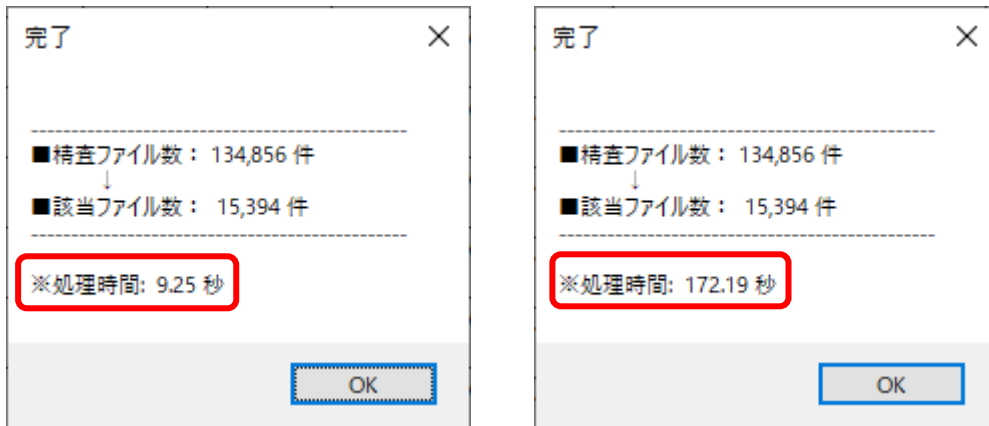


### [01]えっ、何故こんな遅いの？処理速度にこだわる

ExcelVBA ツール開発では、高速化に必要なコーディング技法・作法が存在します。もちろん設計が大切なポイントを占めることにはなりますが、「配列処理」ではなく、「セル操作・シート操作」が大量に繰り返される処理では、実測スピードで、最大 **100** 倍以上の差違が生じてきます。



### [02]えっ、プログラム記述で対応？

営業部門では一般的な「売上集計レポート」。縦計と横計という概念で構成される一覧表のことです。「関数を使用せず」に、プログラム演算結果を出力するようでは、お客様の検証も負荷がかかります、Excel 本来のメリットも生かされません。

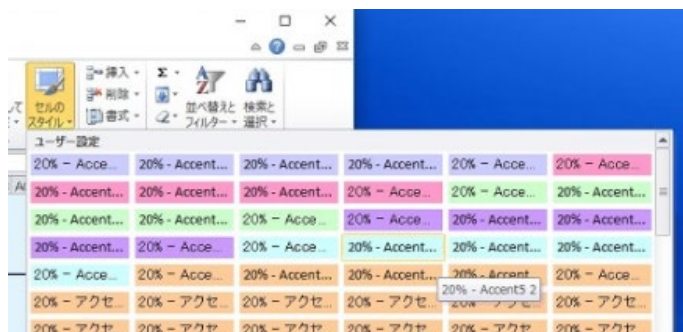
Excel には、便利な関数がたくさん用意されているので、使いこなすことにより、不要なコストを低減することに繋がります。

### [エリア別] 売上集計レポート

エリアCD	エリア名	売上金額
101	北海道・東北	63,823,000
102	関東	128,474,000
103	中部	81,964,000
104	近畿	97,285,000
105	中国・四国	49,853,000
106	九州・沖縄	43,609,000
		465,008,000

### [03]えっ、残額あり。このまま納品？

開発コラム [Excel システム～納品前ブック点検【重要 7 項目】](#)でも紹介していますが、注意が必要な、Excel ブック・シート上にある様々な残額。図形オブジェクト、名前の定義、スタイル書式、クエリ残骸。これらの多くは、お客様自身でも気づかぬまま運用しているケースが大半です。弊社では、開発着手の前に、ブック分析ツールにより、点検のうえ進めて参ります。

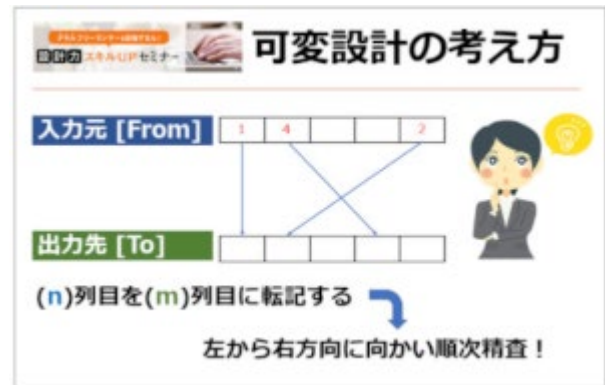


#### [04]えっ、固定でつくったの？

Excel 開発で多く見られる「単純データ転記」という仕様の場合、可変対応という設計が望ましいと考えます。それは、データ転記という点では「何を→どこに」という概念しか存在しないためです。

この為、列番号をお客様が自由に設定できる設計をするだけで、固定設計は可変設計に変わるので

す。基幹系システムから出力される CSV データを使用するケースでは、フォーマット変更はあらかじめ想定範囲内と考えることで、わずかな列の変更などは、プログラム改修が不要になるのです。



#### [05]えっ、列幅自動調整は、大量データに対し厳禁

Excel シート列幅の自動調整ですが、無条件でやってはいけません！大量データの場合、この処理だけで数 10 数秒、スペックの低い PC の場合、

「応答なし」といった事象に見舞われる可能性があります。列幅設定マスター活用により、負荷をかけずに調整したいものです。

```
Sub Macro1()  
    Cells.ColumnWidth = 1.63  
    Cells.EntireColumn.AutoFit  
End Sub
```

#### [06]えっ、何故、プログラムでやるの？

Excel アンケート集計など、円グラフ、棒グラフ、折れ線グラフなどを作成して欲しいとのご依頼が

```
Range("B3:D9").Select  
ActiveSheet.Shapes.AddChart2(201, xlColumnClustered).Select  
ActiveChart.SetSourceData Source:=Range("Sheet1!$B$3:$D$9")  
ActiveSheet.Shapes("グラフ 1").IncrementLeft 64.5  
ActiveSheet.Shapes("グラフ 1").IncrementTop -50.25
```

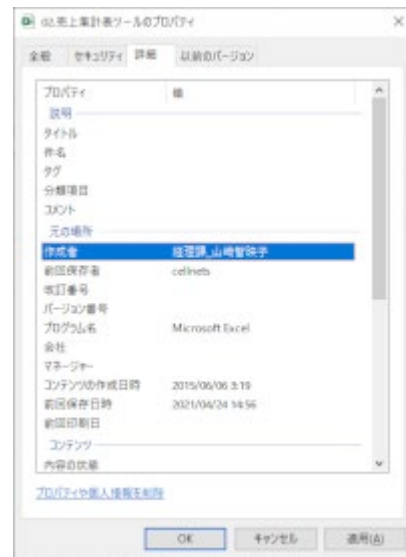
ありますが、プログラミングで生成してはいけません！速度の問題よりも、「メンテナンス性」の問題です。グラフの見た目の色や、フォント、Size など、お客様に自由に設定いただける「ひな型」を活用することで、出力グラフを自由に表現いただくことが可能になります。

#### [07]えっ、セル書式はマクロ記録ではなく書式 Master 活用でしょ

「明細エリア」セル書式の設定ですが、罫線描画や文字サイズ、塗り色など、マクロ記録によるコピペはいただけません。極力、ハードコーディングを避け、書式マスタ適用によるシンプル化が原則。コーディング Step はわずか 1 行です。VBA プログラミング可読性は、後の改修効率を決定づけることとなります。

### [08]えっ、連想配列を使わないの？

弊社のプログラマー採用では、二次元配列活用スキルを必須条件としています。VBAでのシステム開発においてとても重要なテクニックですね。大量のセル操作やシート操作は、速度だけの問題ではなく、「消費メモリ」という面でも大敵です。セル操作しても良いのは、データ件数が少ない場合のみです。



### [09]えっ、個人情報プロパティ削除してない？

残っていたら怖いです。

右クリック> プロパティ(R)> 詳細> 作成者

前任の方のお名前が表示された！？意外と知られていないのですが、要注意ですよ。

弊社では、もちろん基本チェック点検項目です。

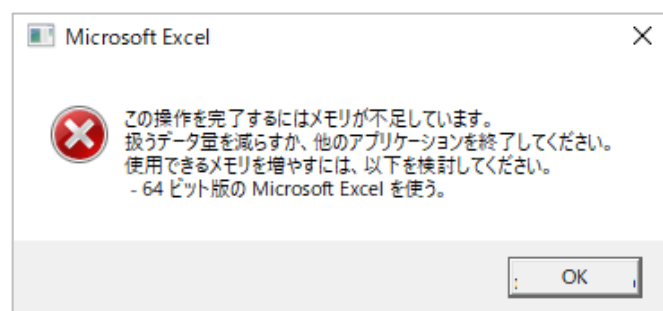
### [10]えっ、罫線描画まさかのマクロ記録！？

罫線描画（四方囲い枠）はマクロ記録の場合、**40** 行程度のステップ数になります。ビジネス文書において、罫線の目的は、読み手にとっての「**見やすさ** & **分かりやすさ**」の配慮ですので、太さや色や種類くらいは、お客様により任意に変更いただける設計が望ましいと考えます。せっかくの Excel 標準機能なのですから。

エリアCD	エリア名	売上金額
101	北海道・東北	63,823,000
102	関東	128,474,000
103	中部	81,964,000
104	近畿	97,285,000
105	中国・四国	49,853,000
106	九州・沖縄	43,609,000
		465,008,000

### [11]えっ、メモリ不足！？ 応答なし

原因は、物理メモリ不足ですが、設計での回避策はあります。ビッグデータであれば配列を垂直分割したり、ThisWorkBook を途中保存したり、ステータスバー描画なら **DoEvents** や **Repaint** など適宜入れるなど…。そもそも、これ



は ExcelVBA という開発言語の問題ではなく、ビッグデータ取り扱いなどの際の「**情報分散処理**」という設計が必要なのです。ビッグデータは珍しくありませんし、弊社でも直近の案件では、「行数 **700,000** 件 × 列数 **3,000** 列」という一般的な業務システムではありえへん！？データを取り扱うこととなりましたが、Excel で開けるファイルではなく、CSV 形式でしたが、横分解での対応で処理は可能になります。

## [12]えっ、シート操作は Object 名が原則

ExcelVBA では、様々なシートに遷移したり、取り扱い方も多岐にわたります。上段は、よく見るワークシートの遷移ですが、内部的にシート名に **Object 名(ws\_51)**を設定することで、少ない文字数でコントロールできます。

```
Worksheets("51売上一覧表").Activate  
ws_51.Activate
```

利点はもう一つ、物理的なシート名が変わってしまっても ("**売上です**")、プログラム処理に影響はありません。これだけで、かなりスッキリしたコーディングになってくるはず。ポイントは、長〜い文字列であるシート名称で管理するのではなく、原則、番号の序列管理ですよね。

## [13]えっ、有効データ範囲の印刷範囲は大丈夫？

「印刷してみたら、何 **10** ページも印刷されてしまった・・・。」これは有効データ範囲の印刷設定のことです。考慮・配慮という点で、『印刷を想定していなかった』は、QA チェック不合格としています。開発技術の前に、ビジネススキル・ビジネスマナーという観点も必要と考えます。

## [14]えっ、そのシート関数バージョン依存は大丈夫？

Excel で使用できる関数は、アプリケーション Version 依存のため意識して活用する必要があります。旧バージョンで動作しない関数は使用してはいけませんし、その逆も然りです。

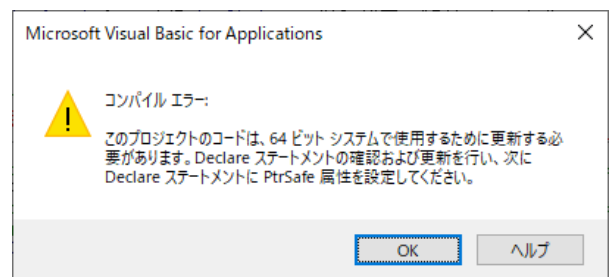
## [15]えっ、実行中 StausBar が無い？

大量データ処理において、全体の**何%**を処理中なのかが不明だと不安になりますよね。また「実行したものの、時間がかかりそうなので、いったん中断したい。」そんな時は、「中断機能」が必要です。その手法として、弊社では、ステータスバーではなく UserForm を活用します。



## [16]えっ、Excel エディション 64 ビット非対応？

昨年迄は **32** ビットが Excel 標準でしたが、Office365 普及に伴い、現在では **64** ビットが Excel インストール時の初期値となりました。API 組み込みにより、**32/64** ビット両対応が図れます。



### [17]えっ、Excel ネットワーク非対応？

これは致命的かもしれません。ローカル環境だけのファイル入出力テストの場合に見受けられます。APIを組み込むことで参照できますが組み込まないと参照できません。

### [18]えっ、Excel は「排他制御」には非対応？

そんなことはありません。ちゃんとできます。更新先の Excel ファイルが「**読取り専用モード**」だった場合、誰かが開いているという事です。ところが、この判定だけでは非常に危険です。何故なら、重い Excel ファイルを開くのに [30 秒] かかる場合、開き終わるまで誰かが開きに行った事実を検知できない場合があるからです。

このケースではシンプル Text で[宣言ファイル.txt]を作成するなど、誰かが更新を試みにいった、その事実を検知する必要があります。それでも駄目な場合は、実行時エラー発生の瞬間をもって更新失敗と判断し、リトライを促すなどの設計により、排他制御が可能です。

### [19]えっ、Excel 旧ファイル形式(\*.xls)は非対応？

Excel2007 発売当時、Excel ファイルの拡張子が新形式(\*.xlsx)に変わりました。

現在、2003 アプリケーションはサポート対象外ですが、旧ファイル形式(\*.xls)には害が無いので、読み込む事が出来ない設計では対応データ範囲が制限されてしまいます。

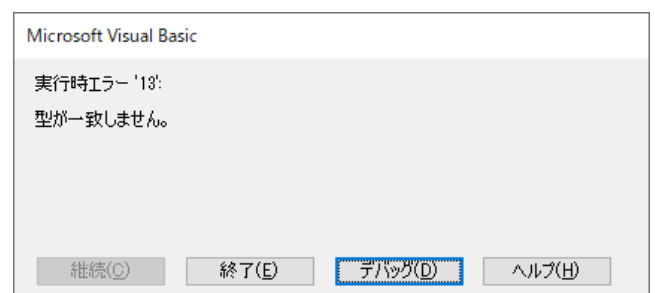
### [20]えっ、ExcelVBA で自動バックアップ機能？

弊社では、外部 DB に Access データベース(accdb)を活用する場合、万一のデータ消失に備え、Excel システム起動時に「**自動世代バックアップ**」機能を実装するなど、万全を期しています。「データ最適化」も忘れずに。



### [21]えっ、UserForm 日付欄に「あいうえお」入力で実行時エラー！？

ユーザーフォームの全フィールド属性についてテストが不十分のための障害で、初心者をはじめ、大ベテランでも考慮のない場合があります。これは、入力値の妥当性チェックが省略されたための不具合ですので、意地悪チェックというテストですぐに発覚する事象です。



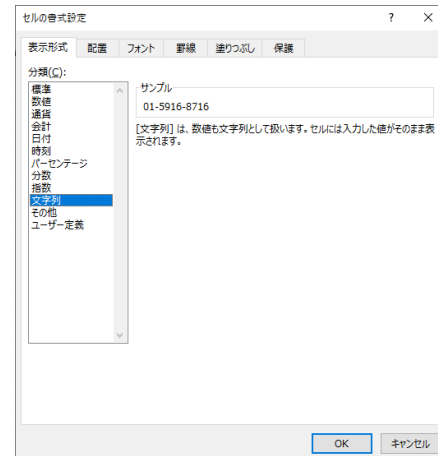
## [22]えっ、マクロボタンが反応しない？

が原因なら、システムを過去日付に戻し、暫定対応いただけます。原因は様々ですが、Excel 専門特化の弊社スタッフが親身に原因究明にあたります。



## [23]えっ、電話番号のゼロ"0"が欠落！？

Excel セル書式には、計 12 種類の属性があり、規定値は[標準]です。これ以外に、代表的なものとして、[数値]、[日付]、[文字列]がありますが、電話番号や郵便番号、場合によっては、0 から始まる可能性がある場合は、原則、[標準]ではなく[文字列]が正解です。[文字列]書式を適用することにより、前ゼロが欠落することは回避できます。



弊社のコード設計が、数値から開始しないようにしているの

は、このためです。JAN コードなど桁数が多い数値の場合、「指数表示」となりますので、回避するためには、予め文字列書式にした後で、値を反映する必要があります。既に入力済セルに書式適用しても適用されないため、プログラム処理が必要な場合があります。

## [24]えっ、PGM ハードコーディング？

データ抽出条件の指定など、リスト選択肢において、改修が想定される要素については、プログラム内での記述は NG です。複雑な計算式などもパターン化～可視化が原則。不備なきよう作ることは当然ですが、不備があったときの「気づく仕組み」も忘れてはいけません。

【比較演算子】=、≠、<、>、> =、< =

【ランク】A、B、C

【クラス】さくら、うめ、たんぽぽ

1、要素を増やしたい 2、要素を減らしたい 3、順番を入れ替えたい

などは、設計段階で想定されるため、「選択肢マスター」での Excel シート管理が基本です。これは、可変対応を心がける Excel システム設計という面で、大切な配慮です。