# 事件

[类](https://msdn.microsoft.com/zh-cn/library/0b0thckt.aspx)或对象可以通过事件向其他类或对象通知发生的相关事情。

发送（或引发）事件的类称为“发行者”

接收（或处理）事件的类称为“订户”。

没有订户的事件永远也不会引发。

事件通常用于通知用户操作，例如，图形用户界面中的按钮单击或菜单选择操作。

如果一个事件有多个订户，当引发该事件时，会同步调用多个事件处理程序。

在 .NET Framework 类库中，事件是基于 [EventHandler](https://msdn.microsoft.com/zh-cn/library/system.eventhandler.aspx) 委托和 [EventArgs](https://msdn.microsoft.com/zh-cn/library/system.eventargs.aspx) 基类的。

## 订阅事件方式

设计视图中,鼠标操作添加

代码添加

this.Load += new System.EventHandler(this.Form1\_Load);

lambda表达式

this.Click += (s,e) => { MessageBox.Show(

((MouseEventArgs)e).Location.ToString());};

请务必注意，如果使用匿名函数订阅事件，事件的取消订阅过程将比较麻烦。

## 取消订阅

publisher.RaiseCustomEvent -= HandleCustomEvent;

## 自定义事件

基本情况

1.创建自己的事件

public class CustomEventArgs : EventArgs

{

public CustomEventArgs(string s)

{

msg = s;

}

private string msg;

public string Message

{

get { return msg; }

}

}

2.在发布类中声明委托

供订阅者添加回调函数;

public delegate void CustomEventHandler(object sender, CustomEventArgs a);

3.声明一个事件字段;或是泛型;

public event CustomEventHandler RaiseCustomEvent;

触发事件方式

EventHandler<CustomEventArgs> handler = RaiseCustomEvent;//回调函数保存地方(委托可以添加方法数量)

e = new CustomEventArgs("Did something");

handler(this, e);//执行回调函数

//回调函数定义格式

void HandleCustomEvent(object sender, CustomEventArgs e)