

```

public sub address ()
Print "Jail Road, Ara"
end sub

```

## → Function :-

A function is also contains some specific code block & similar to sub-routine but, it always returns a value. The statement that made a function are placed in the pair of "Function" & "End Function".

### Syntax

```

[ Private / Public / Friends ] [ Static ] Function-name
[ (arg-list) ] [ AS Type ]

```

```

≡ [ statements ]

```

```

[ Exit Function ]

```

```

≡

```

```

[ statements ]

```

```

≡

```

```

End function

```

The function must be in between function & end function keyword.



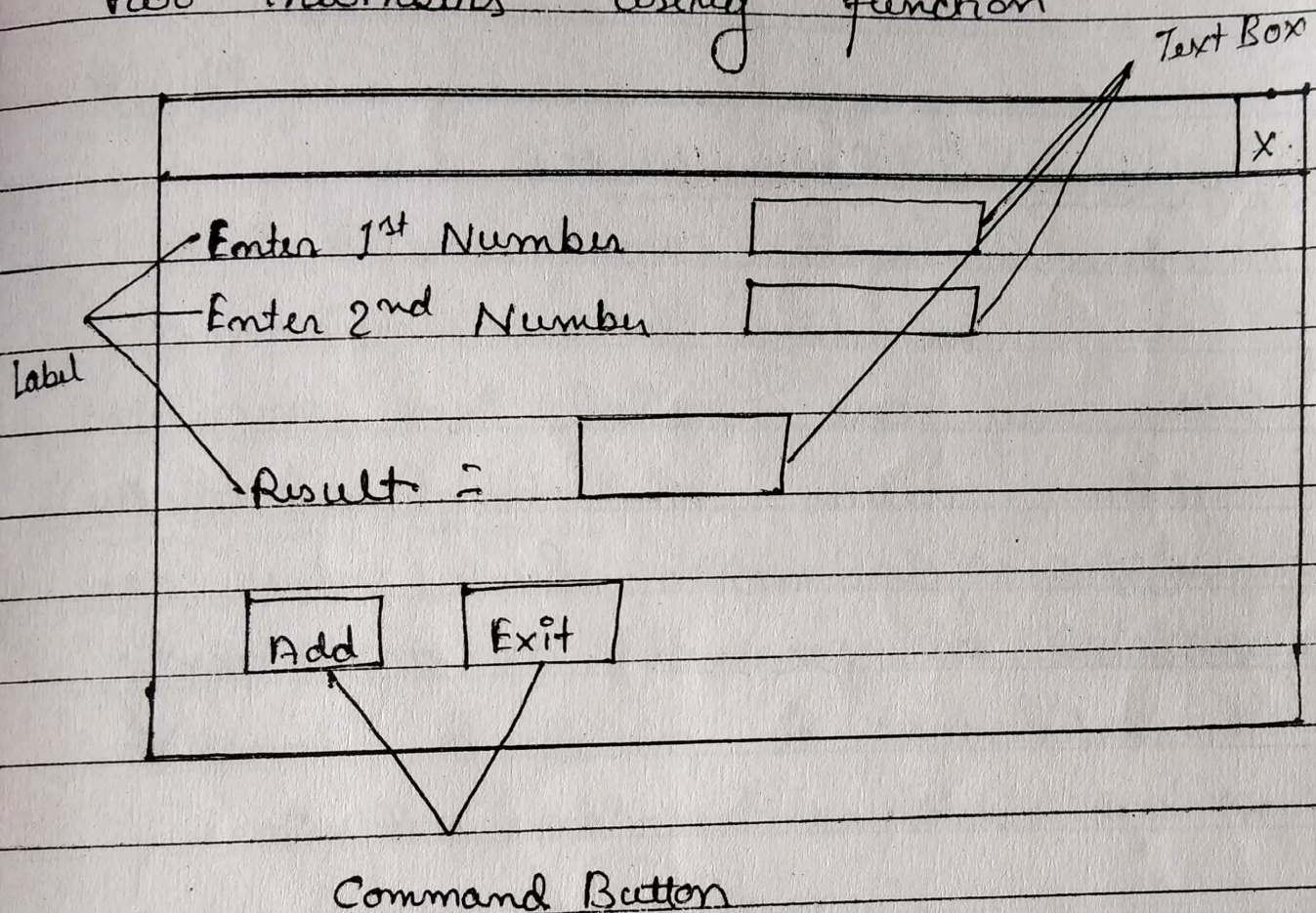
- A Public keyword means a function accessible to all other procedures in all modules & forms.
- The Private keyword makes a function accessible only to other procedures or function in the module or form in which it is declared.
- The Frame keyword is used only in class module.
- The Static keyword specifies that the procedure or functions local variables should be preserved between diff. function calls.
- The name identifier is the name of function the argument list (arg-list) identifies a list of variables represent in the arguments that are passed to function when it is called. We ~~at~~ separate multiple variables with commas (,) •  $\emptyset$
- The TYPE TYPE identifier specifies



the datatype return by the function

Example :

Write a V.B application to add two numbers using function



```
Private sub command1 - click()
```

```
Dim a, b, c As Integer
```

```
a = val (Text1 - text)
```

```
b = val (Text2 - text)
```

```
c = add (a, b)
```

```
Text3 - Text = Str (c)
```

```
end sub
```



Public Function

add (x as Integer, y as Integer) as Integer

Public Function add(x, y) as Integer

add = x + y

end function

21/8/12

### \* Arguments & Parameters :-

Procedure & Function can be called into according to the requirement.

When they are called into execution certain values can be pass to the procedures & Function. Such values are referred as parameter or argument.

Parameter can be passed to function or sub-routines by two ways :-

- 1) Pass by Value.
- 2) Pass by Reference.

#### 1) Pass by Value :-

When a parameter is pass by value a copy of original variable is send to the function or sub-routine. The function can be



change the value of that variable but original value cannot be changed  
 Pass by value is done by "By Val"  
keyword.

Example:-

```
Public Function Add (By Val x As Integer).  

  ≡  

  ≡
```

End Function.

## 2) Pass by Reference or

When a parameter is passed by reference the address of original variable is passed to the function or sub-routine. The function or sub-routine can change the value of variable & the original variable is changed.

Note:-

To use Pass by Reference mechanism we use "ByRef" keyword or if we do not specify any method it by default accessed by reference.



Example :-

Public Function Add (ByRef x as Integer)

=====

End Function.

Q) Write a Visual Basic Function to show the use of ByVal keyword in a function.

Sol.

Private Sub Command1\_Click ()

DIM K, R AS Integer.

K = 100

R = Cal(K)

Print K

Print R

End Sub.

Public Function (ByVal a AS Integer)

a = 500

cal = a

~~End Sub~~; End function

Q) Write a Visual Basic Function to show the use of ByRef keyword in a function.

Private Sub Command1\_Click ()

Dim K, R as Integer

K = 100

R = Cal(R)

Teacher Signature \_\_\_\_\_



```

Print k
Print Print R
End Sub
Public Function (By Ref a As Integer)
a = 500
cal = a
End Sub Function

```

## \* Modules :-

22/8/12

Code in V-Basic stored in modules.

• There are 3 kinds of Modules:-

- 1) Form Module
- 2) Standard Module
- 3) Class Module

## ② Form Modules

A simple application consist of a single form & a code. As our application get larger we add additional forms. We find that the same common code we want to execute



in several form but we don't want to duplicate the code in all form, we create a separate model containing the procedure that implement the common code then we execute it in multiple forms

### 1) Form Module :-

A Form Module are the base of any V-Basic application. They contain graphical description of the form & its control including the property setting. Forms are the part of our application that can be visible at run time. The Form Module have Form

(Form Module में Use एप्लेट only Form में फ़िद है।)

### 2) Standard Module :-

Standard Module are container of procedures & declaration of commonly used variables. Used by other part of our application.

(General It is a general container that contain either Variable, Function or Sub-routine).



They contain Global or Module level declaration of constants, variable, Procedures etc. The Standard Module has

- Bas extension.

Steps to create Standard Module:-

① Step 1:- Right click on Project Explorer window.

Step 2:- Then select add option.

Step 3:- Then select standard module option of add <sup>sub-</sup> menu.

3) Class Module:-

The class Module is the base of object-oriented programming in Visual Basic. We can write code in Class Module to create a new object. The new object can be include to create our own properties & method that can be used by other object in our application. "New" keyword is used to create the object <sup>factory</sup>. The class module have

- cls extension.