PROGRAMMING LANGUAGE 2 (SPM 3112)

PROCEDURES & FUNCTIONS (PART 2)

NOOR AZEAN ATAN
MULTIMEDIA EDUCATIONAL DEPARTMENT
UNIVERSITI TEKNOLOGI MALAYSIA

Topics

- Pre-defined functions
- User-defined functions
- Local vs Global

Introduction

VB functions can be divided into two categories:

- Pre-defined (standard function)
  - Definitions have been written and it is ready to be used.
  - Which means you do not need to declare and define.
- User-defined
  - Function that been created by the user.
  - This functions need to be declared and defined by the user.

Pre-Defined Function

- also known as → built-in function which is already available in VB
- always return exactly 1 value
  - to use this function, you need to call the function with the correct name and value.

FunctionName (Value)

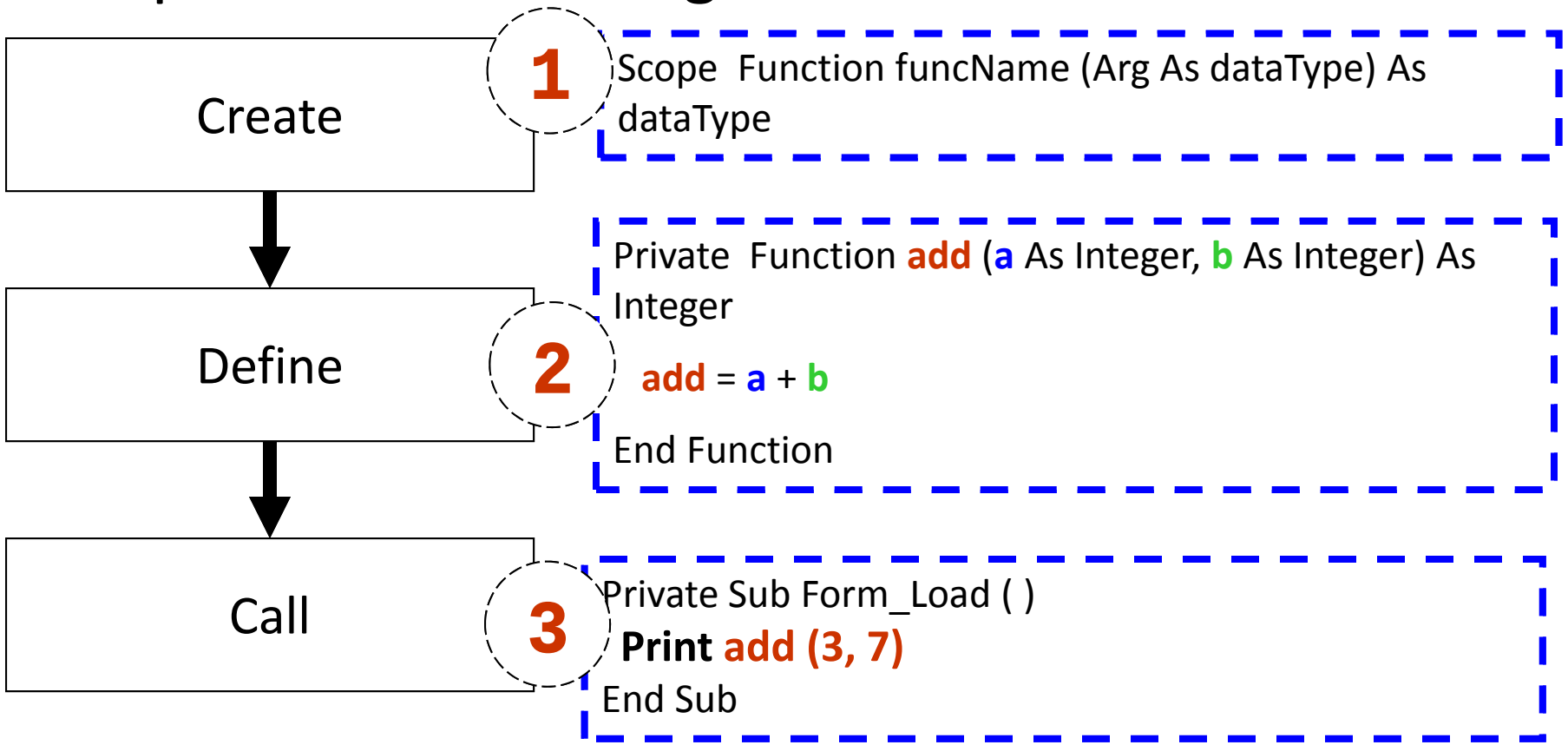
Example: sqr(4)

Built-In functions

- General math
  - ***sqr(number), abs(number), exp(number), log(number)***
- Trigonometric functions
  - ***sin(number), cos(number), tan(number)***

User-Defined Function

The process of creating function



User-Defined Function

Syntax:

Scope Function FunctionName (Arg As Datatype) As Datatype

Statement(s)

End Function

Example:

```
Private Function Add_Two_Numbers (X as Single, Y as Single) as Single
    Add_Two_Numbers = X + Y
End Function
** in calling sub
    Label1.Caption = Add_Two_Numbers(5, 6)
```

Local vs Global

- Three levels of scope:
  - **Local** - Variable can be seen only in an event procedure or sub-procedure and functions
  - **Module-level** - Variable can be seen by all events and procedures in that form
  - **Global** - Variable can be seen by all events and procedures in the entire project

Scope of Variables

In form1:

```
Dim iNum3 as Integer

Private Sub cmdEnter_Click()
   Dim iNum4 as Integer
   iNum4 = iNum1 + iNum3
End Sub

Private Sub cmdCompute_Click()
    Dim iNum5 as Integer
  iNum5 = iNum2 + iNum3
End Sub
```

In form2:

```
Dim iNum3 as Integer

Private Sub cmdPrint_Click()
  iNum3 = iNum1 * iNum2
End Sub
```

In Standard Code Module:

```
Global iNum1 As Integer
Global iNum2 As Integer
```

- **Red** represents module level
- **Blue** represents local
- **Green** represents global