**Q:1 Write the statement that actually construct or instantiates a button object**

**Ans:**

```
using System.Windows.Forms;
Form form = new Form();
Button button = new Button();
button.Text = "Click me";
```

**Q:2 Write the statement that register a button object clickevent with the operating system**

**Ans:**

```
using System;
using System.Windows.Forms;
Form form = new Form();
Button button = new Button();
button.Text = "Click me";
button.Click += (sender, e) => {
   };
```

**Q:3 Identify which property could be set so that a Form object would perform the following function:**

**a. Change the background color for the form.**

**Ans:** form.BackColor = System.Drawing.Color.LightBlue;

**b. Set the default font to Courier for all controls that are added.**

**Ans:** form.Font = new System.Drawing.Font("Courier New", 10, System.Drawing.FontStyle.Regular);

**c. Change the size of the window to 400 by 400.**

**Ans:** form.Size = new System.Drawing.Size(400, 400);

**d. Associate a name of designForm with the Form object.**

**Ans:** form.Name = "designForm";

**Q:4 Write a C# statement or a set of C# statements to accomplish each of the following tasks: Section III**

**a) Sum the odd integers between 1 and 99, using a for structure. Assume that the integer variables sum and count have been declared.**

**Ans**:

```
 int sum = 0;
int count = 0;
for (int i = 1; i <= 99; i += 2)
{
    sum += i;
    count++;
}
Console.WriteLine($"Sum of odd integers between 1 and 99: {sum}");
Console.WriteLine($"Number of odd integers: {count}");
```

**b) Calculate the value of 2.5 raised to the power of 3, using the Math.Pow method.**

**Ans:**

```
 double result = Math.Pow(2.5, 3);
Console.WriteLine($"2.5 raised to the power of 3 is: {result}");
```

**Q:5 Create a Windows application (with diagram) that sets the Form object 05 properties Size, Location, Text, ForeColor,BackColor, and of Accept Button. The form should contain one button labeled Click Me.When the user clicks the button, a Message Box should be displayed indicating the button has been clicked**

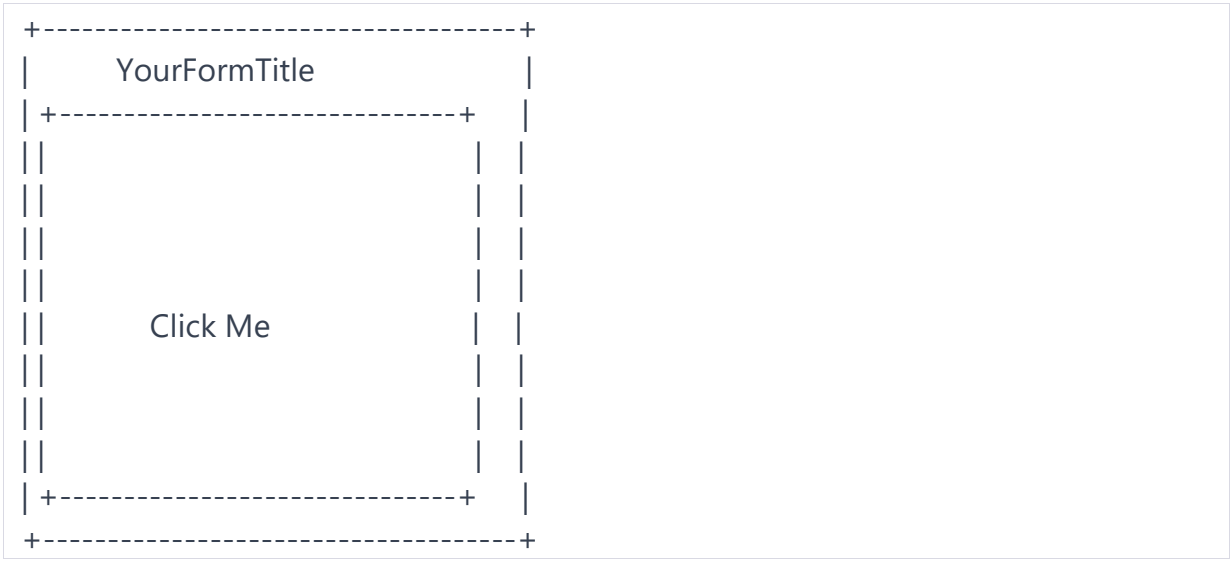
## Ans: Visual Studio Steps:

1. Open Visual Studio and create a new Windows Forms Application project.
2. Drag and drop a `Button` control onto the form.

3. Set the properties as follows:
   - **Form Properties:**
     - Size: 400, 400
     - Location: 100, 100
     - Text: YourFormTitle
     - ForeColor: Choose a color
     - BackColor: Choose a color
     - AcceptButton: YourButton (Select the button you added)
   - **Button Properties:**
     - Text: Click Me
4. Double-click the button to generate the click event handler code.
5. Inside the generated method, add code to display a message box:

```
private void YourButton_Click(object sender, EventArgs e)
{
    MessageBox.Show("Button has been clicked!");
}
```

## Diagram (Text Representation):




**Q:6 Create an Invoice class that could be used by a bookstore. Items to include as data members are Item number, description, unit price, and quantity 05 purchased. Include appropriate constructors and properties plus an additional method that calculates the total cost using the quantity and unit price.Overde the ToString () method to return the item description and i total cost. Create a second class to test your Invoice class.**

**Ans:**

```
using System;
class Invoice
{
   // Data members
   public string ItemNumber { get; set; }
   public string Description { get; set; }
   public decimal UnitPrice { get; set; }
   public int QuantityPurchased { get; set; }

   // Constructors
   public Invoice()
   {
      // Default constructor
   }

   public Invoice(string itemNumber, string description, decimal unitPrice, int quantityPurchased)
   {
      ItemNumber = itemNumber;
      Description = description;
      UnitPrice = unitPrice;
      QuantityPurchased = quantityPurchased;
   }

   // Method to calculate total cost
   public decimal CalculateTotalCost()
   {
      return UnitPrice * QuantityPurchased;
   }

   // Override ToString method
```

```
    public override string ToString()
    {
        return $"{Description}\nTotal Cost: {CalculateTotalCost():C}";
    }
}
class Program
{
    static void Main()
    {
        // Test the Invoice class
        Invoice bookInvoice = new Invoice("001", "Book: Introduction to C#", 29.99m, 3);
        // Display the invoice details
        Console.WriteLine("Invoice Details:\n");
        Console.WriteLine($"Item Number: {bookInvoice.ItemNumber}");
        Console.WriteLine($"Description: {bookInvoice.Description}");
        Console.WriteLine($"Unit Price: {bookInvoice.UnitPrice:C}");
        Console.WriteLine($"Quantity Purchased: {bookInvoice.QuantityPurchased}");
        Console.WriteLine($"Total Cost: {bookInvoice.CalculateTotalCost():C}\n");

        // Using the overridden ToString method
        Console.WriteLine(bookInvoice);
    }
}
```