D Drawが指定された場所のテンプレートを見つけることができない場合は、その他の場所を調べます。

<書き出し>は、テンプレートを保存する場所です。4D Drawは、4D Drawの「ファイル」メニューから「テンプレートとして保存」を選択する、または**DR DO COMMAND**コマンドを使用して、このメニューアイテム（コマンドコード=1006）を呼び出した場合に、この環境設定を使用します。

参照

DR GET GLOBAL PREFERENCES

DR SET PREFERENCES

DR SET PREFERENCES (エリア;プリント順序;PICT;ロック時の警告;自動スクロール;可変プリント領域;確認{;保存方法})

引数	タイプ	説明
エリア	倍長整数型の変数	4D Drawエリア
プリント順序	整数	プリント順序
PICT	整数	0=ピクチャ、1=図形、-1=変更しない
ロック時の警告	整数	0=オフ、1=オン、-1=変更しない
自動スクロール	整数	0=オフ、1=オン、-1=変更しない
可変プリント領域	整数	0=固定、1=可変
確認	整数	0=確認なし、1=確認あり、-1=変更しない
保存方法	整数	0=ピクチャとデータ、1=ピクチャのみ 2=データのみ、-1=変更しない

説明

DR SET PREFERENCESコマンドは、<エリア>の環境設定情報を各引数によって設定します。これらの引数は、「環境設定」ダイアログボックスの環境設定情報に対応します。

<プリント順序>は、<エリア>のプリント時のページのプリント順序を指定します。<プリント順序>が0の場合は、上から下へ、さらに左から右へページをプリントします。1の場合は、左から右へ、さらに上から下へページをプリントします。デフォルトは、左から右にページをプリントします。

メニューアイテムが**DR EXPERT COMMAND**コマンドで使用不可の場合でも、**DR DOCOMMAND**コマンドを使用すれば、メニューアイテムを実行することができます。

<PICT>は、<エリア>における、オープン時、取り込み時、ペースト時のPICT図形とPICTドキュメントの解釈方法を指定します。<PICT>が0の場合は、PICTドキュメントまたはペーストしたPICT図形を1つの図形と見なします。1の場合は、PICT図形を構成要素(4D Drawの図形)に分解します。デフォルトでは、1つの図形としてPICTドキュメントをペーストします。

<ロック時の警告>は、ロックされた図形の属性を変更しようとした場合に警告ボックスを表示するかどうかを指定します。<ロック時の警告>が0の場合は、警告ボックスが表示されません。1の場合は、警告ボックスが表示されます。デフォルトでは、警告ボックスを表示します。

<自動スクロール>は、<エリア>の境界を越えてドラッグした場合に<エリア>のドキュメントを自動的にスクロールするかどうかを指定します。<自動スクロール>が0の場合は、自動スクロール機能がオフになります。1の場合はオンになります。デフォルトでは、自動スクロールになっています。

<可変プリント領域>は、プリント時に<可変プリント領域>オプションを使用するかどうかを指定します。<可変プリント領域>が0の場合は、<エリア>を定義したサイズでエリアをプリントします。1の場合は、ドキュメント内の全図形がプリントできるようにエリアを縦方向に拡張します。デフォルトでは、可変プリント領域でプリントします。

<確認>は、自動保存でない4D Drawエリアを含んだレコードを受け入れる場合に、確認のダイアログボックスを表示するかどうかを指定します。デフォルトでは、4D Drawエリア内のドキュメントを保存したいかどうかを確認するダイアログボックスを表示します。<確認>が0の場合は、ダイアログボックスは表示されません。1の場合は表示されます。

オプション引数の<保存方法>は、4D Drawエリア内のドキュメントの保存方法を指定します。<保存方法>が0の場合は、イメージを再構築するために使用されるピクチャとデータの両方を保存します。そして、これがデフォルトになります。「保存方法」が1の場合は、ピクチャ (PICT) のみ保存します。そのため、保存した図形を個々に操作することができなくなります。2の場合は、4D Drawエリアの図形に関するデータのみを保存します。イメージは、後で保存されたデータ内の情報を使用することにより再構築されます。この保存オプションは、保存を最も高速に行うことができ、使用するメモリ量も最小で済みます。メモリ量が足りない場合は、別の方法を選択するためのダイアログボックスが表示されます。

例題

次の例は、レイアウトメソッドのBeforeフェーズでエリアの表示オプション、ロック警告、自動スクロールをオフにします。プリントに関するオプションは変更しません。

```
If (Form event=On Load)
  DR DISPLAY OPTIONS (Area;-1;0)
  DR SET PREFERENCES (Area;-1;-1;0;0;-1;-1;-1)
End if
```

参照

DR GET PREFERENCES

エリア (Areas)

DR AREA TO AREA

DR AREA TO AREA (送り側;受け側;スコープ)

引数	タイプ	説明
送り側	倍長整数	型の変数送り側の4D Drawエリア
受け側	倍長整数	型の変数受け側の4D Drawエリア
スコープ	倍長整数	1=ドキュメント設定、2=全図形、3=両方

説明

DR AREA TO AREAコマンドは、<スコープ>の指定に従って<送り側>の4D Drawエリアの内容を<受け側>の4D Drawエリアにコピーします。

<スコープ>=1の場合は、描画サイズ、定規設定、表示オプション等のドキュメント設定情報を転送します。

<スコープ>=2の場合は、全図形を転送します。

<スコープ>=3の場合は、全図形とドキュメント設定情報を転送します。

ドキュメント設定情報を転送すると、<受け側>のドキュメント設定が置き換えられます。図形を転送すると、<受け側>のエリアに図形を追加します。

DR AREA TO AREAコマンドは、オフスクリーンエリアを操作するのに便利です。

例題

次の例は、4D Drawエリアの内容を新しいオフスクリーンエリアにコピーします。

```
vOffscreen:=DR New offscreen area  
DR AREA TO AREA (Area;vOffscreen;3)
```

参照

DR New offscreen area、**DR DELETE OFFSCREEN AREA**

DR AREA TO FIELD

DR AREA TO FIELD (エリア;スコープ;テーブル;フィールド;{保存方法})

引数	タイプ	説明
エリア	倍長整数型の変数	4D Drawエリア
スコープ	倍長整数	-2=ドキュメント、-1=全図形、 0=選択図形、>0=図形ID
テーブル	整数	テーブル番号
フィールド	整数	フィールド番号
保存方法	整数	0=ピクチャとデータ、1=ピクチャのみ 2=データのみ、-1=変更なし

説明

DR AREA TO FIELDコマンドは、<テーブル>と<フィールド>で指定したピクチャフィールドに<エリア>の内容を格納します。<フィールド>がピクチャタイプでない場合は、**DR Error**関数にエラーコード31を返します。

<スコープ>は、コピー対象を制御します。

DR AREA TO FIELDコマンドは、リレートしたテーブルのフィールドに複数の図形を格納する、あるいは特定の図形だけを格納する場合に便利です。**DR AREA TO FIELD**コマンドは単にそれらの図形をフィールドに格納するだけです。テーブルのレコードは、保存しなければなりません。

オプション引数である<保存方法>は、4D Drawエリア内のドキュメントの保存方法を指定します。

<保存方法>=0の場合は、イメージを再構築する際に使用したピクチャと内部データの両方を保存します。この値がデフォルトになります。

<保存方法>=1の場合は、ピクチャ (PICT) のみ保存します。そのため、保存した図形を個々に操作することができなくなります。

<保存方法>=2の場合は、4D Drawエリア内の図形に関するデータのみを保存します。イメージは後で保存されたデータ内の情報を使用することにより再構築されます。この保存オプションは、最も高速に保存することができ、使用するメモリ量も最小で済みます。メモリ量が足りない場合は、別の方法を選択するためのダイアログボックスが表示されます。

例題

次の例は、エリアの図形ごとにリレートレコードを作成します。

For (\$i;1; <i>DR Count</i> (Area;-1))	`図形の数だけ繰り返す
CREATE RECORD ([Objects])	`図形を格納するレコードを作成する
[Objects]Key:=[Draw]Name	`リレート値を代入する
\$Temp:= <i>DR Get ID</i> (Area;-1;\$i)	`図形IDを求める
<i>DR AREA TO FIELD</i> (Area;\$Temp;3;2;-1)	`フィールドに図形をコピーする
SAVE RECORD ([Objects])	`レコードを保存する
End for	

参照

DR FIELD TO AREA

DR Area to picture

DR Area to picture (エリア;スコープ) ピクチャ

引数	タイプ	説明
エリア	倍長整数型の変数	4D Drawエリア
スコープ	倍長整数	-2=ドキュメント、-1=全図形、 0=選択図形、>0=図形ID
戻り値	ピクチャ	4th Dimensionのピクチャ

説明

DR Area to picture関数は、<エリア>の図形を4th Dimensionピクチャとして返します。

このピクチャに含まれる図形は<スコープ>で制御できます。

<スコープ>=-2の場合は、ドキュメント全体をコピーします。描画サイズ、定規設定、表示オプション等のドキュメント設定情報も含まれます。

<スコープ>=-1の場合は、<エリア>の全図形をコピーします。ドキュメント設定情報は含みません。

<スコープ>=0の場合は、選択図形だけをコピーします。

<スコープ>>0の場合は、図形IDで指定した図形だけをコピーします。対応する図形が存在しない場合は、**DR Error**関数にエラーコード2を返します。

ドキュメント設定情報を転送すると、<受け側>のドキュメント設定が置き換えられます。図形を転送すると、<受け側>のエリアに図形を追加します。

例題

次の例は、エリアの選択図形からピクチャ配列を作成します。1つの図形が、1つの配列要素になります。

```
$x:=DR Count (Area;0)           `選択図形の数を求める
ARRAY PICTURE (aPicture;$x)    `ピクチャ配列を宣言する
For ($i;1;$x)                  `選択図形の数だけ繰り返す
    $ID:=DR Get ID (Area;0;$i)   `図形IDを求める
    aPicture{$i}:=DR Area to picture (Area;$ID)`配列要素に図形を格納する
End for
```

参照

DR Get ID、**DR PICTURE TO CLIPBOARD**、**DR PLACE PICTURE**

DR DELETE OFFSCREEN AREA

DR DELETE OFFSCREEN AREA (エリア)

引数	タイプ	説明
エリア	倍長整数型の変数	4D Drawエリア

説明

DR DELETE OFFSCREEN AREAコマンドは、**DR New offscreen area**関数で作成した4D Drawオフスクリーンエリアを削除して、使用メモリを開放します。

<エリア>は、レイアウトまたはプラグインウィンドウのエリアではなく、オフスクリーンエリアでなければなりません。オフスクリーンエリアを使い終わったら、必ず**DR DELETE OFFSCREEN AREA**コマンドを呼び出してください。

例題

次の例は、**DR New offscreen area**関数を呼び出したら、必ず**DR DELETE OFFSCREEN AREA**コマンドを呼び出す必要があることを示しています。

```
$Area:=DR New offscreen area           `オフスクリーンエリアを作成する  
`いくつかの処理をここで実行する  
DR DELETE OFFSCREEN AREA ($Area)     `オフスクリーンエリアを削除する
```

参照

DR AREA TO AREA、**DR New offscreen area**

DR FIELD TO AREA

DR FIELD TO AREA (エリア;テーブル;フィールド)

引数	タイプ	説明
エリア	倍長整数型の変数	4D Drawエリア
テーブル	整数	テーブル番号
フィールド	整数	フィールド番号

説明

DR FIELD TO AREAコマンドは、<テーブル>と<フィールド>で指定したピクチャフィールドに含まれる図形を<エリア>に格納します。

<フィールド>は、4D Drawの図形またはピクチャが格納されたピクチャフィールドでなければなりません。<フィールド>が空の場合、このコマンドは無視され、**DR Error**関数にエラーコード39を返します。

例題

次の例は、2番目のファイルの5番目のフィールドに含まれている4D Drawドキュメントを開きます。

If (Before)

```
DR FIELD TO AREA (Area;2;5)
```

```
End if
```

参照

DR AREA TO FIELD

DR NEW DRAWING

DR NEW DRAWING (エリア)

引数	タイプ	説明
エリア	倍長整数型の変数	4D Drawエリア

説明

DR NEW DRAWINGコマンドは、<エリア>の内容をクリアします。「確認」ダイアログボックスが表示されないことを除けば、「ファイル」メニューから「新規」を選択するのと同じことです。**DR NEW DRAWING**コマンドは図形とドキュメント設定情報（ドキュメントサイズ、定規の設定値等）をすべてクリアします。

例題

次の例は、エリアの内容をクリアします。

DR NEW DRAWING (Area)

警告：このコマンドを使用すると、エリア内のカレントドキュメントは保存されません。カレントドキュメントを保存したい場合は、**DR NEW DRAWING**コマンドを呼び出す前に**DRSAVE DOCUMENT**コマンドを呼び出す必要があります。

参照

DR SAVE DOCUMENT

DR New offscreen area

DR New offscreen area 倍長整数

引数	タイプ	説明
----	-----	----

この関数には、引数はありません。

戻り値	倍長整数	エリア
-----	------	-----

説明

DR New offscreen area関数は、4D Drawのオフスクリーンエリアを作成し、そのエリアにアクセスするための値を返します。**DR New offscreen area**関数の返す値は、4D Drawエリアを必要とする任意の4D Drawコマンドで使用することができます。

例題

次の例は、レコードを検索後にオフスクリーンエリアを作成します。さらに、レコードからエリアにドキュメントをコピーし、エリアのプリントを実行します。

```
SEARCH (Table3;[Table3]Field1="レベル1") `レコードを検索する
$Offscreen:=DR New offscreen area `新しいオフスクリーンエリアを作成する
DR FIELD TO AREA ($Offscreen;3;2) `フィールド内に格納されたドキュメントを
コピーする
DR PRINT ($Offscreen;0) `エリアをプリントする
DR DELETE OFFSCREEN AREA ($Offscreen) オフスクリーンエリアを削除する
```

参照

DR DELETE OFFSCREEN AREA、**DR AREA TO AREA**

DR OPEN DOCUMENT

DR OPEN DOCUMENT (エリア;ドキュメント{;モード})

引数	タイプ	説明
エリア	倍長整数型の変数	4D Drawエリア
ドキュメント	文字列	ドキュメント名
モード	整数	0=置き換える、1=追加する

説明

DR OPEN DOCUMENTコマンドは、<ドキュメント>を開いて、その内容を<エリア>に格納します。

<ドキュメント>のタイプは、4DRW、PICT、PNTG、EPSF (Windowsではそれぞれ*.4dw、*.pct、*.pnt、*.eps) のいずれかでなければなりません。<ドキュメント>が空の文字列の場合は、標準の「オープンファイル」ダイアログボックスが表示され、ドキュメントを選択することができます。<ドキュメント>が存在しない場合は、<エリア>の内容を変更せずに、**DR Error**関数にMacintoshのシステムエラーコードを返します。

4D Drawは、データベースフォルダ(データベースストラクチャの存在するフォルダ)内の<ドキュメント>を開きますが、データベースフォルダの外にあるドキュメントを開く場合は、完全なパス名を指定してください。Macintoshのパス名については、『4th Dimension ランゲージリファレンス』を参照してください。また、<ドキュメント>がすでに開かれている場合は、**DR Error**関数にMacintoshのシステムエラーを返します。

Windowsの場合

ディスク名に続いて、コロン(:)を入力し、フォルダ毎に円記号(¥)を入力してください。ファイル名は、“D:¥Folder1¥Folder2¥File.PCT”のように、拡張子を指定してファイルのタイプを決定してください。

Windows 3.xでは、ファイル名は8文字を超えてはいけません(拡張子を除く)。

Macintoshの場合

ディスク名に続いて、フォルダ毎にコロン(:)を入力してください。例えば、“Disk:Folder:Document”のように指定します。

オプション引数の<モード>は、ドキュメントの開き方を制御します。<モード>は<ドキュメント>が空の文字列でない場合にだけ使用できます。ドキュメントタイプがPICT、PNTG、EPSFの場合だけ有効です。<モード>が0または指定しない場合は、<エリア>の内容を開いた<ドキュメント>の内容に置き換えます。1の場合は、現在の<エリア>の内容に開いた<ドキュメント>の内容を追加します。

例題

次の例は、“ Client Type ” フィールドの値を基にドキュメントを開きます。

Case of

¥([Client]Client Type="Distrib") DR OPEN DOCUMENT (Area;"Distrib")	`タイプが特約店の場合 `特約店を開く
¥([Client]Client Type="Construc") DR OPEN DOCUMENT (Area;"Construc")	`タイプが代理店の場合 `代理店を開く
¥([Client]Client Type="Client") DR OPEN DOCUMENT (Area;"Client")	`タイプが一般顧客の場合 `一般顧客を開く

End case

参照

DR SAVE DOCUMENT

DR PICTURE TO AREA

DR PICTURE TO AREA (エリア;ピクチャ)

引数	タイプ	説明
エリア	倍長整数型の変数	4D Drawエリア
ピクチャ	ピクチャ型の変数	ピクチャ変数

説明

DR PICTURE TO AREAコマンドは、<ピクチャ>に含まれる図形を<エリア>に格納します。

<ピクチャ>は、4D Drawの図形またはピクチャが格納されたピクチャ変数でなければなりません。<ピクチャ>が空の場合、このコマンドは無視され、**DR Error**関数にエラーコード39を返します。

例題

1.次の例は、配列要素から作成されたテンプレートをもとにカレント4D Drawドキュメントを設定します。

```
QUERY ([Template];[Template]Type=2)
SELECTION TO ARRAY ([Template]DrawDoc;aTemplate)
```

2.次の例は、配列「aTemplate」のスクリプトです。

```
Confirm ("カレント4D Drawエリアを指定してください。")
If (OK=1)
  DR PICTURE TO AREA (Area;aTemplate{aTemplate})
End if
```

参照

DR Area to picture

DR SAVE DOCUMENT

DR SAVE DOCUMENT (エリア;ドキュメント;タイプ{;スコープ})

引数	タイプ	説明
エリア	倍長整数型の変数	4D Drawエリア
ドキュメント	文字列	ドキュメント名
タイプ	文字列	ドキュメントタイプ
スコープ	整数	0=全図形、1=選択図形

説明

DR SAVE DOCUMENTコマンドは、<エリア>の内容を<ドキュメント>に格納します。

Windowsの場合

ディスク名に続いて、コロン(:)を入力し、フォルダ毎に円記号(¥)を入力してください。ファイル名は、“D:¥Folder1¥Folder2¥File.PCT”のように、拡張子を指定してファイルのタイプを決定してください。

Windows 3.xでは、ファイル名は8文字を超えてはいけません(拡張子を除く)。

Macintoshの場合

ディスク名に続いて、フォルダ毎にコロン(:)を入力してください。例えば、“Disk:Folder:Document”のように指定します。

<ドキュメント>が空の文字列の場合は、標準の「ファイル作成」ダイアログボックスを表示し、ここでドキュメント名、ドキュメントタイプ、スコープを指定することができます。<ドキュメント>が空の文字列でない場合、**DR SAVE DOCUMENT**コマンドは<タイプ>で指定したドキュメントタイプでドキュメントを保存します。

<タイプ>に設定できるドキュメントタイプは、4DWまたは4DRW(4D Draw)、PCTまたはPICT、PNTまたはPNTG(MacPaint)です。<タイプ>が空の文字列の場合は、標準の4D Drawドキュメントを作成します。また、すでに同じ名前の<ドキュメント>が存在する場合は、ドキュメントを上書きします。

オプションの<スコープ>は、<ドキュメント>に保存する対象を制御します。<ドキュメント>が空の文字列でなく<タイプ>にPICTとPNTGのを指定した場合にだけ有効です。

デフォルトでは、データベースフォルダ内にドキュメントを保存します。データベースフォルダの外にドキュメントを保存する場合は、フルパス名を指定してください。Macintoshのパス名については、『4th Dimensionランゲージリファレンス』を参照してください。

ドキュメントがすでに開かれている場合は、DR Error関数にエラーコード-47を返します。

例題

次の例は、PICTドキュメントに描画内容を保存します。

`Windows上

DR SAVE DOCUMENT (Area;"Logo.pct";"pct";1)

`Macintosh上

DR SAVE DOCUMENT (Area;"Logo;pct";1)

参照

DR OPEN DOCUMENT

バインド (Binding)

DR ACTIVATE BIND

DR ACTIVATE BIND (エリア;バインドID;処理)

引数	タイプ	説明
エリア	倍長整数型の変数	4D Drawエリア
バインドID	倍長整数	バインドID
方向	整数	1=エリア フィールド、 2=フィールド エリア

説明

DR ACTIVATE BINDコマンドは、<バインドID>で指定したバインドを<エリア>に結合します。

<バインドID>は、**DR New bind**関数によって返される値です。

バインドは、2つの処理方向(「描画結果 フィールド」と「フィールド 描画結果」)のどちらかで機能します。バインドが「描画結果 フィールド」の処理方向でアクティブの場合は、<エリア>内の選択された図形が変更されると、バインドフィールドも変更されます。バインドが「フィールド 属性」の処理方向でアクティブの場合は、バインドフィールドが変更されると、<エリア>内の選択された図形の属性も変更されます。

<方向>は、バインドを最初にアクティブにした際に何を行うかを指定します。<方向>が1の場合は、選択図形の属性をバインドフィールドに代入します。2の場合は、バインドフィールドの値で図形の属性を更新します。バインドがアクティブになると、バインドは両方の処理方向で機能します。

現在、アクティブ状態にあるバインドを削除する、あるいは1つのエリアに対して複数のバインドを同時にアクティブにすることはできません。別のバインドに変更する場合は、まず**DR DEACTIVATE BIND**コマンドによりバインドをアクティブでない状態にしなければ

ばなりません。バインドは、アクティブでない状態にするか、エリアを閉じるまでアクティブ状態のままです。

例題

次の例は、バインドの作成方法とバインドをアクティブ状態にする方法を示しています。

vBind:=**DR New bind**

DR ADD TO BIND (vBind;0;1;1)

`図形IDを最初のフィールドに結合する

DR ADD TO BIND (vBind;5;1;2)

`図形の幅を2番目のフィールドに結合する

DR ADD TO BIND (vBind;6;1;3)

`図形の高さを3番目のフィールドに結合する

DR ADD TO BIND (vBind;11;1;4)

`図形の回転を4番目のフィールドに結合する

DR ACTIVATE BIND (Area;vBind;1)

`バインドをアクティブ状態にする

参照

DR DEACTIVATE BIND

DR ADD TO BIND

DR ADD TO BIND (バインドID;属性;テーブル;フィールド)

引数	タイプ	説明
バインドID	倍長整数	バインドID
属性	倍長整数	属性コード
テーブル	整数	テーブル番号
フィールド	整数	フィールド番号

説明

DR ADD TO BINDコマンドは、<バインドID>で指定したバインドの<属性>で指定した図形属性を、<テーブル>と<フィールド>で指定したフィールドに結合します。

このコマンドは、フィールドに関連づけられている属性毎に実行してください。27種類の図形属性がバインド可能ですが、1つのバインド内では属性、フィールドはそれぞれ一度しか使用できません。

フィールドと結合した属性を変更する場合は、**DR REMOVE FROM BIND**コマンドで属性とフィールド結合を削除してから、**DR ADD TO BIND**コマンドを新たに使用して、属性とフィールドを再結合します。バインドには、アクティブにする属性とフィールドの結合が最低1つは含まれていなければなりません。バインドがアクティブになると、そのバインドをアクティブでない状態にしてから、アクティブ状態に戻さなければ、新しい属性を追加してもエリアには反映されません。

各属性は、特定のタイプのフィールドに結合する必要があります。適切なタイプに結合しないと、属性は正しく機能しません。表現精度の制約が大きいタイプ（つまり、実数ではなく倍長整数）のフィールドに属性をバインドすると、正しい値が反映されないこともあります。属性に使用するフィールドタイプは、「付録D 図形コード」を参照してください。

注：図形IDのように、修正できない属性もいくつかあります。

属性にバインドしたフィールドに不適切な値（負の幅等）を与えると、そのフィールドは前回の値にリセットされます。

例題

DR ACTIVATE BINDコマンドの例を参照してください。

参照

DR REMOVE FROM BIND

DR DEACTIVATE BIND

DR DEACTIVATE BIND (エリア)

引数	タイプ	説明
エリア	倍長整数型の変数	4D Drawエリア

説明

DR DEACTIVATE BIND コマンドは、<エリア>のバインドをアクティブでない状態にします。バインドを変更する前に、そのバインドを一度アクティブでない状態にしてから、再度アクティブ状態に戻さなければなりません。このコマンドを使用して、バインドをアクティブでない状態にしてからバインドを変更または削除します。

例題

次の例は、エリアのバインドをどのように修正するかを示したものです。

DR DEACTIVATE BIND (Area)

DR ADD TO BIND (vBind;12;1;5)

DR ACTIVATE BIND (Area;vBind;1)

参照

DR ACTIVATE BIND

DR DELETE BIND

DR DELETE BIND (バインドID)

引数	タイプ	説明
バインドID	倍長整数	バインドID

説明

DR DELETE BINDコマンドは、バインドを削除して使用メモリを開放します。バインドは、メモリをほとんど消費しませんが、複雑なデータベースでは多数のバインドを使用することもあります。このコマンドは、バインドに使用されるメモリを開放します。ただし、バインドがアクティブ状態の場合は、削除できません。

例題

次の例は、不要になったバインドを削除します。

```
vBind=DR New bind                                `新しいバインドを作成する
`いくつかの処理をここで実行する
DR DELETE BIND (vBind)                            `バインドを削除してメモリを開放する
```

参照

DR New bind

DR New bind

DR New bind 倍長整数

この関数には、引数はありません。

説明

DR New bind関数は、メモリ内に新しいバインドを作成し、そのバインドにアクセスするために使用する値を返します。

返される値をバインドIDと呼びます。バインドIDは、すべてのバインドコマンドで使用します。バインドIDは、倍長整数です。バインドは、4D Drawエリアに固有のものではありません。

バインドを作成する場合は、フィールドに関連づけする各属性に対して**DR ADD TO BIND**コマンドを使います。

例題

DR ACTIVATE BINDコマンドの例を参照してください。

参照

DR DELETE BIND

DR REMOVE FROM BIND

DR REMOVE FROM BIND (バインドID;属性)

引数	タイプ	説明
バインドID	倍長整数	バインドID
属性	倍長整数	オブジェクト属性

説明

DR REMOVE FROM BINDコマンドは、<属性>に指定した図形の属性を、<バインドID>で指定したバインドから削除します。このコマンドを使用して属性を削除すると、その属性を別のフィールドにバインドすることができます。

バインドがアクティブになると、そのバインドをアクティブでない状態にしてからアクティブ状態に戻さなければ、属性を削除してもエリアには反映されません。

属性に使用するフィールドタイプは、「付録D 図形コード」を参照してください。

参照

DR ADD TO BIND

属性を求める (Get Attributes)

DR GET ARC SPECS

DR GET ARC SPECS (エリア;スコープ;開始角;長さ;半径H;半径V;中心H;中心V)

引数	タイプ	説明
エリア	倍長整数型の変数	4D Drawエリア
スコープ	倍長整数	1=全図形、0=選択図形、>0=図形ID
開始角	整数型の変数	始点の角度 (単位: °)
長さ	整数型の変数	弧の長さ (単位: °)
半径H	実数型の変数	横の半径
半径V	実数型の変数	縦の半径
中心H	実数型の変数	弧の中心の横方向の位置
中心V	実数型の変数	弧の中心の縦方向の位置

説明

DR GET ARC SPECS コマンドは、**<エリア>** の **<スコープ>** で指定した弧の仕様を、**<開始角>**、**<長さ>**、**<半径H>**、**<半径V>**、**<中心H>**、**<中心V>** の各変数に返します。

<スコープ>=-1 の場合は、**<エリア>** の最初の弧の仕様を返します。

<スコープ>=0 の場合は、選択図形内の最初の弧の仕様を返します。

<スコープ>>0 の場合は、図形IDで指定した弧の仕様を返します。対応する図形が存在しない場合は、各属性に“-32000”を返し、**DR Error**関数にエラーコード2を返します。**<スコープ>** で指定した図形が弧でない場合は、各属性に“-32000”を返し、**DR Error**関数にエラーコード47を返します

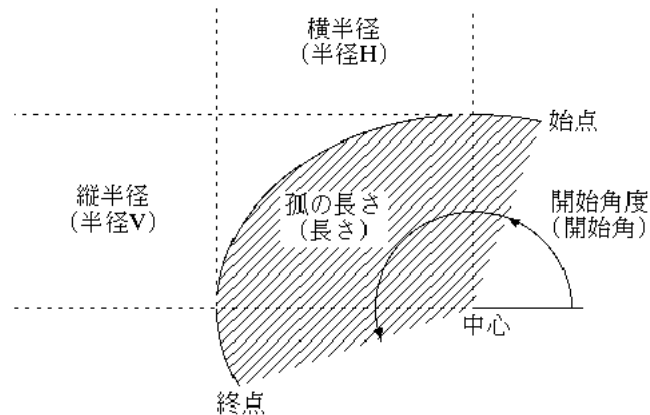
<開始角> は、弧の開始角度 (単位: °) です。この **<開始角>** が弧を描き始める始点で、0° から反時計回りに移動します。

<長さ> は、弧の長さ(単位: °)です。弧の終点は“開始点+長さ%360”で、最大値は359°です。

<半径H>と<半径V>は、弧の横方向と縦方向の半径で、“ベースの単位”で表します。弧は円の一部ではなく楕円の一部です。したがって、2つの数値が異なってもかまいません。<半径H>と<半径V>が等しい場合は、円の一部（弧）になります。

<中心H>と<中心V>は、原点に対する弧の中心の横方向と縦方向の位置で、“ベースの単位”で表します。“ベースの単位”から“目盛の単位”に変換する場合は、**DR Base to scale**関数を使用します。

次の図は、弧に関連する属性を示したものです。



例題

次の例は、**DR GET ARC SPECS**コマンドを使用して、選択された弧の始点と終点を表示するアラートボックスを表示します。

```
DR GET ARC SPECS (Area;0;vStart;vLength;vRadiusH;vRadiusV;vCenterH;vCenterV)
If (0=DR Error)
  ALERT ("弧の始点："+String (vStart)+Char (13)+"弧の終点："+String (vStart+vLength))
Else
  ALERT ("1つの弧を選択してください!")
End if
```

参照

DR Draw arc、**DR SET ARC SPECS**

DR Get attribute lock

DR Get attribute lock (エリア;スコープ;属性コード) 整数

引数	タイプ	説明
エリア	倍長整数型の変数	4D Drawエリア
スコープ	倍長整数	-1 =全図形、0 =選択図形、>0 =図形ID
属性コード	整数	ロック状態を調べたい属性のコード (0 から28)
戻り値	整数	「属性コード」の状態

説明

DR Get attribute lock関数は、<属性コード>で指定した属性の状態を返します。

<属性コード>には、0から28までの整数を指定します。属性コードに関する詳細は、「付録D 図形コード」を参照してください。

この関数が0を返したら、<属性コード>の属性はロックされません。

この関数が1を返したなら、<属性コード>の属性はロックされます。

参照

なし

DR GET BOUNDARY

DR GET BOUNDARY (エリア;スコープ;左;上;右;下)

引数	タイプ	説明
エリア	倍長整数型の変数	4D Drawエリア
スコープ	倍長整数	-1=全図形、0=選択図形、>0=図形ID
左	実数型の変数	左の境界
上	実数型の変数	上の境界
右	実数型の変数	右の境界
下	実数型の変数	下の境界

説明

DR GET BOUNDARY コマンドは、**<エリア>**の**<スコープ>**で指定した図形の境界を、**<左>**、**<上>**、**<右>**、**<下>**の各変数に返します。

境界は図形を囲む最小の矩形の位置で、原点を基準に“ベースの単位”で表します。“ベースの単位”から“目盛の単位”に変換する場合は、**DR Base to scale**関数を使用します。

<スコープ>=-1の場合は、全図形の境界を返します。

<スコープ>=0の場合は、選択図形の境界を返します。

<スコープ>>0の場合は、図形IDで指定した図形の境界を返します。対応する図形が存在しない場合は、各境界に“-32000”を返し、**DR Error**関数にエラーコード2を返します。

例題

次の例は、エリアを含むレイアウト上のボタンのスクリプトです。このボタンをクリックすると、選択図形の左上端が見えるようにドキュメントがスクロールされます。

```
DR GET BOUNDARY (Area;0;$Left;$Top;$Right;$Low)
```

```
DR SCROLL DOCUMENT (Area;$Left;$Top)
```

参照

DR Get ID、**DR GET AREA BOUNDARY**、**DR Base to scale**、**DR SET ORIGIN**

DR Get corner rounding

DR Get corner rounding (エリア;スコープ) 実数

引数	タイプ	説明
エリア	倍長整数型の変数	4D Drawエリア
スコープ	倍長整数	-2=デフォルト、-1=全図形、 0=選択図形、>0=図形ID
戻り値	数値	「スコープ」で指定した図形の角の丸み

説明

DR Get corner rounding関数は、<エリア>の<スコープ>で指定した図形の角の丸みを返します。角の丸みは、“ベースの単位”で表します。

“ベースの単位”から“目盛の単位”に変換する場合は、**DR Base to scale**関数を使用します。

<スコープ>=-2の場合は、角の丸い矩形のデフォルトの角の丸みを返します。

<スコープ>=-1の場合は、全矩形の角の丸みを返します。角の丸みの異なる複数の図形が含まれる場合は、“-32000”を返し、**DR Error**関数にエラーコード29を返します。

<スコープ>=0の場合は、選択図形の角の丸みを返します。角の丸みの異なる複数の図形が含まれる場合は、“-32000”を返し、**DR Error**関数にエラーコード29を返します。

<スコープ>>0の場合は、図形IDで指定した図形の角の丸みを返します。対応する図形が存在しない場合は、“-32000”を返し、**DR Error**関数にエラーコード2を返します。

例題

次の例は、選択図形の角の丸みを求め、角の丸みを“ベースの単位”で0.1増加します。

```
$Temp=DR Get corner rounding (Area;0)
If ($Temp#-32000)
  DR SET CORNER ROUNDING (Area;0;$Temp+0.1)
End if
```

参照

DR Get ID、**DR SET CORNER ROUNDING**

DR GET ENDMARKS

DR GET ENDMARKS (エリア;スコープ;タイプ;端点)

引数	タイプ	説明
エリア	倍長整数型の変数	4D Drawエリア
スコープ	倍長整数	-2=デフォルト、-1=全図形、 0=選択図形、>0=図形ID
タイプ	整数型の変数	1=矢印、2=パーマーク
端点	整数型の変数	0=線のみ、1=始点、2=終点、3=両端

説明

DR GET ENDMARKSコマンドは、<エリア>の<スコープ>で指定した線の終点マークを設定する値を、<タイプ>と<端点>の整数型の変数に返します。

<スコープ>=-2の場合は、デフォルトの終点マークを返します。

<スコープ>=-1の場合は、すべての線の終点マークを返します。

<スコープ>=0の場合は、選択された線の終点マークを返します。

<スコープ>>0の場合は、図形IDで指定した線の終点マークを返します。対応する線が存在しない場合は、各属性に“-32000”を返し、**DR Error**関数にエラーコード2を返します。

<タイプ>は、線の終点マークの種類を表します。たとえ終点マークが表示されなくても、各線は、必ず1つの終点マークを持ちます。<タイプ>が1の場合は、線は矢印の終点マークです。2の場合は、パーマークの終点マークです。

<端点>は、線のどちらの端点が終点マークを持つかを表します。<端点>が0の場合は線だけです。1の場合は線の始点、2の場合は線の終点、3の場合は線の両端に終点マークを持ちます。

<スコープ>で指定した対象に<タイプ>または<端点>の異なる複数の線が含まれる場合は、対応する引数に“-32000”を返し、**DR Error**関数にエラーコード29を返します。

例題

次の例は、選択図形の図形IDとタイプを求め、それが線であれば、線のどちらの端点が終点マークを持つかを表示するアラートボックスを開きます。

```
$Temp:=DR Get ID (Area;0;1)           `選択図形の図形IDを求める
$Type:=DR Get object type (Area;$Temp) ` $Tempのタイプを求める
If ($Type=9)                          `図形が線の場合
    DR GET ENDMARKS (Area;$Temp;vKind;vPoint) `終点マークを求める
    Case of
        ¥(vPoint=1)                    `始点到終点マーク
            ALERT ("この線は始点到終点マークがあります。")
        ¥(vPoint=2)                    `終点到終点マーク
            ALERT ("この線は終点到終点マークがあります。")
        ¥(vPoint=3)                    `両端に終点マーク
            ALERT ("この線は両端に終点マークがあります。")
        Else                          `終点マークなし
            ALERT ("この線は終点マークがありません。")
    End case
End if
```

参照

DR SET ENDMARKS

DR GET FILL ATTRIBUTES

DR GET FILL ATTRIBUTES (エリア;スコープ;パターン;カラー)

引数	タイプ	説明
エリア	倍長整数型の変数	4D Drawエリア
スコープ	倍長整数	-2=デフォルト、-1=全図形、 0=選択図形、>0=図形ID
パターン	整数型の変数	パターン番号
カラー	倍長整数型の変数	カラー設定値

説明

DR GET FILL ATTRIBUTESコマンドは、<エリア>の<スコープ>で指定した図形の塗りつぶし属性を、「パターン」と<カラー>の各変数に返します。

<スコープ>=-2の場合は、デフォルトの塗りつぶし属性を返します。

<スコープ>=-1の場合は、全図形の塗りつぶし属性を返します。

<スコープ>=0の場合は、選択図形の塗りつぶし属性を返します。

<スコープ>>0の場合は、図形IDで指定した図形の属性を返します。対応する図形が存在しない場合は、各属性に“-32000”を返し、**DR Error**関数にエラーコード2を返します。

引数<スコープ>のコードに関する詳細は、第1章「4D Drawランゲージの概要」を参照してください。

<パターン>は、パレット内のパターン番号です。パレットの左から右、上から下の順序で1~36の値が付けられます。

<カラー>は、図形のカラーを表す倍長整数です。**DR SET FILL ATTRIBUTES**コマンドと**DR SET LINE ATTRIBUTES**コマンドでこの数値を使用すると、他の図形に同じカラーを設定することができます。

<スコープ>で指定した対象に属性の異なる複数の図形が含まれる場合は、対応する引数に“-32000”を返します。

例題

次の例は、図形IDをユーザに要求し、IDの正当性を確認します。さらに、図形の塗りつぶしパターンとカラーを求め、図形を強調するためにパターンとカラーを変更するグローバルプロシージャを呼び出します。

```
$Temp:=Num (Request ("どの図形ですか?";"1")) `図形ID番号を求める
If ((OK=1)&($Temp>0)) `正しい番号が得られた場合
  DR GET FILL ATTRIBUTES (Area;$Temp;vPattern;vColor)
  `塗りつぶし属性を求める
  If (vPattern#-32000) `図形が存在する場合
    HIGHLIGHT (Area;$Temp;vPattern;vColor) `グローバルプロシージャに情報を
    `受け渡す
  Else `その他の場合
    ALERT ("このIDの図形がありません!") `ユーザに通知する
  End if
End if
```

参照

DR SET FILL ATTRIBUTES

DR Get handle state

DR Get handle state (エリア;スコープ) 整数

引数	タイプ	説明
エリア	倍長整数型の変数	4D Drawエリア
スコープ	倍長整数	-1 =全図形、0 =選択図形、>0 =図形ID
戻り値	整数	指定した図形のハンドルが表示されているかどうかを示す。

説明

DR Get handle state関数は、<スコープ>内の図形のハンドルが表示されているかどうかを1、または0で示します。

参照

なし

DR GET HIGHLIGHT

DR GET HIGHLIGHT (エリア;先頭;最後)

引数	タイプ	説明
エリア	倍長整数型の変数	4D Drawエリア
先頭	整数型の変数	先頭の文字位置-1
最後	整数型の変数	最後の文字位置

説明

DR GET HIGHLIGHTコマンドは、<エリア>の選択文字列の位置を、<先頭>と<最後>の各変数に返します。

<先頭>は、選択された先頭の文字位置よりも1だけ小さく、<最後>は選択された最後の文字位置です。<先頭>と<最後>が等しい場合、文字は選択されていません。この時挿入ポイントは、<先頭>と<最後>+1の間にあります。

同時に処理できる文字図形は1つだけなので、スコープは必要ありません。<エリア>に選択文字図形が存在しない場合は、<先頭>に“-32000”を返し、**DR Error**関数にエラーコード56を返します。

例題

次の例は、選択文字列の位置を返します。文字列が選択されていない場合は、ユーザに警告します。

```
DR GET HIGHLIGHT (Area;$First;$Last)  
If (56=DR Error)  
  ALERT ("文字列が選択されていません。")  
End if
```

参照

DR SET HIGHLIGHT

DR Get ID

DR Get ID (エリア;スコープ;インデックス) 倍長整数

引数	タイプ	説明
エリア	倍長整数型の変数	4D Drawエリア
スコープ	倍長整数	-1=全図形、0=選択図形、>0=グループID
インデックス	倍長整数	リスト内のインデックス番号
戻り値	倍長整数	図形ID

説明

DR Get ID関数は、<エリア>の<スコープ>と<インデックス>で指定した図形の固有IDを返します。この数値は他の多くの4D Drawコマンドで使用し、“図形ID”として参照されます。

図形IDを求める場合は、参照する図形セットを指定し、そのセットにおける図形の順位を指定します。順位は、背面から前面へ順に付けられます。最背面にある図形のインデックスが1です。

<スコープ>=-1の場合は、<エリア>の図形の順位を<インデックス>で指定します。
<スコープ>=0の場合は、選択図形内の図形の順位を<インデックス>で指定します。
<スコープ>>0の場合は、グループIDで指定したグループ内の図形の順位を<インデックス>で指定します。この構文を使用すれば、グループを解除せずにグループ内の図形を処理することができます。

引数<スコープ>のコードは、第1章「4D Drawランゲージの概要」を参照してください。

例題

1. 次の例は、選択図形のIDの求め方を示しています。

```
vID:=DR Get ID (Area;0;1)           `選択図形のIDを求める
```

2. 次のコードセグメントは、異なる図形セットのループ方法と、それらのIDを求める方法を示しています。

`全図形を処理する

\$Count:=**DR Count** (Area;-1)

For (\$i;1;\$Count)

 vObjectID:=**DR Get ID** (Area;-1;\$i)

`ここで各図形に対する処理を実行する

End for

`選択図形内の全図形を処理する

\$Count:=**DR Count** (Area;0)

For (\$i;1;\$Count)

 vObjectID:=**DR Get ID** (Area;0;\$i)

`ここで各図形に対する処理を実行する

End for

`グループ内の全図形を処理する

\$Count:=**DR Count** (Area;vGrID)

For (\$i;1;\$Count)

 vObjectID:=**DR Get ID** (Area;vGrID;\$i)

`ここで各図形に対する処理を実行する

End for

参照

DR Count

`全図形の数を求める

`図形の数だけ繰り返す

`図形IDを求める

`選択図形の数を求める

`図形の数だけ繰り返す

`図形IDを求める

`グループ内の図形の数を求める

`図形の数だけ繰り返す

`図形IDを求める

DR GET LINE ATTRIBUTES

DR GET LINE ATTRIBUTES (エリア;スコープ;パターン;カラー;幅)

引数	タイプ	説明
エリア	倍長整数型の変数	4D Drawエリア
スコープ	倍長整数	-2=デフォルト、-1=全図形、 0=選択図形、>0=図形ID
パターン	整数型の変数	パターン番号
カラー	倍長整数型の変数	カラーの設定値
幅	実数型の変数線	幅 (単位:ピクセル)

説明

DR GET LINE ATTRIBUTESコマンドは、<エリア>の<スコープ>で指定した図形の線属性を、<パターン>、<カラー>、<幅>の各変数に返します。線属性は図形の境界線にも設定されています。

<スコープ>=-2の場合は、デフォルトの線属性を返します。

<スコープ>=-1の場合は、全図形の線属性を返します。

<スコープ>=0の場合は、選択図形の線属性を返します。

<スコープ>>0の場合は、図形IDで指定した図形の属性を返します。対応する図形が存在しない場合は、各属性に“-32000”を返し、**DR Error**関数にエラーコード2を返します。

<パターン>は、パレット内のパターン番号です。パレットの左から右、上から下の順序で1~36の値が付けられます。

<カラー>は、図形のカラーを表す倍長整数です。**DR SET FILL ATTRIBUTES**コマンドと**DR SET LINE ATTRIBUTES**コマンドでこの数値を使用すると、他の図形に同じカラーを設定することができます。

<幅>は、線幅(線の太さ)を表すピクセル単位の実数です。

<スコープ>で指定した対象に属性の異なる複数の図形が含まれる場合は、対応する引数に“-32000”を返します。

例題

この例題は、線パターン、線カラー、線幅のデフォルトを確認し、線幅が0.25ピクセルの黒の実線でない場合は、線幅が0.25ピクセルの黒の実線にリセットします。

```
DR GET LINE ATTRIBUTES (Area;-2;vPattern;vColor;vWide)
```

```
If ((vPattern#3)|(vColor#0)|(vWide#0.25))
```

```
  DR SET LINE ATTRIBUTES (Area;-2;3;0;0.25)
```

```
End if
```

参照

DR SET LINE ATTRIBUTES

DR GET LINE SPECS

DR GET LINE SPECS (エリア;スコープ;開始H;開始V;終了H;終了V)

引数	タイプ	説明
エリア	倍長整数型の変数	4D Drawエリア
スコープ	倍長整数	-1=全図形、0=選択図形、>0=図形ID
開始H	実数型の変数	横方向の開始位置
開始V	実数型の変数	縦方向の開始位置
終了H	実数型の変数	横方向の終了位置
終了V	実数型の変数	縦方向の終了位置

説明

DR GET LINE SPECSコマンドは、<エリア>の<スコープ>で指定した線の両端の位置を、<開始H>、<開始V>、<終了H>、<終了V>の各変数に返します。

<スコープ>=-1の場合は、<エリア>の最初の図形の両端の位置を返します。

<スコープ>=0の場合は、選択図形内の最初の図形の両端の位置を返します。

<スコープ>>0の場合は、図形IDで指定した線の両端の位置を返します。対応する図形が存在しない場合は、各属性に“-32000”を返し、DR Error関数にエラーコード2を返します。また、<スコープ>で指定した図形が線でない場合は、各属性に“-32000”を返し、DR Error関数にエラーコード47を返します。

<開始H>と<開始V>は、線の始点の横方向と縦方向の位置です。

<終了H>と<終了V>は、線の終点の横方向と縦方向の位置です。

線を作成する場合、その線が描き始められる点が始点で、マウスから指が離される点が終点です。線の始点が必ずしも上とは限りません。また、線の終点の左とも限りません。終点マークの追加する場合は、線の始点と終点を確認することが重要です。

両端の位置は、原点の現在位置を基準に“ベースの単位”で表します。“ベースの単位”から“目盛の単位”に変換する場合は、DR Base to scale関数を使用します。

例題

次の例は、選択された線の周りに楕円を描き、ドキュメントの前面にその線を移します。

```
DR GET LINE SPECS (Area;0;startH;startV;EndH;EndV)
If (0=DR Error)
  Oval:=DR Draw oval (Area;startH-0.25;startV-0.25;EndH-0.25;EndV-0.25;0)
  DR DO COMMAND (Area;5001)           `前面に線を移す
Else
  ALERT ("最初に線を選択してください!")
End if
```

参照

DR SET LINE SPECS、**DR SET ENDMARKS**

DR Get name

DR Get name (エリア;スコープ) 文字列

引数	タイプ	説明
エリア	倍長整数型の変数	4D Drawエリア
スコープ	倍長整数	-2=デフォルト、-1=全図形、 0=選択図形、>0=図形ID
戻り値	文字列	図形の名前

説明

DR Get name関数は、<エリア>の<スコープ>で指定した1つまたは複数の図形の名前を返します。

名前は、図形に付属した文字列で、必ずしも固有のものではありません。名前の長さは最大31文字（バイト）です。また名前は、「属性設定」ダイアログボックスを使用して、または**DR SET NAME**コマンドを使用してメソッドから設定することができます。

<スコープ>=-2の場合は、デフォルトの名前を返します。通常、デフォルトの名前は空の文字列です。メソッド以外では使用できません。

<スコープ>=-1の場合は、全図形の名前を返します。名前の異なる複数の図形が含まれる場合は、“*****”を返し、**DR Error**関数にエラーコード29を返します。

<スコープ>=0の場合は、選択図形の名前を返します。名前の異なる複数の図形が含まれる場合は、“*****”を返し、**DR Error**関数にエラーコード29を返します。

<スコープ>>0の場合は、図形IDで指定した図形の名前を返します。対応する図形が存在しない場合は、空の文字列を返し、**DR Error**関数にエラーコード2を返します。

引数<スコープ>のコードは、第1章「4D Drawランゲージの概要」を参照してください。

例題

次の例は、エリアを含むレイアウト上のボタンのスクリプトです。このスクリプトは、1つ以上の図形が選択されているかどうかを確認し、選択されていれば、変数“\$Name”に図形の名前を代入します。また、名前がすべて同じかどうかを判定して、警告ボックスを表示します。

```
If (DR Count (Area;0)>0)                `1つ以上の図形が選択されている場合
    $Name:=DR Get name (Area;0)          `名前を求め
    If (29=DR Error)                      `名前が同じかどうかを確認する
        ALERT ("選択図形に複数の名前が含まれています!") `ユーザに通知
    Else                                   `名前が同じ場合
        ALERT ("選択図形の名前はすべて<"+$Name+">です。") `ユーザに名前を通知
    End if
End if
```


参照

DR Get ID、DR SET NAME

DR Get object type**DR Get object type (エリア;スコープ) 整数**

引数	タイプ	説明
エリア	倍長整数型の変数	4D Drawエリア
スコープ	倍長整数	-1=全図形、0=選択図形、>0=図形ID
戻り値	整数	図形のタイプ

説明

DR Get object type関数は、<エリア>の<スコープ>で指定した1つまたは複数の図形の図形タイプを返します。

図形タイプは整数コードで構成されます。図形タイプは、図形作成後は変更できません。

<スコープ>=-1の場合は、全図形の図形タイプを返します。図形タイプの異なる複数の図形が含まれる場合は、“-32000”を返し、**DR Error**関数にエラーコード29を返します。
 <スコープ>=0の場合は、選択図形の図形タイプを返します。図形タイプの異なる複数の図形が含まれる場合は、“-32000”を返し、**DR Error**関数にエラーコード29を返します。
 <スコープ>>0の場合は、図形IDで指定した図形の図形タイプを返します。対応する図形が存在しない場合は、“-32000”を返し、**DR Error**関数にエラーコード2を返します。

引数<スコープ>に渡されるコードの詳細は、第1章「4D DRAWランゲージの概要」の「スコープの指定」の節を参照してください。

次の表は、全図形コードのリストです。

コード	図形タイプ
1	文字図形
2	ホットリンク
3	PICT
4	ビットマップ図形
5	矩形 / 角の丸い矩形
6	多角形 / フリーハンド図形
7	楕円
8	弧
9	線
10	グループ

例題

次の例は、エリアを含むレイアウト上のボタンのスクリプトです。このスクリプトは、選択図形が矩形かどうかを判定して、角を丸めます。矩形以外の図形を選択した場合は、ユーザに警告します。

```
If (DR Get object type (Area;0)=5)           `選択図形が矩形かどうかを判定する
    DR SET CORNER ROUNDING (Area;0;0;2)      `選択図形の角を丸める
Else                                           `その他のタイプを選択した場合
    ALERT ("矩形以外の図形は角の丸みを設定できません!") `ユーザに警告する
End if
```

参照

DR Get ID

DR GET POLYGON VERTEX

DR GET POLYGON VERTEX (エリア;スコープ;頂点番号;頂点H;頂点V)

引数	タイプ	説明
エリア倍	倍長整数型の変数	4D Drawエリア
スコープ	倍長整数	-1=全図形、0=選択図形、>0=図形ID
頂点番号	整数	編集する頂点の番号
頂点H	実数型の変数	横方向の位置
頂点V	実数型の変数	縦方向の位置

説明

DR GET POLYGON VERTEXコマンドは、<エリア>の<スコープ>で指定した多角形の<頂点番号>に対応する頂点の位置を、<頂点H>と<頂点V>の各変数に返します。

<スコープ>=-1の場合は、<エリア>の最初の図形の位置を返します。

<スコープ>=0の場合は、選択図形内の最初の図形の頂点位置を返します。

<スコープ>>0の場合は、図形IDで指定した多角形の頂点の位置を返します。対応する図形が存在しない場合は何も実行せずに、**DR Error**関数にエラーコード2を返します。また、<スコープ>で指定した図形が多角形でない場合は、<頂点H>と<頂点V>に“-32000”を返し、**DR Error**関数にエラーコード47を返します。

「<頂点番号>」は、多角形内の頂点の番号で、作成順に番号が付けられています。頂点番号が多角形の頂点の数よりも大きい場合は何も実行せずに、**DR Error**関数にエラーコード49を返します。

<頂点H>と<頂点V>は頂点の位置で、どちらも“ベースの単位”で表します。“ベースの単位”から“目盛の単位”に変換する場合は、**DR Base to scale**関数を使用します。また、<頂点H>と<頂点V>は、原点からのオフセット値を返します。正の値は原点の下または右、負の値は原点の上または左の位置を表します。

例題

DR SET POLYGON VERTEXコマンドの例を参照してください。

参照

DR SET POLYGON VERTEX

DR Get refnum

DR Get refnum (エリア;スコープ) 倍長整数

引数	タイプ	説明
エリア	倍長整数型の変数	4D Drawエリア
スコープ	倍長整数	-2=デフォルト、-1=全図形、 0=選択図形、>0=図形ID
戻り値	倍長整数	図形の参照番号

説明

DR Get refnum関数は、<エリア>の<スコープ>で指定した図形の参照番号を返します。

参照番号は図形に付属した倍長整数で、必ずしも固有のものではありません。参照番号は、プロシージャ以外では使用できません。

<スコープ>=-2の場合は、デフォルトの参照番号を返します。

<スコープ>=-1の場合は、全図形の参照番号を返します。参照番号の異なる複数の図形が含まれる場合は、“-32000”を返し、**DR Error**関数にエラーコード29を返します。

<スコープ>=0の場合は、選択図形の参照番号を返します。参照番号の異なる複数の図形が含まれる場合は、“-32000”を返し、**DR Error**関数にエラーコード29を返します。

<スコープ>>0の場合は、図形IDで指定した図形の参照番号を返します。対応する図形が存在しない場合は、“-32000”を返し、**DR Error**関数にエラーコード2を返します。

例題

次の例は、エリアを含むレイアウト上のボタン用のスクリプトです。このスクリプトは、1つ以上の図形が選択されているかどうかを確認し、対応するレコードを[Parts]テーブルから検索し、その概要を表示します。

```
If (DR Count (Area;0)=1)
  QUERY ([Parts];[Parts]Refnum =DR Get refnum (Area;0))
  ALERT ("この図形は"+[Parts]Description)
End if
```

参照

DR Get ID、**DR SET REFNUM**

DR Get rotation

DR Get rotation (エリア;スコープ) 整数

引数	タイプ	説明
エリア	倍長整数型の変数	4D Drawエリア
スコープ	倍長整数	-1=全図形、0=選択図形、>0=図形ID
戻り値	整数	図形の回転度数

説明

DR Get rotation関数は、<エリア>の<スコープ>で指定した図形の回転角度を0°～359°の範囲で返します。

回転の度数は、0°から反時計回りに求めます。

<スコープ>=-1の場合は、全図形の回転角度を返します。<スコープ>で指定した図形の回転角度が異なる場合は、“-32000”を返し、**DR Error**関数にエラーコード29を返します。

<スコープ>=0の場合は、選択図形の回転角度を返します。<スコープ>で指定した図形の回転角度が異なる場合は、“-32000”を返し、**DR Error**関数にエラーコード29を返します。

<スコープ>>0の場合は、図形IDで指定した図形の回転角度を返します。対応する図形が存在しない場合は、“-32000”を返し、**DR Error**関数にエラーコード2を返します。

例題

次の例は、エリアの図形が回転されているかどうかを確認し、もし回転されていれば、図形の回転角度を0°にリセットします。

```
If (DR Get rotation (Area;-1)=-32000)
  DR ROTATE (Area;-1;0;0)
End if
```

参照

DR Get ID、**DR ROTATE**

DR Get text

DR Get text (エリア;スコープ) テキスト

引数	タイプ	説明
エリア	倍長整数型の変数	4D Drawエリア
スコープ	倍長整数	-3=選択文字列、-1=全図形、 0=選択図形、>0=図形ID
戻り値	テキスト	文字図形の文字列

説明

DR Get text関数は、<エリア>の<スコープ>で指定した文字図形の文字列を返します。

<スコープ>=-3の場合は、選択文字列を返します。挿入ポイントが文字間にある場合は、空の文字列を返します。文字列が選択されていない場合は、**DR Error**関数にエラーコード56を返します。

<スコープ>=-1の場合は、<エリア>の最初の図形の文字列を返します。

<スコープ>=0の場合は、選択図形内の最初の図形の文字列を返します。

<スコープ>>0の場合は、図形IDで指定した文字図形の文字列を返します。対応する文字図形が存在しない場合は、空の文字列を返し、**DR Error**関数にエラーコード2を返します。

引数<スコープ>のコードは、第1章「4D Drawランゲージの概要」を参照してください。

例題

DR SET HIGHLIGHTコマンドの例を参照してください。

参照

DR SET TEXT

DR GET TEXT ATTRIBUTES

DR GET TEXT ATTRIBUTES (エリア;スコープ;フォント;サイズ;書体;行揃え;フレーム;拡張方向)

引数	タイプ	説明
エリア	倍長整数型の変数	4D Drawエリア
スコープ	倍長整数	-3=選択文字列、-2=デフォルト、 -1=全図形、0=選択図形、>0=図形ID
フォント	整数型の変数	フォント番号
サイズ	整数型の変数	フォントサイズ(単位:ポイント)
書体	整数型の変数	フォントの書体
行揃え	整数型の変数	フォントの行揃え
フレーム	整数型の変数	0=可変、1=固定
拡張方向	整数型の変数	0=ダウン、1=アップ

説明

DR GET TEXT ATTRIBUTESコマンドは、<エリア>の<スコープ>で指定した文字図形の文字属性を、<フォント>、<サイズ>、<書体>、<行揃え>、<フレーム>、<フロー>の各整数型の変数に返します。

<スコープ>=-3の場合は、選択文字列の文字属性を返します。挿入ポイントが文字と文字の間にある場合は、挿入ポイントの左側の文字の属性を返します。

<スコープ>=-2の場合は、デフォルトの文字属性を返します。

<スコープ>=-1の場合は、全文字図形の文字属性を返します。

<スコープ>=0の場合は、選択された文字図形の文字属性を返します。

<スコープ>>0の場合は、図形IDで指定した文字図形の文字属性を返します。対応する図形が存在しない場合は、各属性に“-32000”を返し、**DR Error**関数にエラーコード2を返します。<スコープ>に設定できる値は次の通りです。

引数<スコープ>のコードは、第1章「4D Drawランゲージの概要」を参照してください。

<フォント>は、システムのフォントIDで、**DR Font number**関数が返す値と同じです。フォント名(Osaka、Geneva等)を求める場合は、**DR Font name**関数を使用します。

<サイズ>は、選択文字列または文字図形の文字サイズ(単位:ポイント)です。

<書体> は、次の表に示す複数の書体番号を加算した合計値です。

値	書体
0	標準
1	ボールド
2	イタリック
4	アンダーライン
8	アウトライン
16	シャドウ

<行揃え> は、文字図形内の文字列の水平方向の整列で、次の表に示す値で表します。

値	行揃え
0	左揃え
1	中揃え
2	右揃え

<スコープ> で指定した対象に属性の異なる文字図形の図形が含まれる場合は、対応する引数に“-32000”を返します。

<フレーム> は、文字図形が可変フレームであるか固定フレームであるかを返します。<フレーム> が0の場合、文字図形の高さは可変です。文字図形の高さは、文字の量と文字図形の幅によって決まります。また、<フレーム> が1の場合、文字図形の高さは固定です。文字図形に納まらない文字はトランケートします。

<拡張方向> は、改行時に文字図形の拡張方向を返します。<拡張方向> が0の場合は、改行時に文字がページに対して下向きに流れ込みます。1の場合は、改行時に文字がページに対して上向きに流れ込みます。

例題

次の例は、選択図形の文字属性を求めて、フォント名をユーザに知らせる警告ボックスを表示します。

DR GET TEXT ATTRIBUTES

(Area;0;vFont;vSize;vStylelet;vJust;vFrame;vFlow)

If (vFont#-32000)

ALERT ("現在のフォント : "+**DR Font name** (vFont))

End if

参照

DR SET TEXT ATTRIBUTES

DR Get text width

DR Get text width (エリア;スコープ) 整数

引数	タイプ	説明
エリア	倍長整数型の変数	4D Drawエリア
スコープ	倍長整数	-3=選択文字列、-1=全図形、 0=選択図形、>0=図形ID
戻り値	整数	テキスト幅

説明

DR Get text width関数は、<スコープ>内のテキストタイプの図形の改行を含む行の中で、最も幅の広い行の幅を返します。幅はドキュメントの定規に指定された単位になっています。

<スコープ>=-3の場合は、選択された文字の幅を返します。

<スコープ>=-1の場合は、4D Draw エリア内の図形すべてを対象にします。

<スコープ>=0の場合は、その時選択されている図形を対象にします。

<スコープ>>0の場合は、引数に指定されたID番号の図形の改行を含む行の中で最も幅の広い行の幅を返します。このID番号に相当する図形が存在しない場合、コマンドは-32000 を返し、**DR Error**関数にエラーコード2を返します。

引数<スコープ>のコードは、第1章「4D Drawランゲージの概要」を参照してください。

現在の定規設定を指定するには、**DR Get ruler**関数を使用します。

参照

なし

ホットリンク (Hot Links)

DR ADD TO HOT LINK

DR ADD TO HOT LINK (エリア;スコープ;名前)

引数	タイプ	説明
エリア	倍長整数型の変数	4D Drawエリア
スコープ	倍長整数	-1=全図形、0=選択図形、>0=図形ID
名前	文字列	ホットリンクの名前

説明

DR ADD TO HOT LINKコマンドは、<エリア>の<スコープ>で指定した図形を、<名前>で指定したホットリンクに追加します。

<スコープ>=-1の場合は、全図形をホットリンクに追加します。

<スコープ>=0の場合は、選択図形をホットリンクに追加します。

<スコープ>>0の場合、スコープには図形IDを渡します。コマンドはそのオブジェクトをホットリンクに追加します。対応する図形が存在しない場合は、**DR Error**関数にエラーコード2を返します。

<名前>が<エリア>のホットリンクでない場合は、**DR ADD TO HOT LINK**コマンドは何も実行せずに、**DR Error**関数にエラーコード23を返します。

例題

次の例は、ユーザに名前を入力を要求して、指定されたホットリンクに選択図形を追加します。ホットリンクが存在しない場合は、ユーザに警告します。

```
$Temp:=Request ("選択図形をホットリンクに追加します")
If (OK=1)
  DR ADD TO HOT LINK (Area;0;$Temp)
  If (DR Error = 23)
    ALERT ("指定されたホットリンクは存在しません")
  End if
End if
```

参照

DR PUBLISH、**DR Subscribe**

DR PUBLISH

DR PUBLISH (エリア;スコープ;名前)

引数	タイプ	説明
エリア	倍長整数型の変数	4D Drawエリア
スコープ	倍長整数	-1=全図形、0=選択図形、>0=図形ID
名前	文字列	ホットリンクの名前

説明

DR PUBLISHコマンドは、<エリア>の<スコープ>で指定した図形を「名前」で指定した新しいホットリンクとして公開します。

<スコープ>=-1の場合は、全図形を公開します。

<スコープ>=0の場合は、選択図形を公開します。

<スコープ>>0の場合、スコープには図形IDを渡します。コマンドはそのオブジェクトのみを公開します。対応する図形が存在しない場合は、**DR Error**関数にエラーコード2を返します。また、<名前>で指定したホットリンクが既に存在する場合は、**DR Error**関数にエラーコード45を返します。

例題

次の例は、ユーザに名前を入力を要求して、選択図形をホットリンクとして公開します。その名前が既存する場合は、ユーザに警告します。

```
$Temp:=Request ("図形をホットリンクとして公開します")
If (OK=1)
  DR PUBLISH (Area;0;$Temp)
  If (45=DR Error)
    ALERT ("この名前のホットリンクは既存します。別の名前を選択してください")
  End if
End if
```

参照

DR Get ID、**DR Subscribe**、**DR UNPUBLISH**

DR Subscribe

DR Subscribe (エリア;名前) 倍長整数

引数	タイプ	説明
エリア	倍長整数型の変数	4D Drawエリア
名前	文字列	ホットリンクの名前
戻り値	倍長整数	新しい図形ID

説明

DR Subscribe関数は、既存する <名前> のホットリンクを使用して新しい図形を作成し、その図形IDを返します。

この関数は、<エリア> に新しい図形を作成します。

<名前> はPICTタイプのホットリンク名でなければなりません。<名前> が存在しない場合は、“-32000” を返し、**DR Error**関数にエラーコード23を返します。

例題

次の例は、ユーザに名前の入力を要求して、指定されたホットリンクを使用します。ホットリンクの使用に成功した場合は、ホットリンク図形の上に、ホットリンク名を表示する新しい文字図形を作成します。

```
$Temp:=Request ("ホットリンクを使用します") `ホットリンク名を求める
If (OK=1) `入力確認
  DR SET UPDATE MODE (Area;0) `画面更新をオフ
  $ID:=DR Subscribe (Area;$Temp) `ホットリンクを使用
  If ($ID>0) `正しいIDを得た場合
    DR GET BOUNDARY (Area;$ID;$Left;$Top;$Right;$Low)
    `ホットリンクの境界を求める
    $ID:=DR Draw text (Area;$Left;$Top-0.25;$Right;$Top;$Temp) `文字図形を作成
    DR SET TEXT ATTRIBUTES (Area;$ID;-1;-1;0;1) `文字を中揃え
  End if
  DR SET UPDATE MODE (Area;1) `画面更新をオン
End if
```

参照

DR PUBLISH、**DR UNPUBLISH**

DR UNPUBLISH

DR UNPUBLISH (エリア;名前)

引数	タイプ	説明
エリア	倍長整数型の変数	4D Drawエリア
名前	文字列	ホットリンクの名前

説明

DR UNPUBLISHコマンドは、<エリア>から公開したホットリンクを<名前>で指定して公開を解除します。このコマンドは、「ホットリンク公開解除」ダイアログボックスでホットリンクを選択するのと同じです。

<名前>が存在しない場合、または<名前>が<エリア>から公開されたホットリンクでない場合は何も実行しません。

例題

次の例は、ユーザに名前を入力を要求して、指定されたホットリンクの公開を解除します。

```
$Temp:=Request ("ホットリンクの公開を解除します")
If (OK=1)
    DR UNPUBLISH (Area;$Temp)
End if
```

参照

DR PUBLISH、**DR Subscribe**

DR UNSUBSCRIBE

DR UNSUBSCRIBE (エリア;名前)

引数	タイプ	説明
エリア	倍長整数型の変数	4D Drawエリア
名前	文字列	ホットリンクの名前

説明

DR UNSUBSCRIBE コマンドは、<エリア> で使用したホットリンクを <名前> で指定して使用を解除します。

解除すると <エリア> から、すべての <名前> のコピーを消去します。このコマンドは、「ホットリンク使用解除」ダイアログボックスから <名前> を選択するのと同じです。ホットリンク図形を1つだけ消去する場合は、単に、その図形を削除します。

例題

次の例は、ユーザに名前を入力を要求して、指定されたホットリンクの使用を削除します。

```
$Temp:=Request ("ホットリンクの使用を解除します")  
If (OK=1)  
    DR UNSUBSCRIBE (Area;$Temp)  
End if
```

参照

DR PUBLISH、**DR Subscribe**

取り込み / 書き出し (Import and Export)

DR ARRAY TO ATTRIBUTE

DR ARRAY TO ATTRIBUTE (エリア;スコープ;属性;配列)

引数	タイプ	説明
エリア	倍長整数型の変数	4D Drawエリア
スコープ	倍長整数	-1=全図形、0=選択図形、>0=グループID
属性	倍長整数	属性コード
配列	配列	ソース配列

説明

DR ARRAY TO ATTRIBUTEコマンドは、<エリア>の<スコープ>で指定した図形に対して<配列>の各属性を適用して修正します。

最背面にある図形が配列の最初の要素に対応します。

<スコープ>=-1の場合は、全図形に配列を適用します。ただし、グループも1つの図形として扱います。

<スコープ>=0の場合は、選択図形に配列を適用します。ただし、グループも1つの図形として扱います。

<スコープ>>0の場合は、グループIDで指定したグループに含まれる図形に配列を適用します。この構文を使用して、グループを解除しないままグループ内の図形を修正することができます。

DR ARRAY TO ATTRIBUTEコマンドは、配列の要素に対応する図形を修正するだけです。図形よりも配列要素が多く存在する場合でも、図形は作成しません。対応する要素の意味を調べるには、その属性を反映するコマンドを参照してください。例えば、塗りつぶしパターンの意味を知りたいければ、**DR SET FILL ATTRIBUTES**コマンドを参照してください。

DR ARRAY TO ATTRIBUTEコマンドは、図形の1つの“レベル”についてだけ情報を修正します。グループに含まれる図形を修正する場合は、グループIDを<スコープ>として使用します。

また、位置とサイズの各属性は“ベースの単位”で指定します。“ベースの単位”の配列を“目盛の単位”の配列に変換する場合は、**DR ARRAY BASE TO SCALE**コマンドを使用します。属性コードと対応する配列タイプは、「付録D 図形コード」を参照してください。属性には、修正できないものもあるので注意してください。

例題

次の例は、カラー（倍長整数）を含むレコードを検索します。検索結果のカラーから配列を作成し、この配列を選択図形に適用します。選択図形の数、カラー数より少なくします。

```
QUERY ([Colors];[Colors]Kind="標準")
SELECTION TO ARRAY ([Colors]Color;aColor)
If (DR Count (Area;0)<=Size of array (aColor))
  DR ARRAY TO ATTRIBUTE (Area;0;13;aColor)
Else
  ALERT ("選択図形が多すぎます!")
End if
```

参照

DR Get ID、DR ATTRIBUTE TO ARRAY

DR Array to polygon

DR Array to polygon (エリア;配列H;配列V) 倍長整数

引数	タイプ	説明
エリア	倍長整数型の変数	4D Drawエリア
配列H	数値配列	頂点の横方向の値の配列
配列V	数値配列	頂点の縦方向の値の配列
戻り値	倍長整数型	新しい図形ID

説明

DR Array to polygon関数は、<エリア>に<配列H>と<配列V>をもとに新しい多角形を作成し、新しい図形IDを返します。

<配列H>と<配列V>は、多角形の各頂点の位置が格納されていなければなりません。2つの配列のタイプは実数、倍長整数、整数のいずれかで、“ベースの単位”で指定します。“目盛の単位”の配列を“ベースの単位”の配列に変換する場合は、**DR ARRAY SCALE TO BASE**コマンドを使用します。多角形を作成するには各配列に3つ以上の要素が必要です。2つの配列の要素数が異なる場合は、要素数の少ない方の配列の要素数を使用します。また、閉じた多角形を作成する場合は、配列の最初の値と最後の値が一致しなければなりません。

例題

1.次の例は、**DR POLYGON TO ARRAY**コマンドの例と逆の処理を行います。各レコードから2つの配列を作成し、各配列が3つ以上の要素を持つ場合は、配列から多角形を作成します。

```

ARRAY REAL (ArrayH;0)           `配列を宣言する
ARRAY REAL (ArrayV;0)
SELECTION TO ARRAY ([Polygon]FieldH;ArrayH;[Polygon]FieldV;ArrayV) `配列を作成
If (Size of array (ArrayH)>=3)   `配列が3つ以上の要素をもつ場合
    $New:=DR Array to polygon (Area;ArrayH;ArrayV) `多角形を作成する
End if

```

2.次の例は、エリアを含むレイアウト上のボタンのスクリプトです。ユーザに頂点の数を
入力要求し、入力された頂点の数を使用して多角形を作成します。

```
$x:=Num (Request ("頂点の数はいくつですか?")) `頂点の数を入力要求
If (OK=1) `入力確認
  DR SET ORIGIN (Area;2;2;0) `原点を(2,2)に設定
  ARRAY REAL (ArrayH;$x) `配列を宣言
  ARRAY REAL (ArrayV;$x)
  For ($i;1;$x) `配列に値を代入
    ArrayH{$i}:=Sin ($i)
    ArrayV{$i}:=Cos ($i)
  End for
  $New:=DR Array to polygon (Area;ArrayH;ArrayV) `多角形を作成する
End if
```

参照

DR POLYGON TO ARRAY、DR GET POLYGON VERTEX

DR ATTRIBUTE TO ARRAY

DR ATTRIBUTE TO ARRAY (エリア;スコープ;属性;配列)

引数	タイプ	説明
エリア	倍長整数型の変数	4D Drawエリア
スコープ	倍長整数	-1=全図形、0=選択図形、>0=グループID
属性	倍長整数	属性コード
配列	配列	作成する配列

説明

DR ATTRIBUTE TO ARRAYコマンドは、<エリア>の<スコープ>で指定した各図形の属性の値を配列に返します。

<スコープ>=-1の場合は、<エリア>の各図形の属性を返します。ただし、グループも1つの図形として扱います。

<スコープ>=0の場合は、選択図形内の各図形の属性を返します。ただし、グループも1つの図形として扱います。

<スコープ>>0の場合は、グループIDで指定したグループに含まれる個々の図形の属性を返します。この構文によって、グループを解除しなくてもグループ内の図形の情報を探ることができます。ネストしたグループも**DR ATTRIBUTE TO ARRAY**コマンドを使用することにより調べることができます。次ページのこのコマンドの例を参照してください。

一部の図形の属性が不適切な場合は、“-32000”または“*****”を返します。これは<配列>のタイプによって異なります。例えば、<配列>を属性コード24(角の丸み)で作成した場合、図形の1つに線が含まれていると、その線に対応する要素に“-32000”が設定されます。また、属性が異なる図形を含んだグループを指定した場合も、“-32000”または“*****”を返します。

DR ATTRIBUTE TO ARRAYコマンドが返す値は、指定した属性コードに基づきます。

対応する要素の意味を調べるには、その属性を反映するコマンドを参照してください。例えば、塗りつぶしパターンの意味を知りたいければ、**DR SET FILL ATTRIBUTES**コマンドを参照してください。

DR ATTRIBUTE TO ARRAYコマンドは、図形の1つの“レベル”についてだけ情報を返します。例えば、<スコープ>が0で、選択図形がグループ化された図形を含む場合、グループも1つの配列要素になります。さらに、グループIDを<スコープ>に指定して、グループの内部を“覗く”ことができます。

位置とサイズの各属性は“ベースの単位”で返します。“ベースの単位”の配列から“目盛の単位”の配列に変換する場合は、**DR ARRAY BASE TO SCALE**コマンドを使用しません。

属性コードと対応する配列タイプは、「付録D 図形コード」を参照してください。

例題

次の例は、単独の図形またはグループ化された図形に関係なく、エリアの全図形の図形IDと図形タイプの配列を作成する方法を示しています。

```
C_LONGINT ($i;$j)           `ループカウンタとして使用
ARRAY LONGINT (aID;0)       `IDリストを格納
ARRAY LONGINT (aType;0)     `タイプリストを格納
ARRAY LONGINT (aGrpID;0)    `グループ内の図形IDを求めるのに使用
ARRAY LONGINT (aGrpType;0)  `グループ内の図形タイプを求めるのに使用
$i:=0                       `ループカウンタを初期化
`グループ化されていない図形のIDで配列を作成
DR ATTRIBUTE TO ARRAY (Area;-1;0;aID)
`グループ化されていない図形のタイプで配列を作成
DR ATTRIBUTE TO ARRAY (Area;-1;1;aType)
Repeat                       `すべてのグループを“覗き”終るまで繰り返す
  $i:=$i+1                   `カウンタをインクリメント
  If (aType{$i}=10)          `図形がグループの場合
    `グループ内の図形IDで配列を作成する
    DR ATTRIBUTE TO ARRAY (Area;aID{$i};0;aGrpID)
    `グループ内の図形タイプで配列を作成する
    DR ATTRIBUTE TO ARRAY (Area;aID{$i};1;aGrpType)
    `このグループを保持するための要素をID配列に挿入する
    INSERT ELEMENT (aID;$i+1;Size of array (aGrpID))
    `このグループを保持するための要素をタイプ配列に挿入する
    INSERT ELEMENT (aType;$i+1;Size of array (aGrpType))
    For ($j;1;Size of array (aGrpID)) `グループ配列をループする...
      aID{$i+$j}:aGrpID{$j}          `グループ配列内のIDを主配列にコピーする
      aType{$i+$j}:aGrpType{$j}     `グループ配列内のタイプを主配列にコピー
    `する
  End for
End if
Until ($i=Size of array (aID))    `全図形を調べ終わるまで続ける
```

参照

DR ARRAY TO ATTRIBUTE、**DR Get ID**

DR POLYGON TO ARRAY

DR POLYGON TO ARRAY (エリア;スコープ;配列H;配列V;曲線制御ポイント)

引数	タイプ	説明
エリア	倍長整数型の変数	4D Drawエリア
スコープ	倍長整数	-1=全図形、0=選択図形、>0=図形ID
配列H	数値配列	頂点の横方向の値の配列
配列V	数値配列	頂点の縦方向の値の配列
曲線制御ポイント	数値配列	1=曲線制御ポイント、 2=曲線制御ポイントでない

説明

DR POLYGON TO ARRAYコマンドは、<スコープ>で指定した多角形の各頂点の位置を、<配列H>と<配列V>の各配列に返します。

<スコープ>=-1の場合は、<エリア>の最初の図形の各頂点を返します。

<スコープ>=0の場合は、選択図形内の最初の図形の各頂点を返します。

<スコープ>>0の場合は、図形IDで指定した多角形の各頂点を返します。対応する図形が存在しない場合は、空の配列を返し、**DR Error**関数にエラーコード2を返します。

<スコープ>で指定した図形が多角形でない場合は、空の配列を返し、**DR Error**関数にエラーコード47を返します。

<配列H>と<配列V>は、多角形の各頂点の横方向と縦方向の位置で、“ベースの単位”で表します。“ベースの単位”の配列から“目盛の単位”の配列に変換する場合は、**DR ARRAY BASE TO SCALE**コマンドを使用します。また、<配列H>と<配列V>は実数、倍長整数、整数タイプの配列で、**DR POLYGON TO ARRAY**コマンドを呼び出す前に存在しなければなりません。表現精度の制約が大きいタイプ（倍長整数または整数）の配列を使用すると、精度に合わせて値がトランケートされ一部の情報が失われる場合があります。

引数<曲線制御ポイント>は、制御ポイントが曲線制御ポイントであるかどうかを決定します。この曲線制御ポイントは、**DR POLYGON CURVE**コマンドでのみ作成することができます。

例題

次の例は、選択した多角形の頂点を2つの配列に書き込み、4th Dimensionの**ARRAY TO SELECTION**コマンドを使用して、配列要素を各レコードに書き込みます。

```
ARRAY REAL (ArrayH;0)           `配列を宣言する
ARRAY REAL (ArrayV;0)
DR POLYGON TO ARRAY (Area;0;ArrayH;ArrayV) `頂点を求める
If (DR Error =0)
  `各レコードを作成する
  ARRAY TO SELECTION (ArrayH;[Polygon]FieldH;ArrayV;[Polygon]FieldV)
End if
```

参照

DR Array to polygon、**DR Get object type**

図形作成 (Object Creation)

DR Draw arc

DR Draw arc (エリア;開始角;長さ;半径H;半径V;中心H;中心V) 倍長整数

引数	タイプ	説明
エリア	倍長整数型の変数	4D Drawエリア
開始角	整数	開始点の角度 (単位: °)
長さ	整数	弧の長さ (単位: °)
半径H	実数	横の半径
半径V	実数	縦の半径
中心H	実数	弧の中心の横方向の位置
中心V	実数	弧の中心の縦方向の位置
戻り値	倍長整数	新しい図形ID

説明

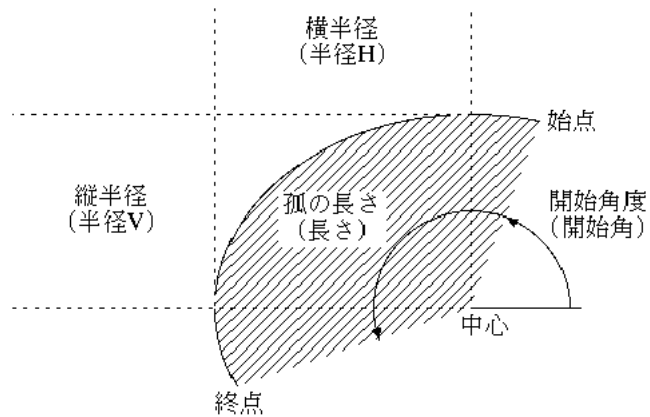
DR Draw arc関数は、<エリア>に新しい弧を作成し、その図形IDを返します。新しい弧は、デフォルトの線属性と塗りつぶし属性を使用して作成されます。

<開始角>は、弧の開始角度 (単位: °) です。この開始角が弧を描き始める始点で、0°から反時計回りに移動します。360°未満でなければなりません。<長さ>は、弧の長さ (単位: °) です。弧の終点は“開始点+長さ%360”で、最大値は359°です。

<半径H>と<半径V>は、弧の横方向と縦方向の半径で、“ベースの単位”で表します。弧は円の一部ではなく楕円の一部です。したがって、2つの数値が異なってもかまいません。<半径H>と<半径V>が等しい場合は、円の一部 (弧) になります。

<中心H>と<中心V>は、原点に対する弧の中心の横方向と縦方向の位置で、“ベースの単位”で表します。“ベースの単位”から“目盛の単位”に変換する場合は、**DR Base to scale**関数を使用します。

次の図は、弧を記述する引数を示したものです。



例題

次の例は、長さ4°の弧を90個作成します。各弧には、それぞれ異なったカラーを使用します。中心の位置は同じです。90個の弧の全体は、レインボーのグラデーションで塗りつぶした円になります。この例は、8ビットのカラーを表示できるカラーモニタが必要です。

For (\$i;0;89)

DR SET FILL ATTRIBUTES (Area;-2;3;**DR Index to color** (\$i+17))

\$ID:=DR Draw arc (Area;\$i*4;4;2;2;2;2)

End for

参照

DR GET ARC SPECS、DR SET ARC SPECS

DR Draw line

DR Draw line (エリア;開始H;開始V;終了H;終了V;モード) 倍長整数

引数	タイプ	説明
エリア	倍長整数型の変数	4D Drawエリア
開始H	実数	横方向の開始位置
開始V	実数	縦方向の開始位置
終了H	実数	横方向の終了位置
終了V	実数	縦方向の終了位置
モード	整数	0=絶対、1=相対
戻り値	倍長整数型	新しい図形ID

説明

DR Draw line関数は、<エリア>に新しい線を作成し、その図形IDを返します。

新しい線は、デフォルトの線属性を使用して作成され、<開始H>、<開始V>、<終了H>、<終了V>によって位置を設定します。4つの位置は、すべて“ベースの単位”で表します。“目盛の単位”から“ベースの単位”に変換する場合は、**DR Scale to base**関数を使用します。

<開始H>と<開始V>は、原点からのオフセット値として指定します。

また、<終了H>と<終了V>は、原点からのオフセット値（絶対）または開始Hと開始Vからのオフセット値（相対）として指定できます。

<モード>が0の場合は、絶対位置、<モード>が1の場合は相対位置になります。正の値は、下または右、負の値は上または左の方向を表します。

例題

1.次の例は、絶対位置で**DR Draw line**関数を使用し、原点をドキュメントの左上端に設定してから、(1,1)で始まり(2,2)で終わる線を描画します。

```
DR SET ORIGIN (Area;0;0;0)
$ID:=DR Draw line (Area;1;1;2;2;0)
```

2.次の例は、相対位置で**DR Draw line**関数を使用し、原点から“ベースの単位”で3だけ右で終わる線を描画します。

```
$ID:=DR Draw line (Area;0;0;3;0;1)
```

参照

DR SET ENDMARKS、**DR SET LINE ATTRIBUTES**

DR Draw oval

DR Draw oval (エリア;開始H;開始V;終了H;終了V;モード) 倍長整数

引数	タイプ	説明
エリア	倍長整数型の変数	4D Drawエリア
開始H	実数	横方向の開始位置
開始V	実数	縦方向の開始位置
終了H	実数	横方向の終了位置
終了V	実数	縦方向の終了位置
モード	整数	0=絶対、1=相対
戻り値	倍長整数型	新しい図形ID

説明

DR Draw oval関数は、<エリア>に新しい楕円を作成し、その図形IDを返します。

新しい楕円は、デフォルトの線属性と塗りつぶし属性で作成され、<開始H>、<開始V>、<終了H>、<終了V>によって位置を設定します。4つの位置は、すべて“ベースの単位”で表します。“目盛の単位”から“ベースの単位”に変換する場合は、**DR Scale to base**関数を使用します。

<開始H>と<開始V>は、原点からのオフセットとして指定します。

また、<終了H>と<終了V>は、原点からのオフセット値（絶対）、または<開始H>と<開始V>からのオフセット値（相対）として指定できます。

<モード>が0の場合は絶対位置、<モード>が1の場合は相対位置になります。正の値は下または右、負の値は上または左の方向を表します。

例題

次の例は、原点を中心に1つの円を作成した後、その円のパターンを変更します。

```
$ID:=DR Draw oval (Area;-0.5;-0.5;0.5;0.5;0)  
DR SET FILL ATTRIBUTES (Area;$ID;3;0)
```

参照

DR SET LINE ATTRIBUTES、**DR SET FILL ATTRIBUTES**

DR Draw rectangle

DR Draw rectangle (エリア;開始H;開始V;終了H;終了V;モード;角の丸み) 倍長整数

引数	タイプ	説明
エリア	倍長整数型の変数	4D Drawエリア
開始H	実数	横方向の開始位置
開始V	実数	縦方向の開始位置
終了H	実数	横方向の終了位置
終了V	実数	縦方向の終了位置
モード	整数	0=絶対、1=相対
角の丸み	実数	角の丸みの大きさ
戻り値	倍長整数型	新しい図形ID

説明

DR Draw rectangle関数は、<エリア>に新しい矩形を作成し、その図形IDを返します。

新しい矩形は、デフォルトの線属性と塗りつぶし属性で作成され、<開始H>、<開始V>、<終了H>、<終了V>によって位置を設定します。4つの位置はすべて“ベースの単位”で表します。“目盛の単位”から“ベースの単位”に変換する場合は、**DR Scale to base**関数を使用します。

<開始H>と<開始V>は、原点からのオフセット値として指定します。

また、<終了H>と<終了V>は、原点からのオフセット値(絶対)、または<開始H>と<開始V>からのオフセット値(相対)として指定できます。

<モード>が0の場合は、絶対位置、<モード>が1の場合は相対位置になります。正の値は下または右、負の値は上または左の方向を表します。

<角の丸み>は、新しい矩形の角の丸みの大きさを“ベースの単位”で指定します。<角の丸み>が0の場合は、角の丸みのない矩形を作成します。

例題

次の例は、ランダムなパターンとカラーで10×10の格子型に矩形を作成します。画面更新をオンのままでメソッドの実行状況を表示します。

```
DR SET ORIGIN (Area;0;0;0)           `原点をドキュメントの左上端に設定する
For ($i;0;9)                         `$iは各矩形の横方向の位置
  For ($j;0;9)                         `$jは各矩形の縦方向の位置
    `デフォルトの塗りつぶし属性をランダムなカラーとパターンに設定する
    DR SET FILL ATTRIBUTES (Area;-2;Random%33+3;Random^2)
    $ID:=DR Draw rectangle (Area;$i;$j;1;1;1;0) `新しい矩形を相対位置で描画する
  End for
End for
```

参照

DR SET LINE ATTRIBUTES、**DR SET FILL ATTRIBUTES**

DR Draw text

DR Draw text (エリア;開始H;開始V;終了H;終了V;テキスト;{;フレーム;フロー})
倍長整数

引数	タイプ	説明
エリア	倍長整数型の変数	4D Drawエリア
開始H	実数	型の変数横方向の開始位置
開始V	実数	型の変数縦方向の開始位置
終了H	実数	型の変数横方向の終了位置
終了V	実数	型の変数縦方向の終了位置
テキスト	テキスト	図形のテキスト
フレーム	整数	0=可変、1=固定
フロー	整数	0=ダウン、1=アップ
戻り値	倍長整数型	新しい図形ID

説明

DR Draw text関数は、<エリア>に新しい文字図形を作成し、その図形IDを返します。

新しい図形は、<開始H>、<開始V>、<終了H>、<終了V>によって位置を設定し、<テキスト>で文字図形内のテキストを設定します。4つの位置は、すべて原点から“ベースの単位”で表します。“目盛の単位”から“ベースの単位”に変換する場合は、**DR Scale to base**関数を使用します。

<テキスト>に、最低1文字(バイト)以上の長さの文字列がなければ図形は作成されません。新しい文字図形のフォント、サイズ、書体、縦揃えは、デフォルトの文字属性によって決まります。**DR SET TEXT ATTRIBUTES**コマンドを参照してください。

オプションの<フレーム>は、文字図形が可変フレームか固定フレームかを指定します。<フレーム>が0、または指定しない場合、図形の高さは可変になります。

図形の高さは、文字の量と文字図形の幅によって決まります。<フレーム>が1の場合、図形の高さは固定で、文字図形内に納まりきれない文字は、トランケートします。また、<フレーム>を省略した場合は、<フロー>も省略しなければなりません。

オプションの<フロー>は、改行時に図形内に文字が流れ込む方向を指定します。フローが0、または指定されない場合は、改行時に文字がページに対して下向きに流れ込みます。<フロー>が1の場合は、改行時に文字がページに対して上向きに流れ込みます。

<フレーム>と<フロー>は、“参照”を含む文字図形を使用する場合、特に便利です。プリント中に、文字列の長さに応じて文字図形のサイズと流れ込む方向を制御できます。

例題

DR Subscribe関数の例を参照してください。

参照

DR GET TEXT ATTRIBUTES、DR SET TEXT ATTRIBUTES

DR End polygon

DR End polygon (エリア) 倍長整数

引数	タイプ	説明
エリア	倍長整数型の変数	4D Drawエリア
戻り値	倍長整数型	新しい図形ID

説明

DR End polygon関数は、<エリア>の多角形の描画を完了して、その図形IDを返します。新しい多角形は、デフォルトの線属性と塗りつぶし属性を使用して作成されます。

例題

次の例は、ページの左上端から2単位だけ下、2単位だけ右に中心を持つ複雑な多角形を作成します。

```
DR START POLYGON (Area)
DR SET ORIGIN (Area;2;2;0)
For ($i;1;51)
  DR POLYGON LINE (Area;Sin ($i);Cos ($i);0)
End for
$ID:=DR End polygon (Area)
```

参照

DR POLYGON LINE、**DR START POLYGON**

DR Objects to bitmap

DR Objects to bitmap (エリア;スコープ;処理) 倍長整数

引数	タイプ	説明
エリア	倍長整数型の変数	4D Drawエリア
スコープ	倍長整数	-1 =全図形、0 =選択図形、>0 =図形ID
処理	整数	「スコープ」の図形の変換方法
戻り値	倍長整数型	新しい図形ID

説明

DR Objects to bitmap関数は、<エリア>と<スコープ>で指定された図形をビットマップに変換します。

<処理>が0の場合、<スコープ>の図形は白黒のビットマップに変換されます。<処理>が1の場合、<スコープ>の図形はカラーのビットマップに変換されます。<処理>が2の場合、<スコープ>の図形は背景が白のビットマップに変換されます。

いずれの場合も、**DR Objects bitmap**関数によって生成されたピクチャタイプの図形IDが返されます。

カラービットマップに使用される色の数は、コントロールパネルのモニタの設定によって変わります。カラービットマップは自動的にピクチャタイプの図形に変換されるので、コントロールパネルのモニタで設定したカラー数に変更されても、ビットマップイメージが影響を受けることはありません。

注：図形からビットマップイメージへの変換を行うと、元に戻すことができなくなります。つまり、ビットマップイメージにすると、それまで持っていた属性が名前も含め、すべて失われてしまいます。

参照

DR ADD TO BITMAP

DR PLACE PICTURE

DR PLACE PICTURE (エリア;ピクチャ;位置)

引数	タイプ	説明
エリア	倍長整数型の変数	4D Drawエリア
ピクチャ	ピクチャ	4Dのピクチャフィールドまたはピクチャ変数
位置	整数	0=標準、1=中央、2=原点

説明

DR PLACE PICTUREコマンドは、<エリア>の<位置>で指定した位置にピクチャをペーストします。

<ピクチャ>は、4th Dimensionで使用できるピクチャでなければなりません。

<位置>が0の場合は、<エリア>の最後にマウスをクリックした位置に<ピクチャ>をペーストします。

<位置>が1の場合は、<エリア>を表示するウィンドウの中央に表示します。

<位置>が2の場合は、原点にペーストします。

例題

次の例は、レコードのピクチャフィールド “[Logos]Logos” の内容をエリアにペーストします。図形は、6単位ごとに配置されます。

```
DR SET UPDATE MODE (Area;0)
ALL RECORDS ([Logos])
For ($i;0;Records in selection ([Logos])-1)
  DR SET ORIGIN (Area;$i%6+1;$i //6+1;0)
  DR PLACE PICTURE (Area;[Logos]Logos;2)
  NEXT RECORD ([Logos])
End for
DR SET UPDATE MODE (Area;1)
```

参照

DR Area to picture、**DR Clipboard to picture**、**DR SET ORIGIN**

DR POLYGON CURVE

DR POLYGON CURVE (エリア;X1;Y1;X2;Y2;終点X;終点Y;モード)

引数	タイプ	説明
エリア	倍長整数型の変数	4D Drawエリア
頂点1.X1	数値	頂点1のX座標
頂点1.Y1	数値	頂点1のY座標
頂点2.X2	数値	頂点2のX座標
頂点2.Y2	数値	頂点2のY座標
終点X	数値	終点のX座標
終点Y	数値	終点のY座標
モード	数値	0=絶対、1=相対

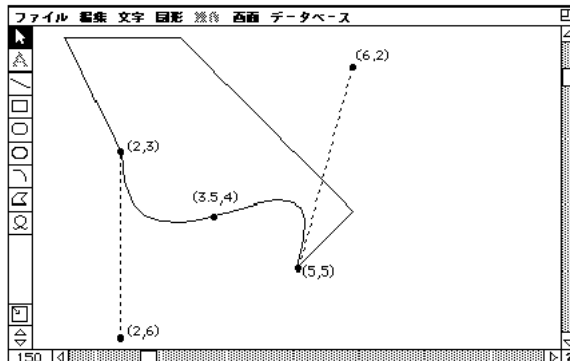
説明

DR POLYGON CURVEコマンドは、曲線を描きます。線は開始点から、<頂点1>の方向に向けて引かれます。2番目の線は終点から、<頂点2>の方向に向かいます。線は、開始点と終点の途中で交わるような曲線になります。

例題

次のメソッドは、数本の直線と1本の曲線を使った多角形を作成します。

```
DR START POLYGON (Area)  
DR POLYGON LINE (Area;1;1;0)  
DR POLYGON LINE (Area;3;1;0)  
DR POLYGON LINE (Area;6;4;0)  
DR POLYGON LINE (Area;5;5;0)  
DR POLYGON CURVE (Area;6;2;2;6;2;3;0)  
DR POLYGON LINE (Area;1;1;0)  
$PolyID:=DR End Polygon (Area)
```



頂点 (3,5,4) は、頂点 (2,3) と頂点 (5,5) の中間点で、この2つの曲線が交わる場所です。

参照

DR START POLYGON、DR POLYGON LINE、DR End polygon

DR POLYGON LINE

DR POLYGON LINE (エリア;頂点H;頂点V;モード)

引数	タイプ	説明
エリア	倍長整数型の変数	4D Drawエリア
頂点H	実数	頂点の横方向の位置
頂点V	実数	頂点の縦方向の位置
モード	整数	0=絶対、1=相対

説明

DR POLYGON LINE関数は、<エリア>に作成中の多角形に頂点を追加します。

<頂点H>と<頂点V>は、原点からのオフセット値(絶対)または前回の頂点からのオフセット値(相対)として指定できます。

<モード>が0の場合は絶対位置、<モード>が1の場合は相対位置になります。

多角形の最初の頂点は、絶対位置以外使用できません。正の値は下または右、負の値は上または左の方向を表します。

<頂点H>と<頂点V>は、どちらも“ベースの単位”で表します。“目盛の単位”から“ベースの単位”に変換する場合は、**DR Scale to base**関数を使用します。

閉じた多角形を作成する場合は、最後の頂点が最初の頂点と一致しなければなりません。また、多角形には最低3つの頂点が必要です。

新しい頂点を追加する場合は、**DR START POLYGON**コマンドを使用して描画過程を開始する必要があります。多角形に複数の頂点を追加してから**DR START POLYGON**コマンドを使用すると、それまでに描画した頂点は消去されます。つまり、多角形の描画を最初からやり直すことができます。**DR POLYGON LINE**コマンドの処理は、ユーザが多角形の描画中にマウスをクリックするのと同じです。

例題

次の例は、**DR START POLYGON**コマンドの例と同じ菱形の多角形を相対位置を使用して作成します。

```
DR START POLYGON (Area)  
DR POLYGON LINE (Area;1.5;1;0)  
DR POLYGON LINE (Area;0.5;0.5;1)  
DR POLYGON LINE (Area;-0.5;0.5;1)  
DR POLYGON LINE (Area;-0.5;-0.5;1)  
DR POLYGON LINE (Area;0.5;-0.5;1)  
$ID:=DR End polygon (Area)
```

参照

DR POLYGON CURVE、**DR START POLYGON**、**DR End polygon**

DR START POLYGON

DR START POLYGON (エリア)

引数	タイプ	説明
エリア	倍長整数型の変数	4D Drawエリア

説明

DR START POLYGONコマンドは、<エリア>に新しい多角形の描画を開始します。**DR START POLYGON**コマンドは、新しい多角形を作成するための作業領域をメモリ内に確保するように4D Drawに通知します。多角形は、**DR End polygon**関数を実行するまで描画されません。多角形を作成中に**DR START POLYGON**コマンドを使用すると、描画処理をリセットできます。

例題

次の例は、菱形の多角形を作成します。

```
DR START POLYGON (Area)
DR POLYGON LINE (Area;1.5;1;0)
DR POLYGON LINE (Area;2;1.5;0)
DR POLYGON LINE (Area;1.5;2;0)
DR POLYGON LINE (Area;1;1.5;0)
DR POLYGON LINE (Area;1.5;1;0)
$ID:=DR End polygon (Area)
```

参照

DR POLYGON CURVE、**DR POLYGON LINE**、**DR End polygon**

図形操作 (Object Manipulation)

DR ADD TO BITMAP

DR ADD TO BITMAP (エリア;スコープ;図形ID)

引数	タイプ	説明
エリア	倍長整数型の変数	4D Drawエリア
スコープ	倍長整数	-1=全図形、0=選択図形、>0=図形ID
図形ID	倍長整数	ビットマップイメージのID番号

説明

DR ADD TO BITMAPコマンドは、<スコープ>内の図形を<図形ID>で指定された図形に追加します。

図形からビットマップイメージへの変換を行うと、元に戻すことができなくなります。つまり、ビットマップイメージにすると、それまで持っていた属性が名前も含め、すべて失われてしまいます。

参照

DR Objects to bitmap

DR ALIGN

DR ALIGN (エリア;スコープ;横揃え;縦揃え)

引数	タイプ	説明
エリア	倍長整数型の変数	4D Drawエリア
スコープ	倍長整数	-1=全図形、0=選択図形
横揃え	整数	0=なし、1=左揃え、2=中揃え、3=右揃え
縦揃え	整数	0=なし、1=上揃え、2=中揃え、3=下揃え

説明




DR ALIGNコマンドは、<エリア>の<スコープ>で指定した図形を整列します。

<スコープ>=-1の場合は、全図形を整列します。

<スコープ>=0の場合は、選択図形を整列します。




<スコープ>で指定した図形を<横揃え>と<縦揃え>の各引数に従って整列します。

次の表は、引数<横揃え>の値がどのような効果をもたらし、その値が「整列」ダイアログボックス内のどのアイコンに相当するのを示しています。

値	整列	アイコン
0	変化なし	
1	左揃え	
2	中揃え	
3	右揃え	

<横揃え>が0の場合は、横方向の整列を行いません。1の場合は左揃え、2の場合は中揃え、3の場合は右揃えにします。

次の表は、引数<縦揃え>の値がどのような効果をもたらし、その値が「整列」ダイアログボックス内のどのアイコンに相当するのを示しています。

値	整列	アイコン
0	変化なし	
1	上揃え	
2	中揃え	
3	下揃え	

<縦揃え> が0の場合は、縦方向の整列を行いません。1の場合は上揃え、2の場合は中揃え、3の場合は下揃えにします。

例題

次の例は、選択図形を縦横ともに中揃えにします。

DR ALIGN (Area;0;2;2)

参照

DR Get ID

DR Count

DR Count (エリア;スコープ) 整数

引数	タイプ	説明
エリア	倍長整数型の変数	4D Drawエリア
スコープ	倍長整数	-1=全図形、0=選択図形、>0=グループID
戻り値	整数型	「スコープ」で指定した「エリア」の 「図形の数

説明

DR Count関数は、<スコープ>で指定した<エリア>の図形の数返します。

<スコープ>=-1の場合は、全図形の数返します。ただし、グループも1つの図形として扱います。

<スコープ>=0の場合は、選択図形内の図形の数返します。ただし、グループも1つの図形として扱います。

<スコープ>>0の場合は、グループIDで指定したグループに含まれる図形の数返します。この構文を使用して、グループを解除しないままグループ内の図形の情報求めることができます。また、ネストしたグループは、続けて**DR Count**関数を使用することによって調べることができます。

例題

次の例は、選択図形内の図形の数を表示する警告ボックスを開きます。

```
$Temp:=DR Count (Area;0)  
ALERT ("選択された"+String ($Temp)+"図形")
```

参照

DR Get ID

DR DELETE

DR DELETE (エリア;スコープ)

引数	タイプ	説明
エリア	倍長整数型の変数	4D Drawエリア
スコープ	倍長整数	-1=全図形、0=選択図形、>0=図形ID

説明

DR DELETEコマンドは、<エリア>の<スコープ>で指定した図形を削除します。

<スコープ>=-1の場合は、全図形を削除します。

<スコープ>=0の場合は、選択図形を削除します。

<スコープ>>0の場合は、図形IDで指定した図形を削除します。対応する図形が存在しない場合は、**DR Error**関数にエラーコード2を返します。

例題

次の例は、エリアの図形IDが5の図形を削除します。

DR DELETE (Area;5)

参照

DR Get ID

DR GROUP

DR GROUP (エリア;スコープ)

引数	タイプ	説明
エリア	倍長整数型の変数	4D Drawエリア
スコープ	倍長整数	-1=全図形、0=選択図形

説明

DR GROUPコマンドは、<エリア>の<スコープ>で指定した図形をグループ化します。

- <スコープ>=-1の場合は、全図形をグループ化します。
- <スコープ>=0の場合は、選択図形をグループ化します。

図形をグループ化すると、新しいグループ図形を作成します。

DR GROUPコマンドを使用すると、新しいグループ図形が自動的に選択されます。この状態で、**DR Get ID**関数を使用すればグループの図形IDを求めることができます。<スコープ>で指定した図形が2つ未満の場合は何も実行しません。

例題

次の例は、複数の図形が選択されているかどうかを確認し、選択されていれば、ユーザに名前を付けるように要求します。次に、選択図形をグループ化し、その名前を新しいグループに与えます。

```
If (DR Count (Area;0)>1)
  $Temp:=Request ("グループに名前を付けてください")
  If (OK=1)
    DR GROUP (Area;0)
    DR SET NAME (Area;0;$Temp)
  End if
Else
  ALERT ("図形の数がありません")
End if
```

参照

DR Is a group、**DR Get ID**、**DR UNGROUP**

DR HIDE

DR HIDE (エリア;スコープ;モード)

引数	タイプ	説明
エリア	倍長整数型の変数	4D Drawエリア
スコープ	倍長整数	-1 =全図形、0 =選択図形、>0 =図形ID
モード	整数	1 =非表示、0 =表示

説明

DR HIDEコマンドは、<スコープ>の図形を表示、または非表示にします。このコマンドは、選択された図形に対する、メニューのすべてを表示や表示しないと同じ機能を持ちます。

<スコープ>が0で<モード>が0の場合、非表示になった図形は選択されたままになっています。制御をユーザに戻す、あるいは他のコマンドを実行する前に、必ず図形の選択を解除しておくようにしてください。

参照

なし

DR LOCK

DR LOCK (エリア;スコープ;コード;処理)

引数	タイプ	説明
エリア	倍長整数型の変数	4D Drawエリア
スコープ	倍長整数	-1=全図形、0=選択図形、>0=図形ID
コード	倍長整数	処理する属性
処理	整数	0=ロック解除、1=ロック、2=切り替え

説明

DR LOCKコマンドは、<エリア>の<スコープ>で指定した図形についてコードで指定した属性をロックまたはロック解除します。

<スコープ>=-1の場合は、全図形を処理します。

<スコープ>=0の場合は、選択図形を処理します。

<スコープ>>0の場合は、図形IDで指定した図形を処理します。対応する図形が存在しない場合は、**DR Error**関数にエラーコード2を返します。

<スコープ>で指定した図形を<コード>と<処理>の各引数に従ってロックまたはロック解除します。<コード>は、属性を指定します。属性は、ロックコードの合計した値で表されます。次の表は、ロックコードと対応する属性です。

値	ロック
-1	すべて
1	削除
2	グループ解除
8	名前
32	サイズ
128	位置
2048	回転
4096	塗りつぶしパターン
8192	塗りつぶしカラー
16384	線パターン
32768	線カラー
65536	線幅
131072	終点マーク
524288	文字のフォント
1048576	文字のサイズ
2097152	文字の書体

4194304	文字の行揃え
8388608	文字の編集
16777216	角の丸み
33554432	変形

<処理> は、コードで指定した属性に対して何を実行するかを指定します。<処理> が0の場合は、<コード> で指定した属性はロック解除され、1の場合はロックされ、2の場合は属性の状態がロックならロック解除に、ロック解除ならロックに切り替えます。ただし、<コード> で指定していない属性は変更されません。

例題

次の例は、エリアの選択図形のすべての属性をロック解除します。

DR LOCK (Area;0;-1;0)

参照

DR Get ID

DR MOVE

DR MOVE (エリア;スコープ;移動H;移動V;モード)

引数	タイプ	説明
エリア	倍長整数型の変数	4D Drawエリア
スコープ	倍長整数	-1=全図形、0=選択図形、>0=図形ID
移動H	実数	横の移動量
移動V	実数	縦の移動量
モード	整数	0=絶対、1=相対

説明

DR MOVEコマンドは、<エリア>の<スコープ>で指定した図形を再配置します。

<スコープ>=-1の場合は、<エリア>の全図形を再配置します。

<スコープ>=0の場合は、選択図形内の全図形を再配置します。

<スコープ>>0の場合は、図形IDで指定した図形を再配置します。対応する図形が存在しない場合は、**DR Error**関数にエラーコード2を返します。

<スコープ>で指定した図形を<移動H>、<移動V>、<モード>の各引数に従って移動します。移動Hと移動Vは、原点からのオフセット値（絶対）または図形の現在位置からのオフセット値（相対）として指定できます。

<モード>が0の場合は絶対位置、<モード>が1の場合は相対位置になります。正の値は下または右、負の値は上または左の方向を表します。<移動H>と<移動V>は、どちらも“ベースの単位”で表します。“目盛の単位”から“ベースの単位”に変換する場合は、**DR Scale to base**関数を使用します。

例題

次の例は、“b左”、“b右”、“b上”、“b下”という名前の4つのボタンのスクリプトです。各ボタンは選択図形をベースの1単位だけ所定の方向に移動します。

DR MOVE (Area;0;-1;0;1)	`b左ボタン
DR MOVE (Area;0;1;0;1)	`b右ボタン
DR MOVE (Area;0;0;-1;1)	`b上ボタン
DR MOVE (Area;0;0;1;1)	`b下ボタン

参照

DDR Get ID、**DR SCALE**、**DR SIZE**

DR ROTATE

DR ROTATE (エリア;スコープ;回転角度;モード)

引数	タイプ	説明
エリア	倍長整数型の変数	4D Drawエリア
スコープ	倍長整数	-1=全図形、0=選択図形、>0=図形ID
回転角度	整数	回転する角度（単位：°）
モード	整数	0=絶対、1=相対

説明

DR ROTATEコマンドは、<エリア>の<スコープ>で指定した図形を回転します。

<スコープ>=-1の場合は、全図形を回転します。

<スコープ>=0の場合は、選択図形を回転します。

<スコープ>>0の場合は、図形IDで指定した図形を回転します。対応する図形が存在しない場合は、DR Error関数にエラーコード2を返します。

<スコープ>で指定した図形を<回転角度>と<モード>に従って回転します。

<回転角度>は、絶対または相対として指定することができます。<モード>が0の場合は、<回転角度>は絶対になり、<回転角度>で指定した角度に図形を回転します。<モード>が1の場合は、<回転角度>は相対になり、<回転角度>で指定した角度だけ図形の回転を増減します。<回転角度>が正の値なら反時計回りの回転、負の値なら時計回りの回転を表します。<回転角度>は±359°の範囲でなければなりません。そうでない場合は何も実行せずに、DR Error関数にエラーコード15を返します。

例題

次の例は、エリアの全図形を回転して0°に戻すボタンのスクリプトです。

DR ROTATE (Area;-1;0;0)

参照

DR Get ID、DR Get rotation

DR SCALE

DR SCALE (エリア;スコープ;アンカH;アンカV;高さ;幅)

引数	タイプ	説明
エリア	倍長整数型の変数	4D Drawエリア
スコープ	倍長整数	-1=全図形、0=選択図形、>0=図形ID
アンカH	整数	0=なし、1=左辺、2=中点、3=右辺
アンカV	整数	0=なし、1=上辺、2=中点、3=下辺
幅	実数	水平スケーリング
高さ	実数	垂直スケーリング

説明

DR SCALEコマンドは、<エリア>の<スコープ>で指定した図形をスケーリングします。

<スコープ>=-1の場合は、全図形をスケーリングします。

<スコープ>=0の場合は、選択図形をスケーリングします。

<スコープ>>0の場合は、図形IDで指定した図形をスケーリングします。対応する図形が存在しない場合は、**DR Error**関数にエラーコード2を返します。

<アンカーH>と<アンカーV>は、スケーリング時に図形のどの位置を固定するかを指定します。<アンカーH>が0の場合は、図形の幅を固定し、<幅>の値を無視します。1の場合は図形の左端、2の場合は図形の中点、3の場合は図形の右端を固定します。

<アンカーV>も同様で、0の場合は図形の高さを固定し、<高さ>の値を無視します。1の場合は図形の上端、2の場合は図形の中心、3の場合は図形の下端を固定します。

<スコープ>で指定した図形を<幅>と<高さ>に従ってスケーリングします。<幅>と<高さ>が1よりも大きい場合は図形のサイズを拡大し、1よりも小さい場合は図形のサイズを縮小します。また、図形の新しい幅は、元の幅と<幅>の積に等しくなります。新しい高さも<高さ>を使用して同様に指定します。

例題

次の例は、“b2倍”と“b半分”という名前の2つのボタンのスクリプトです。最初のボタンは選択図形のサイズを2倍にします。2番目のボタンは選択図形のサイズを半分にします。

DR SCALE (Area;0;2;2;2;2) `b2倍ボタン

DR SCALE (Area;0;2;2;0.5;0.5) `b半分ボタン

参照

DR Get ID、**DR MOVE**、**DR SIZE**

DR SIZE

DR SIZE (エリア;スコープ;アンカH;アンカV;幅;高さ;モード)

引数	タイプ	説明
エリア	倍長整数型の変数	4D Drawエリア
スコープ	倍長整数	-1=全図形、0=選択図形、>0=図形ID
アンカH	整数	0=なし、1=左辺、2=中点、3=右辺
アンカV	整数	0=なし、1=上辺、2=中点、3=下辺
幅	実数	幅
高さ	実数	高さ
モード	整数	0=絶対、1=相対

説明

DR SIZEコマンドは、<エリア>の<スコープ>で指定した図形をサイズ変更します。

<スコープ>=-1の場合は、全図形をサイズ変更します。

<スコープ>=0の場合は、選択図形をサイズ変更します。

<スコープ>>0の場合は、図形IDで指定した図形をサイズ変更します。対応する図形が存在しない場合は、**DR Error**関数にエラーコード2を返します。

<アンカーH>と<アンカーV>は、サイズ変更時に図形のどの位置を固定するかを指定します。<アンカーH>が0の場合は、図形の幅を固定し、<幅>の値を無視します。1の場合は、図形の左端、2の場合は図形の中心、3の場合は図形の右端を固定します。

<アンカV>も同様に、0の場合は、図形の高さを固定し、<高さ>の値を無視します。1の場合は図形の上端、2の場合は図形の中心、3の場合は図形の下端が固定します。

<スコープ>で指定した図形を<幅>、<高さ>、<モード>の各引数に従ってサイズ変更します。<幅>と<高さ>は絶対または相対として指定することができます。

<モード>が0の場合は、幅と高さは絶対になり、図形の幅が<幅>、高さが<高さ>になります。<モード>が1の場合は、幅と高さは相対になり、図形のサイズは<幅>と<高さ>に従って増減します。正の値は下または右、負の値は上または左の方向を表します。<幅>と<高さ>はどちらも“ベースの単位”で表します。“目盛の単位”から“ベースの単位”に変換する場合は、**DR Scale to base**関数を使用します。

例題

次の例は、図形の中心をアンカとして使用し、図形IDが5の図形の高さと幅をそれぞれ1単位ずつ増やします。

DR SIZE (Area;5;2;2;1;1;1)

参照

DR Get ID、**DR MOVE**、**DR SCALE**

DR UNGROUP

DR UNGROUP (エリア;スコープ;レベル)

引数	タイプ	説明
エリア	倍長整数型の変数	4D Drawエリア
スコープ	倍長整数	-1=全図形、0=選択図形、>0=グループID
レベル	整数	グループ解除するレベル番号

説明

DR UNGROUPコマンドは、<エリア>の<スコープ>で指定した図形をグループ解除します。

<スコープ>=-1の場合は、全図形をグループ解除します。

<スコープ>=0の場合は、選択図形をグループ解除します。

<スコープ>>0の場合は、グループIDで指定した図形をグループ解除します。対応する図形が存在しない場合は、**DR Error**関数にエラーコード2を返します。

<レベル>は消去すべきグループのレベル(レイヤ数)を制御します。<レベル>が1の場合は、最初のレベルのグループだけをグループ解除します。<レベル>が2の場合は、最初のレベルのグループをグループ解除し、グループ図形内のグループもグループ解除します。<レベル>が-1の場合は、すべてのレベルのグループをグループ解除します。<操作>メニューから<グループ解除>を選択することは、1つのレベルをグループ解除するのと同じです。

例題

次の例は、エリアの全図形を全レベルにわたってグループ解除します。

DR UNGROUP (Area;-1;-1)

参照

DR Get ID、**DR GROUP**

図形選択 (Object Selection)

DR SELECT

DR SELECT (エリア;スコープ;処理)

引数	タイプ	説明
エリア	倍長整数型の変数	4D Drawエリア
スコープ	倍長整数	-1=全図形、0=選択図形、>0=図形ID
処理	整数	0=選択解除、1=選択、2=切り替え

説明

DR SELECTコマンドは、<エリア>の<スコープ>で指定した図形を選択または選択解除します。

<スコープ>=-1の場合は、全図形を処理します。

<スコープ>=0の場合は、選択図形を処理します。

<スコープ>>0の場合は、図形IDで指定した図形を処理します。対応する図形が存在しない場合は何も実行せずに、**DR Error**関数にエラーコード2を返します。

<スコープ>で指定した図形を<処理>に従って選択または選択解除します。<処理>が0の場合は、<スコープ>で指定した図形を選択解除します。1の場合は選択し、2の場合は図形の状態が選択なら選択解除に、選択解除なら選択に切り替えます。

<スコープ>の範囲外にある図形は処理しません。つまり、<エリア>の選択済みで<スコープ>で指定されていない図形は、選択されたままです。

例題

次の例は、エリアの全図形を選択解除した後、図形IDが1の図形を選択します。

```
DR SELECT (Area;-1;0)
```

```
DR SELECT (Area;1;1)
```

参照

DR Get ID、DR SELECT BY ATTRIBUTES、DR SELECT BY REGION、DR SET HIGHLIGHT

DR SELECT BY ATTRIBUTE

DR SELECT BY ATTRIBUTE (エリア;論理演算子;コード;値)

引数	タイプ	説明
エリア	倍長整数型の変数	4D Drawエリア
論理演算子	整数	0=対象のみを選択、1=対象を追加、 2=判定基準に合った図形を選択、 3=判定基準に合った図形を選択解除
コード	整数	属性コード
値	テキスト	属性と比較する値

説明

DR SELECT BY ATTRIBUTEコマンドは、<エリア>の図形を属性をもとに選択または選択解除します。また、コードで指定した各図形の属性と<値>を比較した後、<論理演算子>に従ってそれらの図形を選択または選択解除します。

<論理演算子>が0の場合は、判定基準に合った図形だけを選択します。

<論理演算子>が1の場合は、前もって選択された図形とともに判定基準に合った図形を選択します。

<論理演算子>が2の場合は、選択図形内の判定基準に合った図形を選択します。

<論理演算子>が3の場合は、選択図形内の判定基準に合った図形を選択解除します。

DR SELECT BY ATTRIBUTEコマンドは4th Dimensionの検索機能とよく似ています。繰り返し実行していくつかの検索の判定基準を組み合わせることができます。

ほとんどの属性は数値で表します。<値>のタイプはテキストです。なぜなら、検索する属性の1つに“名前”があります。“名前”は、文字列で表さなければならないからです。その他の属性の場合は、数値の文字表記（"2"または"4.375"など）として指定します。必要な値を知りたいければ、その属性に反映するコマンドを参照してください。例えば、塗りつぶしパターンの指定方法は、**DR SET FILL ATTRIBUTES**コマンドを参照してください。

<コード>は、属性コードの標準リストから求める整数です。属性コードは、「付録D 図形コード」を参照してください。ただし、属性には検索できないものがあります。注意してください。

DR SELECT BY ATTRIBUTEコマンドは、図形IDによる選択はできません。図形IDで選択する場合は、**DR SELECT**コマンドを使用します。

例題

次の例は、パターン3（実線）で塗りつぶされた矩形と楕円をすべて選択します。これらの図形は回転角度が0°で、“名前#" ”です。

```
DR SET UPDATE MODE (Area;0)           `画面更新をオフにする
DR SELECT BY ATTRIBUTE (Area;0;1;"5") `矩形を検索する
DR SELECT BY ATTRIBUTE (Area;1;1;"7") `楕円を検索する
DR SELECT BY ATTRIBUTE (Area;2;12;"3") `実線パターンで塗りつぶされている
DR SELECT BY ATTRIBUTE (Area;2;11;"0") `回転していない
DR SELECT BY ATTRIBUTE (Area;3;3;"")  `名前が空の文字列ではない
DR SET UPDATE MODE (Area;1)           `画面更新をオンにする
```

参照

DR SELECT

DR SELECT BY REGION

DR SELECT BY REGION (エリア;方法;処理;左;上;右;下)

引数	タイプ	説明
エリア	倍長整数型の変数	4D Drawエリア
方法	整数	0=囲む、1=交差する
処理	整数	0=選択解除する、1=選択する、 2=切り替える
左	実数	左の境界
上	実数	上の境界
右	実数	右の境界
下	実数	下の境界

説明

DR SELECT BY REGIONコマンドは、<エリア>の矩形範囲内の図形を選択または選択解除します。このコマンドは、プロシージャから選択矩形をドラッグして図形を選択するのと同じです。

範囲は<左>、<上>、<右>、<下>で記述します。4つの位置は、すべて原点を基準に“ベースの単位”で指定します。“目盛の単位”から“ベースの単位”に変換する場合は、**DR Scale to base**関数を使用します。

<方法> は、図形を完全に囲むかどうかを決定します。<方法>が0の場合は、範囲に完全に囲まれた図形だけが選択されます。

<方法>が1の場合は、範囲に交差した図形が選択されます。

<処理>に従って図形を選択または選択解除します。<処理>が0の場合は、図形を選択解除します。1の場合は、図形を選択します。2の場合は、状態が選択なら選択解除に、選択解除なら選択に切り替えます。

DR SELECT BY REGIONコマンドは、範囲外の図形には反映しません。つまり、<エリア>の選択済みで、範囲内に存在しない図形は選択されたままです。

例題

次の例は、エリアの全図形を選択解除した後、ドキュメントの最初のページの図形を選択します。

DR SELECT (Area;-1;0)

DR SET ORIGIN (Area;0;0;0)

DR SELECT BY REGION (Area;0;1;0;0;7.67;10.14)

参照

DR GET BOUNDARY

プリント (Printing)

DR PRINT

DR PRINT (エリア;エリア;メッセージ;幅;プリントダイアログ;高さ)

引数	タイプ	説明
エリア	倍長整数型の変数	4D Drawエリア
メッセージ	整数	0=キャンセル不可、1=キャンセル可
幅	実数	ドキュメントの幅
プリントダイアログ	整数	0=ダイアログ表示なし、 1=ダイアログ表示あり
高さ	実数	ドキュメントの長さ

説明

DR PRINTコマンドは、<エリア>のドキュメントとしてプリントします。このコマンドを使用すると、「ファイル」メニューから「プリント...」を選択したのと同じ働きをします。ただし、「用紙設定」ダイアログボックスは表示されません。プリントする前に「用紙設定」ダイアログボックスを表示する場合は、**DR DO COMMAND**コマンドを使用します。

<メッセージ>が1の場合は、ダイアログボックスを表示します。ユーザはプリントをキャンセルすることができます。プリントをキャンセルした場合は、**DR Error**関数にエラーコード55を返します。また、<メッセージ>が0の場合は、ダイアログボックスを表示しません。したがって、プリントをキャンセルすることはできません。

<プリントダイアログ>が0の場合は、標準のプリントファイルダイアログボックスが表示されず、プリントはすぐに始まります。1の場合は、標準の「プリントファイル」ダイアログボックスが表示されます。

例題

次の例は、オフスクリーンエリアを作成して、図形をいくつか描画し、そのエリアをプリントします。

```
$Area:=DR New offscreen area           `ここで、図形をいくつか描画する
DR PRINT ($Area;0)                     `メッセージなしでプリントする
DR DELETE OFFSCREEN AREA ($Area)      `処理が終了すると、必ずオフスクリーン
                                        `エリアを削除する
```

参照

DR DO COMMAND、**DR SET DOCUMENT SIZE**

DR PRINT BACKGROUND

DR PRINT BACKGROUND (エリア;メッセージ;プリントダイアログ)

引数	タイプ	説明
エリア	倍長整数型の変数	4D Drawエリア
メッセージ	整数	0=オフ、1=オン
プリントダイアログ	整数	0=ダイアログ表示なし、 1=ダイアログ表示あり

説明

DR PRINT BACKGROUNDコマンドは、<エリア>のバックグラウンド上の図形をプリントします。

<メッセージ>が1の場合は、ダイアログボックスを表示します。ユーザはプリントをキャンセルすることができます。プリントをキャンセルした場合は、**DR Error**関数にエラーコード55を返します。また、<メッセージ>が0の場合は、ダイアログボックスを表示しません。したがって、プリントをキャンセルすることはできません。

<プリントダイアログ>は、標準の「プリント」ダイアログボックスを表示するかどうかを指定します。<プリントダイアログ>が0の場合は、標準の「プリント」ファイルダイアログボックスは表示されず、プリント処理が即座に始まります。<プリントダイアログ>が1の場合は、「プリントファイル」ダイアログボックスが表示されます。

参照

DR PRINT FOREGROUND

DR PRINT FOREGROUND

DR PRINT FOREGROUND (エリア;メッセージ;プリントダイアログ)

引数	タイプ	説明
エリア	倍長整数型の変数	4D Drawエリア
メッセージ	整数	0=オフ、1=オン
プリントダイアログ	整数	0=ダイアログ表示なし、 1=ダイアログ表示あり

説明

DR PRINT FOREGROUNDコマンドは、上のコマンドの逆で、フォアグラウンド上の図形をプリントします。

<メッセージ> が1の場合は、ダイアログボックスを表示します。ユーザはプリントをキャンセルすることができます。プリントをキャンセルした場合は、**DR Error**関数にエラーコード55を返します。また、<メッセージ> が0の場合は、ダイアログボックスを表示しません。したがって、プリントをキャンセルすることはできません。

<プリントダイアログ> は、標準の「プリント」ダイアログボックスを表示するかどうかを指定します。<プリントダイアログ> が0の場合は、標準の「プリント」ファイルダイアログボックスは表示されず、プリント処理が即座に始まります。

<プリントダイアログ> が1の場合は、「プリントファイル」ダイアログボックスが表示されます。

参照

DR PRINT BACKGROUND

DR PRINT MERGE

DR PRINT MERGE (エリア,ファイル番号;メッセージ;プリントダイアログ)

引数	タイプ	説明
エリア	倍長整数型の変数	4D Drawエリア
ファイル番号	整数	ファイル番号
メッセージ	整数	0=オフ、1=オン
プリントダイアログ ボックス	整数	0=ダイアログ表示なし、 1=ダイアログ表示あり

説明

DR PRINT MERGEコマンドを使うと<ファイル番号>ファイルのカレントセレクションを挿入してプリントすることができます。<エリア>で指定したドキュメントがプリントされます。

<ファイル番号>が0の場合は、標準の「プリントマージ」ダイアログが表示されます。

<メッセージ>が1の場合は、ダイアログボックスを表示します。ユーザはプリントをキャンセルすることができます。プリントをキャンセルした場合は、**DR Error**関数にエラーコード55を返します。また、<メッセージ>が0の場合は、プリントダイアログボックスを表示しません。したがって、プリントをキャンセルすることはできません。

<プリントダイアログ>は、標準の「プリント」ダイアログボックスを表示するかどうかを指定します。<プリントダイアログ>が0の場合は、標準の「プリント」ファイルダイアログボックスは表示されず、プリント処理が即座に始まります。

参照

DR PRINT

参照 (References)

DR INSERT EXPRESSION

DR INSERT EXPRESSION (エリア;スコープ;先頭;最後;値参照{;フォーマット})

引数	タイプ	説明
エリア	倍長整数型の変数	4D Drawエリア
スコープ	倍長整数	-1=全図形、0=選択図形、>0=図形ID
先頭	整数	先頭の文字位置-1
最後	整数	最後の文字位置
値参照	文字列	“ 値参照 ”
フォーマット	文字列	表示フォーマット

説明

DR INSERT EXPRESSIONコマンドは、<エリア>の<スコープ>で指定した文字図形に<値参照>を挿入します。

<スコープ>=-1の場合は、<エリア>で最初の図形に“ 値参照 ”を挿入します。

<スコープ>=0の場合は、選択図形内で最初の図形に“ 値参照 ”を挿入します。

<スコープ>>0の場合は、図形IDで指定した文字図形に“ 値参照 ”を挿入します。対応する図形が存在しない場合は何も実行せずに、**DR Error**関数にエラーコード2を返します。

<スコープ>で指定した図形が文字図形でない場合は、何も実行せずに**DR Error**関数にエラーコード47を返します。

<先頭>と<最後>で“ 値参照 ”を挿入する位置を指定します。<先頭>は置き換える最初の文字位置よりも1だけ小さく、<最後>は置き換える最後の文字の位置です。

<先頭>と<最後>が等しい場合は、文字を置き換えずに<先頭>と<先頭>+1の間に“ 値参照 ”を挿入します。また、<最後>が文字図形内の文字の数よりも大きい場合は、<先頭>から文字図形の最後の文字までを置き換えます。

< 値参照 > は、文字として値を返す4th Dimension値参照のフィールド、変数、4th Dimension関数、ユーザ定義関数（グローバルプロシージャ）、プラグイン関数を“ 値参照 ”にすることができます。次の表は、それぞれの値参照タイプを示したものです。

例	タイプ
[Drawings]Object	フィールド
vCriteria	変数
Current date	4th Dimension関数
GetNum	ユーザ定義関数（グローバルプロシージャ）
DR Count	4D Draw関数
3*"Hello"	ステートメント

オプションの<フォーマット>は“ 値参照 ”の表示フォーマットで、「フォーマット」ダイアログボックスからフォーマットを選択するのと同じです。<フォーマット>は、番号または名前のどちらかで指定します。番号は、「フォーマット」ダイアログボックスのリストに表示される順に付けられています。

<フォーマット>が1桁または2桁の数値文字列の場合は、フィールドに適用するフォーマットをリストから求めることができます。1桁または2桁の文字列でない場合は、フォーマットリストの各フォーマットの文字列と比較し一致するフォーマットを適用します。つまり、“ 19 ”と“ YY:MM:DD ”のどちらでも最初の日付フォーマットを参照することができます。

<フォーマット>がフォーマットリストに存在しない場合は、ユーザ定義の数値フォーマットとして解釈します。また、<フォーマット>が“ フィールド参照 ”に対して不適切な場合は、そのフォーマットは無視します。例えば、数値に日付フォーマットを使用すると、その数値はフォーマットを使用せずに表示されます。

例題

次の例は、新しい文字図形を作成し、4th Dimensionの**Current date**関数の“ 値参照 ”を挿入し、“ YYYY年MM月DD日X曜日 ”の日付フォーマットを使用してフォーマットを設定します。

```
$ID:=DR Draw text (Area;0.5;0.5;3.5;1;"今日の日付は : ")  
DR INSERT EXPRESSION (Area;$ID;32000;32000;"Current date";"YYYY年MM月DD日  
X曜日")
```

参照

DR INSERT FIELD

DR INSERT FIELD

DR INSERT FIELD (エリア;スコープ;先頭;最後;ファイル;フィールド{;フォーマット})

引数	タイプ	説明
エリア	倍長整数型の変数	4D Drawエリア
スコープ	倍長整数	-1=全図形、0=選択図形、>0=図形ID
先頭	整数	先頭の文字位置-1
最後	整数	最後の文字位置
ファイル	整数	ファイル番号
フィールド	整数	フィールド番号
フォーマット	文字列	表示フォーマット

説明

DR INSERT FIELDコマンドは、<エリア>の<スコープ>で指定した文字図形に“フィールド参照”を挿入します。

「スコープ」=-1の場合は、「エリア」の最初の図形に“フィールド参照”を挿入します。
 「スコープ」=0の場合は、選択図形内の最初の図形に“フィールド参照”を挿入します。
 「スコープ」>0の場合は、図形IDで指定した文字図形に“フィールド参照”を挿入します。対応する図形が存在しない場合は何も実行せずに、DR Error関数にエラーコード2を返します。

<スコープ>で指定した図形が文字図形でない場合は何も実行せずに、DR Error関数にエラーコード47を返します。

<先頭>と<最後>で“フィールド参照”を挿入する位置を指定します。<先頭>は置き換える最初の文字位置よりも1だけ小さく、<最後>は置き換える最後の文字の位置です。<先頭>と<最後>が等しい場合は、文字を置き換えずに<先頭>と<先頭>+1の間に“フィールド参照”を挿入します。また、<最後>が文字図形内の文字の数よりも大きい場合は、<先頭>から文字図形の最後の文字までを置き換えます。

<ファイル>と<フィールド>で参照するフィールドを指定します。<ファイル>はファイル番号、<フィールド>はフィールド番号です。ファイル番号とフィールド番号は、作成順に番号が付けられています。

オプションの<フォーマット>は、“フィールド参照”の表示フォーマットで、「フォーマット」ダイアログボックスからフォーマットを選択するのと同じです。<フォーマット>は、番号または名前のどちらかで指定します。番号は、「フォーマット」ダイアログボックスのリストに表示される順に付けられています。

<フォーマット>が1桁または2桁の数値文字列の場合は、フィールドに適用するフォーマットをリストから求めることができます。1桁または2桁の文字列でない場合は、フォーマットリストの各フォーマットの文字列と比較し、一致するフォーマットを適用します。つまり、“19”と“YY:MM:DD”のどちらでも最初の日付フォーマットを参照することができます。

<フォーマット>がフォーマットリストに存在しない場合は、ユーザ定義の数値フォーマットとして解釈します。また、<フォーマット>が“フィールド参照”に対して不適切な場合は、そのフォーマットは無視します。例えば、数値に日付フォーマットを使用すると、その数値はフォーマットを使用せずに表示されます。

例題

次の例は、最初のファイルの最初のフィールドを“フィールド参照”として、図形IDが1の文字図形に挿入し、その図形内の文字を置き換えた後、リスト内の11番目のフォーマットに従ってフォーマットを設定します。

DR INSERT FIELD (Area;1;0;32000;1;1;"11")

4th Dimensionの**Field**関数や**Table**関数を使用すると、フィールド番号やテーブル番号を調べることができます。例えば、上記の例で使用されたフィールドが“[顧客]名前”の場合、コードは次のようになります。

DR INSERT FIELD (Area;1;0;32000;**Table** (^[Customers]);**Field** (^Name);"11")

参照

DR INSERT EXPRESSION

DR Place field

DR Place field (エリア;テーブル番号;フィールド番号;フォーマット;位置) 倍長整数

引数	タイプ	説明
エリア	倍長整数型の変数	4D Drawエリア
テーブル番号	整数	テーブル番号
フィールド番号	整数	フィールド番号
フォーマット	文字列	使用するフォーマット(「フィールド番号」のフィールドがピクチャタイプの場合は使われません)
位置	整数	0=標準、1=中央、2=原点
戻り値	倍長整数	図形ID

説明

DR Place field関数は<テーブル番号>ファイルの「フィールド番号」フィールドから図形を作成し、そのID番号を返します。ピクチャフィールドの場合の図形のタイプはピクチャに、その他の場合はすべてテキストタイプになります。**DR Place field**関数は、エリア内のフィールド位置を選択できることを除けば、「データベース」メニューの「フィールドをペースト」メニューアイテムと同じです。

<位置>が0の場合は、4D Drawエリア内で最後にクリックした位置にフィールドが配置されます。

<位置>が1の場合は、4D Drawエリアの表示領域の中央に配置されます。

<位置>が2の場合は、原点、つまり座標(0,0)の位置に配置されます。

ファイル番号やフィールド番号の設定方法に関する詳細は、**DR INSERT FIELD**コマンドを参照してください。

参照

DR INSERT FIELD

DR SET FORMAT

DR SET FORMAT (エリア;スコープ;フォーマット)

引数	タイプ	説明
エリア	倍長整数型の変数	4D Drawエリア
スコープ	倍長整数	-3=選択文字列、-1=全“参照”、 0=選択図形、>0=図形ID
フォーマット	文字列	“参照”の表示フォーマット

説明

DR SET FORMATコマンドは、フォーマットを<エリア>の<スコープ>で指定した“参照”の表示フォーマットに設定します。

<スコープ>=-3の場合は、選択文字列の“参照”にフォーマットを設定します。

<スコープ>=-1の場合は、すべての“参照”にフォーマットを設定します。

<スコープ>>0の場合は、図形IDで指定した図形内のすべての“参照”にフォーマットを設定します。対応する図形が存在しない場合は、**DR SET FORMAT**コマンドは何も実行せず、**DR Error**関数にエラーコード2を返します。また、「スコープ」に“参照”が含まれていない場合は、**DR SET FORMAT**コマンドは何も実行しません。

<フォーマット>は“参照”の表示フォーマットで、「フォーマット」ダイアログボックスからフォーマットを選択するのと同じです。<フォーマット>は、番号または名前のどちらかで指定します。番号は、「フォーマット」ダイアログボックスのリストに表示される順に付けられています。

<フォーマット>が1桁または2桁の数値文字列の場合は、フィールドに適用するフォーマットをリストから求めることができます。1桁または2桁の文字列でない場合は、フォーマットリストの各フォーマットの文字列と比較し、一致するフォーマットを適用します。

つまり、“19”と“YY:MM:DD”のどちらでも最初の日付フォーマットを参照することができます。<フォーマット>がフォーマットリストに存在しない場合は、ユーザ定義の数値フォーマットとして解釈します。また、<フォーマット>が“フィールド参照”に対して不適切な場合は、そのフォーマットは無視します。例えば、数値に日付フォーマットを使用すると、その数値はフォーマットを使用しないで表示されます。

例題

次の例は、文字図形内の選択された文字列を“ Current date ”に置き換えた後、**DR DO COMMAND**コマンドを使用して“ 値参照 ”に変換し、日付フォーマット“ YYYYY年MM月DD日 X 曜日 ”を適用します。

DR SET TEXT (Area;-3;"Current date")

DR DO COMMAND (Area;7003)

DR SET FORMAT (Area;-3;"YYYY年MM月DD日 X 曜日")

参照

DR SET TEXT

定規 (Rulers)

DR ARRAY BASE TO SCALE

DR ARRAY BASE TO SCALE (エリア;ベース配列;目盛配列)

引数	タイプ	説明
エリア	倍長整数型の変数	4D Drawエリア
ベース配列	数値配列	ベースの値の配列
目盛配列	数値配列	目盛の値の配列

説明

DR ARRAY BASE TO SCALE コマンドは、<エリア>の定規設定に従ってベースの値の配列を目盛の値の配列に変換します。配列はどちらも数値（整数、倍長整数、実数）でなければなりません。また同じ配列を指定してもかまいません。

DR ARRAY BASE TO SCALE コマンドを実行すると、<目盛配列>には<ベース配列>と同数の要素が設定されます。このコマンドは、**DR POLYGON TO ARRAY** コマンド、**DR Array to polygon** 関数、**DR ATTRIBUTE TO ARRAY** コマンド、**DR ARRAY TO ATTRIBUTE** コマンド等を使用する場合に特に便利です。このコマンドを使用すると、各配列要素を個々に変換する必要はありません。

例題

次の例は、選択された多角形の頂点の位置を求め、それらを“目盛の単位”に変換して画面に表示します。2行目では、ベース配列と目盛配列に同じ配列を使用します。

```
DR POLYGON TO ARRAY (Area;0;ArrayH;ArrayV)
DR ARRAY BASE TO SCALE (Area;ArrayH;ArrayH)
DR ARRAY BASE TO SCALE (Area;ArrayV;ArrayV)
```

参照

DR ARRAY SCALE TO BASE、**DR GET RULER**、**DR SETRULER**

DR ARRAY SCALE TO BASE

DR ARRAY SCALE TO BASE (エリア;目盛配列;ベース配列)

引数	タイプ	説明
エリア	倍長整数型の変数	4D Drawエリア
目盛配列	数値配列	目盛の値の配列
ベース配列	数値配列	ベースの値の配列

説明

DR ARRAY SCALE TO BASEコマンドは、<エリア>の定規設定に従って目盛の値の配列をベースの値の配列に変換します。配列はどちらも数値（整数、倍長整数、実数）でなければなりません。このコマンドを使用すると、各配列要素を個々に変換する必要はありません。

DR ARRAY SCALE TO BASEコマンドを実行すると、<ベース配列>には<目盛配列>と同数の要素が設定されます。このコマンドは、**DR POLYGON TO ARRAY**コマンド、**DR Array to polygon**関数、**DR ATTRIBUTE TO ARRAY**コマンド、**DR ARRAY TO ATTRIBUTE**コマンドを使用する場合に特に便利です。

例題

次の例は、**DR ARRAY BASE TO SCALE**コマンドの例で求めた配列を使用します。配列を“ベースの単位”に戻した後、求めた値を使用して新しい多角形を作成します。

DR ARRAY SCALE TO BASE (Area;ArrayH;ArrayH)

DR ARRAY SCALE TO BASE (Area;ArrayV;ArrayV)

\$ID:=**DR Array to polygon** (Area;ArrayH;ArrayV)

参照

DR ARRAY BASE TO SCALE、**DR GET RULER**、**DR SET RULER**

DR Base to scale

DR Base to scale (エリア;値) 実数

引数	タイプ	説明
エリア	倍長整数型の変数	4D Drawエリア
値	実数	“ベースの単位”で表示される値
戻り値	実数	“目盛りの単位”で表示される値

説明

DR Base to scale関数は、<エリア>におけるベース単位の値を“目盛の単位”の値で返します。<値>は“ベースの単位”で指定します。ベースの1単位は、

目盛の値 / ベースの値

の“目盛の単位”に相当します。**DR Base to scale**関数は、“ベースの単位”の値を返すコマンドと併用すると便利です。

例題

次の例は、選択図形の境界を求め、その値を使用して“目盛の単位”で幅と高さを表示します。

```
DR GET BOUNDARY (Area;0;$Left;$Top;$Right;$Low)
vWidth:=DR Base to scale (Area;$Right-$Left)
vHeight:=DR Base to scale (Area;$Low-$Top)
```

参照

DR GET RULER、**DR SET RULER**、**DR Scale to base**

DR GET ORIGIN

DR GET ORIGIN (エリア;横;縦)

引数	タイプ	説明
エリア	倍長整数型の変数	4D Drawエリア
横	実数型の変数	横方向の位置
縦	実数型の変数	縦方向の位置

説明

DR GET ORIGINコマンドは、<エリア>の原点を<横>と<縦>に返します。

<横>と<縦>の値は、“ベースの単位”で返します。“ベースの単位”から“目盛の単位”に変換する場合は、**DR Base to scale**関数を使用します。また、<横>と<縦>はプリントできるエリアの左上端からのオフセット値として返され、正の値は下または右、負の値は上または左の位置を表します。デフォルトの原点の位置はドキュメントのプリントできるエリアの左上端です。

例題

次の例は、ボタンのスクリプトです。このボタンを押すと、“vHorizontal”と“vVertical”のグローバル変数に原点の位置を表示します。

DR GET ORIGIN (Area;vHorizontal;vVertical)

参照

DR SET ORIGIN

DR GET RULER

DR GET RULER (エリア;ベースU;目盛U;ベースV;目盛V;目盛の刻み)

引数	タイプ	説明
エリア	倍長整数型の変数	4D Drawエリア
ベースU	整数変数	ベースの単位 (単位コード)
目盛U	整数変数	目盛の単位
ベースV	実数変数	ベースの値 (単位コード)
目盛V	実数変数	目盛の値
目盛の刻み	整数変数	目盛の刻み

説明

DR GET RULERコマンドは、<エリア>の定規設定情報を、<ベースU>、<目盛U>、<ベースV>、<目盛V>、<目盛の刻み>の各変数に返します。

<ベースU>は<エリア>の“ベースの単位”を表し、インチ、センチメートル、ピクセルが設定されます。また、<目盛U>は<エリア>の“目盛の単位”を表します。“目盛の単位”は、座標パネルにも表示されます。

<目盛U>は、単位を表します。<ベースV>と<目盛V>は、“ベースの単位”と“目盛の単位”の関係を表します。ベースの1単位は、“目盛V / ベースV”の“目盛の単位”に相当します。<目盛の刻み>は、画面に表示される定規の目盛の刻み数を表します。

次の表は、単位コードを一覧表示したものです。

コード	単位
1	インチ
2	フィート
3	ヤード
4	マイル
5	ミリメートル
6	センチメートル
7	デシメートル
8	メートル
9	デカメートル
10	キロメートル
11	ピクセル

例題

次の例は、エリアの定規を示すアラートボックスを表示します。

ARRAY STRING (20;aUnit;11)

aUnit{1}:="Inches"

aUnit{2}:="Feet"

aUnit{3}:="Yards"

aUnit{4}:="Miles"

aUnit{5}:="Millimeters"

aUnit{6}:="Centimeters"

aUnit{7}:="Decimeters"

aUnit{8}:="Meters"

aUnit{9}:="Decameters"

aUnit{10}:="Kilometers"

aUnit{11}:="Pixels"

DR GET RULER (Area;\$BaseU;\$ScaleU;\$BaseV;\$ScaleV;\$Divisions)

\$Temp:="定規"+Char(13)+"1"

\$Temp:=\$Temp+aUnit{\$BaseU}+"="+String (\$ScaleV / \$BaseV)+" "+aUnit{\$ScaleU}

ALERT (\$Temp)

参照

DR SET RULER

DR GET RULER OPTIONS

DR GET RULER OPTIONS (エリア;グリッド;表記;間隔)

引数	タイプ	説明
エリア	倍長整数型の変数	4D Drawエリア
グリッド	整数型の変数	0=標準、1=図形を通して
表記	整数型の変数	0=小数、1=分数
間隔	実数型の変数	グリッド間隔 (“ 目盛の単位 ”)

説明

DR GET RULER OPTIONS コマンドは、<エリア> の定規に関する情報を、<グリッド>、<表記>、<間隔> の各変数に返します。

<グリッド> は、<エリア> のグリッドの取り扱い方を表します。<グリッド> が0の場合は、“なし” 以外のパターンで塗りつぶされた図形上でグリッドが見えなくなります。1の場合は、塗りつぶしパターンに関係なく、図形を通してグリッドを表示します。

<表記> は、座標パネル内の値の表示方法を表します。<表記> が0の場合は、座標に小数表記で表示します。1の場合は、“ベースの単位” がインチの時に分数表記 (分数とフット'インチ') で表示します。

<間隔> は<エリア> のグリッド間隔 (“ 目盛の単位 ”) を表します。“目盛の単位” から “ベースの単位” に変換する場合は、**DR Scale to base**関数を使用します。

例題

次の例は、グリッド間隔を半分にして、図形の配置精度を高めます。

DR GET RULER OPTIONS (Area;\$Grid;\$Notation;\$Lines)

DR SET RULER OPTIONS (Area;-1;-1;\$Lines/2)

参照

DR SET RULER OPTIONS

DR Scale to base

DR Scale to base (エリア;値) 実数

引数	タイプ	説明
エリア	倍長整数型の変数	4D Drawエリア
値	実数	“目盛りの単位”で表示される値
戻り値	実数	“ベースの単位”で表示される値

説明

DR Scale to base関数は、<エリア>における目盛単位の値を“ベースの単位”の値で返します。

<値>は“目盛の単位”で指定します。“目盛の単位”と“ベースの単位”の比は定規設定によって決まります。詳細は、**DR SET RULER**コマンドを参照してください。**DR Scale to base**関数は、引数として“ベースの単位”の値を使用するコマンドと併用すると便利です。

例題

次の例は、**DR Base to scale**関数の例で定義された変数を使用します。その変数の値を取り、それらを“ベースの単位”に戻した後、選択図形のサイズ指定に使用します。

```
$Wide:=DR Scale to base (Area;vWide
```

```
$High:=DR Scale to base (Area;vHige)
```

```
DR SIZE (Area;0;0;2;2;$Wide;$High)
```

参照

DR GET RULER、**DR SET RULER**、**DR Base to scale**

DR SET ORIGIN

DR SET ORIGIN (エリア;横;縦;モード)

引数	タイプ	説明
エリア	倍長整数型の変数	4D Drawエリア
横	実数	横方向の位置
縦	実数	縦方向の位置
モード	整数	0=絶対、1=相対

説明

DR SET ORIGINコマンドは、<エリア>の原点を<横>、<縦>、<モード>に従って、設定します。

<横>と<縦>は“ベースの単位”で指定します。“目盛の単位”から“ベースの単位”に変換する場合は、**DR Scale to base**関数を使用します。<横>と<縦>は原点からのオフセット値（相対）またはドキュメントのプリントできるエリアの左上端からのオフセット値（絶対）として指定できます。

<モード>が0の場合は、絶対位置になります。

<モード>が1の場合は、相対位置になります。正の値は下または右、負の値は上または左の方向を表します。デフォルトの原点の位置はドキュメントのプリントできるエリアの左上端です。4D Drawのほとんどのコマンドがこの原点の位置を使用します。

例題

次の例は、原点をドキュメントの左上端にリセットします。

DR SET ORIGIN (Area;0;0;0)

参照

DR GET ORIGIN

DR SET RULER

DR SET RULER (エリア;ベースU;目盛U;ベースV;目盛V;目盛の刻み)

引数	タイプ	説明
エリア	倍長整数型の変数	4D Drawエリア
ベースU	整数	ベースの単位
目盛U	整数	目盛の単位
ベースV	実数	ベースの値
目盛V	実数	目盛の値
目盛の刻み	整数	目盛の刻み

説明

DR SET RULER コマンドは、<エリア>のベースの定規設定を、<ベースU>、<目盛の刻み>、<目盛U>、<ベースV>、<目盛V>の各引数に従って変更します。

<ベースU>は<エリア>の“ベースの単位”を指定し、インチ、センチメートル、ピクセルが設定できます。

<目盛U>は、<エリア>の“目盛の単位”を指定します。“目盛の単位”は、座標パネルに表示されます。また、<目盛U>には単位を設定できます。

<ベースV>と<目盛V>は、“ベースの単位”と“目盛の単位”の関係を指定し、ベースの1単位は“目盛V/ベースV”の“目盛の単位”に相当します。<目盛の刻み>は、画面に表示される定規の目盛の刻み数を指定します。

単位コードは、**DR GET RULER**コマンドの説明のところを参照してください。

例題

次の例は、1インチ=1フィート、目盛の刻みが8になるようにエリアの定規設定を変更します。

DR SET RULER (Area;1;2;1;1;8)

参照

DR GET RULER

DR SET RULER OPTIONS

DR SET RULER OPTIONS (エリア;グリッド;表記;間隔)

引数	タイプ	説明
エリア	倍長整数型の変数	4D Drawエリア
グリッド	整数	-1=変更しない、0=標準、 1=図形を通して
表記	整数	-1=変更しない、0=10進、1=12進
間隔	実数	グリッド間隔 (“ 目盛の単位 ”)、 -1=変更しない

説明

DR SET RULER OPTIONS コマンドは、<エリア>の定規の各オプションを<グリッド>、<表記>、<間隔>に従って変更します。

<グリッド>は、<エリア>のグリッドの取り扱い方を指定します。<グリッド>が0の場合は、“なし”以外のパターンで塗りつぶされた図形上でグリッドが見えなくなります。1の場合は、塗りつぶしパターンに関係なく、図形を通してグリッドを表示します。-1の場合は、グリッドを変更しません。

<表記>は、座標パネル内の値の表示方法を指定します。<表記>が0の場合は、座標に小数表記で表示します。1の場合は、“ベースの単位”がインチの時に分数表記（分数とフィート'インチ”）で表示します。-1の場合は、表記を変更しません。

<間隔>は、<エリア>のグリッド間隔 (“ 目盛の単位 ”)を指定します。“目盛の単位”から“ベースの単位”に変換する場合は、**DR Scale to base**関数を使用します。<間隔>が-1の場合は、グリッドを変更しません。

DR GET RULER OPTIONS コマンドの例を参照してください。

参照

DR GET RULER OPTIONS

属性設定 (Set Attributes)

DR SET ARC SPECS

DR SET ARC SPECS (エリア;スコープ;開始角;長さ)

引数	タイプ	説明
エリア	倍長整数型の変数	4D Drawエリア
スコープ	倍長整数	-1=全図形、0=選択図形、>0=図形ID
開始角	整数	開始点の角度 (単位 : °)
長さ	整数	弧の長さ (単位 : °)

説明

DR SET ARC SPECSコマンドは、<エリア>の<スコープ>で指定した弧を変形します。

<スコープ>=-1の場合は、<エリア>の最初の図形に反映します。

<スコープ>=0の場合は、選択図形内の最初の図形に反映します。

<スコープ>>0の場合は、図形IDで指定した弧に反映します。対応する図形が存在しない場合は何も実行せずに、**DR Error**関数にエラーコード2を返します。

<スコープ>で指定した図形が弧でない場合は何も実行せずに、**DR Error**関数にエラーコード47を返します。

<開始角>は、弧の開始角度 (単位 : °) です。この<開始角>が弧を描き始める始点で、0°から反時計回りに移動します。

<長さ>は、弧の長さ (単位 : °) です。弧の終点は“開始点+長さ%360”で、最大値は359°です。

DR Draw arc関数の図を参照してください。

例題

次の例は、エリアの1～10番目の図形を変形します。10個の図形は、すべて弧です。それぞれの弧は、異なる開始角で18°の弧長になります。

```
For ($i;0;9)
  DR SET ARC SPECS (Area;$i+1;$i*18;18)
End for
```

参照

DR Draw arc、DR Get ID、DR GET ARC SPECS

DR SET ATTRIBUTE LOCK

DR SET ATTRIBUTE LOCK (エリア;スコープ;属性コード;処理)

引数	タイプ	説明
エリア	倍長整数型の変数	4D Drawエリア
スコープ	倍長整数	-1=全図形、0=選択図形、>0=図形ID
属性コード	整数	ロックする属性のコード(0から28)
処理	整数	0=ロック解除、1=ロック、2=切り替える

説明

DR SET ATTRIBUTE LOCKコマンドは、<スコープ>内の図形の<属性コード>の属性のロック/ロック解除を行います。

引数<属性コード>には0から28までの値のいずれかを指定します。属性コードに関する詳細は、「付録D 図形コード」を参照してください。

参照

なし

DR SET CORNER ROUNDING

DR SET CORNER ROUNDING (エリア;スコープ;角の丸み)

引数	タイプ	説明
エリア	倍長整数型の変数	4D Drawエリア
スコープ	倍長整数	-2=デフォルト、-1=全図形 0=選択図形、>0=図形ID
角の丸み	実数	角の丸みの大きさ

説明

DR SET CORNER ROUNDINGコマンドは、**<エリア>**の**<スコープ>**で指定した図形の角の丸みを変更します。角の丸みは“ベースの単位”で指定します。“目盛の単位”から“ベースの単位”に変換する場合は、**DR Scale to base**関数を使用します。

<スコープ>=-2の場合は、角の丸い矩形のデフォルトの角の丸みを設定します。

<スコープ>=-1の場合は、全図形の角の丸みを設定します。

<スコープ>=0の場合は、選択図形の角の丸みを設定します。

<スコープ>>0の場合は、図形IDで指定した図形の角の丸みを設定します。対応する図形が存在しない場合は何も実行せずに、**DR Error**関数にエラーコード2を返します。

例題

DR Get corner rounding関数の例題を参照してください。

参照

DR Get ID、**DR Get corner rounding**

DR SET ENDMARKS

DR SET ENDMARKS (エリア;スコープ;タイプ;端点)

引数	タイプ	説明
エリア	倍長整数型の変数	4D Drawエリア
スコープ	倍長整数	-2=デフォルト、-1=全図形、 0=選択図形、>0=図形ID
タイプ	整数	1=矢印、2=バーマーク
端点	整数	0=線のみ、1=始点、2=終点、3=両端

説明

DR SET ENDMARKSコマンドは、<エリア>の<スコープ>で指定した1本または複数の線の終点マークを変更します。

<スコープ>=-2の場合は、デフォルトの終点マークを設定します。

<スコープ>=-1の場合は、すべての線の終点マークを設定します。

<スコープ>=0の場合は、選択された線の終点マークを設定します。

<スコープ>>0の場合は、図形IDで指定した線の終点マークを設定します。対応する図形が存在しない場合は何も実行せずに、**DR Error**関数にエラーコード2を返します。

<タイプ>は、線の終点マークの種類を指定します。たとえ終点マークが表示されなくても、各線は、必ず1つの終点マークを持ちます。<タイプ>が1の場合は、線は矢印の終点マークです。2の場合は、バーマークの終点マークです。

<端点>は、線のどちらの端点が終点マークを持つかを指定します。<端点>が0の場合は線だけです。1の場合は線の始点、2の場合は線の終点、3の場合は線の両端に終点マークをもちます。

例題

次の例題は、選択された線の始点を矢印にします。

DR SET ENDMARKS (Area;0;1;1)

参照

DR GET ENDMARKS

DR SET FILL ATTRIBUTES

DR SET FILL ATTRIBUTES (エリア;スコープ;パターン;カラー)

引数	タイプ	説明
エリア	倍長整数型の変数	4D Drawエリア
スコープ	倍長整数	-2=デフォルト、-1=全図形、 0=選択図形、>0=図形ID
パターン	整数	パターン番号
カラー	倍長整数	カラーの設定値

説明

DR SET FILL ATTRIBUTESコマンドは、<エリア>の<スコープ>で指定した図形の塗りつぶし属性を変更します。

<スコープ>=-2の場合は、デフォルトの塗りつぶし属性を設定します。

<スコープ>=-1の場合は、全図形の塗りつぶし属性を設定します。

<スコープ>=0の場合は、選択図形の塗りつぶし属性を設定します。

<スコープ>>0の場合は、図形IDで指定した図形の属性を設定します。対応する図形が存在しない場合は何も実行せずに、**DR Error**関数にエラーコード2を返します。

<パターン>は、パレット内のパターン番号です。パレットの左から右、上から下の順序で1~36の値が付けられます。

<カラー>は、図形のカラーを表す倍長整数です。この数値は、**DR RGB to color**関数と**DR Index to color**関数で求めることができます。

<パターン>と<カラー>に-1を指定すると、その属性を変更しません。

例題

次の例題は、エリアの全図形の塗りつぶし属性を設定します。図形パターンを実線に変更します。カラーは、変更しません。

DR SET FILL ATTRIBUTES (Area;0;3;-1)

参照

DR GET FILL ATTRIBUTES

DR SET HANDLE STATE

DR SET HANDLE STATE (エリア;スコープ;処理)

引数	タイプ	説明
エリア	倍長整数型の変数	4D Drawエリア
スコープ	倍長整数	-1=全図形、0=選択図形、>0=図形ID
処理	整数	1=表示、0=非表示

説明

DR SET HANDLE STATEコマンドは、<スコープ>の図形のハンドルを表示するかどうか指定します。ハンドルは、図形が選択された場合に図形の周囲に表示される黒色の矩形で、図形のサイズ変更に使います。

<処理>が1の場合は、図形のハンドルが表示され、0の場合は表示されません。

ハンドルの状態は、データベース終了後も記録されます。

参照

なし

DR SET HIGHLIGHT

DR SET HIGHLIGHT (エリア;スコープ;先頭;最後)

引数	タイプ	説明
エリア	倍長整数型の変数	4D Drawエリア
スコープ	倍長整数	-1=全図形、0=選択図形、>0=図形ID
先頭	整数	先頭の文字位置-1
最後	整数	最後の文字位置

説明

DR SET HIGHLIGHTコマンドは、<エリア>の<スコープ>で指定した文字図形内の文字を選択します。

<スコープ>=-1の場合は、<エリア>の最初の図形の文字列を選択します。

<スコープ>=0の場合は、選択図形内の最初に図形の文字列を選択します。

<スコープ>>0の場合は、図形IDで指定した図形内の文字列を選択します。対応する図形が存在しない場合は何も実行せずに、**DR Error**関数にエラーコード2を返します。

<スコープ>で指定した図形が<エリア>で選択されている唯一の図形です。

<スコープ>で指定した図形が文字図形でない場合は、何も実行せずに、**DR Error**関数にエラーコード47を返します。

<先頭>と<最後>で選択する文字列の範囲を指定します。<先頭>は選択する最初の文字位置よりも1だけ小さく、最後は選択する最後の文字位置です。<先頭>と<最後>が等しい場合は、文字を選択しません。挿入ポイントは、<先頭>と<先頭>+1の間になります。また、<最後>が文字図形の文字の数よりも大きい場合は、文字図形の最後の文字までを選択します。

DR SET HIGHLIGHTコマンドは、“参照”の一部だけを選択することはできません。“参照”の一部を選択すると“参照”全体が選択されます。

DR SET HIGHLIGHTコマンドは、<エリア>が100% (実サイズ) で表示されている場合にだけ有効です。描画が他の倍率で表示されている場合は、何も実行せずに、**DR Error**関数にエラーコード48を返します。

例題

次の例題は、選択された文字図形から“4th Dimension”という文字列を探し、それを選択してからボールドにします。

```
$Temp:=DR Get text (Area;0)
$Search:=Position ("4th Dimension";$Temp)
If ($Search#0)
    DR SET HIGHLIGHT (Area;0;$Search-1;$Search+12)
    DR SET TEXT ATTRIBUTES (Area;-3;-1;-1;1;-1;-1)
End if
```

参照

DR Get text、**DR SET TEXT**、**DR SET TEXT ATTRIBUTES**、**DR GET TEXT ATTRIBUTES**

DR SET LINE ATTRIBUTES

DR SET LINE ATTRIBUTES (エリア;スコープ;パターン;カラー;幅)

引数	タイプ	説明
エリア	倍長整数型の変数	4D Drawエリア
スコープ	倍長整数	-2=デフォルト、-1=全図形、 0=選択図形、>0=図形ID
パターン	整数	パターン番号、-1=変更しない
カラー	倍長整数	カラーの設定値、-1=変更しない
幅	実数	線幅(単位:ピクセル)、-1=変更しない

説明

DR SET LINE ATTRIBUTESコマンドは、<エリア>の<スコープ>で指定した図形の線属性を変更します。

<スコープ>=-2の場合は、デフォルトの線属性を設定します。

<スコープ>=-1の場合は、全図形の線属性を設定します。

<スコープ>=0の場合は、選択図形の線属性を設定します。

<スコープ>>0の場合は、図形IDで指定した図形の属性を設定します。対応する図形が存在しない場合は何も実行せずに、**DR Error**関数にエラーコード2を返します。

<パターン>は、パレット内のパターン番号です。パレットの左から右、上から下の順序で1~36の値が付けられます。

<カラー>は、図形のカラーを表す倍長整数です。この数値は、**DR RGB to color**関数と**DR Index to color**関数で求めることができます。

<幅>は、線幅(線の太さ)を表す実数です(単位:ピクセル)。

<パターン>、<カラー>、<幅>の各引数に-1を指定すると、その属性を変更しません。

例題

DR GET LINE ATTRIBUTESコマンドの例題を参照してください。

参照

DR GET LINE ATTRIBUTES

DR SET LINE SPECS

DR SET LINE SPECS (エリア;スコープ;開始H;開始V;終了H;終了V)

引数	タイプ	説明
エリア	倍長整数型の変数	4D Drawエリア
スコープ	倍長整数	-1=全図形、0=選択図形、>0=図形ID
開始H	実数	横方向の開始位置
開始V	実数	縦方向の開始位置
終了H	実数	横方向の終了位置
終了V	実数	縦方向の終了位置

説明

DR SET LINE SPECSコマンドは、<エリア>の<スコープ>で指定した線の端点をリセットします。

<スコープ>=-1の場合は、<エリア>の最初の図形の端点に反映します。

<スコープ>=0の場合は、選択図形内の最初の図形の端点に反映します。

<スコープ>>0の場合は、図形IDで指定した線の端点に反映します。対応する図形が存在しない場合は何も実行せずに、DR Error関数にエラーコード2を返します。

<スコープ>で指定した図形が線でない場合は何も実行せずに、DR Error関数にエラーコード47を返します。

<開始H>、<開始V>、<終了H>、<終了V>に従って線を再配置します。4つの位置は、すべて“ベースの単位”で表します。“目盛の単位”から“ベースの単位”に変換する場合は、DR Scale to base関数を使用します。また、<開始H>、<開始V>、<終了H>、<終了V>は、原点からのオフセット値として指定します。

正の値は原点の下または右、負の値は原点の上または左の位置を表します。

例題

DR GET LINE SPECSコマンドの例題を参照してください。

参照

DR SET ENDMARKS、DR SET LINE ATTRIBUTES、DR GET LINE SPECS

DR SET NAME

DR SET NAME (エリア;スコープ;名前)

引数	タイプ	説明
エリア	倍長整数型の変数	4D Drawエリア
スコープ	倍長整数	-2=デフォルト、-1=全図形、 0=選択図形、>0=図形ID
名前	文字列	設定する名前

説明

DR SET NAMEコマンドは、<エリア>の<スコープ>で指定した図形に<名前>を設定します。

名前は図形に付属した文字列で、必ずしも固有のものではありません。名前の長さは最大31文字（バイト）です。

また、名前は「属性設定」ダイアログボックスを使用しても設定することができます。

<スコープ>=-2の場合は、デフォルトの名前を設定します。

<スコープ>=-1の場合は、全図形に名前を設定します。

<スコープ>=0の場合は、選択図形に名前を設定します。

<スコープ>>0の場合は、図形IDで指定した図形に名前を設定します。対応する図形が存在しない場合は何も実行せずに、**DR Error**関数にエラーコード2を返します。

例題

次の例は、エリアの全図形の名前を空の文字列に設定します。

```
DR SET NAME (Area;-1;"")
```

参照

DR Get ID、**DR Get name**

DR SET POLYGON VERTEX

DR SET POLYGON VERTEX (エリア;スコープ;頂点番号;頂点H;頂点V)

引数	タイプ	説明
エリア	倍長整数型の変数	4D Drawエリア
スコープ	倍長整数	-1=全図形、0=選択図形、>0=図形ID
頂点番号	整数	編集する頂点の番号
頂点H	実数	横方向の位置
頂点V	実数	縦方向の位置

説明

DR SET POLYGON VERTEXコマンドは、<エリア>の<スコープ>で指定した多角形の<頂点番号>で指定した頂点を修正します。

<スコープ>=-1の場合は、「エリア」の最初の図形に反映します。

<スコープ>=0の場合は、選択図形内の最初の図形に反映します。

<スコープ>>0の場合は、図形IDで指定した多角形に反映します。対応する図形が存在しない場合は何も実行せずに、**DR Error**関数にエラーコード2を返します。

<スコープ>で指定した図形が多角形でない場合は何も実行せずに、**DR Error**関数にエラーコード47を返します。

<頂点番号>は、多角形内の頂点の番号で、作成順に番号が付けられています。<頂点番号>が多角形の頂点の数よりも大きい場合は何も実行せずに、**DR Error**関数にエラーコード49を返します。

<頂点H>と<頂点V>は、頂点の新しい位置で、どちらも“ベースの単位”で表します。“目盛の単位”から“ベースの単位”に変換する場合は、**DR Scale to base**関数を使用します。また、<頂点H>と<頂点V>は、原点からのオフセット値として指定します。正の値は原点の下または右、負の値は原点の上または左の位置を表します。

例題

次の例は、ID番号が1の多角形の最初の頂点の位置を求め、その頂点を下と右にそれぞれ1単位ずつ移動します。

DR GET POLYGON VERTEX (Area;1;1;\$VertexH;\$VertexV)

DR SET POLYGON VERTEX (Area;1;1;\$VertexH+1;\$VertexV+1)

参照

DR GET POLYGON VERTEX

DR SET REFNUM

DR SET REFNUM (エリア;スコープ;参照番号)

引数	タイプ	説明
エリア	倍長整数型の変数	4D Drawエリア
スコープ	倍長整数	-2=デフォルト、-1=全図形、 0=選択図形、>0=図形ID
参照番号	倍長整数	参照番号

説明

DR SET REFNUMコマンドは、<エリア>の<スコープ>で指定した図形の参照番号に<参照番号>を設定します。

<スコープ>=-2の場合は、デフォルトの参照番号を設定します。

<スコープ>=-1の場合は、全図形に参照番号を設定します。

<スコープ>=0の場合は、選択図形に参照番号を設定します。

<スコープ>>0の場合は、図形IDで指定した図形に参照番号を設定します。対応する図形が存在しない場合は何も実行せずに、**DR Error**関数にエラーコード2を返します。

参照番号は図形に付属した倍長整数で、メソッド以外では使用できません。参照番号のデフォルト値は0です。参照番号は、必ずしも固有のものではありません。

例題

次の例は、選択図形の参照番号をグローバル変数“vNum”の値に変更します。

```
DR SET REFNUM (Area;0;vNum)
```

参照

DR Get ID、**DR Get refnum**

DR SET TEXT

DR SET TEXT (エリア;スコープ;テキスト)

引数	タイプ	説明
エリア	倍長整数型の変数	4D Drawエリア
スコープ	倍長整数	-3=選択文字列、-1=全図形、 0=選択図形、>0=図形ID
テキスト	テキスト	設定するテキスト

説明

DR SET TEXTコマンドは、<エリア>の<スコープ>で指定した文字図形を<テキスト>で置き換えます。

<スコープ>=-3の場合は、選択文字列を<テキスト>に置き換えます。挿入ポイントが文字間にある場合は、挿入ポイントの位置に文字を挿入します。

<スコープ>=-1の場合は、<エリア>の最初の図形のテキストを置き換えます。

<スコープ>=0の場合は、選択図形内の最初の図形のテキストを置き換えます。

<スコープ>>0の場合は、図形IDで指定した文字図形のテキストを置き換えます。対応する図形が存在しない場合は何も実行せずに、**DR Error**関数にエラーコード2を返します。

<スコープ>で指定した図形が文字図形でない場合は何も実行せずに、**DR Error**関数にエラーコード47を返します。

例題

次の例は、選択された文字図形のテキストをグローバル変数“v文字”のテキストに置き換えます。

DR SET TEXT (Area;0;vText)

参照

DR Get text

DR SET TEXT ATTRIBUTES

DR SET TEXT ATTRIBUTES (エリア;スコープ;フォント;サイズ;書体;行揃え{;フレーム;フロー})

引数	タイプ	説明
エリア	倍長整数型の変数	4D Drawエリア
スコープ	倍長整数	-3=選択文字列、-2=デフォルト、 -1=全図形、0=選択図形、>0=図形ID
フォント	整数	フォント番号
サイズ	整数	フォントサイズ(単位:ポイント) (-1=変更しない)
書体	整数	フォントの書体(-1=変更しない)
行揃え	整数	フォントの桁揃え(-1=変更しない)
フレーム	整数型の変数	0=可変、1=固定
フロー	整数型の変数	0=ダウン、1=アップ

説明

DR SET TEXT ATTRIBUTESコマンドは、<エリア>の<スコープ>で指定した文字または図形の文字属性を変更します。

<スコープ>=-3の場合は、選択文字列の文字属性を設定します。

<スコープ>=-2の場合は、デフォルトの文字属性を設定します。

<スコープ>=-1の場合は、全図形の文字属性を設定します。

<スコープ>=0の場合は、選択図形の文字属性を設定します。

<スコープ>>0の場合は、図形IDで指定した文字図形の文字属性を設定します。対応する図形が存在しない場合は何も実行せずに、**DR Error**関数にエラーコード2を返します。

<スコープ>に設定できる値は、次の通りです。

スコープ	対象
-3	選択文字列の文字属性
-2	デフォルトの文字属性
-1	全図形の文字属性
0	選択図形の文字属性
>0	特定の図形の文字属性

<フォント>は、システムのフォントIDです。フォントIDは、**DR Font number**関数にフォント名を指定して求めます。

<サイズ>は、選択文字列または文字図形のサイズ(単位:ポイント)です。

<書体> は、次に示す複数の書体番号を加算した値です。

値	書体
0	標準
1	ボールド
2	イタリック
4	アンダーライン
8	アウトライン
16	シャドウ

<行揃え> は、文字ブロック内の文字の整列で、設定できる値は次の通りです。

値	行揃え
0	左揃え
1	中揃え
2	右揃え

オプションの<フレーム> は、文字図形が可変フレームか固定フレームかを指定します。<フレーム> が0、または指定しない場合、図形の高さは可変になります。図形の高さは、文字の量と文字図形の幅によって決まります。<フレーム> が1の場合、図形の高さは固定で、文字図形内に納まりきれない文字は、トランケートします。また、<フレーム> を省略した場合は、<フロー> も省略しなければなりません。

オプションの<フロー> は、改行時に図形内に文字が流れ込む方向を指定します。フローが0、または指定されない場合は、改行時に文字がページに対して下向きに流れ込みます。<フロー> が1の場合は、改行時に文字がページに対して上向きに流れ込みます。

<フレーム> と<フロー> は、“参照” を含む文字図形を使用する場合、特に便利です。プリント中に、文字列の長さに応じて文字図形のサイズと流れ込む方向を制御できます。

<フォント>、<サイズ>、<書体>、<行揃え>、<フレーム>、<フロー> の各引数に-1を指定すると、その属性を変更しません。

例題

次の例は、エリアのすべての文字図形の文字属性を設定します。すべての文字を12ポイント、標準、左揃えにします。フォントは、変更しません。

DR SET TEXT ATTRIBUTES (Area;-1;-1;12;0;0;-1;-1)

参照

DR GET TEXT ATTRIBUTES

ユーティリティ (Utilities)

DR Calculate area

DR Calculate area (エリア;スコープ) 実数

引数	タイプ	説明
エリア	倍長整数型の変数	4D Drawエリア
スコープ	倍長整数	-1=全図形、0=選択図形、>0=図形ID
戻り値	実数	図形の面積

説明

DR Calculate area関数は、<エリア>の<スコープ>で指定した図形の面積を返します。

<スコープ>=-1の場合は、全図形の面積を返します。

<スコープ>=0の場合は、選択図形の面積を返します。

<スコープ>>0の場合は、図形IDで指定した図形の面積を返します。

対応する図形が存在しない場合は、**DR Calculate area**に-3を返し、**DR Error**関数にエラーコード2を返します。

例題

次の例は、エリアの全図形の面積を求めます。

```
$SurfArea:=DR Calculate area (Area;-1)
```

参照

DR Get ID、**DR Calculate perimeter**

DR Calculate perimeter

DR Calculate perimeter (エリア;スコープ) 実数

引数	タイプ	説明
エリア	倍長整数型の変数	4D Drawエリア
スコープ	倍長整数	-1=全図形、0=選択図形、>0=図形ID
戻り値	実数	スコープで指定した図形の周辺の長さ

説明

DR Calculate perimeter関数は、<スコープ>で指定した図形の周辺の長さを返します。

<スコープ>=-1の場合は、全図形の周囲の長さを返します。

<スコープ>=0の場合は、選択図形の周囲の長さを返します。

<スコープ>>0の場合は、図形IDで指定した図形の周囲の長さを返します。

対応する図形が存在しない場合は、“-32000”を返し、**DR Error**関数にエラーコード2を返します。

例題

次の例は、エリアの選択図形の周囲の長さを求めます。

\$周囲:=**DR Calculate perimeter** (Area;0)

参照

DR Get ID、**DR Calculate area**

DR Clipboard to picture

DR Clipboard to picture ピクチャ

引数	タイプ	説明
		このコマンドは、引数はありません。
戻り値	ピクチャ	4th Dimensionのピクチャ

説明

DR Clipboard to picture関数は、クリップボードの内容を返します。また、このピクチャは4th Dimensionの標準のピクチャとして使用することができます。クリップボードにピクチャが含まれていない場合は、空のピクチャを返し、**DR Error**関数にエラーコード14を返します。

例題

次の例は、クリップボードの内容をピクチャフィールドに格納します。

```
[Drawing]Object=DR Clipboard to picture
```

参照

DR PICTURE TO CLIPBOARD

DR Color to index

DR Color to index (カラー) 整数

引数	タイプ	説明
カラー	倍長整数	カラーの設定値
戻り値	整数	4Dのカラーパレットでもっとも近いカラーのインデックス

説明

DR Color to index関数は、4th Dimensionのカラーパレット上でカラーに最も近いカラーのインデックスを返します。

例えば、<カラー>に特定の値を指定した場合、4th Dimensionのカラーパレットで最も近いカラーのインデックスを返します。4th Dimensionのカラーパレット上のカラーには1～256の番号が付いています。

例題

次の例は、赤に最も近いカラーパレットのインデックスを変数“Color”に格納します。

```
Color:=DR Color to index (DR RGB to color (56683;2242;1698))
```

参照

DR Index to color

DR COLOR TO RGB

DR COLOR TO RGB (カラー;赤;緑;青)

引数	タイプ	説明
カラー	整数	カラー
赤	倍長整数型の変数	赤の値
緑	倍長整数型の変数	緑の値
青	倍長整数型の変数	青の値

説明

DR COLOR TO RGB コマンドは、`<カラー>` で指定されたカラーを3つの成分に分解し、`<赤>`、`<緑>`、`<青>` に返します。

`<カラー>` は、4D Drawで使用する内部番号で、**DR RGB to color**関数、**DR Index to color**関数、**DR GET FILL ATTRIBUTES**コマンド、**DR GET LINE ATTRIBUTES**コマンドから求めることができます。

例題

次の例は、選択図形の塗りつぶしカラーを求め、そのカラーを3つの成分に分解します。

```
DR GET FILL ATTRIBUTES (Area;0;$Pattern;$Color)
```

```
DR COLOR TO RGB ($Color;$Red;$Green;$Blue)
```

参照

DR RGB to color

DR Font name

DR Font name (フォントID) 文字列

引数	タイプ	説明
フォントID	整数	フォントのID番号
戻り値	文字列	フォント名

説明

DR Font name関数は、<フォントID>に対応するフォント名を文字列として返します。フォントIDは、**DR Font number**関数から返される値です。

<フォントID>が存在しない場合は、空の文字列を返します。

例題

次の例は、フォントID3のフォントを変数“vName”に格納します。

```
vName:=DR Font name (3)
```

参照

DR Font number

DR Font number

DR Font number (フォント名) 整数

引数	タイプ	説明
フォント名	テキスト	フォントの名前
戻り値	整数	フォントID

説明

DR Font number関数は、<フォント名>に対応するフォントIDを整数として返します。フォントIDは、**DR SET TEXT ATTRIBUTES**コマンドで使用します。

例題

次の例は、エリアの選択図形のフォントを“Times”に変更します。

```
$FontID:=DR Font number ("Times")  
DR SET TEXT ATTRIBUTES (Area;0;$FontID;-1;-1;-1;-1)
```

参照

DR Font name

DR Index to color

DR Index to color (インデックス) 倍長整数

引数	タイプ	説明
インデックス	整数	パレットの指標
戻り値	倍長整数	パレットのインデックスに対応する カラー

説明

DR Index to color関数は、<インデックス>で指定したカラーパレットのインデックスに対応するカラーを返します。

<インデックス>は、4th Dimensionのカラーパレット上のカラーを指定する整数です。

DR Index to color関数を使用すれば、カラー成分（赤、緑、青）がわからなくてもカラーを求めることができます。4th Dimensionのカラーパレット上のカラーには、1～256の番号が付いています。

例題

次の例は、エリアの選択図形のカラーにシアンを設定します。

DR SET FILL ATTRIBUTES (Area;0;3;**DR Index to color** (8))

参照

DR Color to index

DR PICTURE TO CLIPBOARD

DR PICTURE TO CLIPBOARD (ピクチャ)

引数	タイプ	説明
ピクチャ	ピクチャ	ピクチャ

説明

DR PICTURE TO CLIPBOARDコマンドは、<ピクチャ>をクリップボードにコピーします。

クリップボード上にコピーしたピクチャは、任意の場所にペーストすることができます。

例題

次の例は、ピクチャフィールド “[Drawing]Object ” の内容をクリップボードにコピーします。

DR PICTURE TO CLIPBOARD ([Drawing]Object)

参照

DR Clipboard to picture

DR RGB to color

DR RGB to color (赤;緑;青) 倍長整数

引数	タイプ	説明
赤	倍長整数	赤の値 Macintoshの場合: (0 ~ 65535) Windowsの場合: (0 ~ 255)
緑	倍長整数	緑の値 Macintoshの場合: (0 ~ 65535) Windowsの場合: (0 ~ 255)
青	倍長整数	青の値 Macintoshの場合: (0 ~ 65535) Windowsの場合: (0 ~ 255)
戻り値	倍長整数	カラー成分に対応する数値

説明

DR RGB to color関数は、4D Drawでカラー制御に使用する数値を返します。**DR RGB to color**関数は、<赤>、<緑>、<青>で指定した3つのカラー成分に対応する数値を返します。これらの引数は、使用しているOSの標準カラー調整で使用する値と同じです。

一般的なカラーで使われる赤、緑、青の値は次の通りです。

カラー	Macintosh			Windows		
	赤	緑	青	赤	緑	青
赤	56576	2048	1536	221	8	6
緑	0	32768	4352	0	128	17
青	0	0	54272	0	0	212
シアン	577	43776	59904	2	171	234
マゼンタ	64512	62208	1280	252	243	5
黄	61952	2948	33792	242	8	132

例題

次の例は、エリアの選択図形を赤に設定します。

```
$Color:=DR RGB to color (56683;2242;1698)  
DR SET FILL ATTRIBUTES (Area;0;3;$Color)
```

参照

DR COLOR TO RGB

付録A 属性コード

塗りつぶしパターンや塗りつぶしカラー等の図形の属性は、いくつかのコマンドで数値コードを使用して指定または参照します。この付録は、**DR ATTRIBUTE TO ARRAY**コマンドや**DR ARRAY TO ATTRIBUTE**コマンドで属性をフィールドや配列に格納する、または図形を修正する場合に参照してください。

「推奨タイプ」が最良の選択肢ですが、「その他のタイプ」も状況に応じて使用できます。しかし、データの一部が失われる場合がありますのでご注意ください。

コード	属性	推奨タイプ	その他のタイプ	制約
0	ID	倍長整数	整数、実数	修正不可、検索不可
1	図形タイプ	整数	倍長整数、実数	修正不可
2	参照番号	倍長整数	整数、実数	
3	名前	文字列	テキスト	31文字 (バイト)
4	ロックコード	倍長整数	整数、実数	検索不可
5	幅	実数	整数、倍長整数	検索不可
6	高さ	実数	整数、倍長整数	検索不可
7	左の境界	実数	整数、倍長整数	検索不可
8	右の境界	実数	整数、倍長整数	検索不可
9	上の境界	実数	整数、倍長整数	検索不可
10	下の境界	実数	整数、倍長整数	検索不可
11	回転	整数	倍長整数、実数	
12	塗りつぶしパターン	整数	倍長整数、実数	
13	塗りつぶしカラー	倍長整数	整数、実数	
14	線パターン	整数	倍長整数、実数	
15	線カラー	倍長整数	整数、実数	

16	線幅	実数	整数、倍長整数	
17	終点マークの種類	整数	倍長整数、実数	検索不可
18	終点マークの端点	整数	倍長整数、実数	検索不可
19	文字のフォント	整数	倍長整数、実数	
20	文字のサイズ	整数	倍長整数、実数	
21	文字の書体	整数	倍長整数、実数	
22	文字の行揃え	整数	倍長整数、実数	
23	文字図形内の文字	テキスト	文字列225バイト	検索不可
24	角の丸み	実数	整数、倍長整数	
25	エリア	実数	整数、倍長整数	修正不可、検索不可
26	周囲/線の長さ	実数	整数、倍長整数	修正不可、検索不可
27	オブジェクト	整数	整数	0=見えない 1=見える
28	オブジェクト	整数	整数	0=背景外 1=背景内

付録B エラーコード

次の表は、4D Drawエラーメッセージ用のエラーコードです。

エラー	メッセージ
1	4D Drawエリアが正しくありません。
2	図形IDが正しくありません。
3	フォントが正しくありません。
4	選択操作が正しくありません。
5	図形インデックスが正しくありません。
6	コマンド番号が正しくありません。
7	DR SELECT BY ATTRIBUTEコマンドの値が正しくありません。
8	メモリが足りません。
9	引数が正しくありません。
10	表示コードが正しくありません。
11	グループ解除のレベルが正しくありません。
12	カラーインデックスが正しくありません。
13	RGBの値が正しくありません。
14	クリップボードにピクチャがありません。
15	角度の値が正しくありません。
16	この属性にデフォルト値は設定できません。
17	配列タイプが正しくありません。
18	配列サイズが正しくありません。
19	パターンインデックスが正しくありません。
20	線幅が正しくありません。
21	角の丸みの値が正しくありません。
22	属性コードが正しくありません。
23	ホットリンクが見つかりません。
24	ファイルまたはフィールドの番号が正しくありません。
25	終点マークのタイプが正しくありません。
26	終点マークが正しくありません。
27	文字の行揃えが正しくありません。

28 指定された図形にこの属性はありません。
29 値の個数が多すぎます。
30 ピクチャを作成できませんでした。
31 フィールドタイプが正しくありません。
32 DR AREA TO AREAコマンドの記述が正しくありません。
33 図形をコピーできませんでした。
34 最後のクリックポイントがセットされていません。
35 DR Draw text関数の引数には最低1文字必要です。
36 “送り側”と“受け側”のエリアが同じです。
37 図形が撰択されていません。
38 この属性は全ての図形でロックされています。
39 ピクチャを取り込めません。
40 ピクチャをセットできません。
41 図形の境界が正しくありません。
42 バインドIDが正しくありません。
43 このバインドは、使用中です。
44 この属性は、すでにこのバインドに設定されています。
45 同じ名前とタイプを持ったホットリンクがすでに存在します。
46 このドキュメントには図形がありません。
47 この処理は、このタイプの図形に対しては行なえません。
48 文字は実サイズのときだけ編集することができます。
49 多角形の頂点の数が正しくありません。
50 多角形には最低3つの頂点が必要です。
51 処理を完了するの十分にメモリがありません。
52 ファイルタイプが正しくありません。
53 この4D Drawドキュメントのバージョンは、サポートしていません。
54 この4D Drawドキュメントは、新しいバージョンで作成されています。
55 ダイアログがユーザにキャンセルされました。
56 文字編集モードではありません。
57 スコープが正しくありません。
58 値が範囲外です。
59 グループ化するのに必要な図形がありません。
60 文字位置が正しくありません。
61 文字を回転や反転することはできません。
62 コーディネイトが正しくありません。
63 書類サイズが正しくありません。
64 図形サイズが正しくありません。
65 ホットリンクのタイプが正しくありません。
66 このホットリンクを重複してにホットリンクチェーンに追加することはできません。
67 カラーの値が正しくありません。

68	ドキュメントの外に図形を作成したり、ドキュメントの外に図形を移動することはできません。
69	このコマンドは使用できません。
70	スケーリングの値が正しくありません。
71	この図形の編集属性はロックされています。
72	この処理を行なうと図形数の最大値を越える可能性があります。
73	すべてのグループがロックされています。
74	フォアグラウンドの図形がありません。
75	バックグラウンドの図形がありません。
76	ビットマップイメージが大きすぎます。
77	表示されている図形がありません。
78	非表示になっている図形がありません。
79	ファイルがありません。
80	ズーム倍率が範囲を越えています。
81	ファイルパス名が255文字を越えています。
82	多角形には始点が必要です。
32767	その他のエラー
<0	Macintoshのシステムエラー

付録C イベントコード

次の表は、引数<イベント>用のコードを示したものです。

値	イベント
-1	何らかのイベント
0	イベントなし
1	エリアの作成
2	エリアの削除
4	エリアのアクティベート（クリックまたは最前面へ出す）
8	エリアのディアクティベート（エリアがアクティブでない）
16	図形の作成（作成、ペースト、複製）
32	図形の削除（削除、カット、クリア）
64	commandキーを押しながらクリック（図形上でなくてもよい）
128	図形の移動（上下左右に移動、整列など）
256	図形のサイズ変更（矢印キー、ドラッグなど）
512	図形の回転
1024	選択図形の変更
2048	ダブルクリック
4096	図形の変形 この変形イベントは、多角形や線、弧を変形したり、スムージング / スムージング解除した際に発生します。また、多角形の頂点を追加 / 削除した際にも発生します。

付録D 図形コード

次の表は、4D Drawの図形コードを示したものです。

コード	説明書き
1	文字図形
2	ホットリンク
3	イメージ
4	ビットマップ
5	矩形 / 角の丸い矩形
6	多角形 / フリーハンド図形
7	楕円
8	弧
9	線
10	グループ

付録E ユニットコード

次の表は、DR SET RULERやDR GET RULERで使用するユニットコードを示したものです。

コード	説明
1	インチ
2	フィート
3	ヤード
4	マイル
5	ミリメートル
6	センチメートル
7	ディシメートル
8	メートル
9	デカメートル
10	キロメートル
11	ピクセル

付録F コマンドコード









次の表は、引数<コマンド>用のコードを示したものです。

メニュー	コマンド	コード	
ファイル	新規	1001	
	開く	1002	
	保存	1003	
	新規保存	1004	
	テンプレートとして保存	1006	
	用紙設定	1008	
	プリント	1009	
	フルウインドウ	1011	
	マージプリント	1012	
	インポート	1013	
	イクスポート	1014	
	編集	取り消し	2001
		カット	2003
		コピー	2004
ペースト		2005	
クリア		2006	
複製		2007	
すべてを選択		2009	
文字	フォントメニュー	3001	
	独自のフォント名	8001-8999	
	サイズメニュー	3002	
	独自のフォントサイズ	9001-9999	
	書体メニュー	3003	
	標準	10001	
	ボールド イタリック	10002 10003	

	アンダーライン	10004
	アウトライン	10005
	シャドウ	10006
	行揃えメニュー	3004
	左揃え	11001
	中央揃え	11002
	右揃え	11003
	設定	3005
図形	塗りつぶしパターンメニュー	4001
	独自の塗りつぶしパターン	12001-12033
	塗りつぶしカラーメニュー	4002
	独自の塗りつぶしカラー	13001-13256
	線パターンメニュー	4004
	独自の線パターン	14001-14033
	線カラーメニュー	4005
	独自の線カラー	15001-15256
	線幅メニュー	4007
	ヘアライン	16001
	1ピクセル	16002
	2ピクセル	16003
	4ピクセル	16004
	6ピクセル	16005
	その他	16006
	終点マークメニュー	4008
	マーク線のみ	17001
	マーク始点に矢印	17002
	マーク終点に矢印	17003
	マーク両端に矢印	17004
	マーク	17006
	マーク	17007
	スムージング	4010
	スムージング解除	4011
	変形	4012
	回転	4013
	角の丸み	4015
	ロック	4017
	属性設定	4018
	表示しない	4020
	すべてを表示	4021
操作	最前面へ	5001
	最背面へ	5002

	前面へ	5003
	背面へ	5004
	整列	5006
	グリッドに合わせる	5007
	横反転	5009
	縦反転	5010
	グループ化	5011
	グループ解除	5012
	バックグラウンドに追加	5014
画面	実サイズ	6001
	ウインドウに合わせる	6002
	縮小	6003
	拡大	6004
	表示	6006
	定規	18001
	グリッド	18002
	ページブレイク	18003
	座標	18004
	メニューバー	18005
	ツールパレット	18006
	スクロールバー	18007
	グリッドに吸着	6008
	描画サイズ設定	6010
	定規設定	6011
	環境設定	6012
	バックグラウンド解除	6014
データベース	フィールドをペースト	7001
	フォーマット	7002
	データ参照	7003
	データ参照解除	7004
	値を表示 / 参照名を表示	7006
	ホットリンク使用	7008
	ホットリンク公開	7009
	ホットリンク使用解除	7010
	ホットリンク公開解除	7011
	ホットリンクに追加	7012
	バインド	7014

次の表は、引数<コマンド>用のコードを示したものです。

コード	アイコン	アイテム
1	 「ポインタ」	ポインタ
2	 「文字」	文字
3	 「線」	線
4	 「矩形」	矩形
5	 「角の丸い矩形」	角の丸い矩形
6	 「円/楕円」	円 / 楕円
7	 「弧」	弧
8	 「多角形」	多角形
9	 「フリーハンド」	フリーハンド

A

DR ACTIVATE BIND (エリア;バインドID ;処理)	79
DR ADD TO BACKGROUND (エリア;スコープ)	25
DR ADD TO BIND (バインドID ;属性;ファイル;フィールド)	81
DR ADD TO BITMAP (エリア;スコープ;図形ID)	145
DR ADD TO HOT LINK (エリア;スコープ;名前)	115
DR ALIGN (エリア;スコープ;横揃え;縦揃え)	146
DR AREA TO AREA (送り側;受け側;スコープ)	65
DR AREA TO FIELD (エリア;スコープ;ファイル;フィールド)	66
DR Area to picture (エリア;スコープ) ピクチャ	68
DR ARRAY BASE TO SCALE (エリア;ベース配列;目盛配列)	175
DR ARRAY SCALE TO BASE (エリア;目盛配列;ベース配列)	176
DR ARRAY TO ATTRIBUTE (エリア;スコープ;属性;配列)	121
DR Array to polygon (エリア;配列H ;配列V) 倍長整数	123
DR ATTRIBUTE TO ARRAY (エリア;スコープ;配列;属性)	125

B

DR Base to scale (エリア;値) 実数	177
-----------------------------	-----

C

DR Calculate area (エリア;スコープ) 実数	203
DR Calculate perimeter (エリア;スコープ) 実数	204
DR Clipboard to picture ピクチャ	205
DR Color to index (カラー) 整数	206
DR COLOR TO RGB (カラー;赤;緑;青)	207
DR COORDINATES (エリア;座標;モード)	49
DR Count (エリア;スコープ) 整数	148

D

DR DEACTIVATE BIND (エリア)	82
DR DELETE (エリア;スコープ)	149

DR DELETE BIND (バインドID)	83
DR DELETE OFFSCREEN AREA (エリア)	69
DR DISPLAY OPTIONS (エリア;オプション;モード)	50
DR DO COMMAND (エリア;コマンド)	26
DR Draw arc (エリア;開始角;長さ;半径H;半径V;中心H;中心V) 倍長整数	129
DR Draw line (エリア;開始H;開始V;終了H;終了V;モード) 倍長整数	131
DR Draw oval (エリア;開始H;開始V;終了H;終了V;モード) 倍長整数	132
DR Draw rectangle (エリア;開始H;開始V;終了H;終了V;モード;角の丸み) 倍長整数	133
DR Draw text (エリア;開始H;開始V;終了H;終了V;テキスト{;フレーム;フロー}) 倍長整数	135
E	
DR End polygon (エリア) 倍長整数	137
DR Error ({メッセージ}) 倍長整数	27
DR EVENT FILTER (エリア;マスク)	28
DR EXPERT COMMAND (エリア;コマンド;状態)	29
DR EXPERT MODE (エリア;モード)	30
F	
DR FIELD TO AREA (エリア;ファイル;フィールド)	70
DR Font name (フォントID) 文字列	208
DR Font number (フォント名) 整数	209
G	
DR GET ARC SPECS (エリア;スコープ;開始角;長さ;半径H;半径V;中心H;中心V)	87
DR GET AREA BOUNDARY (エリア;左;上;右;下)	31
DR Get attribute lock (エリア;スコープ;属性コード) 整数	89
DR GET BOUNDARY (エリア;左;上;右;下)	90
DR Get corner rounding (エリア;スコープ) 実数	91
DR Get display (エリア;オプションコード) 整数	52
DR GET DOCUMENT SIZE (エリア;幅;高さ)	53
DR Get draw mode (エリア) 整数	54
DR GET ENDMARKS (エリア;スコープ;タイプ;端点)	92
DR GET FILL ATTRIBUTES (エリア;スコープ;パターン;カラー)	94
DR Get handle state (エリア;スコープ)	96
DR GET HIGHLIGHT (エリア;先頭;最後)	97
DR Get ID (エリア;スコープ;インデックス) 倍長整数	98

DR GET LINE ATTRIBUTES (エリア;スコープ;パターン;カラー;幅)	100
DR GET LINE SPECS (エリア;スコープ;開始H;開始V;終了H;終了V)	102
DR Get name (エリア;スコープ) 文字列	104
DR Get object type (エリア;スコープ) 整数	105
DR GET ORIGIN (エリア;横;縦)	178
DR GET POLYGON VERTEX (エリア;スコープ;頂点番号;頂点H;頂点V)	107
DR GET PREFERENCES (エリア;プリント順序;PICT;ロック時の警告; 自動スクロール;可変プリント領域)	56
DR Get refnum (エリア;スコープ) 倍長整数	108
DR Get rotation (エリア;スコープ) 整数	109
DR GET RULER (エリア;ベースU;目盛U;ベースV;目盛V;目盛の刻み)	179
DR GET RULER OPTIONS (エリア;グリッド;表記;間隔)	181
DR Get text (エリア;スコープ) テキスト	110
DR GET TEXT ATTRIBUTES (エリア;スコープ;フォント;サイズ;書体; 行揃え;フレーム;フロー)	111
DR Get text width (エリア;スコープ) 整数	113
DR Get update mode (エリア) 整数	32
DR Get zoom (エリア) 数値	33
DR GROUP (エリア;スコープ)	150
H	
DR HIDE (エリア;スコープ;モード)	151
I	
DR Index to color (インデックス) 倍長整数	210
DR INSERT EXPRESSION (エリア;スコープ;先頭;最後;値参照 {;フォーマット})	167
DR INSERT FIELD (エリア;スコープ;先頭;最後;ファイル;フィールド {;フォーマット})	169
L	
DR LAST CLICK (エリア;横;縦)	34
DR Last event (エリア) 倍長整数	35
DR LOCK (エリア;スコープ;コード;処理)	152
M	
DR MENU STATUS (エリア;コマンド;チェック;アクティブ;名前)	36
DR MOVE (エリア;スコープ;移動H;移動V;モード)	154

N

DR New bind 倍長整数	84
DR NEW DRAWING (エリア)	71
DR New offscreen area 倍長整数	72

O

DR Objects to bitmap (エリア;スコープ;処理) 倍長整数	138
DR ON ERROR (プロシージャ)	37
DR ON EVENT (プロシージャ)	38
DR ON MENU (エリア;プロシージャ)	40
DR OPEN DOCUMENT (エリア;ドキュメント{;モード})	73

P

DR PICTURE TO AREA (エリア;ピクチャ)	75
DR PICTURE TO CLIPBOARD (ピクチャ)	211
DR Place field (エリア;ファイル番号;フィールド番号;フォーマット;位置) 倍長整数	171
DR PLACE PICTURE (エリア;ピクチャ;位置)	139
DR POLYGON CURVE (エリア;頂点1.X1 ;頂点1.Y1 ;頂点1.X2 ;頂点1.Y2 ; 終点X ;終点Y ;モード)	140
DR POLYGON LINE (エリア;頂点H ;頂点V ;モード)	142
DR POLYGON TO ARRAY (エリア;スコープ;配列H ;配列V)	127
DR PRINT (エリア;メッセージ{;ダイアログ})	163
DR PRINT BACKGROUND (エリア;メッセージ{;ダイアログ})	164
DR PRINT FOREGROUND (エリア;メッセージ{;ダイアログ})	165
DR PRINT MERGE (エリア;メッセージ{;ダイアログ})	166
DR PUBLISH (エリア;スコープ;名前)	117

R

DR REDRAW (エリア)	41
DR RELEASE BACKGROUND (エリア)	42
DR REMOVE FROM BACKGROUND (エリア;図形ID)	43
DR REMOVE FROM BIND (バインドID ;属性)	85
DR RGB to color (赤;緑;青) 倍長整数	212
DR ROTATE (エリア;スコープ;回転角度;モード)	155

S

DR SAVE DOCUMENT (エリア;文書;タイプ{;スコープ})	76
--------------------------------------	----

DR SCALE (エリア;スコープ;アンカH;アンカV;高さ;幅)	156
DR Scale to base (エリア;値) 実数	182
DR SCROLL DOCUMENT (エリア;横;縦;モード)	44
DR SELECT (エリア;スコープ;処理)	159
DR SELECT BY ATTRIBUTE (エリア;論理演算子;コード;値)	160
DR SELECT BY REGION (エリア;方法;処理;左;上;右;下)	161
DR SET ARC SPECS (エリア;スコープ;開始角;長さ)	187
DR SET ATTRIBUTE LOCK (エリア;スコープ;属性コード;状態)	188
DR SET CORNER ROUNDING (エリア;スコープ;角の丸み)	189
DR SET DISPLAY (エリア;オプション番号;表示)	58
DR SET DOCUMENT SIZE (エリア;幅;高さ)	60
DR SET DRAW MODE (エリア;モード)	61
DR SET ENDMARKS (エリア;スコープ;タイプ;端点)	190
DR SET ENTERABLE (エリア;モード)	45
DR SET FILL ATTRIBUTES (エリア;スコープ;パターン;カラー)	191
DR SET FORMAT (エリア;スコープ;フォーマット)	172
DR SET GLOBAL PREFERENCES (読み込み;書き込み)	62
DR SET HANDLE STATE (エリア;スコープ;処理)	192
DR SET HIGHLIGHT (エリア;スコープ;先頭;最後)	192
DR SET LINE ATTRIBUTES (エリア;スコープ;パターン;カラー;幅)	194
DR SET LINE SPECS (エリア;スコープ;開始H;開始V;終了H;終了V)	195
DR SET NAME (エリア;スコープ;名前)	196
DR SET ORIGIN (エリア;横;縦;モード)	183
DR SET POLYGON VERTEX (エリア;スコープ;頂点番号;頂点H;頂点V)	197
DR SET PREFERENCES (エリア;プリント順序;PICT;ロック時の警告; 自動スクロール;可変プリント領域)	63
DR SET REFNUM (エリア;スコープ;参照番号)	198
DR SET RULER (エリア;ベースU;目盛U;ベースV;目盛V;目盛の刻み)	184
DR SET RULER OPTIONS (エリア;グリッド;表記;間隔)	185
DR SET TEXT (エリア;スコープ;テキスト)	199
DR SET TEXT ATTRIBUTES (エリア;スコープ;フォント;サイズ;書体; 行揃え;フレーム;フロー)	200
DR SET UPDATE MODE (エリア;モード)	46
DR SIZE (エリア;スコープ;アンカH;アンカV;高さ;幅;モード)	157
DR START POLYGON (エリア)	144
DR Subscribe (エリア;名前) 倍長整数	118

U

DR UNGROUP (エリア;スコープ;レベル) 158

DR UNPUBLISH (エリア;名前) 119

DR UNSUBSCRIBE (エリア;名前) 120

Z

DR ZOOM (エリア;ズーム;横;縦;位置) 47