**PRACTICAL # 12**

**OBJECT:**

GUI Click Event Handling

**THEORY:**

C# GUI applications support event driven programming model. That is, the application responds to different actions the user performs on the UI. For instance, if a user wants to login in the login form, the application will execute the login code only when the user clicks the login button. The button defines an onClick event handler code for login action. Here, Login button click is a user action that generates an onClick event. Corresponding to this event, there is a code that handles this event.

**Program:**

First create a new Windows Form application in Visual Studio,

Choose File → New → Project.

Choose Visual C# in the Installed Templates list, and select Windows Forms Application. Name the new project, and press OK button.

This creates a new project in Visual Studio with the following interface:

**Add Button**

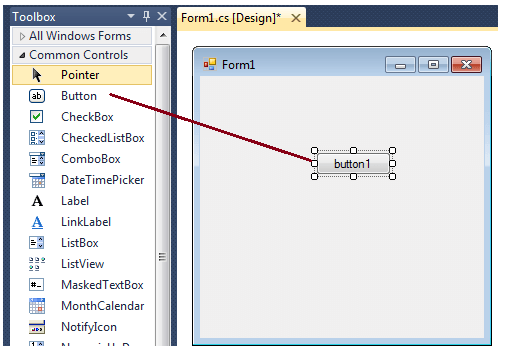
A button is used to allow the user to click on a button which would then start the processing of the form. We will add a simple button called 'Submit' which will be used to submit all the information on the form.

Step 1) The first step is to drag the button control onto the Windows Form from the toolbox.

Change the text property of the button control. Go the properties windows and change the text to 'submit'. Also change the name property of the control. Go the properties windows and change the name to 'btnSubmit'.

**Event Handling for Controls**

When working with windows form, you can add events to controls. An event is generated with different conditions, like when a user clicks on the application or presses a keyboard button or some other internal event of the application. A common action is the clicking of a button on a form. In Windows Forms, you can add code to perform certain actions when a button is clicked.



**Adding Controls to a form**

You can open toolbox and select GUI elements to be created on the form. There are a number of different UI controls and containers available to design your application interface. Simply drag and drop a control from the toolbox onto the form and it will be created on that specific location.

The Windows Form in Designer view is a visual representation of the window that will open when your application is opened. You can switch between this view and Code view at any time by right-clicking the design surface or code window and then clicking View Code or View Designer.

**ACTIVITIES**

**Activity 1**

Create a simple calculator application with four arithmetic operators (+, -, \*, /) buttons. Where each button click performs the corresponding operation between the two operands.

**Activity 2**

Create a simple application that changes the background color of a button when it gains focus and loses focus.

**REVIEW QUESTIONS**

1. What is the advantage of event driven programming model?
2. How are events generated?
3. How do you add a button click event handler code in C#?
4. Can a UI element have more than one event handlers for different events?