

البرمجة المعتمدة على الأحداث Event-driven programming هي نوع من البرمجة التي تعتمد على الأحداث زي ضغطات الماوس، أو ضغطات الكيبورد، أو إشارات من أجهزة استشعار، أو رسائل من برامج ثانية. في لغة C#، ده بيتنفذ عادة باستخدام الـ delegates والـ events.

ده بيشتغل في:

1. Delegates الـ delegate هو نوع بيمثل مراجع للطرق methods التي ليها قائمة معينة من المعاملات ونوع معين من القيم المرجعة. الـ delegates بتستخدم لتعريف طرق الاستدعاء العكسي callback methods وتنفيذ معالجة الأحداث.
2. Events الحدث event هو رسالة ببيعته object للإشارة لحدوث فعل معين. الكائن الذي بيرفع الحدث بيتسمى الناشر publisher والكائن الذي بيستقبل الحدث بيتسمى المشترك subscriber الأحداث بتعتمد على الـ delegates

```

using System;

public class Publisher
{
    public delegate void Notify();

    public event Notify OnNotify;

    public void DoSomething()
    {
        OnNotify?.Invoke();
    }
}

public class Subscriber
{
    public void Subscribe(Publisher publisher)
    {
        publisher.OnNotify += HandleNotification;
    }

    private void HandleNotification()
    {
        Console.WriteLine("تم استقبال الحدث");
    }
}

class Program
{
    static void Main()
    {
        Publisher publisher = new Publisher();
        Subscriber subscriber = new Subscriber();

        subscriber.Subscribe(publisher);

        publisher.DoSomething();
    }
}

```

فئة الـ Publisher بتعلن عن delegate اسمه Notify وحدث اسمه OnNotify

فئة الـ Subscriber بتشارك في الحدث OnNotify وبتوفر طريقة HandleNotification لمعالجة الحدث.

لما يتم استدعاء طريقة DoSomething في فئة الـ Publisher، بيرفع الحدث OnNotify، اللي بيشغل طريقة HandleNotification في فئة الـ Subscriber.