



Национална програма  
"Обучение за ИТ умения и кариера"  
<https://it-kariera.mon.bg>

Министерството на  
образованието и науката  
<https://www.mon.bg>



# Дефиниране на КЛАСОВЕ

Defining Classes

# Съдържание

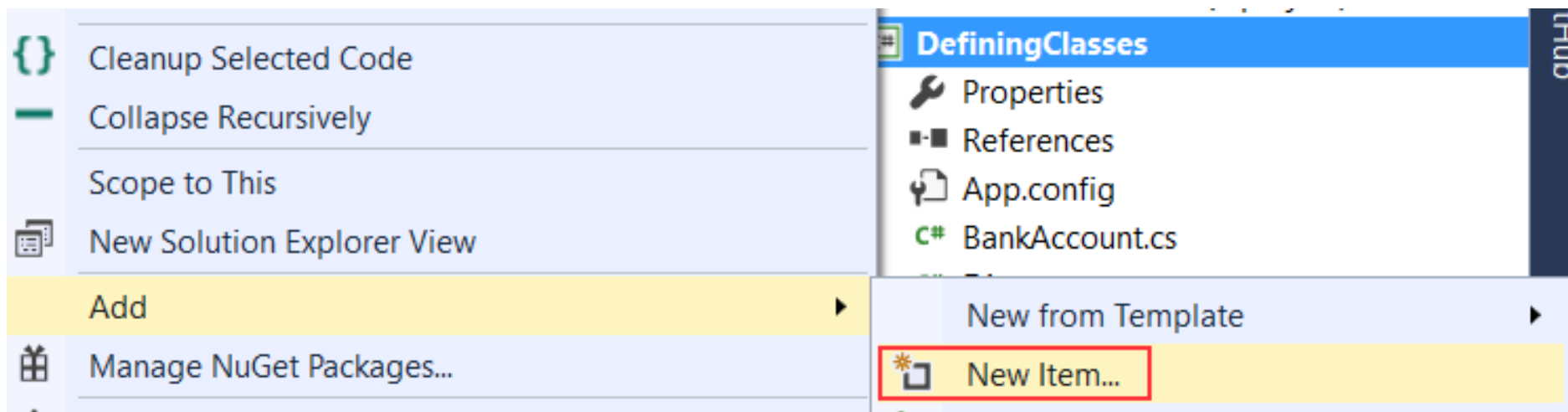
1. Дефиниране на клас **Person**
2. Използване на обекти
3. Дефиниране на по-сложен клас

# 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 отпечатава представяне на човека. След това създайте и използвайте обект от класа **Person**.

# Решение: Дефинирайте клас Person [1/4]

- 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 [2/4]

```
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);  
    }  
}
```

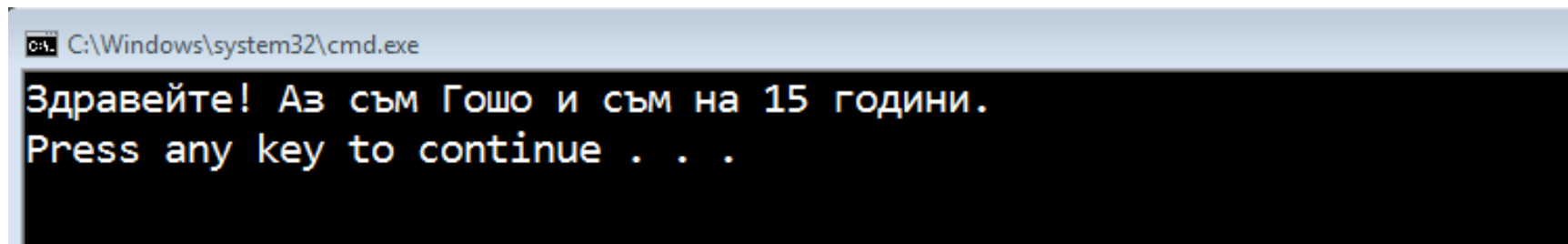
## Решение: Дефинирайте клас Person [3/4]

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 [4/4]

- 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
- За момента няма да акцентираме на теоретичния смисъл на кода от по-рано – той ще се уточни допълнително в следващите теми

# Задача: Човекът и неговите пари

Задача: Дефинирайте клас **Person**, като за него пазете информация за името и възрастта на човек, както и информация за банкови сметки (клас **BankAccount**). Направете метод **GetBalance()**, който дава информация каква е общата стойност на парите, които притежава човек.



# Решение: Човекът и неговите пари [1/5]

- Първо ще създадем файл за клас **BankAccount**.
- След това ще направим възможно в **Person** да се пази списък от банковите сметки на човека
- Последно в класа **Person** ще добавим метод, който изчислява сумата от балансите на всички сметки в списъка

## Решение: Човекът и неговите пари [2/5]

```
class BankAccount {  
    private int id;  
    private double balance;  
  
    public int ID {  
        get { return id; }  
        set { id = value; }  
    }  
  
    public double Balance {  
        get { return balance; }  
        set { balance = value; }  
    }  
}
```

# Решение: Човекът и неговите пари [3/5]

В [Person.cs](#) добавете поле, в което ще се пази списък от банковите сметки, както и свойство:

```
class Person {  
    // TODO: добавете полета за име и възраст  
    private List<BankAccount> accounts =  
        new List<BankAccount>();  
    //TODO: добавете свойства за име и възраст  
    public List<BankAccount> Accounts {  
        get { return accounts; }  
        set { accounts = value; }  
    }  
}
```

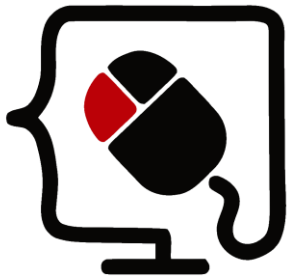
## Решение: Човекът и неговите пари [4/5]

В [Person.cs](#) добавете метода, който ще изчислява стойността на сумата от балансите на всички сметки:

```
class Person {  
    //...  
    public double GetBalance() {  
        return accounts.Sum(element => element.Balance);  
    }  
}
```

# Решение: Човекът и неговите пари [5/5]

- Създайте три обекта от клас **Person** в **Main()**.
- Създайте един обект от клас **BankAccount**, задайте му **ID** и баланс и го добавете към първия човек, използвайки метода **Add()** - не забравяйте, че **.Accounts** е списък и има метод **Add()**
- Аналогично за втория човек създайте две сметки и му ги добавете, а за третия – три;
- Изкарайте трите суми – състоянието на първия, втория и третия човек



Национална програма  
"Обучение за ИТ умения и кариера"  
<https://it-kariera.mon.bg>

Министерството на  
образованието и науката  
<https://www.mon.bg>



Документът е разработен за нуждите на Национална програма "Обучение за ИТ умения и кариера" на Министерството на образованието и науката (МОН) и се разпространява под свободен лиценз CC-BY-NC-SA (Creative Commons Attribution-Non-Commercial-Share-Alike 4.0 International).