C# - Event

In general terms, an event is something special that is going to happen. For example, Microsoft launches events for developers, to make them aware about the features of new or existing products. Microsoft notifies the developers about the event by email or other advertisement options. So in this case, Microsoft is a publisher who launches (raises) an **event** and **notifies** the developers about it and developers are the **subscribers** of the event and attend (**handle**) the event.

Events in C# follow a similar concept. An event has a publisher, subscriber, notification and a handler. Generally, UI controls use events extensively. For example, the button control in a Windows form has multiple events such as click, mouseover, etc. A custom class can also have an event to notify other subscriber classes about something that has happened or is going to happen. Let's see how you can define an event and notify other classes that have event handlers.

using System;

public delegate void EventHandler();

class Program

{

public static **event** EventHandler \_show;

static void Main()

{

// Add event handlers to Show event.

\_show += new EventHandler(Dog);

\_show += new EventHandler(Cat);

\_show += new EventHandler(Mouse);

\_show += new EventHandler(Mouse);

// Invoke the event.

\_show.Invoke();

}

static void Cat()

{

Console.WriteLine("Cat");

}

static void Dog()

{

Console.WriteLine("Dog");

}

static void Mouse()

{

Console.WriteLine("Mouse");

}

}