

**PROCEDURE :**

## 1 . Create an Simple Calculator

**STEP-I :** Open Visual Basic 6.0 and create a new Standard EXE project.

**STEP-II :**Add and arrange two Label controls (lblFirstNum and lblSecondNum) and two TextBox controls (txtFirstNum and txtSecondNum) on the form.

**STEP-III :**Add five CommandButton controls, setting their Name and Caption properties to cmdAdd (Add), cmdSub (Subtract), cmdMul (Multiply), cmdDiv (Divide), and cmdExit (Exit).

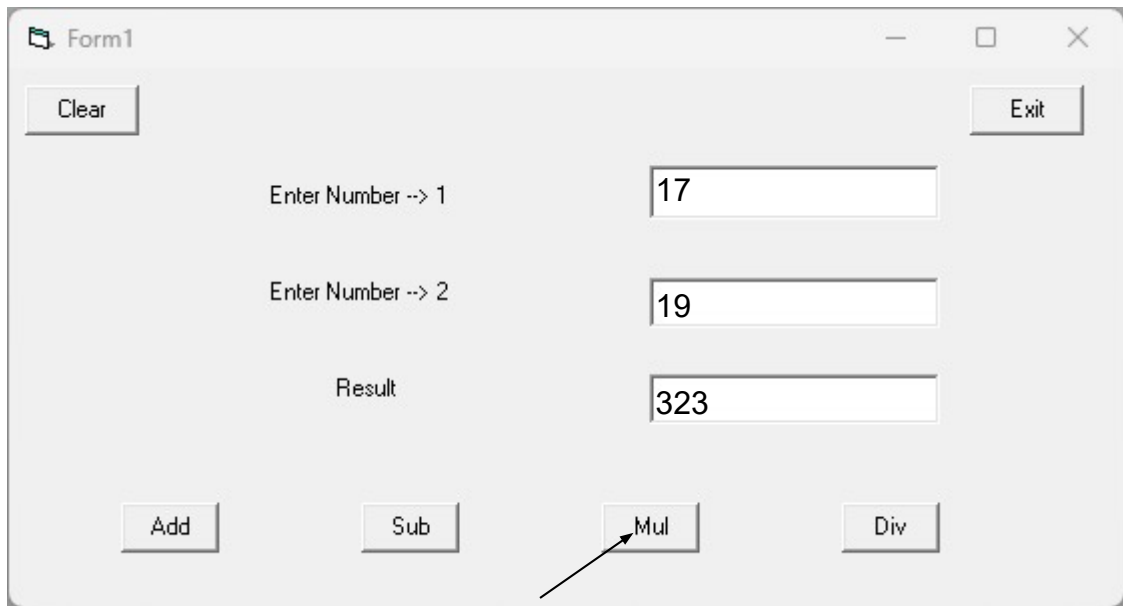
**STEP-IV :** Double-click each button and write code in the click event to perform the respective arithmetic operation or exit the application.

**STEP-V :**Run the form to test functionality.

**CODING :**

Private Sub Command1_Click() Text3.Text = Val(Text1.Text) + Val(Text2.Text) End Sub	"ADDITION"
Private Sub Command2_Click() Text3.Text = Val(Text1.Text) - Val(Text2.Text) End Sub	"SUBTRACT"
Private Sub Command3_Click() Text3.Text = Val(Text1.Text) * Val(Text2.Text) End Sub	"MULTIPLY"
Private Sub Command4_Click() Text3.Text = Val(Text1.Text) / Val(Text2.Text) End Sub	"DIVISION"
Private Sub Command5_Click() Text3.Text = "" Text2.Text = "" Text1.Text = "" End Sub	"CLEAR"
Private Sub Command6_Click() End End Sub	"EXIT"

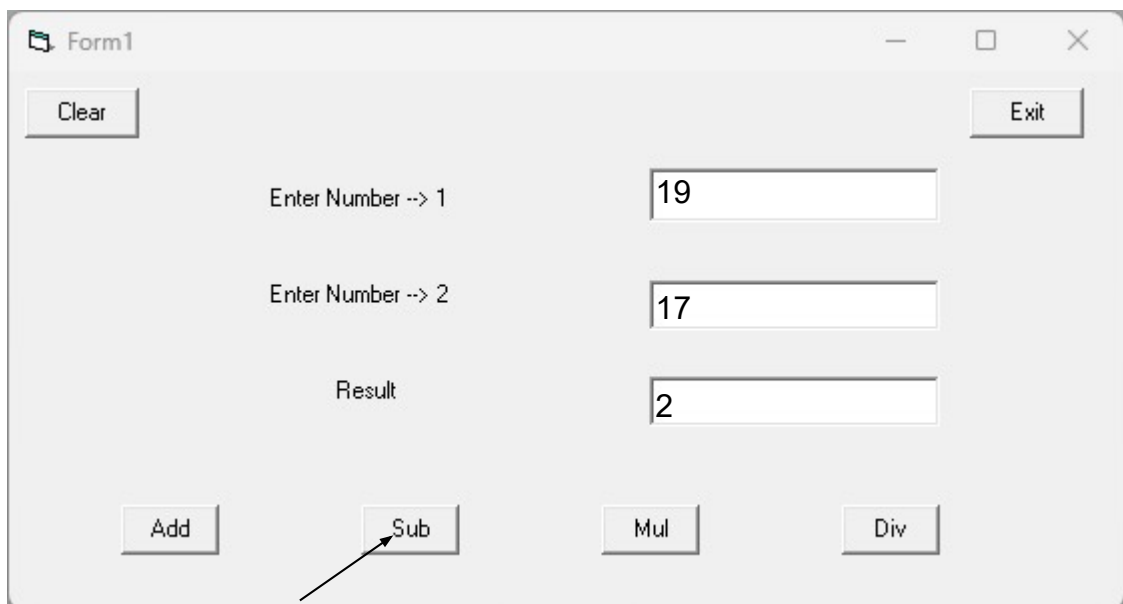
## EXECUTION :



A screenshot of a Windows application window titled "Form1". The window contains a "Clear" button at the top left and an "Exit" button at the top right. Below these, there are two input fields. The first is labeled "Enter Number --> 1" and contains the value "17". The second is labeled "Enter Number --> 2" and contains the value "19". Below these is a "Result" label and a text box containing "323". At the bottom, there are four buttons: "Add", "Sub", "Mul", and "Div". An arrow points to the "Mul" button, indicating it was just clicked.

Label	Value
Enter Number --> 1	17
Enter Number --> 2	19
Result	323

Buttons: Add, Sub, Mul, Div



A screenshot of the same "Form1" application window. The input fields now contain "19" for "Enter Number --> 1" and "17" for "Enter Number --> 2". The "Result" text box now contains "2". The "Add", "Sub", "Mul", and "Div" buttons are still at the bottom. An arrow points to the "Sub" button, indicating it was just clicked.

Label	Value
Enter Number --> 1	19
Enter Number --> 2	17
Result	2

Buttons: Add, Sub, Mul, Div

## **PROCEDURE :**

### **1 . Converting Font Style**

**STEP-I** : Open Visual Basic 6.0 and create a new Standard EXE project.

**STEP-II** : Add and arrange two Label controls ("Input Text:" and "Formatted Text:") and two TextBox controls (txtInput for input, txtFormatted for displaying formatted text).

**STEP-III** : Add four CommandButton controls with captions: Bold, Italic, Underline, and Exit.

**STEP-IV** : Write code in the click events of the buttons to toggle the corresponding font styles for txtFormatted, and sync txtFormatted with txtInput.

**STEP-V** : Run the form to test the functionality.

## **CODING :**

```
Private Sub Command1_Click()  
Text2.Text = Text1.Text  
Text2.FontBold = True  
End Sub  
Private Sub Command2_Click()  
Text2.Text = Text1.Text  
Text2.FontItalic = True  
End Sub  
Private Sub Command3_Click()  
End  
End Sub  
Private Sub Command4_Click()  
Text2.Text = Text1.Text  
Text2.FontUnderline = True  
End Sub  
Private Sub Command5_Click()  
Text2.Text = ""  
End Sub
```

“BOLD”

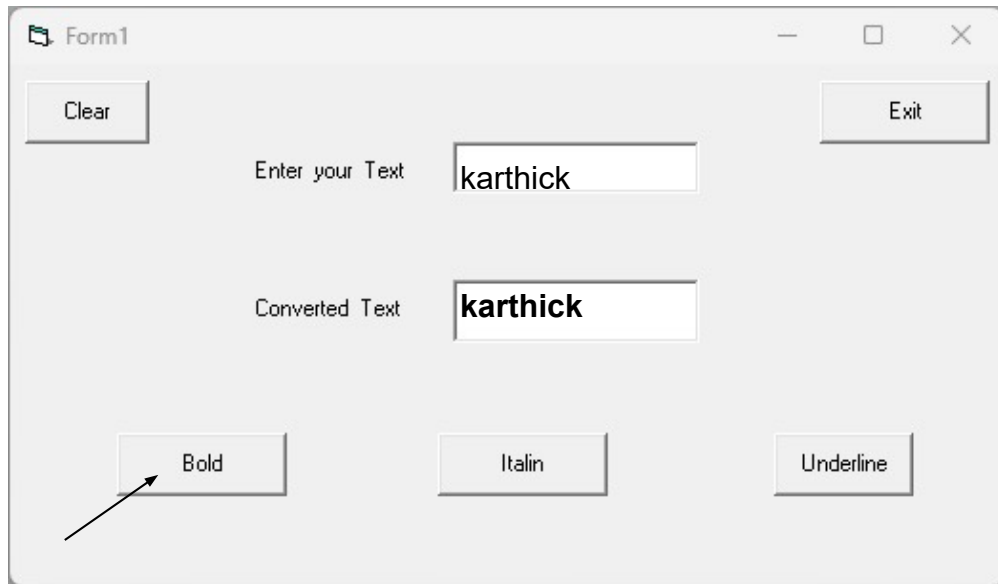
“ITALIC”

“EXIT”

“UNDERLINE”

“CLEAR”

## EXECUTION :



Form1

Clear

Exit

Enter your Text

karthick

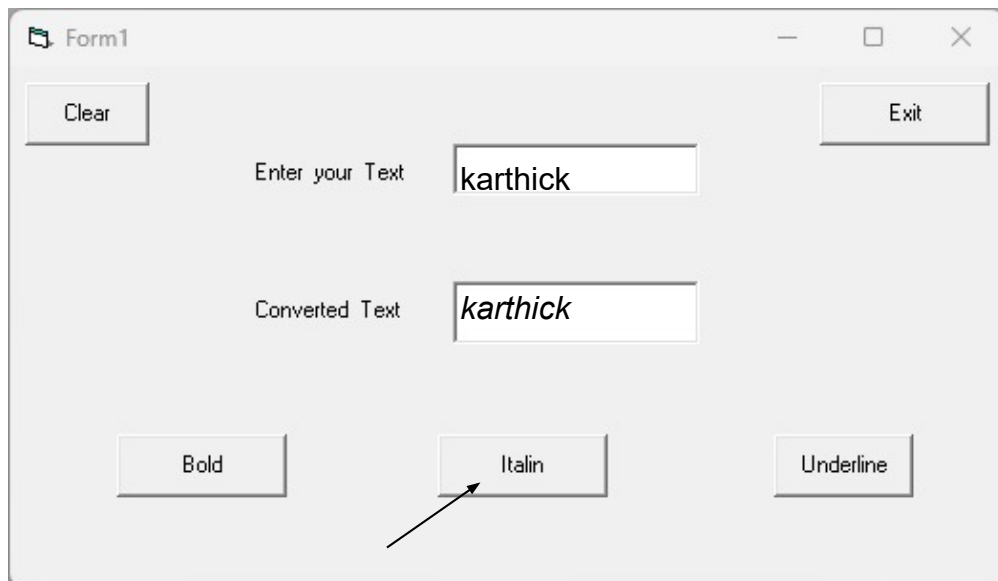
Converted Text

karthick

Bold

Italin

Underline



Form1

Clear

Exit

Enter your Text

karthick

Converted Text

**karthick**

Bold

Italin

Underline

## **PROCEDURE :**

### **1 . Various Component of VB**

**STEP-I :** Open Visual Basic 6.0 and create a new Standard EXE project.

**STEP-II :** Add and arrange two Label controls ("Input Text:" and "Formatted Text:") and two TextBox controls (txtInput for input, txtFormatted for displaying formatted text).

**STEP-III :** Add One CommandButton controls with captions: 'Apply'

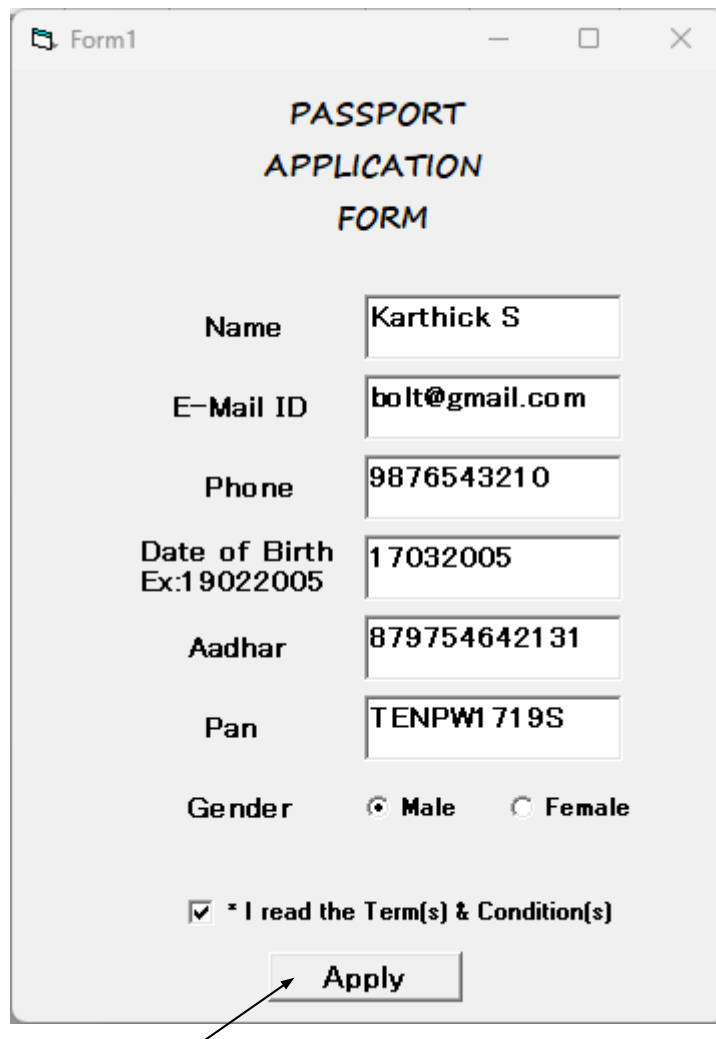
**STEP-IV :** Write code in the click events of the buttons to toggle the corresponding Action

**STEP-V :** Run the form to test the functionality.

## **CODING :**

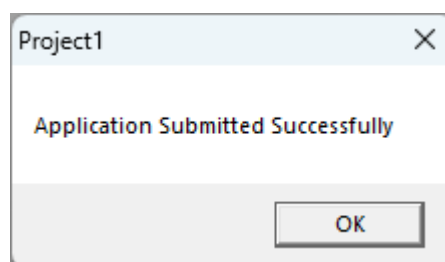
```
Private Sub Command1_Click()  
Text1.Text = ""  
Text2.Text = ""  
Text3.Text = ""  
Text4.Text = ""  
Text5.Text = ""  
Text6.Text = ""  
Check1.Value = 0  
Option1.Value = False  
Option2.Value = False  
MsgBox "Application Submitted Successfully"  
End  
End Sub
```

## EXECUTION :



The screenshot shows a Windows application window titled "Form1". The form is titled "PASSPORT APPLICATION FORM" in bold, italicized capital letters. It contains several input fields for personal information, each with a label to its left. The fields are: Name (Karthick S), E-Mail ID (bolt@gmail.com), Phone (9876543210), Date of Birth (17032005, with an example "Ex:19022005"), Aadhar (879754642131), and Pan (TENPW1719S). Below these fields is a "Gender" section with two radio buttons: "Male" (selected) and "Female". At the bottom, there is a checkbox labeled "\* I read the Term(s) & Condition(s)" which is checked. An "Apply" button is located at the bottom right, with an arrow pointing to it from below the window.

Name	Karthick S
E-Mail ID	bolt@gmail.com
Phone	9876543210
Date of Birth Ex:19022005	17032005
Aadhar	879754642131
Pan	TENPW1719S
Gender	<input checked="" type="radio"/> Male <input type="radio"/> Female
<input checked="" type="checkbox"/> * I read the Term(s) & Condition(s)	
<input type="button" value="Apply"/>	



The screenshot shows a small dialog box titled "Project1" with a close button (X) in the top right corner. The message "Application Submitted Successfully" is displayed in the center. At the bottom right, there is an "OK" button.

Project1

Application Submitted Successfully

OK