

# Шаблонни (типизирани) методи и интерфейси



Учителски екип

Обучение за ИТ кариера

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



# Съдържание

1. Шаблонни интерфейси
2. Шаблонни методи
3. Използване на default



# Шаблонни интерфейси

- Подобни са на шаблонните класове

```
interface IBox<T> {  
    void Add (T element);  
    ...  
}
```

```
class MyList : IBox<MyClass> {...}
```

```
class MyList<T> : IBox<T> {...}
```

# Шаблонни методи

- Може да имат какъв да е вход и връщан резултат

```
public List<T> createList(T item, int count) {  
    List<T> list = new List<T>();  
    for (int i = 0; i < count; i++) {  
        list.Add(item);  
    }  
    return list;  
}
```

- Статичните методи също **могат** да бъдат шаблонни
- Конструкторите и свойствата – **не**



# Задача: Създател на шаблонен масив

- Създайте клас **ArrayCreator** с един-единствен метод:
  - **static T[] create(int length, T item)**
- Той трябва да връща масив
  - С указаната дължина
  - Всички елементи трябва да бъдат от типа, подаден като параметър

# Решение: Създател на шаблонен масив

```
public static class ArrayCreator
{
    public static T[] Create<T>(int lenght, T item)
    {
        T[] array = new T[length];
        return array;
    }
}
```

# Задача: Универсална везна

- Създайте клас **Scale<T>** който:
  - Съдържа два елемента: **left** и **right**
  - Получава елементите чрез своя единствен конструктор:
    - **Scale(T left, T right)**
  - Има метод: **T GetHeavier()**
- По-големият от двата елемента е по-тежък
- **Проблем:** Ако елементите са равни, какво да върне?
  - Ако T е референтен тип, трябва да е **null**, а ако е числов - **0**



# Оператор Default(T)

- Връща **подразбиращата се стойност** за конкретния тип:
  - за референтни типове: **null**
  - за числови типове: **0**
  - за булев тип: **false**, за символен: **'\0'** и т.н.
- Т.е. нашата везна трябва да връща **default**, ако елементите са равни





# Решение: Универсална везна

```
public class Scale<T> where T : IComparable<T>
{
    private T left;
    private T right;

    public Scale(T left, T right)
    {
        this.left = left;
        this.right = right;
    }
    //TODO: продължава на следващия слайд
}
```

## Решение: Универсална везна (2)

```
public T GetHavier()  
{  
    if (left.CompareTo(right) > 0)  
    { return left; }  
    else if (left.CompareTo(right) < 0)  
    { return right; }  
    return default(T);  
}
```

# Обобщение

- Шаблонните интерфейси улесняват **взаимодействието** с кода
  - В същото време позволяват **проверка на типа**
- Шаблонните методи са **по-универсални**
- Статичните методи **могат** да бъдат шаблонни
- Конструкторите и свойствата – **не**
- **Default** връща подразбиращата се стойност на параметъра за типа



# Шаблонни (типизирани) методи и интерфейси



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

