**54.Събития. Кратък пример.**

**Събитията** могат да се разглеждат като съобщения за настъпване на някакво действие. В компонентно-ориентираното програмиране компонен-тите изпращат събития (events) към своя притежател за да го уведомят за настъпването на интересна за него ситуация. Този модел е много характе-рен например за графичните потребителски интерфейси, където контро-лите уведомяват чрез събития други класове от програмата за действия от страна на потребителя. Например, когато потребителят натисне бутон, бутонът предизвиква събитие, с което известява, че е бил натиснат. Раз-бира се, събития могат да се предизвикват не само при реализиране на потребителски интерфейси. Нека вземем за пример програма, в която като част от функционалността влиза трансфер на файлове. Приключването на трансфера на файл може да се съобщава чрез събитие.

**Изпращачи и получатели**

Обектът, който предизвиква дадено събитие се нарича **изпращач на събитието (event sender)**. Обектът, който получава дадено събитие се нарича **получател на събитието (event receiver)**. За да могат да получават дадено събитие, получателите му трябва преди това да се абонират за него (subscribe for event).

За едно събитие могат да се абонират произволен брой получатели. Изпращачът на събитието не знае кои ще са получателите на събитието, което той предизвиква. Затова чрез механизма на събитията се постига по-ниска степен на свързаност (coupling) между отделните компоненти на програмата.

**Събитията в .NET Framework**

В компонентния модел на .NET Framework абонирането, изпращането и получаването на събития се поддържа чрез делегати и събития. Реализа-цията на механизма на събитията е едно от главните приложения на делегатите. Класът, който публикува събитието, дефинира делегат, който абонатите на събитието трябва да имплементират. Когато събитието бъде предизвикано, методите на абонатите се извикват посредством делегата. Тези методи обикновено се наричат **обработчици** на събитието. Делега-тът е multicast делегат, за да могат чрез него да се извикват много обра-ботващи методи (на всички абонати).

Извикването на събитие може да стане само в класа, в който то е дефинирано. Това означава, че само класът, в който се дефинира събитие, може да предизвика това събитие. Това е наложително, за да се спази шаблонът на Публикуващ/Абонати – абонираните класове се информират при промяна на състоянието на публикуващия и именно публикуващият е отговорен за разпращане на съобщенията за промяната, настъпила у него.