

# Microsoft® Excel® マクロ/VBA

Office 2021/2019/2016/Microsoft 365 対応

## 練習問題・総合問題 標準解答


練習問題 標準解答 .....	1
総合問題 標準解答 .....	10

# 練習問題 標準解答

## 第1章 練習問題

①

①《開発》タブを選択

②《コード》グループの  (マクロの記録) をクリック

③《マクロ名》に「集計削除」と入力

④《マクロの保存先》の  をクリックし、一覧から《作業中のブック》を選択



⑤《OK》をクリック

⑥ セル【B4】をクリック

※表内のセルであれば、どこでもかまいません。

⑦《データ》タブを選択

⑧《アウトライン》グループの  (小計) をクリック


※《アウトライン》グループが  (アウトライン) で表示されている場合は、 (アウトライン) をクリックすると、  
《アウトライン》グループのボタンが表示されます。


⑨《すべて削除》をクリック

⑩ セル【B4】をクリック

※表内のセルであれば、どこでもかまいません。

「表内のセルをクリックする」という操作を記録するため、表内のセルを再度クリックします。

⑪《並べ替えとフィルター》グループの  (並べ替え) をクリック

⑫《先頭行をデータの見出しとして使用する》を  にする

⑬《並べ替え》ダイアログボックスの《最優先されるキー》の《列》の一覧から「No.」を選択

⑭ 2021/2019/365

《並べ替えのキー》が《セルの値》になっていることを確認

2016

《並べ替えのキー》が《値》になっていることを確認

⑮ 2021/2019/365

《順序》の一覧から《小さい順》を選択

2016

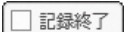
《順序》の一覧から《昇順》を選択

⑯《OK》をクリック

⑰ セル【A1】をクリック

※アクティブセルをホームポジションに戻します。

⑱《開発》タブを選択

⑲《コード》グループの  (記録終了) をクリック

## 第5章 練習問題

### 練習問題1

①

- ① VBEを起動
- ② メニューの《挿入》をクリック
- ③ 《標準モジュール》をクリック
- ④ モジュール「Module1」に次のプロシーダを入力

#### ■「練習1」プロシーダ

```
1. Sub 練習1 ()
2.     Dim kyuyo As Currency
3.     If Range ("D10").Value = "" Then
4.         Range ("E10").Value = "休み"
5.     Else
6.         If Range ("C10").Value = "平日" Then
7.             kyuyo = 1100 * Range ("D10").Value
8.             Range ("E10").Value = kyuyo
9.         ElseIf Range ("C10").Value = "休日" Then
10.            kyuyo = 1200 * Range ("D10").Value
11.            Range ("E10").Value = kyuyo
12.        End If
13.    End If
14. End Sub
```







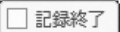
#### ■プロシーダの意味

1. 「練習1」プロシーダ開始
2. 通貨型の変数「kyuyo」を使用することを宣言
3. セル【D10】が空白の場合は
4. セル【E10】に「休み」と代入
5. それ以外の場合は、9行目の条件で処理を分岐する
6. セル【C10】が「平日」の場合は
7. 変数「kyuyo」に「1100×セル【D10】」の結果を代入
8. セル【E10】に変数「kyuyo」の値を代入
9. セル【C10】が「休日」の場合は
10. 変数「kyuyo」に「1200×セル【D10】」の結果を代入
11. セル【E10】に変数「kyuyo」の値を代入
12. 6行目からのIfステートメント終了
13. 3行目からのIfステートメント終了
14. プロシーダ終了

- ⑤ Excellに切り替えて、シート「練習問題1」の「練習1」ボタンをクリック

## 練習問題2

①

- ① Excelの《開発》タブを選択
- ② 《コード》グループの  (マクロの記録) をクリック
- ③ 《マクロ名》に「練習2\_1」と入力
- ④ 《マクロの保存先》の  をクリックし、一覧から《作業中のブック》を選択
- ⑤ 《OK》をクリック
- ⑥ セル範囲【B11:E16】を選択
- ⑦ 《挿入》タブを選択
- ⑧ 《グラフ》グループの  (縦棒/横棒グラフの挿入) をクリック
- ⑨ 《2-D縦棒》の《集合縦棒》をクリック
- ⑩ 作成したグラフが選択されていることを確認
- ⑪ 《グラフのデザイン》タブを選択
- ※お使いの環境によっては、《デザイン》タブと表示される場合があります。
- ⑫ 《種類》グループの  (グラフの種類の変更) をクリック
- ⑬ 《すべてのグラフ》タブを選択
- ⑭ 左側の一覧から《横棒》を選択
- ⑮ 右側の一覧から左側の《集合横棒》を選択
- ⑯ 《OK》をクリック
- ⑰ 《場所》グループの  (グラフの移動) をクリック
- ⑱ 《新しいシート》を  にし、「グラフ」と入力
- ⑲ 《OK》をクリック
- ⑳ 《開発》タブを選択
- ㉑ 《コード》グループの  (記録終了) をクリック

②

- ① VBEに切り替える
- ② プロジェクトエクスプローラーのモジュール「Module2」をダブルクリック
- ③ 次のようにプロシージャを編集

## ■「練習2\_1」プロシージャ

```
1. Sub 練習2_1 ()
2.     Dim syurui As Integer
3.     Select Case Range ("G5").Value
4.         Case "縦棒"
5.             syurui = xlColumnClustered
6.         Case "横棒"
7.             syurui = xlBarClustered
8.         Case "折れ線"
9.             syurui = xlLine
10.        Case "面"
11.            syurui = xlArea
12.        Case Else
13.            MsgBox "正しいグラフ名を入力してください"
14.        Exit Sub
15.    End Select
16.    Range ("B11:E16").Select
17.    ActiveSheet.Shapes.AddChart2 (201, xlColumnClustered).Select
18.    ActiveChart.SetSourceData Source:=Range ("練習問題2!$B$11:$E$16")
19.    ActiveChart.ChartType = syurui
20.    ActiveChart.Location Where:=xlLocationAsNewSheet, Name:="グラフ"
21. End Sub
```

## ■プロシージャの意味

```
1. 「練習2_1」プロシージャ開始
2.     整数型の変数「syurui」を使用することを宣言
3.     セル【G5】の値が
4.         「縦棒」の場合は
5.             変数「syurui」に「xlColumnClustered」を代入
6.         「横棒」の場合は
7.             変数「syurui」に「xlBarClustered」を代入
8.         「折れ線」の場合は
9.             変数「syurui」に「xlLine」を代入
10.        「面」の場合は
11.            変数「syurui」に「xlArea」を代入
12.        それ以外の場合は
13.            「正しいグラフ名を入力してください」のメッセージを表示
14.        Subプロシージャを抜ける
15.    Select Caseステートメント終了
16.
17.
18. } グラフの追加と移動   グラフの種類には変数「syurui」に代入されている値を設定
19.
20.
21. プロシージャ終了
```

④ Excelに切り替えて、グラフシート「グラフ」を削除

⑤ シート「練習問題2」の「練習2\_1」ボタンをクリック

③

- ① VBEに切り替える
- ② 「End Sub」の下の方にカーソルを移動
- ③ 次のプロシーダを入力

#### ■「練習2\_2」プロシーダ

1. Sub 練習2\_2 ()
2.     Sheets ("グラフ").Delete
3. End Sub

#### ■プロシーダの意味

1. 「練習2\_2」プロシーダ開始
2.     グラフシート「グラフ」を削除
3. プロシーダ終了

- ④ Excelに切り替えて、シート「練習問題2」の「練習2\_2」ボタンをクリック
- ⑤ 《削除》をクリック

## 練習問題3

①

- ① VBEに切り替える
- ② メニューの《挿入》をクリック
- ③ 《標準モジュール》をクリック
- ④ モジュール「Module3」に次のプロシーダを入力

#### ■「練習3」プロシーダ

1. Sub 練習3 ()
2.     Dim i As Integer
3.     Range ("E6").Select
4.     For i = 1 To 5
5.         If ActiveCell.Offset (0, -1).Value >= 80 Then
6.             ActiveCell.Value = "合格"
7.         Else
8.             ActiveCell.Value = "不合格"
9.         End If
10.        ActiveCell.Offset (1, 0).Select
11.     Next
12. End Sub

### ■プロシージャの意味

1. 「練習3」プロシージャ開始
2. 整数型の変数「i」を使用することを宣言
3. セル【E6】を選択
4. 変数「i」が「1」から「5」になるまで処理を繰り返す
5. アクティブセルの1列左のセルの値が「80」以上の場合は
6. アクティブセルに「合格」を代入
7. それ以外の場合は
8. アクティブセルに「不合格」を代入
9. Ifステートメント終了
10. 1行下のセルを選択
11. 変数「i」に変数「i」+1の結果を代入し、4行目に戻る
12. プロシージャ終了

- ⑤ Excellに切り替えて、シート「練習問題3」の「練習3」ボタンをクリック

## 練習問題4

①

- ① VBEに切り替える
- ② メニューの《挿入》をクリック
- ③ 《標準モジュール》をクリック
- ④ モジュール「Module4」に次のプロシージャを入力

### ■「練習4」プロシージャ

1. Sub 練習4 ()
2. Dim total As Long
3. Range ("C6").Select
4. Do While ActiveCell.Value <> ""
5. total = total + ActiveCell.Value
6. ActiveCell.Offset (0, 1).Value = total
7. ActiveCell.Offset (1, 0).Select
8. Loop
9. End Sub

### ■プロシージャの意味

1. 「練習4」プロシージャ開始
2. 長整数型の変数「total」を使用することを宣言
3. セル【C6】を選択
4. アクティブセルが空白でない間は
5. 変数「total」に変数「total」+アクティブセルの値の結果を代入
6. アクティブセルの1列右のセルに変数「total」の値を代入
7. アクティブセルの1行下のセルに移動
8. 4行目に戻る
9. プロシージャ終了

- ⑤ Excellに切り替えて、シート「練習問題4」の「練習4」ボタンをクリック

### 練習問題1

①

- ① VBEを起動
- ② プロジェクトエクスプローラーの《標準モジュール》をダブルクリック
- ③ モジュール「アンケート」をダブルクリック
- ④ 「End Sub」の下の方にカーソルを移動
- ⑤ 次のプロシーダを入力

#### ■「検索」プロシーダ

```
1. Sub 検索 ()
2.     Dim koumoku As String
3.     Dim jyouken As String
4.     Dim retu As Byte
5.     koumoku = InputBox ("検索する項目を番号で選択してください" & Chr(10) & _
6.         "1.性別" & Chr(10) & "2.職業" & Chr(10) & "3.価格" & Chr(10) & _
7.         "4.飲みやすさ" & Chr(10) & "5.購入予定", "検索")
8.     Select Case koumoku
9.         Case 1
10.             jyouken = InputBox ("性別を入力してください", "性別")
11.             retu = 2
12.         Case 2
13.             jyouken = InputBox ("職業を入力してください", "職業")
14.             retu = 3
15.         Case 3
16.             jyouken = InputBox ("価格を入力してください", "価格")
17.             retu = 4
18.         Case 4
19.             jyouken = InputBox ("飲みやすさを入力してください", "飲みやすさ")
20.             retu = 5
21.         Case 5
22.             jyouken = InputBox ("購入予定を入力してください", "購入予定")
23.             retu = 6
24.         Case Else
25.             MsgBox "入力したデータが間違っています"
26.     End Select
27.     If jyouken <> "" Then
28.         Range ("B4").Select
29.         Selection.AutoFilter retu, jyouken
30.     End If
31. End Sub
```

※5～7行目はコードが長いので、行継続文字「\_ (半角スペース+半角アンダースコア)」を使って行を複数に分割しています。行継続文字を使わずに1行で記述してもかまいません。

※「&」の前のスペースは直接入力します。



## ■プロシージャの意味

1. 「検索」プロシージャ開始
2. 文字列型の変数「koumoku」を使用することを宣言
3. 文字列型の変数「jyouken」を使用することを宣言
4. バイト型の変数「retu」を使用することを宣言
5. タイトルバー「検索」、メッセージ「検索する項目を番号で選択してください (改行)
6. 1.性別(改行)2.職業(改行)3.価格(改行)
7. 4.飲みやすさ(改行)5.購入予定」とテキストボックスを表示し、入力された値を変数「koumoku」に代入
8. 変数「koumoku」が
9. 「1」の場合は
10. タイトルバー「性別」、メッセージ「性別を入力してください」とテキストボックスを表示し、入力された値を変数「jyouken」に代入
11. 変数「retu」に「2」を代入
12. 「2」の場合は
13. タイトルバー「職業」、メッセージ「職業を入力してください」とテキストボックスを表示し、入力された値を変数「jyouken」に代入
14. 変数「retu」に「3」を代入
15. 「3」の場合は
16. タイトルバー「価格」、メッセージ「価格を入力してください」とテキストボックスを表示し、入力された値を変数「jyouken」に代入
17. 変数「retu」に「4」を代入
18. 「4」の場合は
19. タイトルバー「飲みやすさ」、メッセージ「飲みやすさを入力してください」とテキストボックスを表示し、入力された値を変数「jyouken」に代入
20. 変数「retu」に「5」を代入
21. 「5」の場合は
22. タイトルバー「購入予定」、メッセージ「購入予定を入力してください」とテキストボックスを表示し、入力された値を変数「jyouken」に代入
23. 変数「retu」に「6」を代入
24. それ以外の場合は
25. 「入力したデータが間違っています」のメッセージを表示
26. Select Caseステートメント終了
27. 変数「jyouken」が空白以外の場合は
28. セル【B4】を選択
29. 変数「retu」の列を変数「jyouken」を条件として抽出
30. Ifステートメント終了
31. プロシージャ終了

⑥ Excellに切り替えて、シート「アンケート」の「検索」ボタンをクリック

## 練習問題2

①

- ① VBEに切り替える
- ② プロジェクトエクスプローラーのモジュール「お客様リスト」をダブルクリック
- ③ 次のプロシージャを入力



### ■「割引後料金」プロシージャ

```
1. Function 割引後料金(継続月数, 通常料金)
2.     Select Case 継続月数
3.         Case Is >= 36
4.             割引後料金 = 通常料金 - Int(通常料金 * 0.05)
5.         Case Is >= 24
6.             割引後料金 = 通常料金 - Int(通常料金 * 0.03)
7.         Case Is >= 12
8.             割引後料金 = 通常料金 - Int(通常料金 * 0.02)
9.         Case Else
10.            割引後料金 = 通常料金
11.     End Select
12. End Function
```

### ■プロシージャの意味

```
1. 「割引後料金」プロシージャ開始(引数に「継続月数」と「通常料金」を指定)
2.     「継続月数」が
3.         「36」以上の場合は
4.             「通常料金 - (通常料金 × 0.05)の小数点以下を切り捨てた値」の結果を「割引後料金」に代入
5.         「24」以上の場合は
6.             「通常料金 - (通常料金 × 0.03)の小数点以下を切り捨てた値」の結果を「割引後料金」に代入
7.         「12」以上の場合は
8.             「通常料金 - (通常料金 × 0.02)の小数点以下を切り捨てた値」の結果を「割引後料金」に代入
9.         それ以外の場合は
10.            「通常料金」を「割引後料金」に代入
11.     Select Caseステートメント終了
12. プロシージャ終了
```

②

- ① Excellに切り替える
- ② シート「お客様リスト」のセル【I7】をクリック
- ③  (関数の挿入) をクリック
- ④ 《関数の分類》の  をクリックし、一覧から《ユーザー定義》を選択
- ⑤ 《関数名》の一覧から「割引後料金」を選択
- ⑥ 《OK》をクリック
- ⑦ 《継続月数》に「E7」と入力
- ⑧ 《通常料金》に「H7」と入力
- ⑨ 《OK》をクリック
- ⑩ セル【I7】をセル範囲【I8:I27】にコピー

# 総合問題 標準解答

## 総合問題 1

①



- ① VBEを起動
- ② 《挿入》をクリック
- ③ 《標準モジュール》をクリック
- ④ モジュール「Module1」に次のプロシージャを入力

### ■「第1四半期へ」プロシージャ

1. Sub 第1四半期へ ()
2.     Worksheets ("第1四半期") .Select
3. End Sub

### ■プロシージャの意味

1. 「第1四半期へ」プロシージャ開始
2.     シート「第1四半期」を選択
3. プロシージャ終了

- ⑤ 同様に、「第2四半期へ」、「上半期計へ」、「メニューへ」プロシージャを入力
- ⑥ Excellに切り替える
- ⑦ シート「メニュー」を選択
- ⑧ 《開発》タブを選択
- ⑨ 《コントロール》グループの  (コントロールの挿入) をクリック
- ⑩ 《フォームコントロール》の  (ボタン (フォームコントロール)) をクリック
- ⑪ 任意の大きさにドラッグ
- ⑫ 《マクロ名》の一覧から「第1四半期へ」を選択
- ⑬ 《OK》をクリック
- ⑭ ボタンの表示名を「第1四半期」に変更
- ⑮ 同様に、「第2四半期」ボタン、「上半期計」ボタンを作成し、プロシージャを登録
- ⑯ シート「第1四半期」、「第2四半期」、「上半期計」に「メニュー」ボタンを作成し、プロシージャを登録

## ②



- ① VBEに切り替える
- ② モジュール「Module1」が表示されていることを確認
- ③ 最終行の「End Sub」の下の方にカーソルを移動
- ④ 次のプロシージャを入力

### ■「印刷プレビュー」プロシージャ





1. Sub 印刷プレビュー ()
2.     ActiveSheet.PrintPreview
3. End Sub



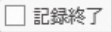


### ■プロシージャの意味

1. 「印刷プレビュー」プロシージャ開始
2.     アクティブシートを印刷プレビューで表示
3. プロシージャ終了

- ⑤ Excellに切り替える
- ⑥ シート「上半期計」を選択
- ⑦ 《開発》タブを選択
- ⑧ 《コントロール》グループの  (コントロールの挿入) をクリック
- ⑨ 《フォームコントロール》の  (ボタン (フォームコントロール)) をクリック
- ⑩ 任意の大きさにドラッグ
- ⑪ 《マクロ名》の一覧から「印刷プレビュー」を選択
- ⑫ 《OK》をクリック
- ⑬ ボタンの表示名を「印刷プレビュー」に変更

## ③

- ① シート「上半期計」を選択
- ② 《開発》タブを選択
- ③ 《コード》グループの  マクロの記録 (マクロの記録) をクリック
- ④ 《マクロ名》に「トップ3」と入力
- ⑤ 《マクロの保存先》の  をクリックし、一覧から《作業中のブック》を選択
- ⑥ 《OK》をクリック
- ⑦ セル【B4】をクリック  
※表内のセルであれば、どこでもかまいません。
- ⑧ 《データ》タブを選択
- ⑨ 《並べ替えとフィルター》グループの  (フィルター) をクリック
- ⑩ 「上半期計」フィールドの  をクリック
- ⑪ 《数値フィルター》をポイント
- ⑫ 《トップテン》をクリック
- ⑬ 《上位》《3》《項目》に設定
- ⑭ 《OK》をクリック
- ⑮ セル範囲【C4:C18】を選択

- ⑫  を押しながら、セル範囲【E4:E18】を選択
- ⑬ 《挿入》タブを選択
- ⑭ 《グラフ》グループの  (縦棒/横棒グラフの挿入) をクリック
- ⑮ 《2-D縦棒》の《集合縦棒》をクリック
- ⑯ 《開発》タブを選択
- ⑰ 《コード》グループの  (記録終了) をクリック
- ⑱ 《コントロール》グループの  (コントロールの挿入) をクリック
- ㉑ 《フォームコントロール》の  (ボタン (フォームコントロール)) をクリック
- ㉒ 任意の大きさにドラッグ
- ㉓ 《マクロ名》の一覧から「トップ3」を選択
- ㉔ 《OK》をクリック
- ㉕ ボタンの表示名を「グラフ」に変更

#### ④



- ① VBEに切り替える
- ② プロジェクトエクスプローラーのモジュール「Module2」をダブルクリック
- ③ 「End Sub」の下の方にカーソルが表示されていることを確認
- ④ 次のプロシージャを入力

##### ■「グラフ削除」プロシージャ

1. Sub グラフ削除 ()
2.     ActiveSheet.ChartObjects.Delete
3.     Range ("B4").Select
4.     Selection.AutoFilter
5.     Range ("A1").Select
6. End Sub

##### ■プロシージャの意味

1. 「グラフ削除」プロシージャ開始
2.     アクティブシートのグラフを削除
3.     セル【B4】を選択
4.     フィルターを解除
5.     セル【A1】を選択
6. プロシージャ終了

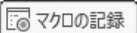
- ⑤ Excellに切り替える
- ⑥ 《開発》タブを選択
- ⑦ 《コントロール》グループの  (コントロールの挿入) をクリック
- ⑧ 《フォームコントロール》の  (ボタン (フォームコントロール)) をクリック
- ⑨ 任意の大きさにドラッグ
- ⑩ 《マクロ名》の一覧から「グラフ削除」を選択
- ⑪ 《OK》をクリック
- ⑫ ボタンの表示名を「グラフ削除」に変更

## 総合問題 2

①

① セル【B5】をクリック



② 《開発》タブを選択

③ 《コード》グループの  (マクロの記録) をクリック


④ 《マクロ名》に「日報」と入力


⑤ 《マクロの保存先》の  をクリックし、一覧から《作業中のブック》を選択

⑥ 《OK》をクリック

⑦ 《コード》グループの  (相対参照で記録) をオン (濃い灰色の状態) にする  
※ 選択するセルの位置を相対的に記録するため、 (相対参照で記録) をオンにします。

⑧ セル【B5】に「6/1」と入力

⑨  を2回押す

※  を押すと、アクティブセルが下に移動するので注意しましょう。

⑩ セル【D5】に「晴れ」と入力

⑪  を押す

⑫ セル【E5】に「512」と入力


⑬  を押す

⑭ セル【F5】に「921600」と入力

⑮  を押す

⑯ セル【G5】に「松岡」と入力

⑰  を押す

⑱  を5回押す

⑲ 《コード》グループの  (記録終了) をクリック

⑳ VBEを起動

㉑ プロジェクトエクスプローラーの《標準モジュール》をダブルクリック

㉒ モジュール「Module1」をダブルクリック

㉓ プロシージャを次のように編集

## ■「日報」プロシージャ

```
1. Sub 日報()  
2.     If Range("B5").Value = "" Then  
3.         Range("B5").Select  
4.     Else  
5.         Range("B4").Select  
6.         Selection.End(xlDown).Select  
7.         ActiveCell.Offset(1, 0).Select  
8.     End If  
9.     ActiveCell.Value = InputBox("今日の日付を入力してください", "日報", , 200, 200)  
10.    ActiveCell.Offset(0, 2).Select  
11.    ActiveCell.Value = InputBox("天候を入力してください", "日報", , 200, 200)  
12.    ActiveCell.Offset(0, 1).Select  
13.    ActiveCell.Value = InputBox("来場者数を入力してください", "日報", , 200, 200)  
14.    ActiveCell.Offset(0, 1).Select  
15.    ActiveCell.Value = InputBox("売上金額を入力してください", "日報", , 200, 200)  
16.    ActiveCell.Offset(0, 1).Select  
17.    ActiveCell.Value = InputBox("担当者名を入力してください", "日報", , 200, 200)  
18.    ActiveCell.Offset(1, -5).Select  
19. End Sub
```

※「FormulaR1C1」は「R1C1形式」の数式を受け取るプロパティです。マクロで自動的に記述されたコードは、行（Row）と列（Column）を相対的な位置関係で記録するため、FormulaR1C1プロパティが使われます。本書では、FormulaR1C1プロパティの代わりにオブジェクトの値を代入するValueプロパティを使います。



※「PhoneticCharacters」プロパティは指定した文字列のフリガナの情報を返すためのコードで、マクロで自動的に記述されます。

※Offsetプロパティに続く「Range("A1")」は、Offsetプロパティによる参照先のセルをセル【A1】（左上端）とみなすためのコードで、マクロで自動的に記述されます。



10行目の「ActiveCell.Offset(0,2).Range("A1").Select」は「アクティブセルを基準に、2列右のセルを基準セル【A1】として、そのセル【A1】を選択」という意味になります。

VBAで記述する場合は「Range("A1")」を省略できるため、本書では省略しています。

## ■プロシージャの意味

1. 「日報」プロシージャ開始
2. セル【B5】が空白の場合は
3. セル【B5】を選択
4. それ以外の場合は
5. セル【B4】を選択
6.  +  でデータの下端のセルを選択
7. アクティブセルの1行下のセルを選択
8. Ifステートメント終了
9. 画面上端200、左端200の位置に、タイトルバー「日報」、メッセージ「今日の日付を入力してください」とテキストボックスを表示し、入力された値をアクティブセルに代入
10. 2列右のセルにアクティブセルを移動
11. 画面上端200、左端200の位置に、タイトルバー「日報」、メッセージ「天候を入力してください」とテキストボックスを表示し、入力された値をアクティブセルに代入
12. 1列右のセルにアクティブセルを移動
13. 画面上端200、左端200の位置に、タイトルバー「日報」、メッセージ「来場者数を入力してください」とテキストボックスを表示し、入力された値をアクティブセルに代入
14. 1列右のセルにアクティブセルを移動
15. 画面上端200、左端200の位置に、タイトルバー「日報」、メッセージ「売上金額を入力してください」とテキストボックスを表示し、入力された値をアクティブセルに代入
16. 1列右のセルにアクティブセルを移動
17. 画面上端200、左端200の位置に、タイトルバー「日報」、メッセージ「担当者名を入力してください」とテキストボックスを表示し、入力された値をアクティブセルに代入
18. 1行下、5列左のセルにアクティブセルを移動
19. プロシージャ終了

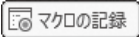

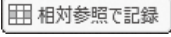

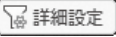



## ②

- ① Excelに切り替える
- ② 《開発》タブを選択
- ③ 《コントロール》グループの  (コントロールの挿入) をクリック
- ④ 《フォームコントロール》の  (ボタン (フォームコントロール)) をクリック
- ⑤ 任意の大きさにドラッグ
- ⑥ 《マクロ名》の一覧から「日報」を選択
- ⑦ 《OK》をクリック
- ⑧ ボタンの表示名を「日報入力」に変更



## 総合問題 3

①



- ①《開発》タブを選択
- ②《コード》グループの  マクロの記録 (マクロの記録) をクリック
- ③《マクロ名》に「未入荷リスト作成」と入力
- ④《マクロの保存先》の  をクリックし、一覧から《作業中のブック》を選択
- ⑤《OK》をクリック
- ⑥《コード》グループの  相対参照で記録 (相対参照で記録) をオフ (標準の色の状態) にする  
※ 選択するセルの位置をそのまま記録するため、 相対参照で記録 (相対参照で記録) をオフにします。
- ⑦ シート「商品リスト」を選択
- ⑧ セル【B4】をクリック
- ⑨《データ》タブを選択
- ⑩《並べ替えとフィルター》グループの  詳細設定 (詳細設定) をクリック
- ⑪《抽出先》の《指定した範囲》を  にする
- ⑫《リスト範囲》を「\$B\$4:\$G\$50」に設定
- ⑬《検索条件範囲》を「入荷待ちリスト!\$B\$3:\$B\$4」に設定
- ⑭《抽出範囲》を「商品リスト!\$I\$4」に設定  
※ 空いているセルを仮に指定します。
- ⑮《OK》をクリック
- ⑯ シート「入荷待ちリスト」を選択
- ⑰《開発》タブを選択
- ⑱《コード》グループの  記録終了 (記録終了) をクリック
- ⑲ シート「商品リスト」を選択
- ⑳ セル範囲【I4:N8】を選択
- ㉑《ホーム》タブを選択
- ㉒《編集》グループの  (クリア) をクリック
- ㉓《すべてクリア》をクリック
- ㉔ VBEを起動
- ㉕《標準モジュール》をダブルクリック
- ㉖ モジュール「Module1」をダブルクリック
- ㉗ プロシージャを次のように編集

## ■「未入荷リスト作成」プロシージャ

```
1. Sub 未入荷リスト作成()  
2.     Worksheets("入荷待ちリスト").Select  
3.     Range("B7").Select  
4.     ActiveCell.CurrentRegion.Clear  
5.     Worksheets("商品リスト").Select  
6.     Range("B4:G50").AdvancedFilter Action:=xlFilterCopy, CriteriaRange:=Sheets(_  
7.         "入荷待ちリスト").Range("B3:B4"), CopyToRange:=Worksheets("入荷待ちリスト") _  
8.         .Range("B7"), Unique:=False  
9.     Sheets("入荷待ちリスト").Select  
10. End Sub
```

※6～8行目はコードが長いので、行継続文字「\_」（半角スペース+半角アンダースコア）を使って行を複数に分割しています。行継続文字を使わずに1行で記述してもかまいません。

## ■プロシージャの意味

1. 「未入荷リスト作成」プロシージャ開始
2. シート「入荷待ちリスト」を選択
3. セル【B7】を選択
4. アクティブ領域をクリア
5. シート「商品リスト」を選択
6. 
7. フィルターによる抽出の実行（シート「入荷待ちリスト」のセル【B7】に抽出）
8. 
9. シート「入荷待ちリスト」を選択
10. プロシージャ終了

## 総合問題 4

①

- ① VBEを起動
- ②《挿入》をクリック
- ③《標準モジュール》をクリック
- ④ モジュール「Module1」に次のプロシージャを入力



### ■「表面利回り」プロシージャ

```
1. Function 表面利回り (家賃, 販売価格)
2.     If 家賃 = "" Then
3.         表面利回り = ""
4.     Else
5.         表面利回り = Round (家賃 * 12 / 販売価格 * 100, 1)
6.     End If
7. End Function
```

### ■プロシージャの意味

1. 「表面利回り」プロシージャ開始 (引数に「家賃」と「販売価格」を指定)
2. 「家賃」が空白の場合は
3. 空白を「表面利回り」に代入
4. それ以外の場合は
5. 「家賃 × 12 ÷ 販売価格 × 100」の小数第2位を四捨五入した結果を「表面利回り」に代入
6. Ifステートメント終了
7. プロシージャ終了

②

- ① Excellに切り替える
- ② セル【D9】をクリック
- ③  (関数の挿入) をクリック
- ④ 《関数の分類》の  をクリックし、一覧から《ユーザー定義》を選択
- ⑤ 《関数名》の一覧から「表面利回り」を選択
- ⑥ 《OK》をクリック
- ⑦ 《家賃》に「B9」と入力
- ⑧ 《販売価格》に「\$B\$6」と入力
- ⑨ 《OK》をクリック
- ⑩ セル【D9】をセル範囲【D10:D11】にコピー
- ⑪ 同様に、セル範囲【H9:H11】にユーザー定義関数「表面利回り」を入力

## 総合問題5

①

- ① VBEを起動
- ② 《挿入》をクリック
- ③ 《標準モジュール》をクリック
- ④ モジュール「Module1」に次のプロシーダを入力



### ■「金種計算」プロシーダ

```
1. Function 金種計算 (金額, 単位)
2.     Select Case 単位
3.         Case 10000
4.             金種計算 = Int (金額 / 10000)
5.         Case 5000
6.             金種計算 = Int ((金額 Mod 10000) / 5000)
7.         Case 1000
8.             金種計算 = Int ((金額 Mod 5000) / 1000)
9.         Case 500
10.            金種計算 = Int ((金額 Mod 1000) / 500)
11.        Case 100
12.            金種計算 = Int ((金額 Mod 500) / 100)
13.        Case 50
14.            金種計算 = Int ((金額 Mod 100) / 50)
15.        Case 10
16.            金種計算 = Int ((金額 Mod 50) / 10)
17.        Case 5
18.            金種計算 = Int ((金額 Mod 10) / 5)
19.        Case 1
20.            金種計算 = 金額 Mod 5
21.    End Select
22. End Function
```

### ■プロシーダの意味

1. 「金種計算」プロシーダ開始 (引数に「金額」と「単位」を指定)
2. 「単位」が
3. 「10000」の場合は
4. 「金額 ÷ 10000」の小数点以下を切り捨てた結果を「金種計算」に代入
5. 「5000」の場合は
6. 「(金額 ÷ 10000 の余り) ÷ 5000」の小数点以下を切り捨てた結果を「金種計算」に代入
7. 「1000」の場合は
8. 「(金額 ÷ 5000 の余り) ÷ 1000」の小数点以下を切り捨てた結果を「金種計算」に代入
9. 「500」の場合は
10. 「(金額 ÷ 1000 の余り) ÷ 500」の小数点以下を切り捨てた結果を「金種計算」に代入
11. 「100」の場合は
12. 「(金額 ÷ 500 の余り) ÷ 100」の小数点以下を切り捨てた結果を「金種計算」に代入
13. 「50」の場合は
14. 「(金額 ÷ 100 の余り) ÷ 50」の小数点以下を切り捨てた結果を「金種計算」に代入
15. 「10」の場合は
16. 「(金額 ÷ 50 の余り) ÷ 10」の小数点以下を切り捨てた結果を「金種計算」に代入
17. 「5」の場合は
18. 「(金額 ÷ 10 の余り) ÷ 5」の小数点以下を切り捨てた結果を「金種計算」に代入
19. 「1」の場合は
20. 「金額 ÷ 5 の余り」の結果を「金種計算」に代入
21. Select Caseステートメント終了
22. プロシーダ終了

## ②

- ① Excelに切り替える
- ② セル【D4】をクリック
- ③  (関数の挿入) をクリック
- ④ 《関数の分類》の  をクリックし、一覧から《ユーザー定義》を選択
- ⑤ 《関数名》の一覧から「金種計算」を選択
- ⑥ 《OK》をクリック
- ⑦ 《金額》に「\$C4」と入力
- ⑧ 《単位》に「D\$3」と入力
- ⑨ 《OK》をクリック
- ⑩ セル【D4】をセル範囲【D4:L8】にコピー

## 総合問題 6

①

- ① VBEを起動
- ② 《挿入》をクリック
- ③ 《標準モジュール》をクリック
- ④ モジュール「Module1」に次のプロシージャを入力

### ■「顧客名入力」プロシージャ

```
1. Sub 顧客名入力 ()
2.     Dim kokyaku As String
3.     kokyaku = InputBox ("顧客名を入力します", "顧客名")
4.     If kokyaku = "" Then
5.         Range ("B4").Value = ""
6.     Else
7.         Range ("B4").Value = kokyaku & "□御中"
8.     End If
9. End Sub
```

※「&」の前のスペースは直接入力します。

※□は全角スペースを表します。

### ■プロシージャの意味

1. 「顧客名入力」プロシージャ開始
2. 文字列型の変数「kokyaku」を使用することを宣言
3. タイトルバー「顧客名」、メッセージ「顧客名を入力します」とテキストボックスを表示し、入力された値を変数「kokyaku」に代入
4. 変数「kokyaku」が空文字列の場合は
5. セル【B4】に空白を代入
6. それ以外の場合は
7. 変数「kokyaku」と「 御中」を連結した文字列をセル【B4】に代入
8. Ifステートメント終了
9. プロシージャ終了

②

- ① VBEが表示されていることを確認
- ② 《ファイル》をクリック
- ③ 《ファイルのインポート》をクリック
- ④ フォルダー「Excel2021／2019／2016／365マクロVBA」が開かれていることを確認  
※フォルダー「Excel2021／2019／2016／365マクロVBA」が開かれていない場合は、《ドキュメント》→「Excel2021／2019／2016／365マクロVBA」を選択します。
- ⑤ フォルダー「総合問題」を選択
- ⑥ 《開く》をクリック
- ⑦ 「リスト\_B.bas」を選択  
※ブック「総合問題6\_A」を開いている場合は「リスト\_A.bas」を選択します。
- ⑧ 《開く》をクリック

③

- ① モジュール「Module1」が表示されていることを確認
- ② 「End Sub」の下の方にカーソルを移動
- ③ 次のプロシーダを入力

#### ■「データ入力」プロシーダ

```

1. Sub データ入力 ()
2.     Dim kataban As String
3.     Range ("C15").Select
4.     Do While ActiveCell.Offset (0, -1).Value <> ""
5.         kataban = InputBox ("型番を入力します" & Chr (10) & _
6.             "終了する場合は* (アスタリスク) を入力してください", "型番")
7.         If kataban = "*" Then
8.             ActiveCell.Value = "*"
9.             Exit Do
10.        Else
11.            ActiveCell.Value = kataban
12.            ActiveCell.Offset (0, 3).Select
13.            ActiveCell.Value = InputBox ("数量を入力します", "数量")
14.            Application.ScreenUpdating = True
15.            ActiveCell.Offset (1, -3).Select
16.        End If
17.    Loop
18. End Sub

```

※5～6行目はコードが長いので、行継続文字「\_ (半角スペース+半角アンダースコア)」を使って行を複数に分割しています。行継続文字を使わずに1行で記述してもかまいません。

※6～8行目の「\*」は半角で入力します。

#### ■プロシーダの意味

```

1. 「データ入力」プロシーダ開始
2.     文字列型の変数「kataban」を使用することを宣言
3.     セル【C15】を選択
4.     アクティブセルの1列左のセルが空白でない間は
5.         タイトルバー「型番」、メッセージ「型番を入力します (改行)
6.         終了する場合は* (アスタリスク) を入力してください」とテキストボックスを表示し、
           入力された値を変数「kataban」に代入
7.         変数「kataban」が「*」の場合は
8.             アクティブセルに「*」を代入
9.             Do～Loopステートメントから抜ける
10.    それ以外の場合は
11.        変数「kataban」の値をアクティブセルに代入
12.        3列右にアクティブセルを移動
13.        タイトルバー「数量」、メッセージ「数量を入力します」とテキストボックスを表示し、
           入力された値をアクティブセルに代入
14.        画面の更新を有効にする
15.        1行下、3列左のセルにアクティブセルを移動
16.    Ifステートメント終了
17.    4行目に戻る
18. プロシーダ終了

```

④

- ① モジュール「Module1」が表示されていることを確認
- ② 最終行の「End Sub」の下の方にカーソルを移動
- ③ 次のプロシージャを入力

#### ■「プレビュー」プロシージャ

1. Sub プレビュー ()
2.     MsgBox "プレビューを実行後、印刷ボタンをクリックします"
3.     ActiveSheet.PrintPreview
4. End Sub

#### ■プロシージャの意味

1. 「プレビュー」プロシージャ開始
2.     「プレビューを実行後、印刷ボタンをクリックします」のメッセージを表示
3.     アクティブシートを印刷プレビューで表示
4. プロシージャ終了

⑤

- ① モジュール「Module1」が表示されていることを確認
- ② 最終行の「End Sub」の下の方にカーソルを移動
- ③ 次のプロシージャを入力

#### ■「データ削除」プロシージャ

1. Sub データ削除 ()
2.     Range ("B4,C15:C29,F15:F29").ClearContents
3.     Range ("A1").Select
4. End Sub





#### ■プロシージャの意味

1. 「データ削除」プロシージャ開始
2.     セル【B4】とセル範囲【C15:C29】とセル範囲【F15:F29】のデータを削除
3.     セル【A1】を選択
4. プロシージャ終了



## 総合問題7

①

- ①《開発》タブを選択
- ②《コード》グループの  (マクロの記録) をクリック
- ③《マクロ名》に「シート保護」と入力
- ④《マクロの保存先》の  をクリックし、一覧から《作業中のブック》を選択
- ⑤《OK》をクリック
- ⑥《ホーム》タブを選択
- ⑦《セル》グループの  (書式) をクリック
- ⑧《シートの保護》をクリック
- ⑨《シートとロックされたセルの内容を保護する》が ☒ になっていることを確認
- ⑩《OK》をクリック
- ⑪《開発》タブを選択
- ⑫《コード》グループの  (記録終了) をクリック

②

- ① VBEを起動
- ② プロジェクトエクスプローラーの《標準モジュール》をダブルクリック
- ③ モジュール「Module1」をダブルクリック
- ④ 「End Sub」の下の方にカーソルを移動
- ⑤ 次のプロシージャを入力

### ■「シート保護解除」プロシージャ

```
1. Sub シート保護解除 ()
2.     Dim i As Integer
3.     Dim mypass As String
4.     For i = 1 To 3
5.         mypass = InputBox ("パスワードを入力 (大文字小文字を認識します)", _
6.             "パスワード入力")
7.         If mypass = "password" Then
8.             MsgBox "シート保護を解除します"
9.             ActiveSheet.Unprotect
10.            Exit For
11.        Else
12.            MsgBox "パスワードが違います"
13.        End If
14.    Next
15. End Sub
```

※「To」の前後のスペースは直接入力します。

※5～6行目はコードが長いので、行継続文字「\_ (半角スペース+半角アンダースコア)」を使って行を複数に分割しています。行継続文字を使わずに1行で記述してもかまいません。

## ■プロシージャの意味

1. 「シート保護解除」プロシージャ開始
2.     整数型の変数「i」を使用することを宣言
3.     文字列型の変数「mypass」を使用することを宣言
4.     変数「i」が「1」～「3」になるまで処理を繰り返す
5.         タイトルバー「パスワード入力」、メッセージ「パスワードを入力（大文字小文字を認識します）」
6.         とテキストボックスを表示し、入力された値を変数「mypass」に代入
7.         変数「mypass」の値が「password」の場合は
8.             「シート保護を解除します」のメッセージを表示
9.             アクティブシートのシート保護を解除
10.            For Nextステートメントを抜ける
11.         それ以外の場合は
12.             「パスワードが違います」のメッセージを表示
13.         Ifステートメント終了
14.     変数「i」に変数「i」+ 1の結果を代入し、4行目に戻る
15. プロシージャ終了

よくわかる  
**Microsoft® Excel® マクロ/VBA**  
Office 2021/2019/2016/Microsoft 365 対応  
(FPT2220)

**練習問題・総合問題 標準解答**

2023年 4 月 4 日 初版発行

著作／制作：株式会社富士通ラーニングメディア

- 
- Microsoft、Excelは、米国Microsoft Corporationの米国およびその他の国における登録商標または商標です。
  - その他、記載されている会社および製品などの名称は、各社の登録商標または商標です。
  - 本文中では、TMや®は省略しています。
  - 本文中のスクリーンショットは、マイクロソフトの許可を得て使用しています。
  - 本資料は、構成・文章・プログラム・画像・データなどのすべてにおいて、著作権法上の保護を受けています。  
本資料の一部あるいは全部について、いかなる方法においても複写・複製など、著作権法上で規定された権利を侵害する行為を行うことは禁じられています。
  - 本製品に起因してご使用者に直接または間接的損害が生じてても、株式会社富士通ラーニングメディアはいかなる責任も負わないものとし、一切の賠償などは行わないものとします。
  - 本資料に記載された内容などは、予告なく変更される場合があります。
  - 購入者自らが使用になる場合に限り、複製を許諾します。