

Капсулация

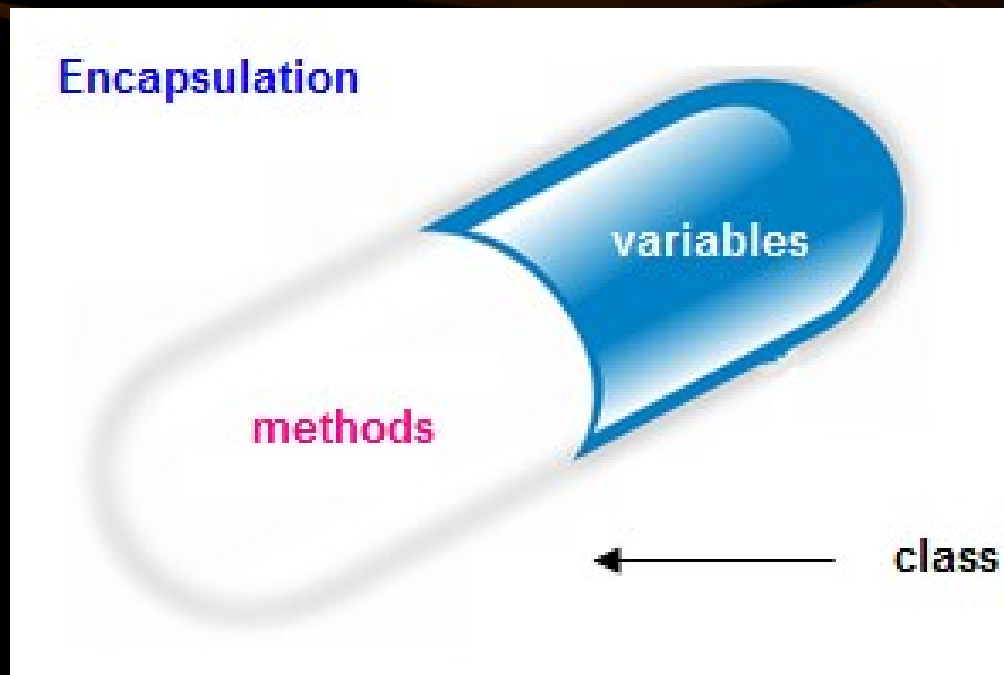
Капсулацията и ползите от нея



Учителски екип

Обучение за ИТ кариера

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



Съдържание

1. Какво е капсулация?
2. Ключова дума **this**



Капсулация

- Процесът на **обединяване** на кода и данните в едно **цяло** (обект)
- Полетата на обекта **трябва да са private**

```
private int age;
```



- Използване на **getters** и **setters** за достъп до данните

```
class Person  
{  
    public int Age => return this.age  
}
```



Капсулация – Примери

Полетата трябва да са **private**

Person
<code>-name: string</code> <code>-age: int</code>
<code>+Person(string name, int age)</code> <code>+Name: string</code> <code>+Age: int</code>

Задача: Клас Creature

- Създайте клас **Creature**
- Creature трябва да има полета **name**, **years**, **areal**, съответно за име, възраст и местообитание
- Създайте методи за достъп до обектите **getters** и **setters** за полетата

Creature
<pre>-name : string -years: int -areal: string</pre>
<pre>+setName(string value) +setYears(int value) +setAreal(string value) +getName(): string +getYears(): int +getAreal(): string</pre>

Решение: Клас Creature

```
private string name;
private int years;
private string areal;

public string Name
{
    get { return this.name; }
    set { this.name = value; }
}
public int Years
{
    get => this.years;
    set => this.years = value;
}
```

```
public string Areal
{
    get
    {
        return this.areal;
    }

    set
    {
        this.areal = value;
    }
}
```

Ключова дума This

- **this** е препратка към текущия обект
- **this** може да сочи към променлива, която е инстанция (представител) на текущия клас

```
public Person(string name)
{
    this.name = name;
}
```

- **this** може да се предава като аргумент в метод или като извикване на конструктор
- **this** може да се върне като стойност на метод

Ключова дума This (2)

- **this** може да извика метод на текущия клас

```
private string FirstName
{
    get { return this.fname }
}
public string FullName
{
    return this.FirstName + " " + this.LastName
}
```

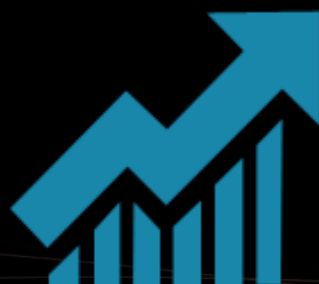

Ключова дума This (3)

- **this** може да извиква конструктор на текущия клас

```
public Person(string firstName, string lastName)
{
    this.firstName = firstName;
    this.lastName = lastName;
}
public Person (string fname, string lname, int age)
    :this (fname, lname);
{
    this.age = age;
}
```

Обобщение – ползи от капсулацията

- Капсулацията скрива имплементацията (реализацията на обекта - неговите компоненти – полета, свойства, методи)
- Капсулацията намалява сложността
- Осигурява структурните промени да останат локални
- С **this** можем да управляваме инстанциите



Капсулация – This



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

