

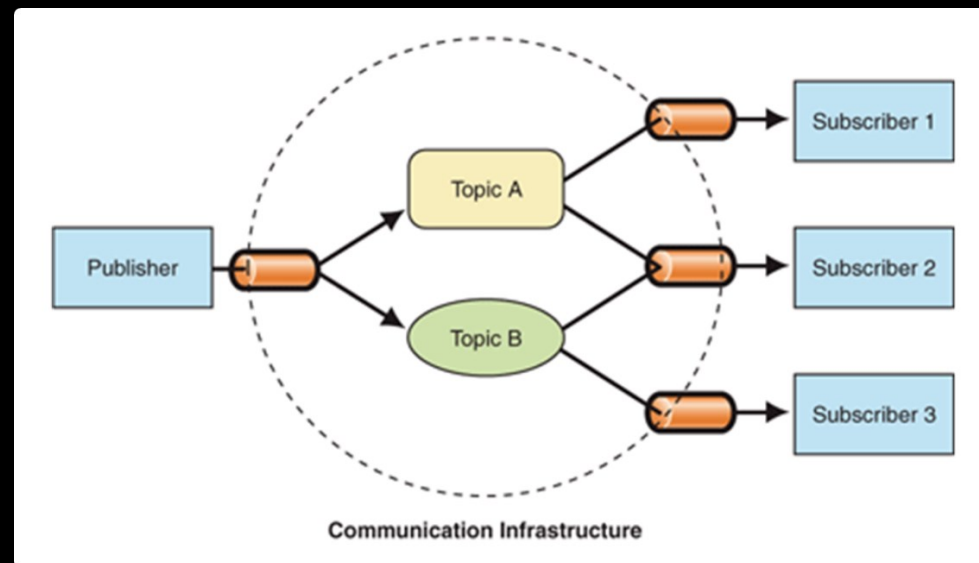
Комуникация между обекти. Въведение в събитийното програмиране. Делегати



Учителски екип

Обучение за ИТ кариера

<https://it-kariera.mon.bg/e-learning/>



Съдържание

1. Въведение в събитийното програмиране
2. Делегати



Какво са събитията?

- Събитията са уведомления (**известия**)
- Играят централна роля в .NET framework
- Дават начина за **задействане** на **уведомления** от потребителя или от обект



Какво са събитията?

- Събитията **сигнализират** появата на действие/обявяване
- Обектите, създаващи събития **не е необходимо изрично да знаят** кои **обекти** ще обработват събитието
- Събитията дават **EventArgs**(данни за събитието)



Какво са делегатите?

- Делегатът е специализиран клас, често наричан “указател към функция”
 - Той е променлива
- Лепилото/тръбопровода между събитието и обработчика му
- На основата на класа **`MulticastDelegateBase`**



Какво е обработчик на събитето?

- Обработчикът отговаря за получаване и обработване на данните от делегата
- Нормално получава два параметъра:
 - изпращач
 - данни на събитието (EventArgs)
- EventArgs отговарят за капсулирането на данните за събитието



Какво са делегатите?

- Делегатите са специални типове в C#, които държат референция (указател) към метод
- Типът данни, съдържащ функцията (метода) като нейна стойност
- Описва параметрите – приемани и връщаната стойност (сигнатура на метода)
- Делегатите наподобяват указателите към функции в C и C++
- В JavaScript променливите могат да държат функция

Създаване на делегати

- Потребителски дефинираните делегати съдържат ключовата дума **delegate**

```
public delegate void WorkPerformedHandler  
    (int hours, WorkType workType)
```

Трябва да е сходна с метода,
който я обработва (еднаква
сигнатура)

```
public void ManagerWorkPerformed  
    (int workHours, WorkType wType)
```


Какво е Делегат за пакетна обработка (Multicast)?

- Може да реферира повече от една делегирана функция

- Поддържа делегираните

Референции, използвайки

списък на извикванията

- Делегатите в списъка се извикват последователно



Създаване на инстанция на делегат

```
public delegate void WorkPerformedHandler  
                    (int hours, WorkType workType);
```

Делегат

```
WorkPerformedHandler dele =  
    new WorkPerformedHandler(WorkPerformed);
```

Инстанция

```
static void WorkPerformed(int hours, WorkType workType)  
{  
    Console.WriteLine("WorkPerformed called " +  
                      hours.ToString());  
}
```

обработчик

Присвояване на ламбда към делегат

- Ламбда изразите могат да се присвояват на делегат

```
delegate int AddDelegate(int a, int b);  
static void Main(string[] args)  
{  
    AddDelegate ab = ( a, b ) => a + b;  
    int result = ab(1, 1);  
    //result = 2  
}
```

Операции с делегати

- Извикване

```
WorkPerfHandler dele = new WorkPerfHandler(WorkPerformed);  
dele(5, WorkType.Golf);
```

- Добавяне към списъка с извиквания

```
var first = new WorkPerfHandler(WorkPerformed);  
var second = new WorkPerfHandler(WorkPerformedSecond);  
  
first += second;  
first(5, WorkType.Golf);
```


Ограничения за делегат

- Могат да сочат статични методи или инстанции на методи
- Могат да сочат към последователност от множество методи
- Използват се за изпълнение на извиквания от тип обратно извикване (callback)
- Използват се за реализацията на модела „публикация-абониране“
- Компонентите публикуват събитията си
 - Напр. **Button** публикува събития **Click** и **MouseOver**
 - Други компоненти се абонират за събитията
 - напр. **LoginForm** се абонира за **LoginButton.Click**

Примери за делегати

```
static void DoWork(WorkPerformedHandler del)
{
    del(5, WorkType.Golf);
}
static void WorkPerformed(int hours, WorkType workType)
{
    Console.WriteLine("WorkPerformed " + hours.ToString()
                      + " hours of " + workType);
}
//TODO: имплементирайте логика в следващите два метода
static void WorkPerformedSecond(int hours, WorkType workType)
static void WorkPerformedThird(int hours, WorkType workType)
```

Какво научихме?

- Събитията ни информират за настъпило действие/състояние в даден обект и дават информация, чрез **EventArgs**
- **Обработчикът на събитието** се е абонира за него и го обработва след като събитието настъпи
- Делегатът е **специализиран клас**, често наричан “указател към функция” използва се за обработка на събития



Комуникация между обекти. Делегати



Въпроси?



Министерство на образованието и науката (МОН)

- Настоящият курс (презентации, примери, задачи, упражнения и др.) е разработен за нуждите на Национална програма "**Обучение за ИТ кариера**" на МОН за подготовка по професия "Приложен програмист"



Министерство
на образованието
и науката



Национална
програма
„Обучение за
ИТ кариера“

- Курсът е базиран на учебно съдържание и методика, предоставени от **фондация "Софтуерен университет"** и се разпространява под свободен лиценз **CC-BY-NC-SA**



SoftUni
Foundation

