ЛЕКЦИЯ I

JAVA ООП: Основи на ООП

Теодор Костадинов

Съдържание

Що е то?

- Парадигма начин на структуриране на кода чрез класове и обекти
- ООП моделира обектите от реалния свят и взаимоотношенията между тях

Стол - предмет - обект

SOFTWARE ACADEMY
Educational Center Net It

- Цвят
- Материал
- Брой крака
- Позиция
- Може да бъде местен
- Празен или зает

Защо се ползва?

SOFTWARE ACADEMY
Educational Center Net It

- Добро структуриране на кода
- Намалява сложността на кода
- Позволява преизползване на кода
- Може да се постигне абстрактност

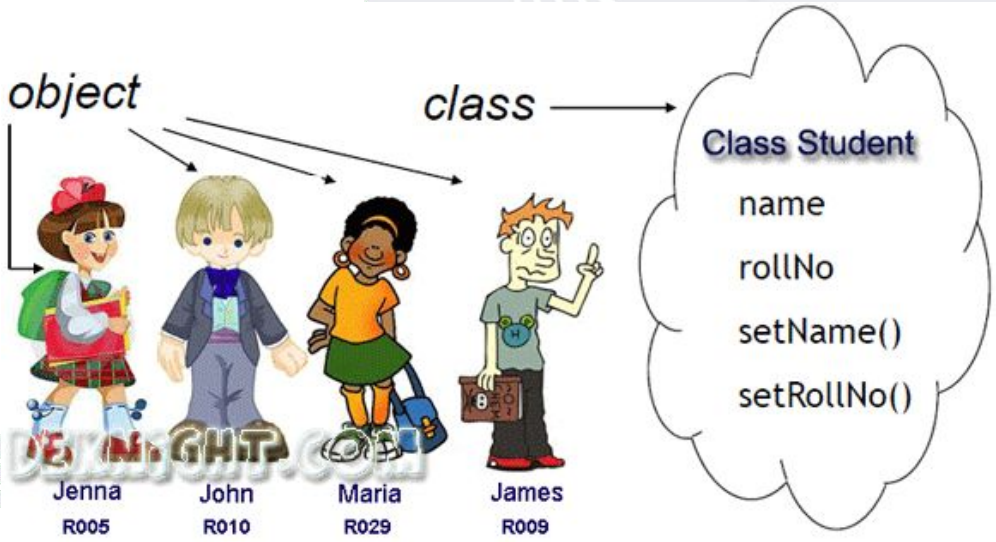
Принципи на ООП

- Капсулация: знае се КАКВО може да прави компонента, а не как
- Наследяване: един обект със същите свойства като друг може да го наследи
- Полиморфизъм: позволява унифицирано извършване на действия над различни обекти
- Абстракция: процесът по скриване на детайлите и показвайки само необходимото на потребителя

Какво е клас?

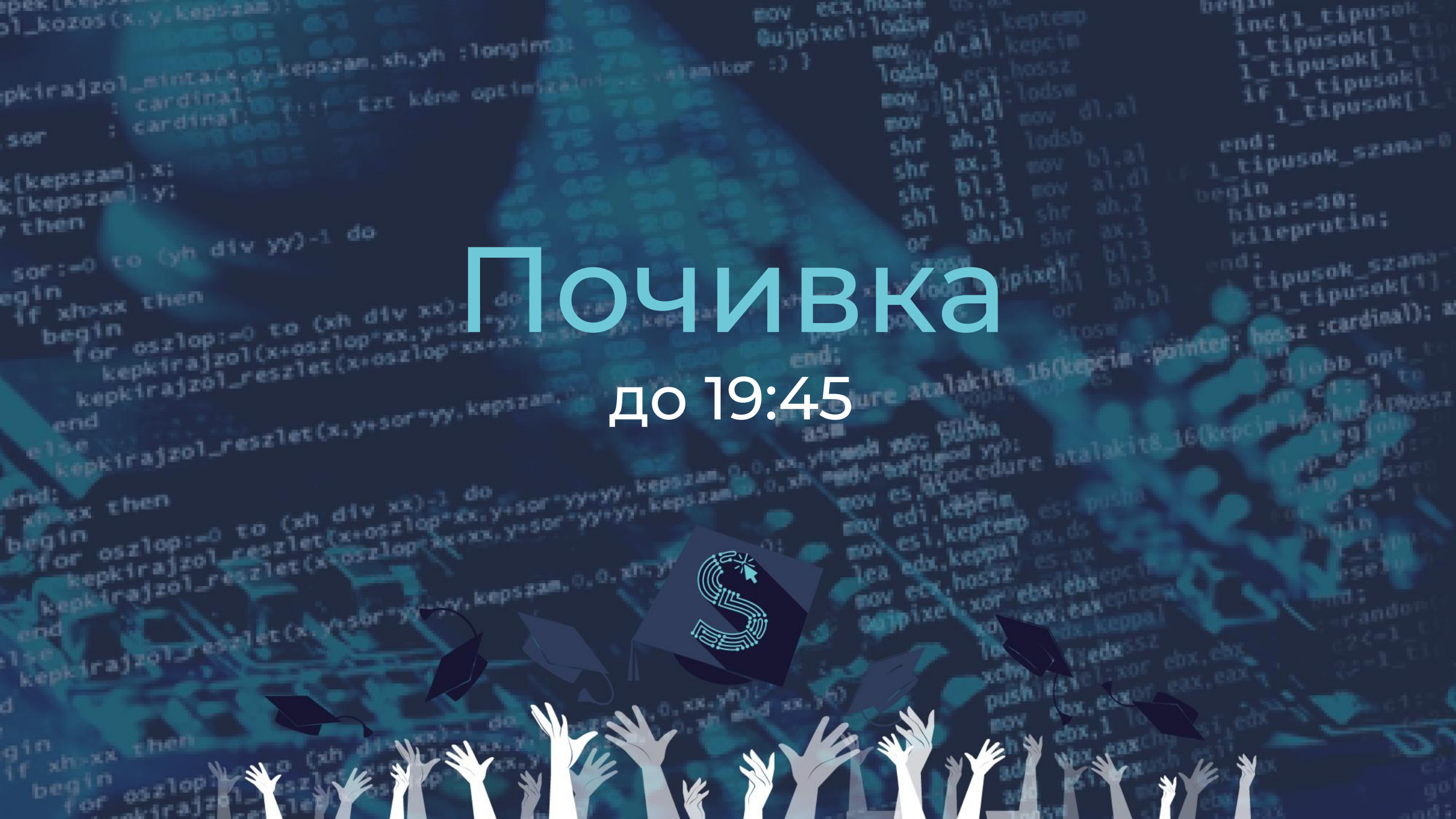
- Класът е шаблон от който създаваме обект
- Обекта е конкретен елемент от даден клас. Нарича се още инстанция на класа

```
Student pesho = new Student();
Chair officeChair = new Chair();
officeChair.color = ”green”;
```



Почивка

до 19:45



Какво е клас?

```
class Chair {
        String material = “wood”;
        int positionX = 5;
        int positionY = 4;
        String matColor = “red”

void moveChair ( int x, int y) {...}

}
```

Какво има в един клас?

SOFTWARE ACADEMY
Educational Center Net It

Полета:

- String name;
- int age;

Методи:

- void study(){...}
- void doHomework(){...}

Конструктор

Getter-и и Setter-и

Конструктор

SOFTWARE ACADEMY
Educational Center Net It

Конструктора е метод, който се извиква автоматично при създаването на обект от класа и само тогава

Използва се за да се зададат първоначални стойности на полетата и за да се направят първоначални настройки на класа

Един клас може да има много конструктори

SOFTWARE ACADEMY
Educational Center Net It

- Напишете клас Dog, да има поне 4 полета и поне 2 метода. Да има и конструктор.

Ами сега?

Задача за упражнение

Нива на достъп

Modifier	Class	Package	Subclass	World
public	✓	✓	✓	✓
protected	✓	✓	✓	✗
no modifier	✓	✓	✗	✗
private	✓	✗	✗	✗

Мутатори (getters & setters)

SOFTWARE ACADEMY
Educational Center Net It

Ограничават достъпа до полетата.

Предотвратяват унищожаване на данни

Или подмяната им (още по-лошо!!!)

```
private int age;

public getAge() {
    return age;
}

public setAge(int newAge) {
    age = newAge;
}
```

Оператор this

Позволява обръщание към полетата на класа от вътрешността на класа (обекта)

```
private int age;

public getAge() {
    return this.age;
}

public setAge(int newAge) {
    this.age = newAge;
}
```

- Направете кучкарник. Нека може да се добавя ново куче и да се осиновява. Нека кучетата имат собствени характеристики. Нека имат картинка.

ВЪПРОСИ?

Резюме

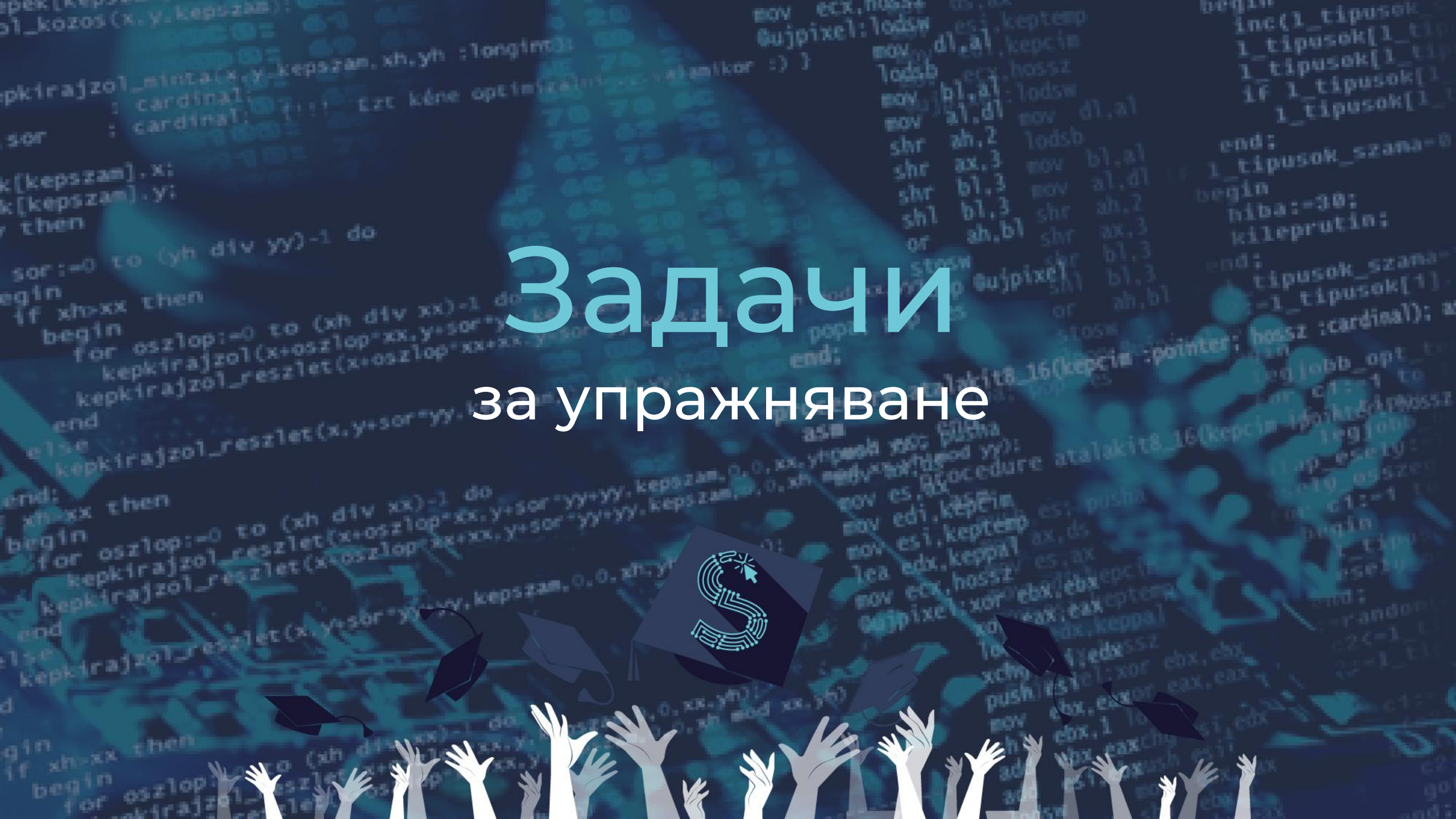
- Всичко в ООП са обекти
- Кода трябва да се разделя по обекти, така че да има смисъл за конкретния обект.
- Трябва да се спазват принципите на ООП.

Ресурси

- Docs
- GitHub Repo with Demos
-

Задачи

за упражняване



Задача 1

Напишете клас Circle с поле радиус и методи getPerimeter и getArea

Напишете клас Rectangle, да е подобен на Circle

Домашно

Качвайте домашното си в ГитХъб и слагайте линка тук:

https://forms.gle/AcvCptCbSDizr2Ay6

Задача 1

SOFTWARE ACADEMY
Educational Center Net It

Дефинирайте клас Human с полета name, EGN, age и съответните getter-и и setter-и. Дефинирайте класове Student и Teacher със същите полета. Измислете по 2 специфични полета и по 1 метод за тях. Създайте обекти от тип Student и Teacher в тестов клас. Задайте стойности на полетата, извикайте методите.

БЛАГОДАРЯ
ЗА ВНИМАНИЕТО!