

Методи

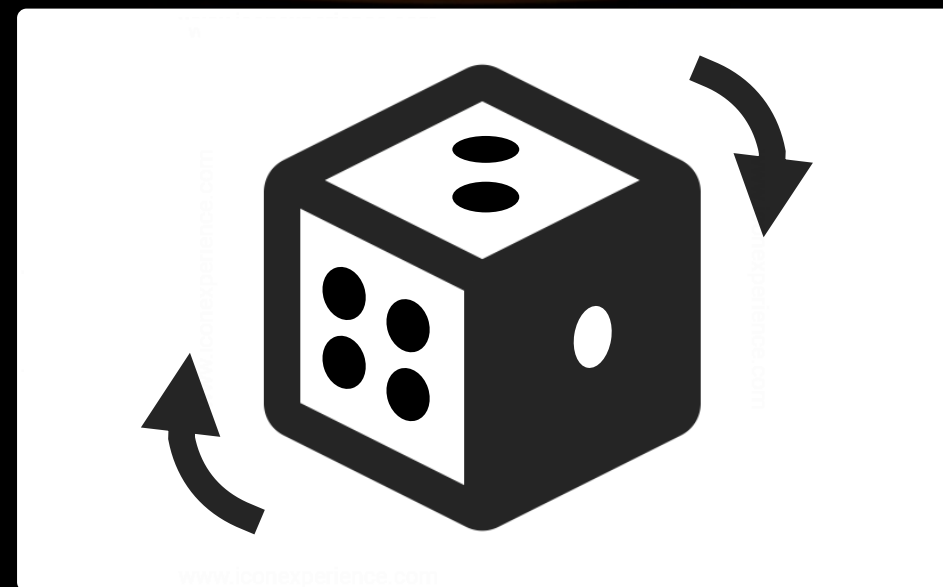
Описване на поведението на класа



Учителски екип

Обучение за ИТ кариера

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



Съдържание

1. Методи
2. Getter и Setter методи



Елементи на класа

- Клас се дефинира чрез **състояние и поведение**
- Полетата **съхраняват състоянието**
- Методите **описват поведението**

```
class Dice {  
    int sides;  
    string type;
```

Полета

```
    void Roll(){ ... }  
}
```

Метод

Методи

- Те са **изпълним код** (алгоритъм), който променя състоянието

```
class Dice {  
    public int sides;  
    private Random rnd = new Random();  
    public int Roll()  
    {  
        int rollResult = rnd.Next(1, this.sides + 1);  
        return rollResult;  
    }  
}
```

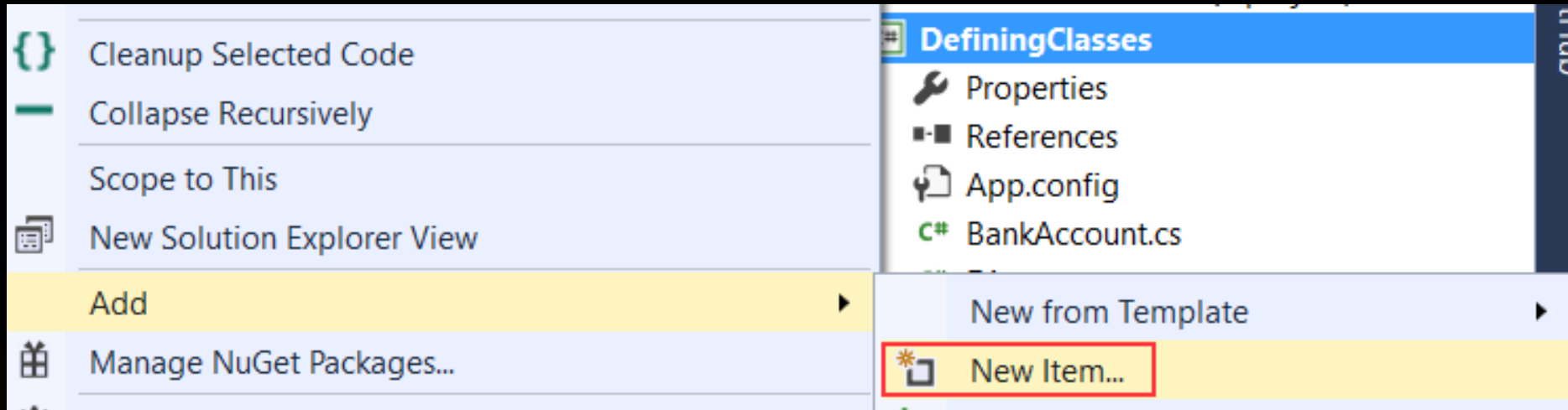
this сочи към
тази инстанция

Задача: Дефинирайте клас `Person`

- Задача: Дефинирайте клас `Person`, като за него пазете информация за името и възрастта на човек и реализирайте единствено действието `IntroduceYourself()`, което отпечатва представяне на човека. След това създайте и използвайте обект от класа `Person`.

Решение: Дефинирайте клас Person (1)

- Нека първо да създадем файл за този клас:
[Project] → [Add Class] или:
десен бутон върху проекта [Add] → [New Item] → [Class]



- Внимавайте с именуването на класа

Решение: Дефинирайте клас Person (2)

```
class Person {  
    private string name;  
    private int age;  
    public String Name { //реализираме свойство Name  
        get { return name; }  
        set { name = value; }  
    }  
    public int Age { //реализираме свойство Age  
        get { return age; }  
        set { age = value; }  
    }  
    public void IntroduceYourself() {  
        Console.WriteLine("Здравейте! Аз съм {0} и съм на {1} години.",  
name, age);  
    }  
}
```

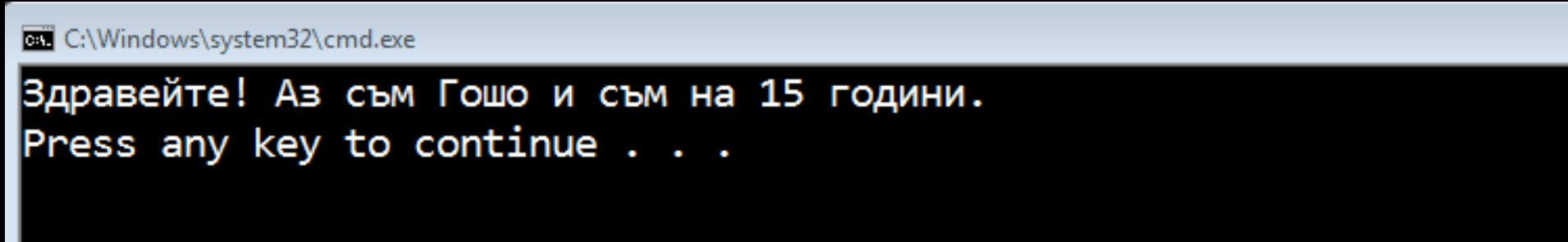
Решение: Дефинирайте клас Person (3)

- Сега е време да използваме класа и да направим обект в Main метода ни в **Program.cs**:

```
static void Main(string[] args) {  
    Person firstPerson = new Person();  
    firstPerson.Name = "Гошо";  
    firstPerson.Age = 15;  
  
    firstPerson.IntroduceYourself();  
}
```


Решение: Дефинирайте клас Person (4)

- Ако сте работили правилно ще получите:

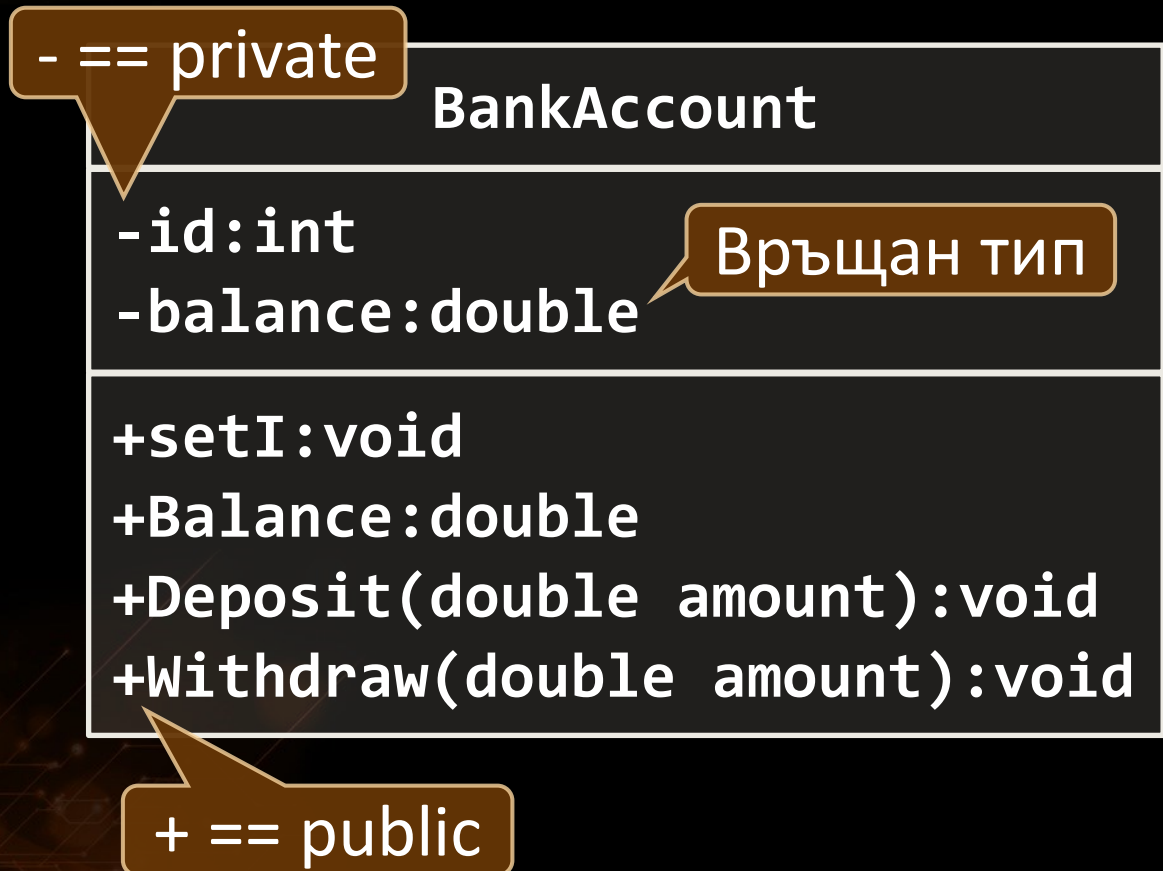


```
C:\Windows\system32\cmd.exe
Здравейте! Аз съм Гошо и съм на 15 години.
Press any key to continue . . .
```

- Аналогично създайте `secondPerson` и `thirdPerson` и извикайте `IntroduceYourself` и за тях

Задача: Getter-и и Setter-и

- Създайте клас **BankAccount**



```
public static void Main()
{
    BankAccount acc = new BankAccount();

    acc.ID = 1;
    acc.Deposit(15);
    acc.Withdraw(5);

    Console.WriteLine(acc.ToString());
}
```

Предефинирайте
toString()

Решение: Getter-и и Setter-и

```
private double balance;  
public void Deposit(double amount)  
{  
    this.balance += amount;  
}  
public void Withdraw(double amount)  
{  
    this.balance -= amount;  
}  
public override string ToString()  
{  
    return $"Account {this.id}, balance {this.balance}";  
}
```

Какво научихме днес?

- Методите описват поведението на обектите
- Методите може да променят състоянието на обектите или да го достъпват и анализират
- Getter / setter методите са за достъп до и промяна на полетата на обекта



Методи



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

