

Lect2

Advance programming

Level 3 IT

IBB UN



```
1 A delegate is a type that represents references to methods
2 with a particular parameter list and return type. When you
3 instantiate a delegate, you can associate its instance with
4 any method with a compatible signature and return type.
5 You can invoke (or call) the method through the delegate
6 instance.
7
```



```
1 Syntax:
2 [modifier] delegate [return_type] [delegate_name] ([parameter_list]);
3
```

المفوضات (Delegates) في لغة C#

المفوضات هي عبارة عن مؤشرات إلى دوال (Function Pointers) تسمح لك باستدعاء دوال بشكل غير مباشر. تُستخدم المفوضات في العديد من السيناريوهات البرمجية لتحقيق المرونة وفصل الكود. فيما يلي أهم استخدامات المفوضات:

1. استدعاء الدوال بشكل غير مباشر
المفوضات تسمح لك بتعيين دالة ما واستدعائها لاحقاً دون الحاجة إلى معرفة تفاصيل تنفيذها.

مثال :

```
1  delegate void MyDelegate(string message);
2
3  static void ShowMessage(string message)
4  {
5      MessageBox.Show(message);
6  }
7      static MyDelegate msg =(s) => MessageBox.Show(s);
8  static void Main()
9  {
10     MyDelegate del = ShowMessage;
11     del("Hello, World!");
12 }
13
```

الشرح:

- هنا، del هو **delegate** يشير إلى الدالة ShowMessage.
- عند استدعاء del("Hello, World!")، يتم تنفيذ الدالة ShowMessage .

1. (UI Elements) إضافة عناصر واجهة المستخدم

نقوم بإضافة عناصر واجهة المستخدم مثل الأزرار

```

1 class Program : Form
2 {
3     Button b1, b2, b3;
4     TextBox t;
5
6     public Program()
7     {
8         CreateTool(t = new TextBox(), new Rectangle(50, 20, 60, 20), "fact", this);
9         CreateTool(b1 = new Button(), new Rectangle(10, 50, 60, 20), "", this);
10        CreateTool(b2 = new Button(), new Rectangle(120, 50, 60, 20), "abs", this);
11    }
12
13    void CreateTool(Control control, Rectangle rectangle, String txt, Control parent)
14    {
15        control.Text = txt;
16        control.Bounds = rectangle;
17        parent.Controls.Add(control);
18    }
19 }

```

2. (Events) الأحداث و (Event Handlers)

```

1 class Program : Form
2 {
3     Button b1, b2, b3;
4     TextBox t;
5
6     public Program()
7     {
8         CreateTool(t = new TextBox(), new Rectangle(50, 20, 60, 20), "fact", this);
9         CreateTool(b1 = new Button(), new Rectangle(10, 50, 60, 20), "fact", this);
10        CreateTool(b2 = new Button(), new Rectangle(120, 50, 60, 20), "abs", this);
11
12        b1.Click += (sender, e) => MessageBox.Show("Button 1 Clicked!");
13        b2.Click += (sender, e) => MessageBox.Show("Button 2 Clicked!");
14    }
15
16    void CreateTool(Control control, Rectangle rectangle, String txt, Control parent)
17    {
18        control.Text = txt;
19        control.Bounds = rectangle;
20        parent.Controls.Add(control);
21    }
22 }

```

3. Delegates والأحداث المخصصة (Custom Events)

```
1 delegate void ButtonClickHandler();
2
3 class Program : Form
4 {
5     Button b1, b2, b3;
6     TextBox t;
7     public event ButtonClickHandler OnClick;
8
9     public Program()
10    {
11        CreateTool(t = new TextBox(), new Rectangle(50, 20, 60, 20), "fact", this);
12        CreateTool(b1 = new Button(), new Rectangle(10, 50, 60, 20), "fact", this);
13        CreateTool(b2 = new Button(), new Rectangle(120, 50, 60, 20), "abs", this);
14
15        this.OnClick += SayHello;
16        b1.Click += (sender, e) => OnClick?.Invoke();
17    }
18
19    void SayHello() => MessageBox.Show("Hello!");
20
21    void CreateTool(Control control, Rectangle rectangle, String txt, Control parent)
22    {
23        control.Text = txt;
24        control.Bounds = rectangle;
25        parent.Controls.Add(control);
26    }
27 }
28
```

4. الديلجيت العام (Generic Delegates)

يمكنه التعامل مع أنواع مختلفة من البيانات (Generic Delegate) نقوم بتعريف ديلجيت عام



```
1  public delegate TResult MyFunc<in T1, in T2, out TResult>(T1 arg1, T2 arg2);
2
3  class Program : Form
4  {
5      MyFunc<int, int, int> add = (x, y) => x + y;
6      MyFunc<float, float, float> addd = (x, y) => x + y;
7
8      public Program()
9      {
10         MessageBox.Show(add(2, 2).ToString());
11         MessageBox.Show(addd(2.2f, 2.3f).ToString());
12     }
13 }
14
```

5. (Generic Classes)

يمكنه التعامل مع أنواع مختلفة من البيانات (Generic Class) نقوم بتعريف فصل عام

```
1  class Myint<T>
2  {
3      T value;
4      public void setValue(T value) { this.value = value; }
5      public T getValue() { return this.value; }
6  }
7
8  class Program : Form
9  {
10     Myint<int> myint = new Myint<int>();
11
12     public Program()
13     {
14         myint.setValue(222);
15         MessageBox.Show(myint.getValue().ToString());
16     }
17 }
```

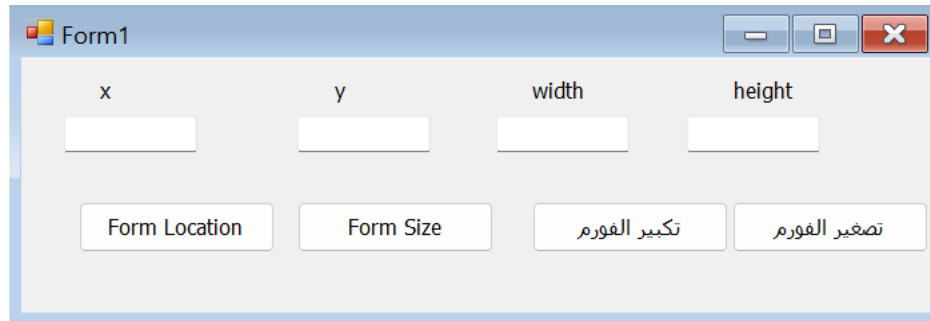
```

1  using System;
2  using System.Drawing;
3  using System.Windows.Forms;
4
5  namespace ConsoleApp3
6  {
7      delegate void ButtonClickHandler();
8      public delegate TResult MyFunc<in T1, in T2, out TResult>(T1 arg1, T2 arg2);
9
10     class Myint<T>
11     {
12         T value;
13
14         public void setValue(T value) { this.value = value; }
15         public T getValue() { return this.value; }
16     }
17
18     class Program : Form
19     {
20         Button b1, b2, b3;
21         TextBox t;
22         public event ButtonClickHandler OnClick;
23
24         MyFunc<int, int, int> add = (x, y) => x + y;
25         MyFunc<float, float, float> addd = (x, y) => x + y;
26         Myint<int> myint = new Myint<int>();
27
28         void SayHello() => MessageBox.Show("Hello!");
29         void SayGoodbye() => MessageBox.Show("Goodbye!");
30
31         public Program()
32         {
33             CreateTool(t = new TextBox(), new Rectangle(50, 20, 60, 20), "fact", this);
34             CreateTool(b1 = new Button(), new Rectangle(10, 50, 60, 20), "fact", this);
35             CreateTool(b2 = new Button(), new Rectangle(120, 50, 60, 20), "abs", this);
36
37             this.OnClick += SayHello;
38             this.OnClick += SayGoodbye;
39
40             b1.Click += (sender, e) => OnClick?.Invoke();
41             b2.Click += (sender, e) =>
42             {
43                 MessageBox.Show(add(2, 2).ToString());
44                 MessageBox.Show(addd(2.2f, 2.3f).ToString());
45                 myint.setValue(222);
46                 MessageBox.Show(myint.getValue().ToString());
47             };
48         }
49
50         void CreateTool(Control control, Rectangle rectangle, String txt, Control parent)
51         {
52             control.Text = txt;
53             control.Bounds = rectangle;
54             parent.Controls.Add(control);
55         }
56
57         static void Main(string[] args)
58         {
59             new Program().ShowDialog();
60         }
61     }
62 }
63

```

الواجب:

- 1- تطبيق المحاضرة
- 2- انشا الفورمات التالية في الكنسول ابليكيشن:
 - فورم يحتوي على الازرار الموضحة للتحكم بمكان وحجم الفورم وتكبيره وتصغيره.



- فورم يحتوي على الازرار الموضحة لعرض خصائص الفورم داخل MessageBox وأيضا لتحريكه على حسب الابعاد.



- فورم يحتوي على الازرار الموضحة لعرض خصائص البوتون داخل MessageBox وأيضا لتحريكه على حسب الابعاد.

