

7-4 実行時エラーの発生時に別の処理を実行するには？

※ VBEを起動し、《標準モジュール》→「Module1」を開いておきましょう。

① 「原価率計算」プロシージャを、次のように修正します。

■ 「原価率計算」プロシージャ

```
1. Sub 原価率計算()  
2.     Dim kakaku As Double  
3.     Dim genka As Double  
4.     Dim ritu As Double  
5.     Dim i As Integer  
6.     On Error GoTo ErrorKakaku  
7.     For i = 0 To 9  
8.         kakaku = 0  
9.         genka = 0  
10.        kakaku = Range("B4").Offset(i, 0).Value  
11.        genka = Range("C4").Offset(i, 0).Value  
12.        ritu = genka / kakaku * 100  
13.        Range("D4").Offset(i, 0).Value = ritu  
14.    Next  
15.    Exit Sub  
16. ErrorKakaku:  
17.    MsgBox "0による除算はできません。" & Chr(10) & "処理を継続します."  
18.    kakaku = genka * 2  
19.    Resume  
20. End Sub
```

■ プロシージャの意味

```
1. 「原価率計算」プロシージャ開始  
2.     倍精度浮動小数点数型の変数「kakaku」を使用することを宣言  
3.     倍精度浮動小数点数型の変数「genka」を使用することを宣言  
4.     倍精度浮動小数点数型の変数「ritu」を使用することを宣言  
5.     整数型の変数「i」を使用することを宣言  
6.     エラー処理を開始(実行時エラーが発生したら「ErrorKakaku」に制御を移す)  
7.     変数「i」が「0」から「9」になるまで次の行以降の処理を繰り返す  
8.         変数「kakaku」に「0」を代入  
9.         変数「genka」に「0」を代入  
10.        セル【B4】から変数「i」だけ下に移動したセルの値を変数「kakaku」に代入  
11.        セル【C4】から変数「i」だけ下に移動したセルの値を変数「genka」に代入  
12.        変数「genka」/変数「kakaku」*100の結果を変数「ritu」に代入  
13.        セル【D4】から変数「i」だけ下に移動したセルに変数「ritu」の値を代入  
14.    変数「i」に変数「i」+1の結果を代入し、7行目に戻る  
15.    プロシージャを抜け出す  
16. エラー処理ルーチン(行ラベル ErrorKakaku)  
17.     メッセージを表示  
18.     変数「kakaku」に変数「genka」を2倍した値を代入  
19.     7行目に制御を戻す  
20. プロシージャ終了
```

※ 再度「原価率計算」プロシージャを実行します。メッセージが2回表示され、エラーが発生せず、最後まで実行されることを確認します。確認後、上書き保存しておきましょう。