

Полиморфизм

Профессия Java-разработчик на Hexlet

Преподаватель: Яковлев Егор

Вопросы к лекции:

- Как представлять программу как объекты?
- Как обращаться с разными типами как с одним?
- Какие еще задачи решает полиморфизм?

План

1. Принципы ООП - начало
2. Полиморфизм
3. Зачем нужен полиморфизм?

Принципы ООП - начало

- вокруг нас объекты
- каждый объект создается по чертежу, схеме или шаблону - *классу*

Принципы ООП:

- полиморфизм
- наследование
- инкапсуляция

Полиморфизм

“ Полиморфизм - способность функции обрабатывать данные разных типов

Википедия

”

“ Параметрический полиморфизм - способность функции, исполнять один и тот же код для всех допустимых типов аргументов (аргументов разных классов или примитивов)

Википедия

”

Полиморфизм: пример

```
// Car - Bmw, Kia, Renault  
// Car - interface  
public void goCar(Car car) {  
    car.go();  
    System.out.println("Car is going.");  
}
```

Полиморфизм

Демо

Зачем нужен полиморфизм?

- подмена объектов
- расширяемость программы
- за счет общего интерфейса можно "сгруппировать" объекты разных классов

Домашнее задание

```
hexlet program download java subtyping  
hexlet program submit java subtyping
```

Вопросы?