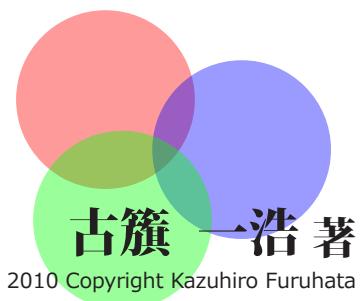


# Adobe Illustrator JavaScript 自動化サンプル集

2010.01.01 版

## Version CS3 (CS4) 対応



- Adobe、Adobe Creative Suite は Adobe Systems Incorporated（アドビシステムズ社）の米国ならびに他の国における商標または商標登録です。
- Macintosh、MacOS X は米国 Apple 社の商標または商標登録です。
- Microsoft Windows および Microsoft Excel の各バージョンは米国 Microsoft Corporation の登録商標です。
- その他、本文中に記載されている製品名、会社名等は関係各社の商標または登録商標です。
- 本 PDF に掲載されているスクリプトを使用して発生したいかなる不具合、トラブルについても責任を負わないものとします。

## はじめに

2004年にIllustrator 10自動化作戦 with JavaScriptが出版されてから5年以上が経過しました。アマゾンではいまだに高い価格で取引されている珍しい本の1つですが、さすがに1冊が万単位の金額にもなると書いた方がびっくりしますし買う方も躊躇しています。

そこで、今回新たにIllustrator CS3(CS4)を対象にして自動化に関するPDFを作成することにしました。普通のIllustrator本であれば一定の需要が見込めるので書籍として出版することができますが、このような自動化に関するものは書籍としては採算が合わないのでPDFとして個人的に出すことにしました。

少しでも手間を削減する方法としてコンピューターが得意なことはコンピューターにやらせる、という方法があります。つまり手作業を自動化してしまおうということです。やり方によっては大幅な時間短縮＝コスト削減になります。ただし、Illustratorの場合はInDesignなどと異なり大量のページものを作成することは少ないため、InDesignと比べて劇的に効率化できるわけではありません。それでも自動化によってかなりの手間が軽減できることも確かです。

自動化といってもプログラムを作らなければいけないとは限りません。特にIllustrator CS3/CS4は機能が大幅に向上し定型処理ならアクション機能／バッチ機能を使えばできてしまいます。このため、プログラムを作成しないと駄目な場面はIllustrator 10と比べてかなり減っています。既存のアクション、バッチ処理で可能なものは極力それらで処理すべきです。また、予算があれば販売されているプラグインを使うのもよいですし、フリーのプラグインで処理可能であればそれでも構いません。

というのも、今からIllustratorを動かすためのプログラム言語であるJavaScript（ジャバスクリプト）を勉強するというのは現実的ではないでしょう。そもそも、学習するだけの時間が捻出できない人も多いはずです。とにかく寝たい！という人もいるかもしれません。

そこで、このPDFではなるべく多くのサンプル／パターンを用意し、個人や社内の状況に合わせて自由に修正し使ってもらうという考え方で作成しました。

自動化で全てが解決することはできませんが、本PDFが何らかの手助けになればと思います。

2010年1月 古旗一浩

### 【注意】

本PDFはMacOS X版のIllustrator CS3を基準にしています。Windows版での動作確認は行っていません。ファイルパスなど一部の指定やMacOS X固有のものを除けばほとんどのスクリプトはWindows版でも動作します。また、CS3以前のCS/CS2ではスクリプトの動作が怪しいため、CS3で動作してもCS/CS2では期待通り動かないことがあります。CS3より前のバージョンでの動作は全く保証しませんし質問なども受け付けていません。CS4はGUI部分の一部を除いて、ほぼそのまま動作するはずです。ただし、CS4独自の機能を利用したスクリプトは掲載していません。

掲載しているスクリプトに不具合が存在することもあります。その場合は「学習編」の「スクリプトの利用について」の説明に従ってください。

## 目次

■スクリプトの利用について .....	3
■本 PDF の構成 .....	4
■ Illustrator の JavaScript の情報源.....	5
■掲載スクリプトについての注意事項.....	6
■掲載されているスクリプトを業務内容にあわせて書き換えるには....	7
■自動化でどのくらい早くなるのか？ .....	10
■こんな処理は自動化できない .....	11
■ Illustrator10 で作成されたスクリプトとの違い .....	12
■ Illustrator の座標系 .....	13
■ファイルパスの指定方法について.....	14
■ Illustrator から実行する .....	16
■ ESTK2 から実行する .....	18
■ Bridge から実行する .....	19
■ ESTK2 (Extend Script ToolKit 2) とは .....	20
■ ESTK2 の起動.....	22
■プログラムを入力してみよう .....	23
■指定した行で実行を一時的に止める（ブレークポイントの設定） .....	26
■正常に動作しなくなった場合 .....	28
■簡単なプログラムを作成しよう .....	29
■日本語を表示する .....	30
■計算させる .....	31
■ Illustrator を制御する .....	34
■新規にドキュメントを作成する.....	34
■テキストフレームを作成する .....	36
■後から処理しやすいようにプログラムの書き方を変える .....	37
■テキストフレームの位置を指定する .....	39
■ドキュメントを保存する .....	40
■JavaScript を学習しよう .....	42
■バッチ処理とプログラムの違い.....	43
■プログラムのルール .....	45
■改行について .....	46
■プログラムの実行順序について.....	47
■コメント（注釈）について .....	48
■変数について .....	49
■配列変数／連想配列（ハッシュ）について .....	51
■条件判断について .....	53
■繰り返し処理について .....	56
■条件付き繰り返し処理について.....	59
■計算について .....	60
■ true、false、undefined、null について .....	63
■関数について .....	64
■例外処理について（エラー処理） .....	67
■自動型変換について .....	68
■変数、関数のスコープについて.....	70
■オブジェクト、プロパティ、メソッドについて.....	72
■ Illustrator のオブジェクト階層図 .....	74
■ JavaScript の基本オブジェクト（ビルトインオブジェクト）.....	75
■ Array オブジェクト（配列） .....	76
■ Math オブジェクト（高度な計算）.....	77
■ Boolean (true か false か／真偽値) .....	78
■ Number (数値を扱う) .....	79
■ String (文字列を扱う) .....	80
■ Date (日付や時間を扱う) .....	81
■ 正規表現（複雑な検索や置換を行う） .....	82

## 目次

■ツールパネル	3
■ファイルメニュー	4
■編集メニュー	5
■オブジェクトメニュー	6
■オブジェクトメニュー>変形	7
■オブジェクトメニュー>アレンジ	8
■オブジェクトメニュー>パス	9
■オブジェクトメニュー>ライブトレース	10
■オブジェクトメニュー>複合パス	11
■書式>フォント	12
■書式>組み方向	13
■選択	14
■線パネル	15
■文字パネル	16
■段落パネル	17
■タブパネル	18
■カラーパネル	19
■スウォッチカラーパネル	20
■グラデーションパネル	21
■透明パネル	22
■レイヤーパネル	23
■透明パネル	24
■シンボルパネル	25
■ブラシパネル	26
■グラフィックスタイルパネル	27
■ドキュメント	28
■各種オブジェクト	29
■パスオブジェクト	30
■プリント（印刷）	31

## 目次

■コメントを入れる .....	3
■変数を用意する .....	4
■変数に数値や文字を入れる .....	5
■配列変数を用意する .....	6
■配列のサイズ、要素の総数を調べる .....	8
■基本的な演算を行う .....	9
■高度な演算を行う .....	10
■小数点以下の処理を行う .....	12
■乱数値を求める .....	14
■奇数、偶数を調べる .....	15
■文字列を連結する（文字列同士の加算） .....	16
■文字列の長さを調べる .....	17
■特定の文字列があるかどうか調べる .....	18
■文字列を抜き出す .....	19
■条件判断を行う .....	21
■複数の条件を満たす条件判断を行う .....	23
■一定回数繰り返し処理を行う .....	25
■条件付で繰り返し処理を行う .....	27
■内容によって分岐させる .....	28
■例外（エラー）処理を行う .....	30
■関数を定義する .....	32
■プロパティやメソッドを削除する .....	34
■プロパティを作成する .....	35
■メソッドを定義する .....	38
■文字から数値に変換する .....	41
■数値から文字に変換する .....	42
■文字を文字コードに変換する .....	43
■英文字を大文字、小文字に変換する .....	44
■日付処理を行う .....	45
■時間処理を行う .....	47

## 目次

基本編	1
アプリケーション	5
■ アプリケーション名を求める	6
■ バージョンを求める	7
■ スクリプトエンジンのバージョンを求める	9
■ ビープ音を鳴らす	10
■ コピーする	11
■ カットする	12
■ ペーストする、貼り付ける	13
■ 取り消し処理を行う	14
■ 実行しているスクリプトのファイル名を求める	15
■ マトリックスオブジェクトを作成する	16
■ マトリックスオブジェクトを連結する	17
■ イラストレータを終了させる	18
ドキュメント	19
■ 新規にドキュメントを作成する	20
■ 新規にドキュメントを作成する（プリセット設定）	21
■ 最前面の（アクティブ）ドキュメントを指定する	23
■ 開かれているドキュメントの総数を求める	24
■ ドキュメントが保存されたかどうか調べる	25
■ ドキュメントを保存する	26
■ ドキュメントの一部を PNG 画像として保存する	28
■ ドキュメント名を調べる	29
■ ドキュメントサイズを調べる	30
■ ドキュメントのカラーモードを調べる	31
■ ドキュメントの原点座標を求める	32
■ トリムエリアを設定する	33
■ ドキュメントを閉じる	34
カラー	35
■ RGB カラーで指定する	36
■ CMYK カラーで指定する	37
■ グレーカラーで指定する	38
■ 透明（カラーなし）を指定する	39
■ Lab カラーで指定する	40
■ グラデーションカラーを指定する	41
■ パターンカラーを指定する	43
■ スポットカラーを指定する	44
■ オブジェクトのカラーを読み出す	45
■ スウォッチカラーの総数を求める	47
■ スpottカラーの総数を求める	48
■ グラデーションカラーの総数を求める	49
■ パターンの総数を求める	50
■ 特定のスポットカラーを削除する	51
■ スpottカラーの種類を調べる	52
■ RGB カラーを HSV カラーに変換する	53
■ HSV カラーを RGB カラーに変換する	54
■ RGB カラーを YCbCr( 輝度と赤青の色差 ) カラーに変換する	55
■ YCbCr カラーを RGB カラーに変換する	56
フォント／書体	57
■ 登録フォント（書体）数を求める	58
■ フォント名、フォントファミリー名を求める	59
■ 全てのフォント名を表示する	60
■ 特定のフォントを指定する	61
テキストフレーム	62
■ テキストフレームを作成する	63
■ エリアテキストを作成する	64

■ パステキストを作成する	65
■ 段組を作成する	66
■ テキストフレーム内に文字を表示する	67
■ テキストフレームの位置を指定する	68
■ テキストフレームを回転する	70
■ テキストフレームのスケールを変更する	71
■ テキストフレームを複製する	72
■ テキストのアウトラインを作成する	73
■ テキストフレームの不透明度を指定する	74
■ テキストフレームの種類を調べる	75
■ テキストフレームの前後関係（Z 座標）を変更する	76
段落	78
■ 段落を作成する	79
■ 段落を挿入する	80
■ 段落を削除する	81
■ 段落スタイルを作成する	82
■ 段落スタイルを反映させる	83
■ 行揃えを指定する	84
■ 行単位で文字サイズを指定する	85
■ 行単位でフォントを指定する	86
■ 行単位でベースラインを指定する	87
■ 行単位で文字色を指定する	88
■ 行間を指定する	89
■ 段落の総数を求める	90
■ 段落スタイル名を求める	91
■ 段落の禁則処理名を求める	92
■ 段落のインデントを指定する	93
テキスト／文字	94
■ 文字のサイズを指定する	95
■ 文字の色を指定する	96
■ 文字のフォント（書体）を指定する	98
■ 文字の長体、平体を設定する	99
■ 文字のトラッキングを設定する	100
■ 文字のカーニングを設定する	101
■ 文字のベースラインを設定する	102
■ 英文字のキャップスを設定する	103
■ 文字のスタイル名を求める	104
正規表現	105
■ 特定の文字が含まれるかどうか調べる	106
■ 特定の文字が含まれるかどうか調べる	107
■ マッチした文字を取り出す	108
■ 文字を置換する	109
パス、図形	110
■ パスの総数を求める	111
■ 選択されたパスのパスポイントの総数を求める	112
■ 選択されたパスポイントの座標値を求める	113
■ 選択されたパスポイントのハンドル座標を求める	114
■ 選択された中でパスを削除する	115
■ ドキュメント内の全てのパスを削除する	116
■ 特定のレイヤー内の全てのパスを削除する	117
■ 特定のパスポイントを削除する	118
■ 特定のパスポイントを選択する	119
■ 四角形を描く	120
■ 角丸四角形を描く	121
■ 楕円形を描く	122
■ 多角形を描く	123
■ 星形を描く	124

■ 直線を描く	125
■ 点線を描く	127
■ ベジエ曲線を描く	129
■ 半円形を描く	130
■ 円弧を描く	131
■ 複合パスを作成する	132
■ 複合パスを削除する	134
■ 複合パスの総数を求める	135
■ クリッピングパスを作成する	136
■ ガイドを作成する	137
■ 前後関係(Z座標)を変更する	138
■ 図形の不透明度を指定する	140
■ 図形の描画モードを指定する	141
画像／イメージ	142
■ 配置された画像の数を求める	143
■ 選択された画像を移動させる	144
■ 選択された画像を回転させる	146
■ 選択された中で画像を削除する	148
■ 画像だけ選択する	149
■ JPEG画像や特定の種類の画像だけ選択する	150
■ 特定の画像形式以外の画像を選択する	151
■ 画像を配置する	152
■ 選択された画像をトレースする	153
■ 選択された画像をトレースし、しきい値を設定する	154
■ 選択されたトレース画像のしきい値を設定する	155
■ 選択されたトレース画像のトレーシングモードを設定する	156
■ 選択されたトレース画像の最大色数を設定する	157
■ 選択された画像の座標値を求める	158
■ 配置された画像のファイルパスを求める	159
グループ	161
■ グループ化する	162
■ 選択されたオブジェクトをグループ化する	163
■ グループを解除する	164
レイヤー	165
■ レイヤーを作成する	166
■ レイヤーを名前で指定する	167
■ レイヤー数を取得する	168
■ レイヤーを削除する	169
■ アクティブライヤーにアクセスする	170
■ レイヤーをアクティブにする	171
■ サブレイヤーを作成する(レイヤー内にレイヤーを作成する)	172
■ サブレイヤーを削除する	173
選択	174
■ 選択を解除する	175
■ 選択されたオブジェクトの種類を求める	176
■ 選択されたテキストフレーム内の文字を求める	177
■ 選択したテキストフレーム内の文字を削除する	178
■ テキストフレームだけ選択する	179
■ 選択されたテキストフレームの3行目を選択する	180
■ テキストフレームだけ選択を解除する	181
■ 各種オブジェクトを選択する	182
■ 特定の注釈を持つオブジェクトのみ選択する	184
■ 選択されたテキストフレーム内の文字カラーを変更する	185
■ 選択されたテキストフレーム内の文字サイズを変更する	187
■ テキストフレーム内で選択された文字の位置を求める	188
■ テキストフレーム内で任意の位置から文字を選択する	189
■ 特定の破線パターンのパスだけ選択する	191

プリント／印刷	194
■ プリンターを求める	195
■ プリントプリセットを求める	196
■ アクティブドキュメントを印刷する	197
フォルダ	198
■ 現在のフォルダ(カレントフォルダ)を取得する	199
■ ドキュメントフォルダを取得する	200
■ デスクトップフォルダを取得する	201
■ ゴミ箱フォルダを取得する	202
■ テンポラリフォルダを取得する	203
■ フォルダ内のファイル一覧を取得する	204
■ フォルダ内のフォルダだけの一覧を取得する	208
■ フォルダ内のファイル一覧だけを取得する	210
■ サブフォルダ内のファイル等も取得する	212
ファイル処理	214
■ Illustratorで開くことができるファイルをファイル選択ダイアログを表示して開く	215
■ ファイルパスを指定して開く	216
■ テキストファイルを開いて内容を全て読み込む	217
■ テキストファイルを一行だけ読み込む	219
■ テキストファイルから一文字だけ読み込む	220
■ テキストファイルから4文字だけ読み込む	221
■ テキストファイルで任意の位置から4文字だけ読み込む	222
■ ファイルに一行書き込む	223
■ ファイルをコピーする	224
■ ファイルを削除する	225
■ ファイル名を変更する	226
■ ファイル名を求める	227
■ ファイルのパスを求める	228
■ ファイルの読み出し位置(ファイルポジション)を求める	229
■ ファイルサイズを求める	230
■ 行数を求める	231
■ ファイルがエイリアス／ショートカットかどうか調べる	232
■ ファイルの作成日、修正日を求める	233
■ ファイルが存在するかどうか調べる	234
■ 改行コードの種類を調べる	235
■ ファイルが書き込み禁止になっているか調べる	236
■ 親フォルダを求める	237
■ アプリケーションを実行する	238
ファイル保存	239
■ アクティブドキュメントを保存する	240
■ EPS形式で保存する	241
■ GIF形式で保存する	243
■ PNG形式で保存する	244
■ JPEG形式で保存する	246
■ Photoshop形式で保存する	247
■ ファイルをPDFとして保存する	248
■ Flash形式で保存する	249
■ SVG形式で保存する	250
単位／単位変換	251
■ 単位付きのオブジェクトを作成する	252
■ 単位を変換する	253
■ 異なる単位同士で計算する	254
■ 異なる単位での値の比較をチェックする	255
XMLデータ処理	256
■ XMLデータを作成する	257
■ 特定のXMLタグにアクセスする	258

■ 指定した XML タグの数を求める .....	261
■ XML タグの属性にアクセスする .....	262
■ XML データをファイルから読み込む .....	264
PDF .....	266
■ ページを指定して PDF を開く .....	267
■ ファイルを PDF として保存する .....	268
■ パスワード付の PDF として保存する.....	269
ダイアログ .....	270
■ アラートダイアログを表示する.....	271
■ 確認ダイアログを表示する .....	272
■ 入力ダイアログを表示する .....	273
■ アラートダイアログを表示しないようにする .....	274
■ ファイル選択ダイアログを表示する .....	275
■ ファイル保存ダイアログを表示する.....	277
■ フォルダ選択ダイアログを表示する .....	278
■ カラーピッカーを表示する .....	279
GUI .....	280
■ ウィンドウやダイアログ、パネルを表示する .....	281
■ クローズボタンを表示する .....	283
■ 画面の中央にウィンドウを表示する .....	284
■ ウィンドウを閉じる .....	285
■ ウィンドウを閉じた時に処理を行う .....	286
■ イベントを設定、解除する .....	288
■ OK ボタン、キャンセルボタンを表示する .....	289
■ ボタンを表示する .....	291
■ 固定テキストを表示する .....	292
■ 編集可能なテキストを表示する.....	293
■ チェックボックスを表示する .....	294
■ ラジオボタンを表示する .....	295
■ ドロップダウンリストを表示する.....	297
■ リストボックスを表示する .....	298
■ パネルを表示する .....	299
■ グループを作成する .....	300
■ アイコンボタンを表示する .....	301
デバッグ .....	302
■ スクリプトの実行を一時停止する（ブレークポイント） .....	303
■ コンソールに文字などを出力する.....	304
■ 環境変数を使用する .....	305
■ 一定時間処理を行わずに待つ（スリープ） .....	306

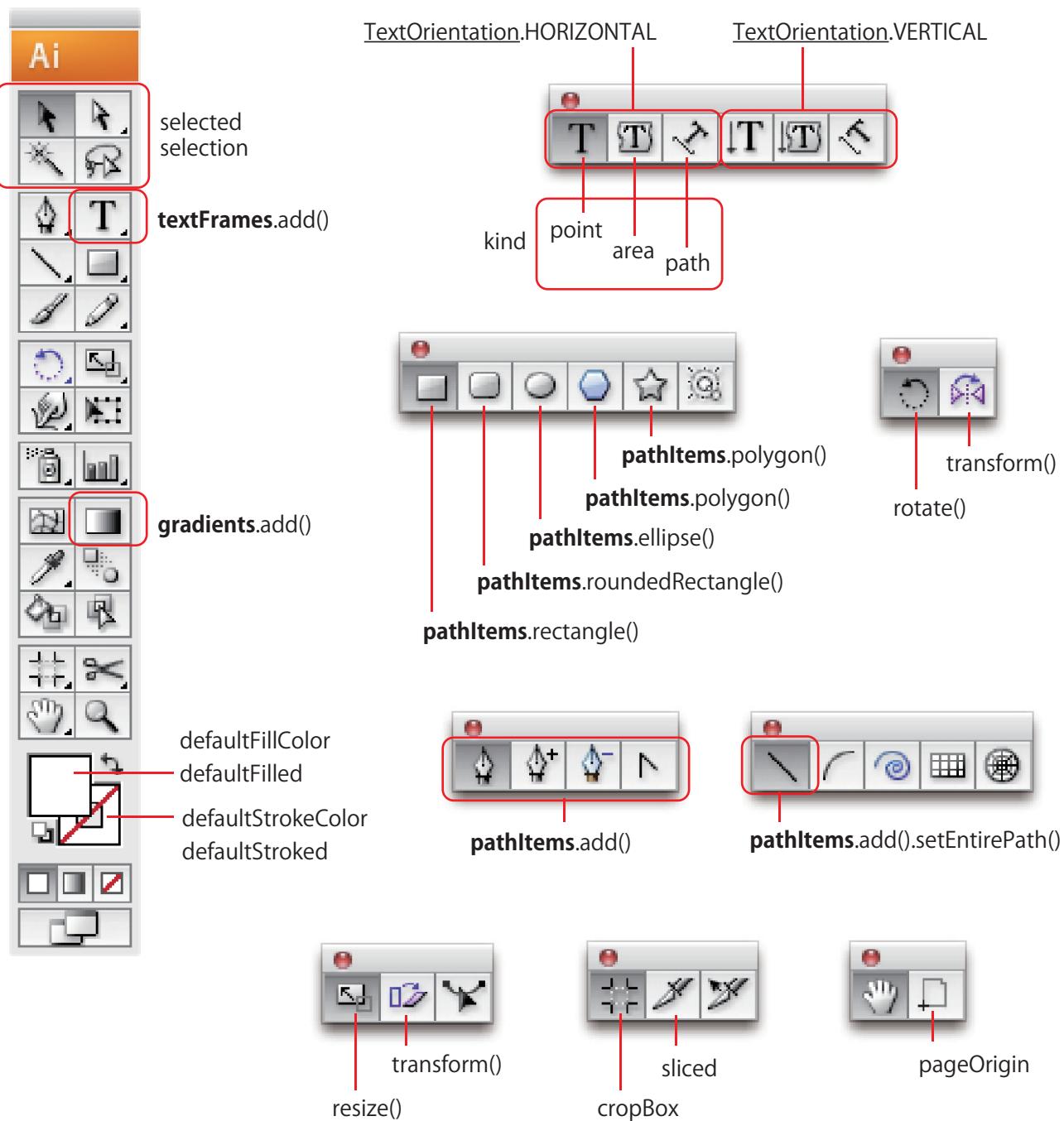
## 目次

応用編.....	1
ドキュメント処理.....	4
■開かれているドキュメントを保存せず全て閉じる.....	5
■保存済みのドキュメントだけ閉じる.....	6
■ドキュメントにタイムスタンプを表示する .....	7
■ドキュメントを分割して PNG 形式で保存する.....	8
テキストフレーム処理.....	10
■テキストフレームの回転角度を表示する .....	11
■テキストフレームに背景色を付ける .....	12
■テキストフレームを複製して影を付ける .....	13
■テキストフレーム内の番号を 1 つ増やす .....	14
■テキストフレーム内の価格に消費税を加える .....	15
■テキストフレーム内の価格のフォーマット（円←→¥）を変更する	17
■テキストフレーム内の日付フォーマットを変更する.....	20
■テキストフレームをテキストファイルとして書き出す .....	22
■アウトライン化する際にフォント名をメモに埋め込む .....	23
■テキストフレーム内の横の文字数を指定する .....	24
■テキストフレームを一行ごと分解する .....	25
■テキストフレームの文字を 1 文字ずつ分解する.....	26
■ファイルを読み込み 1 文字単位で配置する .....	27
■選択した行を連結する .....	33
■テキストのみ新規ドキュメントにコピーする .....	34
■図形とテキストが選択されたらテキストを前面に出す .....	35
■タブ区切りテキストを読み込み縦に分割して配置する .....	36
■タブ区切りテキストをエクセルのようにマス目で表示する .....	37
■選択されたテキストフレーム内容を指定ファイル内容に入れ替える	42
■選択された HTML ファイルのタグを削除して配置する .....	46
テキスト／文字処理.....	47
■数字だけサイズを変える .....	48
■英文字だけサイズを変える .....	50
■平仮名、片仮名だけサイズを変える .....	51
■漢字だけサイズを変える .....	52
■数字だけフォントを変える .....	53
■英文字だけフォントを変える .....	54
■平仮名、片仮名だけフォントを変える .....	55
■漢字だけフォントを変える .....	56
■テキストの不透明度をランダムに変える .....	57
■数字や英文字など特定の文字のみ色を変える .....	59
■特定の文字列のみ色を変える .....	61
■特定の文字列の両側にカッコを付ける .....	64
■文字を円に沿って配置する .....	65
■文字を螺旋状に配置する .....	67
■文字をサインカーブに沿って配置する .....	70
■文字の色をグラデーションにする.....	72
■文字の色をランダムに変える .....	75
■文字別にトラッキングを設定する.....	77
■文字を縦中横にする .....	78
■常用漢字以外が含まれているか調べる .....	82
■文字のフォントを調べる .....	86
■特定のフォントが含まれているかどうか調べる.....	87
■引用符を変換する .....	88
■語尾を変更する.....	89
■ファイルに記載されたデータで字詰め処理を行う .....	90
■特定のスウォッチカラーを削除する .....	92
■規則性を持ったスウォッチカラーを作成する .....	93
■図形の色相や輝度を変える .....	94

■カラーのイエローを消す(2 色にする) .....	96
■ドキュメント内の CMYK カラーのオブジェクトを選択する .....	97
ファイル処理 .....	99
■ファイル内容を読み込み配置する.....	100
■フォルダ内にある Illustrator ファイルを開く .....	102
■サブフォルダ内にある全ての AI ファイルを開く .....	103
■フォルダ内の全ての EPS ファイルを指定バージョンで保存しなおす.	104
■フォルダ内の全ての EPS ファイルを PDF 形式で保存しなおす ...	105
■複数ページある PDF を分解し 1 ページごとの PDF に保存しなおす ..	106
■複数ページある PDF を分解し 1 ページごとの JPEG に保存しなおす	108
■PDF 内のテキストを読み出してテキストファイルとして保存する	111
■フォルダ内の全ての EPS ファイル内の文字をテキストファイルとして	
保存する .....	113
■フォルダ内にある EPS ファイル内のテキストを検索する .....	114
■ドキュメント情報をファイルに出力する .....	115
イメージ／画像処理 .....	116
■選択された画像の回転角度を表示する .....	117
■選択された画像のスケール（拡大縮小率）を表示する .....	118
■選択された画像の傾きを表示する .....	119
■選択された画像の幅や傾きをリセットする .....	120
■選択された画像を傾ける .....	121
■画像のみ新規ドキュメントにコピーする .....	122
■ランダムな位置に画像を配置する .....	123
■画像のまわりに枠を付ける .....	127
■画像の上と左に幅を表示する .....	130
■ファイル選択ダイアログで選択した画像をまとめて配置する .....	133
■画像の座標などの情報をファイルに書き出す .....	134
■リンク画像のファイルパスを表示する .....	135
■リンク画像の横幅と縦幅を表示する .....	138
■フォルダ内にある画像をタイル状に 1 ページだけ割り付ける .....	140
■フォルダ内にある全ての画像をタイル状に割り付けページ単位で保存	
する .....	144
■サブフォルダも含むフォルダ内にある全ての画像をタイル状に割り付	
けページ単位で保存する .....	147
■埋め込み画像の不透明度を変える .....	150
■選択されたパスを图形に置換する .....	152
■ゲームのマップを描く .....	158
■画像をタイル状に分割する .....	160
パス／图形処理 .....	163
■選択された图形を絶対座標系で回転させる .....	164
■图形のバウンディングボックスから引き出し線を描く .....	165
■任意の数だけ罫線を描く .....	166
■市松模様を描く .....	176
■ランダムな位置に图形を描く .....	180
■ドキュメント内にランダムに線を描く .....	186
■階段状の線を描く .....	190
■图形を複製して影を付ける .....	191
■图形のパスポイントに●を表示する .....	192
■图形のパスポイントに●、ハンドルに■を表示する .....	193
■四角形を円に沿って配置する .....	195
■四角形を螺旋状に配置する .....	198
■カラーランプを描く .....	200
■雪を描く .....	202

■ 雨を描く .....	204
■ 緯度経度の線を描く .....	206
■ 選択されたパスを個別の線に分解する .....	208
■ 選択した四角形を橢円形に置き換える .....	209
■ レイヤー処理 .....	211
■ 表示されていないレイヤーのみ削除する .....	212
■ ロックされたレイヤーのみ削除する .....	214
■ 空っぽのレイヤーを削除する .....	215
■ レイヤー状態を保存する .....	217
■ サブレイヤーも含めたレイヤー一覧を作成する .....	218
■ サブレイヤーもフラットに指定できるようにする .....	221
変換、チェック .....	222
■ オーバープリントのチェックを行う .....	223
■ テキストフレーム内の金額がオーバーしているか調べる .....	225
■ 全角英数字が含まれているかどうか調べる .....	227
■ 半角英数字を全角英数字にする .....	229
■ 全角英数字を半角英数字にする .....	230
■ アラビア数字を漢数字に置き換える .....	231
■ 漢数字をアラビア数字に置き換える .....	232
■ テキストフレーム内のディスク容量の単位を変更する .....	233
■ 句読点をピリオドとカンマ変換する .....	234
■ 3行ごとにカンマを付ける .....	235
■ レガシーテキストがあるかどうか調べる .....	236
■ レガシーテキストを一括変換し保存する .....	237
フォント／書体 .....	238
■ フォント一覧を作成する .....	239
■ ドキュメント内で使用されているフォントをファイルに書き出す .....	241
■ インストールされていないフォントを調べてファイルに書き出す .....	244
プリント／印刷処理 .....	250
■ 開かれているドキュメントを全て印刷する .....	251
■ 特定のフォルダ内にある AI または EPS ファイルを印刷する .....	252
一括／大量処理 .....	254
■ フォルダ内にある AI または EPS ファイルのカラー mode をチェックする .....	255
■ フォルダ内にある AI または EPS ファイルにウォーターマークを入れる .....	257
■ フォルダ内にある AI または EPS ファイルにタイムスタンプを入れる .....	259
■ フォルダ内にある AI または EPS ファイルに任意の文字を入れる .....	261
■ フォルダ内にある AI ファイル内の文字を検索し、一致したファイルを開く .....	263
■ シリアル番号を付けファイルとして保存する .....	265
■ フォルダ内にあるテキストファイルを HTML にする .....	268
■ フォルダ内にある全 EPS ファイル内に特定のフォントが使われているかどうか調べる .....	271
■ フォルダ内にある EPS ファイルがアウトライン化されているか調べる .....	273
その他 .....	275
■ 選択されたオブジェクトをランダムにペーストする .....	276
■ ドキュメント内のテキスト以外のオブジェクトを全てロックする .....	277
■ オブジェクトに注釈（ノート）を入れる .....	279
■ オブジェクトの座標を表示する .....	281
■ 名刺の名前などを自動的に入れ替え EPS ファイルとして保存する .....	283
■ 住所氏名のデータを元に葉書を自動生成し保存する .....	285
■ スクラッチカードの番号を読み込み配置する .....	287
■ カレンダーを作成する .....	289
■ RSS データを読み込み配置する .....	295
■ オブジェクトを HTML5 の canvas で描画できる命令に変換する .....	297
■ パスの座標値を KML に変換する .....	299
■ KML データを読み込み表示する .....	302
■ アンドウを利用してアニメーション用連番ファイルを作成する .....	304
■ ドキュメントの表示倍率を指定する .....	306
■ 処理が終わったらメールを送信して知らせる .....	307

## ■ツールパネル



## ■コメントを入れる

コメントを入れるには // または /\* ~ \*/ を使います。一行だけコメントにしたい場合は // を使います。この場合、// から行末（改行）までがコメントとなります。複数行をまとめてコメントにしたい場合は /\* と \*/ の間にコメントしたい部分を入れます。また、コメントにしてプログラムの一部を動作させないようにすることを「コメントアウト」すると言います。

コメントにはプログラムがどういう動作をするのか、制作したのは誰か、関数を呼び出すパラメータ（引数）などを書いておくとよいでしょう。また、ESTK2 には選択した行をまとめてコメントにしたり解除したりするメニューが用意されています（編集メニューの「選択行をコメント化とコメント解除」）。



### ●サンプル1

```
// コメントを使う
// ここは一行のコメントになります。
/*
    alert("OK");
    ここは複数行のコメントになります。
    コメント部分は何も処理されません。
*/
```

## ■テキストフレームの前後関係(Z座標)を変更する

テキストフレームの前後関係(Z座標)を変更するにはmove()メソッドを使います。move()の最初のパラメータには位置を変更するための基準となるオブジェクトを指定します。これはグループオブジェクトやドキュメントオブジェクトなどになります。

move()の2番目のパラメータには以下の表に示すものを指定できます。オブジェクトメニューのアレンジにある「最前面へ」「最背面へ」はドキュメントオブジェクトを基準にして処理を行います。

【表】

ElementPlacement.INSIDE	内側
ElementPlacement.PLACEATBEGINNING	最前面へ
ElementPlacement.PLACEATEND	最背面へ
ElementPlacement.PLACEBEFORE	前へ
ElementPlacement.PLACEAFTER	後へ

サンプル1では4つのテキストフレームを描き、マゼンタ色のテキストフレームを最前面に移動させています。サンプル2では逆にマゼンタ色のテキストフレームを最背面に移動させています。

### ●サンプル1

```
// テキストフレームを最前面に移動させる
(function(){
    var text1 = drawText(140, 0, setCMYKColor(100,0,0,0));
    var text2 = drawText(100, 20, setCMYKColor(0,100,0,0));
    var text3 = drawText(60, 40, setCMYKColor(100,100,0,0));
    var text4 = drawText(20, 60, setCMYKColor(100,0,100,0));
    text2.move(app.activeDocument, ElementPlacement.PLACEATBEGINNING);
})();
// 文字を描く
function drawText(x,y, color){
    var txtObj = app.activeDocument.textFrames.add();
    txtObj.contents = "Illustrator";
    txtObj.paragraphs[0].size = 48;
    txtObj.paragraphs[0].fillColor = color;
    txtObj.paragraphs[0].textFont = app.textFonts["Arial-Black"];
    txtObj.translate(x,y);
    return txtObj;
}
// CMYK カラーを設定し、CMYK カラーオブジェクトを返す
function setCMYKColor(c,m,y,k){
    var CMYK = new CMYKColor();
    CMYK.cyan = c;
    CMYK.magenta = m;
    CMYK.yellow = y;
    CMYK.black = k;
    return CMYK;
}
```



## ■サブフォルダも含むフォルダ内にある全ての画像をタイル状に割り付けページ単位で保存する

サブフォルダも含むフォルダ内にある全ての画像をタイル状に割り付けページ単位で保存するにはフォルダ選択ダイアログを表示して基準(親／ルート)となるフォルダを決定します。このフォルダを基準としてフォルダ内にあるファイル一覧を取得します。ファイル一覧の中にフォルダがある場合は、再帰を使ってそのフォルダ内にあるファイルを全て取得します。このため、ファイルが多い場合やフォルダ階層が深い場合には処理時間がかかる場合があります。ファイル数が多い場合や、デバイス(SDカード等)が低速な場合には数分から数十分かかることもあります。

サンプル1ではファイル形式はEPSで「1.eps」「2.eps」のように連番ファイルとして保存されます。サンプルではprompt()を使ってユーザーに配置する枚数を入力するようになっています。枚数が固定されている場合はvar xCount = 5; var yCount = 8;のように直接数値を指定してください。また、画像と画像の余白なども同様にプログラム内にある変数の値を変更することで行えます。

サンプル1ではjpg,png,gif,eps,.psdの5種類を配置する対象にしていますが、これを変更したい場合は「var extType = /\.jpg|\.png|\.gif|\.eps|\.psd/g;」の正規表現を変更します。正規表現でマッチするかどうかを指定しているだけですので、応用すればABCで始まるファイル名を配置することもできます。

保存するEPSのオプションはsaveEPS()関数内で指定していますので、状況に合わせて設定してください。また、ファイルの枚数によっては最終ページが空白になることがあります。

**【注意】**配置する画像によっては途中でダイアログが表示されることがあります。また、MacOS Xの場合ファイル名やフォルダ名に日本語が含まれるとファイル情報が取得できずエラーで停止したり、ハングアップ状態になることがあります。このため、MacOS Xではファイルやフォルダ名に日本語が含まれないようにしてください。

### ●サンプル1

```
// サブフォルダも含むフォルダ内にある全ての画像をタイル状に割り付けページ単位で保存する
(function(){
    var folderObj = Folder.selectDialog("JPEG画像ファイルがあるフォルダを選択して下さい");
    if (!folderObj) return;
    var saveObj = Folder.selectDialog("EPSファイルを保存するフォルダを選択して下さい");
    if (!saveObj) return;
    var fileList = new Array();
    getFileList(folderObj); // 全てのファイルを対象にする
    var xCount = parseInt(prompt("横の表示枚数をいれてください", 5));
    if (!xCount) return;
    var yCount = parseInt(prompt("縦の表示枚数をいれてください", 8));
    if (!yCount) return;
    var extType = /\.jpg|\.png|\.gif|\.eps|\.psd/g; // 配置したい画像ファイルの拡張子
    var imageW = 100; // 画像の横幅は100pt
    var imageH = 80; // 画像の縦幅は80pt
    var marginW = 10; // 横の余白(単位はポイント)
    var marginH = 20; // 縦の余白(単位はポイント)
    var startX = 20; // 横の開始座標(単位はポイント)
    var startY = app.activeDocument.height; // ドキュメントの一番上から
    startY = startY - 20; // 一番上の余白
    var x = startX;
    var y = startY;
    var tempY = 0;
    var pageNo = 1; // ファイル名として保存するページ番号
    var imageData = []; // 画像データ
    var imgLayer = app.activeDocument.layers.add();
    for(var i=0; i<fileList.length; i++){
        if (fileList[i].name.match(extType)){
            var myImage = app.activeDocument.placedItems.add();
            myImage.file = fileList[i]; // Mac OS Xではファイル名が取得できず、ここでエラーになる事があり！
            var w = myImage.width;
            var h = myImage.height;
            var percent = calcSize(w, h, imageW, imageH);
            myImage.resize(percent*100, percent*100);
            myImage.left = x;
            myImage.top = y;
            myImage.embed(); // 埋め込み画像にする
            imageData.push(myImage); // 画像情報を配列に入れておく
            x = x + (imageW+marginW); // 横幅を加算
            if (x >= xCount * (imageW+marginW)){
                tempY = 0;
                x = startX;
                y = y + marginH;
            }
        }
    }
})()
```

## ■選択された画像の回転角度を表示する

選択された画像の回転角度を表示するにはオブジェクトの matrix の各プロパティを読み出します。マトリックスのプロパティ値から画像の回転角度を求めます。

サンプル 1 は選択された画像の回転角度をアラートダイアログに表示します。

### ●サンプル 1

```
// 選択された画像の回転角度を表示する
(function(){
    var selObjList = activeDocument.selection;
    for(var i=0; i<selObjList.length; i++){
        var tp = selObjList[i].typename;
        if ((tp == "RasterItem")|| (tp == "PlacedItem")){
            var mtx = selObjList[i].matrix;
            var a = mtx.mValueA;
            var b = mtx.mValueB;
            var c = mtx.mValueC;
            var d = mtx.mValueD;
            var scaleW = Math.sqrt(Math.pow(a, 2)+Math.pow(b, 2));
            var deg = (Math.acos(a/scaleW)) * 180 / Math.PI;
            if ((b <= 0) && (c >= 0)) deg = -deg;
            alert(deg+" 度回転しています");
        }
    }
})();
```

