

Answer on Question#50511 – Programming - Visual Basic

Which of the following statements would you use to pass a variable named *message* by reference to a procedure named *DisplayMessage*?

Select one:

- a. *DisplayMessage(ByRef message)*
- b. *DisplayMessage(message ByRef)*
- c. *DisplayMessage(message As Reference)*
- d. *DisplayMessage(message)*

Answer: D

In Visual Basic, you can pass an argument to a procedure by value or by reference. This is known as the passing mechanism, and it determines whether the procedure can modify the programming element underlying the argument in the calling code. The procedure declaration determines the passing mechanism for each parameter by specifying the *ByVal* or *ByRef* keyword.

The procedure declaration specifies the passing mechanism for each parameter. The calling code can't override a *ByVal* mechanism. If a parameter is declared with *ByRef*, the calling code can force the mechanism to *ByVal* by enclosing the argument name in parentheses in the call. The default in Visual Basic is to pass arguments by value.

When to Pass an Argument by Reference

- If the procedure has a genuine need to change the underlying element in the calling code, declare the corresponding parameter *ByRef* (Visual Basic);
- If the correct execution of the code depends on the procedure changing the underlying element in the calling code, declare the parameter *ByRef*. If you pass it by value, or if the calling code overrides the *ByRef* passing mechanism by enclosing the argument in parentheses, the procedure call might produce unexpected results.

Example:

```
Sub Calculate(ByVal rate As Double, ByRef debt As Double)
    debt = debt + (debt * rate / 100)
End Sub
```

Correct call:

```
Calculate(10, debtWithInterest)
```

So, type “by referenced”, is described on description function, and on call just write name variable (if type variable is by reference)