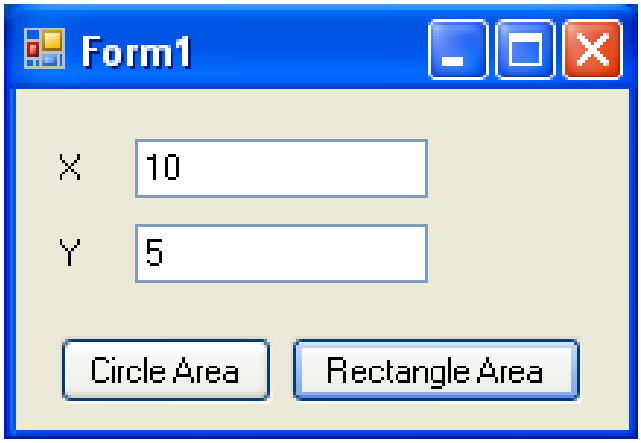
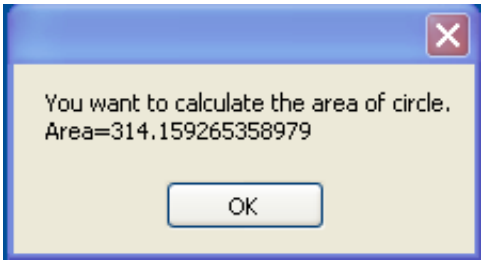
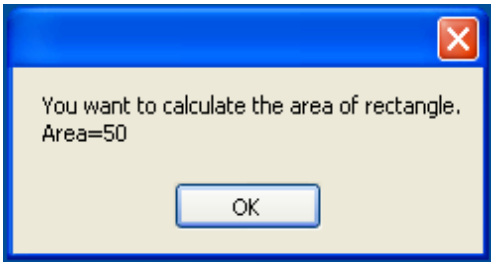
Method Overloading

- Method overloading commonly is used to create several methods with the same name that perform similar tasks, but on different data types. C# enables several methods of the same name to be defined in the same class, as long as these methods have different sets of parameters.
- When an overloaded method is called, the C# compiler selects the proper method by examining the number, types and order of the call's arguments.


```
class MyClass
{
    public static void Get(double Number, out double Square, out double Cube)
    {
        Square = Number * Number;
        Cube = Number * Number * Number;
    }
}
private void ButCalculate_Click(object sender, EventArgs e)
{
   double x, y, z;
   x = double.Parse(TxtX.Text);
   MyClass.Get(x, out y, out z);
   TxtSquare.Text = y.ToString( );
   TxtCube.Text = z.ToString( );
}
```








```
private void ButCircleArea_Click(object sender, EventArgs e)
{
   double x, y, z;
   x = double.Parse(TxtX.Text);
   y = double.Parse(TxtY.Text);
   z = MyClass.Area(x);
   MessageBox.Show("You want to calculate the area of circle. \nArea="
               + z.ToString());
}

private void ButRectangleArea_Click(object sender, EventArgs e)
{
   double x, y, z;
   x = double.Parse(TxtX.Text);
   y = double.Parse(TxtY.Text);
   z = MyClass.Area(x,y);
   MessageBox.Show("You want to calculate the area of rectangle. \nArea="
               + z.ToString());
}
```