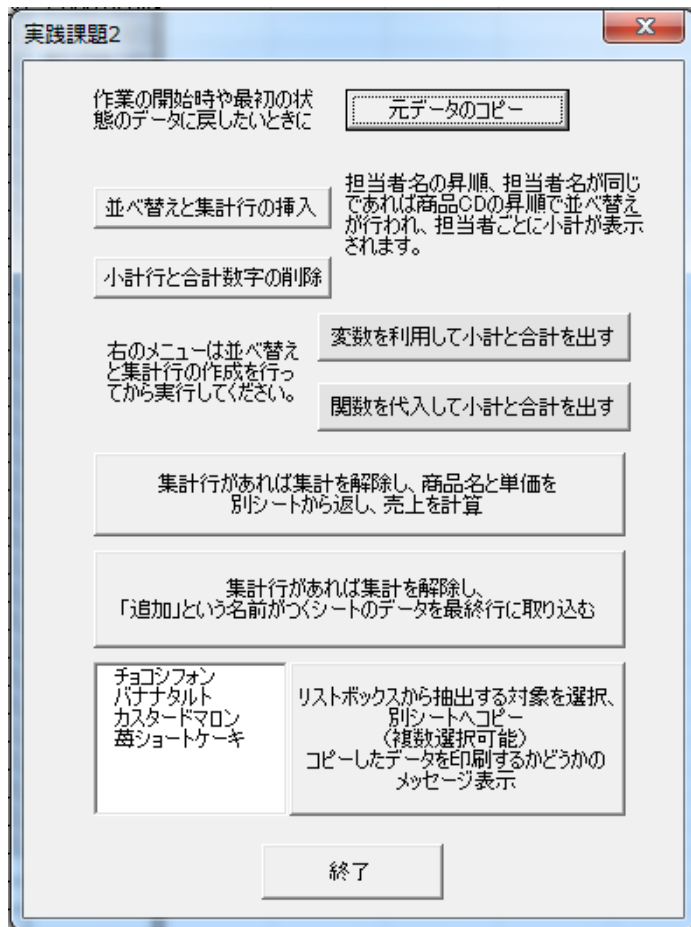


■ 制作課題2 使用ブック:Practice2-2.xlsx

集計を主な目的としたプログラムを作成しましょう。「Practice2_2ans.xlsm」の処理の動きを確認しながら、システムを行きましょう。

▶ 完成例



■ HINT - 考え方のフロー

■ 共通のプロシージャは標準モジュールに作成

- ▶ シートの存在をチェックする
Function シート存在チェック (wsname As string) As Boolean
- ▶ 集計が行われているかどうかのチェック
Function 集計されているかチェック () As Boolean
- ▶ 罫線を整える
Sub 罫線を整える
- ▶ ユーザーフォームの表示
Sub ユーザーフォーム表示

■ 共通の変数は宣言セクションで宣言

- ▶ ワークシート作業で利用するワークシート (wsname)
- ▶ 抽出結果で利用するワークシート (wsname2)
- ▶ 行番号を代入するための変数 (iRow)
- ▶ 列番号を代入するための変数 (iCol)
- ▶ 繰り返し処理に利用する変数 (i)

Memo

解説と解答は一例です。
プログラムの流れに沿って記述できれば、テキストの回答例に合わせて作らなくても問題はありません。

Point

▶ **プロシージャの呼び出し**
シートの存在と集計が行われているかをチェックした結果は、呼び出し元のプロシージャで利用しますので、Function プロシージャ、罫線を整える作業は、プロシージャ内で処理が完了しますので、Sub プロシージャで作成します。

Memo

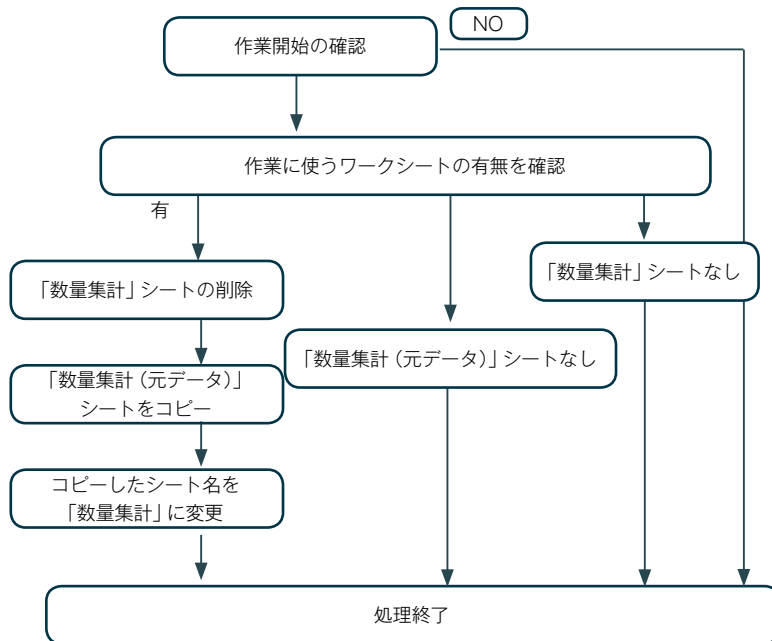
プロシージャ名は任意の名前でかまいません。
共通の変数は最初の段階で宣言しておくとうわりやすいです。
行番号や列番号は変数で取らなくても、プログラム実行時に代入する方法を利用してもかまいません。
(解答ブックは後者で作成されています。)

■ 作成の流れ

- Step 1** ▶ ユーザーフォームを作成します。
リストボックスの値は複数選択可能にします。
オブジェクト名はプロシージャを割り当てる際にわかりやすい名前にします。
- Step 2** ▶ ブックを開くと同時に、ユーザーフォームを表示するプロシージャを作成します。なお、作業中でもワークシートの作業ができる状態で開きます。
- Step 3** ▶ [終了] のコマンドボタンにユーザーフォームを閉じるプロシージャを作成します。
- Step 4** ▶ HINT を参考にしながらそれぞれのコマンドボタンに処理を記述します。
- Step 5** ▶ さまざまなケースを想定して動作をテストします。

■ HINT- [元データのコピー] ボタンをクリックしたときの処理

処理の流れを整理します



■ 作業開始の確認

- ① 「処理を行っても良いかどうか」のメッセージボックスを表示します。



- ② 「いいえ」を選択すると次図のメッセージボックスを表示して処理を中止します。



Point

▶ユーザーフォームを開く

ThisWorkbook モジュールに「Workbook_Open ()」を記述します。ThisWorkbook の Workbook オブジェクトの Open イベントを選択し、コードを記述します。

記述例

```
ユーザーフォーム名 .show
```

Point

▶vbModeless

Show メソッドの引数に指定すると、ユーザーフォームを表示しながらアプリケーションの他の部分の操作を行うことができるモードレスな状態になります。

Memo

処理の流れがつかみづらい場合は、完成例を実行して確認しましょう。

Point

▶ユーザーフォームを閉じる

記述例

```
Unload.Me
```

Point

▶MsgBox関数

引数 buttons でボタンとアイコンの種類を指定します。

- ボタンとアイコンを組み合わせる場合は定数を足し算した式を引数に指定します。

例) [はい] ボタンと [いいえ] ボタン (定数 4) と情報メッセージアイコン (定数 64) を同時に表示する場合には「4+64」と指定します。

- ②では情報メッセージアイコンと、[OK] ボタンが表示されていますが、[OK] ボタンは既定のボタンなので、[vbInformation] または [64] を指定します。