

DTPでの表作りは、頭を悩ます作業です。エクステンションやイラストレータを利用して作成している方も多いと思います。表計算ソフトの定番「Excel」はAppleScriptに十二分に対応していることは意外と知られていません。Excelでの自動処理では

AppleScriptよりもVisual Basicが一般的だからです。しかし、Visual Basicで出来ることは、大抵AppleScriptでもできるのです。今回は、Excelで作成したワークシート上の表をそのままのイメージでQuarkXPressで作成するスクリプトにトライしてみます。

No.	製品名称	価格A	価格B	価格C
1	ASD-500	100,000	¥ 80,000	¥85,000
2	AGF-912	250,000	¥ 200,000	¥212,500
3	CCD-440	185,000	¥ 148,000	¥157,250
4	DGX-50	55,000	¥ 44,000	¥46,750
5	KEY-10	25,000	¥ 20,000	¥21,250
6	LMM-100	154,200	¥ 123,360	¥131,070
7	LMN-200	189,000	¥ 151,200	¥160,650
8	LMN-300	258,000	¥ 206,400	¥219,300
9	REN-400	356,000	¥ 284,800	¥302,600
10	BBV-900	98,000	¥ 78,400	¥83,300
11	BCL-600	123,000	¥ 98,400	¥104,550
12	XYZ-200	298,000	¥ 238,400	¥253,300

スクリプト実行

No.	製品名称	価格A	価格B	価格C
1	ASD-500	100,000	¥ 80,000	¥85,000
2	AGF-912	250,000	¥ 200,000	¥212,500
3	CCD-440	185,000	¥ 148,000	¥157,250
4	DGX-50	55,000	¥ 44,000	¥46,750
5	KEY-10	25,000	¥ 20,000	¥21,250
6	LMM-100	154,200	¥ 123,360	¥131,070
7	LMN-200	189,000	¥ 151,200	¥160,650
8	LMN-300	258,000	¥ 206,400	¥219,300
9	REN-400	356,000	¥ 284,800	¥302,600
10	BBV-900	98,000	¥ 78,400	¥83,300
11	BCL-600	123,000	¥ 98,400	¥104,550
12	XYZ-200	298,000	¥ 238,400	¥253,300

Excelの表をQuarkXPressでどう作成するか
 Excelのワークシートの表を、どのようにQuarkXPressのページ上で作表するかを考えてみましょう。
 テキストボックスの作成方法
 選択したセル範囲の個々のセルごとに、テキストボックスを作成する
 選択したセル範囲を1つのテキストボックスで作成する
 選択したセル範囲の列を1つのテキストボックスで作成する
 罫線の作成方法
 セルの周囲の罫線を、テキストボックスの周囲にラインとして描く
 セルの周囲の罫線の連結状況を読み取り、連結されたラインで描く

セルの幅、高さ、セル内の文字、フォント、フォントサイズ、セル内での文字縦位置と横位置の配置
 以上が最低限必要です。その他、フォントカラー、バックグラウンドカラー、文字飾りなどが考えられますが、今回は省略します。
 罫線を作るために
 上下左右の罫線の種類
 Excelの罫線の指定では、線種だけなので、セルの周囲にどの罫線があるかの情報で十分です。

分かり易さという点では、テキストボックス、罫線とも が良いでしょう。ただし、大きな表ではテキストボックスやラインばかりが増え、動作が重くなります。小さな表向きといえます。
 大きな表の場合は、テキストボックス、罫線とも がいいでしょう。作成するオブジェクト数が最も少なくなるからです。
 表の列幅を変更したり、列の書式を変更することが頻繁に起こるといならば、 が良いでしょう。

QuarkXPressで作成する表のExcelでの指定方法
 作成したい表のワークシートを画面に表示して、QuarkXPressのページ上に作成したい範囲をドラッグして選択しましょう。それから、スクリプトを起動するという手順にします。

作者としては、テキストボックス、罫線とも が気に入っています（処理が早いから）。しかし、今回は「分かり易さ」という点で、テキストボックス、罫線とも の方法を選択します。

表を作成する範囲をドラッグして指定する

Excelのプロパティ、何が必要か？
 では、Excelから何のなプロパティを取得すれば、QuarkXPressで作表できるでしょうか？
 テキストボックスを作るために

Excelでのスクリプティング
 ExcelもQuarkXPressと同じようにオブジェクトの階層構造を持

っています。Workbook, Worksheet, Range, Row, Column, Cell 等です。ここで、Excelのスク립ティングについて詳しく述べるスペースがありませんので割愛します。知りたい方は後述する資料をお読み下さい。

まず、選択したセル範囲のセル属性を取得するスク립トです。

```
tell application "Microsoft Excel"
  set ROW_LIST to Row of Selection
  set COL_LIST to Column of Selection
  set ROW_BEGIN to item 1 of (item 1 of ROW_LIST)
  set ROW_END to item 1 of (item -1 of ROW_LIST)
  set COL_BEGIN to item 1 of (item 1 of COL_LIST)
  set COL_END to item 1 of (item -1 of COL_LIST)
  repeat with R from ROW_BEGIN to ROW_END
    repeat with C from COL_BEGIN to COL_END
      tell ActiveSheet of Workbook 1
        tell Cell C of Row R
          set H_LIST to Height
          set W_LIST to Width
          set TEXT_LIST to every text
          set FONT_LIST to Font of Selection
          set SIZE_LIST to Size
          set H_ALIGN to HorizontalAlignment
          set V_ALIGN to VerticalAlignment
          set B_BRDRS to Bottom Border
          set L_BRDRS to Left Border
          set R_BRDRS to Right Border
          set T_BRDRS to Top Border
        end tell
      end repeat
    end repeat
  end tell
```

セルのプロパティとして、Height (高さ), Width (幅), text (内容), Font (フォント名), Size (フォントサイズ), HorizontalAlignment (横配置), VerticalAlignment (縦配置), BottomBorder (下罫線), LeftBorder (左罫線), RightBorder (右罫線), TopBorder (上罫線) を取得しています。この情報をもとにするわけです。

このスク립トは個々のセルに対して、そのプロパティを得るという方法です。分かり易いのですが、処理スピードが遅いのが欠点です。

処理の早いスク립ト

```
tell application "Microsoft Excel"
  tell Selection
    set H_LIST to Height
    set W_LIST to Width
    set TEXT_LIST to every text
    set FONT_LIST to Font of Selection
    set SIZE_LIST to Size
    set H_ALIGN to HorizontalAlignment
    set V_ALIGN to VerticalAlignment
    set B_BRDRS to Bottom Border
    set L_BRDRS to Left Border
    set R_BRDRS to Right Border
    set T_BRDRS to Top Border
  end tell
end tell
```

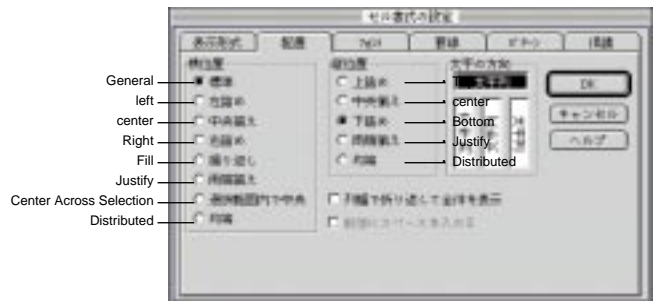
上記のスク립トで行うと、倍の処理速度でプロパティ取得が完了します。

これは、セルごとではなく、選択した範囲のプロパティを得ているためです。処理スピードは重要なので、今回はこの方法で行います。

このスク립トを、先の選択したワークシートで実行すると、変数TEXT_LISTには、{{"No.", "製品名称", "価格A", "価格B", "価格C"}, {"1", "ASD-500", "100,000", "¥¥ 80,000", "¥¥85,000"}, {"2", "AGF-912", "250,000", "¥¥ 200,000", "¥¥212,500"}, {"3", "CCD-440", "185,000", "¥¥ 148,000", "¥¥157,250"}, {"4", "DGX-50", "55,000", "¥¥ 44,000", "¥¥46,750"}, {"5", "KKY-10", "25,000", "¥¥ 20,000", "¥¥21,250"}, {"6", "LMM-100", "154,200", "¥¥ 123,360", "¥¥131,070"}, {"7", "LMN-200", "189,000", "¥¥ 151,200", "¥¥160,650"}, {"8", "LMN-300", "258,000", "¥¥ 206,400", "¥¥219,300"}, {"9", "DKN-400", "356,000", "¥¥ 284,800", "¥¥302,600"}, {"10", "BBV-900", "98,000", "¥¥ 78,400", "¥¥83,300"}, {"11", "JKL-600", "123,000", "¥¥ 98,400", "¥¥104,550"}, {"12", "XYZ-200", "298,000", "¥¥ 238,400", "¥¥253,300"}} が入ります。処理としては、「変数の何番目のリストの何番目のアイテムが処理対象データ」として、各変数からプロパティを取得していくわけです。

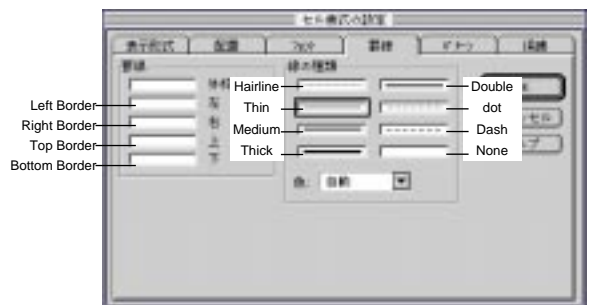
配置と罫線

セルのプロパティで、分かりにくいのが、セル内での文字の横配置を表す HorizontalAlignment と、縦配置を表す VerticalAlignment (縦配置) と、各罫線, BottomBorder (下), LeftBorder (左), RightBorder (右), TopBorder (上) です。



セルの書式設定<配置>画面とそのプロパティ

配置では、標準の場合、文字ならば右寄せ、数字ならば左寄せの考慮が必要になります。また、繰り返し、選択範囲内で中央、均等とは QuarkXPress では出来ませんので、別の行揃えにしましょう。



セルの書式設定<罫線>画面とそのプロパティ

罫線の太さの指定がないため、QuarkXPress で線幅を決める必要があります。

では、これらを考慮して、QuarkXPress側での処理を考えてみます。

まずは、メイン処理です。Excelで取得した大きさのサイズはポイントですから、現在のルーラ単位を一時保存してからポイントに変更します。

QuarkXPressでのスクリプティング

```

tell application "QuarkXPress3.3J"
  activate
  tell document 1
    set OLD_HM to horizontal measure
    set OLD_VM to vertical measure
    set horizontal measure to points
    set vertical measure to points
    set BOX_H1 to 0
    repeat with H from 1 to CNT_ROW
      set BOX_V1 to 0
      set BOX_H0 to BOX_H1
      set BOX_H1 to BOX_H1 + (item 1 of (item H of H_LIST))
      repeat with V from 1 to CNT_CLM
        set BOX_V0 to BOX_V1
        set BOX_V1 to BOX_V1 + (item V of (item H of W_LIST))
        set BOX_CNTNTS to (item V of (item H of TEXT_LIST))
        set BOX_FONT to (item V of (item H of FONT_LIST))
        set BOX_SIZE to (item V of (item H of SIZE_LIST))
        set BOX_HALIGN to (item V of (item H of H_ALIGN))
        set BOX_VALIGN to (item V of (item H of V_ALIGN))
        set BOX_BLINE to (item V of (item H of B_BRDRS))
        set BOX_LLINE to (item V of (item H of L_BRDRS))
        set BOX_RLINE to (item V of (item H of R_BRDRS))
        set BOX_TLINE to (item V of (item H of T_BRDRS))
        my MAKE_TEXTBOX(BOX_H0, BOX_V0, BOX_H1, BOX_V1, BOX_CNTNTS, BOX_FONT, BOX_SIZE, BOX_VALIGN, BOX_HALIGN)
        my MAKE_LINEBOX(BOX_H0, BOX_V0, BOX_H1, BOX_V1, BOX_LLINE, BOX_RLINE, BOX_TLINE)
      end repeat
    end repeat
    set horizontal measure to OLD_HM
    set vertical measure to OLD_VM
  end tell
end tell
  
```

メイン処理

}メジャー単位の保存とポイントへの変換

}テキストボックスとラインボックス作成
サブルーチン

変数から処理対象データを選択

```

on MAKE_TEXTBOX(BOX_H0, BOX_V0, BOX_H1, BOX_V1, BOX_CNTNTS, BOX_FONT, BOX_SIZE, BOX_VALIGN, BOX_HALIGN)
  tell application "QuarkXPress3.3J"
    tell current page of document 1
      make text box at beginning with properties {bounds:{BOX_H0, BOX_V0, BOX_H1, BOX_V1}} }ボックス作成
      tell text box 1
        set story 1 to BOX_CNTNTS }テキスト設定
        set properties of story 1 to {font:{BOX_FONT}, size:{BOX_SIZE}} }フォント・サイズの設定
        if BOX_VALIGN is «constant ****#top» then set vertical justification to top justified
        if BOX_VALIGN is centered then set vertical justification to centered
        if BOX_VALIGN is «constant ****#bot» then set vertical justification to bottom justified
        if BOX_VALIGN is «constant ****#just» then set vertical justification to fully justified
        if BOX_VALIGN is «constant ****#dst» then set vertical justification to fully justified
        if BOX_HALIGN is «constant ****#gnrl» then
          try
            BOX_CNTNTS as number
            set justification of paragraph 1 to right justified
          on error
            set justification of paragraph 1 to left justified
          end try
        end if
        if BOX_HALIGN is left then set justification of paragraph 1 to left justified
        --«constant ****#left» }<注意>参照
        if BOX_HALIGN is centered then set justification of paragraph 1 to centered
        if BOX_HALIGN is «constant ****#right» then set justification of paragraph 1 to right justified
        if BOX_HALIGN is «constant ****#fill» then set justification of paragraph 1 to left justified
        if BOX_HALIGN is «constant ****#just» then set justification of paragraph 1 to left justified
        if BOX_HALIGN is «constant ****#cnsl» then set justification of paragraph 1 to centered
        if BOX_HALIGN is «constant ****#dst» then set justification of paragraph 1 to left justified
      end tell
    end tell
  end MAKE_TEXTBOX
  
```

テキストボックス作成ルーチン

}ボックス内の行位置の設定

}プロパティが"general"の処理

行揃えの設定

そして、先の変数のリスト値から値を一つづつとって、テキストボックス作成ルーチンと、ラインボックス（罫線）作成ルーチンに処理を移します。

テキストボックス作成ルーチン











テキストボックスを作成して、ボックス内の行位置と行揃えをおこないます。

ここで、アレッ！と思われるのが、「constant ****#top」等という記述です。これは、Excelのプロパティの配置縦位置の上詰め（Top）を表しています。Excel内では定数値として持っている値ですが、QuarkXPressでは用意されていない定数値なので、こういう記述になってしまいます。逆に、たまたま用意されていた、leftやcenteredは、「constant ****left」、「constant ****cent」と記入してもコンパイル後にleft、centerと変わってしまうのです。

<注意>このスクリプトは訂正した場合、最後に、leftの表記を「constant ****left」に書き換えてからコンパイルし直して下さい。それを行わないと、leftの部分でエラーとなります。

ラインボックス（罫線）作成ルーチン

ここでも、罫線の種類が、「constant ****thin」等となっています。罫線の幅を変数にセットしています。これを表にあった太さに書き換えて、利用して下さい。罫線の種類を、solid line、double line、dotted line、dashed lineでセットしています。ここも、任意に書き換えてください。ちなみに、QuarkXPressでは、次のようなプロパティが用意されています。

solid line	
dashed line	
densely dashed line	
sparsely dashed line	
dotted line	
double line	
thin thick line	
thick thin line	
thin thick thin line	
thick thin thick line	

thin thin thin line



実行にあたって

QuarkXPressでページのルーラ原点に、表の左上が来るように設定していきますので、実行前に、作成したい場所にルーラの原点を移動してから実行して下さい。

以前、私はフリーウェアとしてExcelからの表コンバートツールを作成しました（「QuarkXPressユーザのためのAppleScript入門」や、P DTP 96/06号に収録されています）が、それは、ExcelのプロパティをVisual Basicで処理し、しかもテキストファイルとして出力した中間ファイルをQuarkXPress側でスクリプトで設定するというものでした。今回は、そのバージョンアップ版という位置づけで、すべてをAppleScriptで行ない、かつ実用に耐える処理速度をもつものです。時間をみて、版を作りたいと考えています。出来次第、CD-ROMに収録したいと思います。

Excelのスクリプティング資料

「Excel5.0 VisualBasicとAppleScript」 新井雅行 著

「快傑AppleScript」MacUser 1997年4・5月号記事 箕原辰夫著

収録スクリプト

Excel Table to QXP

記事内スクリプト。メインルーチンとサブルーチンをまとめ、加筆したものです

Excel Cell属性取得（slow）とExcel Cell属性取得（fast）

記事内スクリプト

Excel5.0サンプルワークシート

```
on MAKE_LINEBOX(BOX_H0, BOX_V0, BOX_H1, BOX_V1, BOX_LLINE, BOX_RLINE, BOX_BLINE, BOX_TLINE)
```

```
set W_THIN to 1
set W_HAIR to 0.5
set W_MED to 1.5
set W_THCK to 2
set W_DBL to 2
set W_DOT to 1
set W_DASH to 1
tell application "QuarkXPress3.3J"
```

罫線の幅を設定

```
tell current page of document 1
```

```
if BOX_LLINE is «constant ****thin» then make line box at beginning with properties {style:solid line, width:W_THIN, bounds:{BOX_H0, BOX_V0, BOX_H1, BOX_V0}}
if BOX_LLINE is «constant ****hair» then make line box at beginning with properties {style:solid line, width:W_HAIR, bounds:{BOX_H0, BOX_V0, BOX_H1, BOX_V0}}
if BOX_LLINE is «constant ****med» then make line box at beginning with properties {style:solid line, width:W_MED, bounds:{BOX_H0, BOX_V0, BOX_H1, BOX_V0}}
if BOX_LLINE is «constant ****thck» then make line box at beginning with properties {style:solid line, width:W_THCK, bounds:{BOX_H0, BOX_V0, BOX_H1, BOX_V0}}
if BOX_LLINE is «constant ****dbl» then make line box at beginning with properties {style:double line, width:W_DBL, bounds:{BOX_H0, BOX_V0, BOX_H1, BOX_V0}}
if BOX_LLINE is «constant ****dot» then make line box at beginning with properties {style:dotted line, width:W_DOT, bounds:{BOX_H0, BOX_V0, BOX_H1, BOX_V0}}
if BOX_LLINE is «constant ****dash» then make line box at beginning with properties {style:dashed line, width:W_DASH, bounds:{BOX_H0, BOX_V0, BOX_H1, BOX_V0}}
```

罫線の作成とプロパティの設定

以下上、下、右の罫線作成のコーディングは省略します

```
end tell
end tell
end MAKE_LINEBOX
```