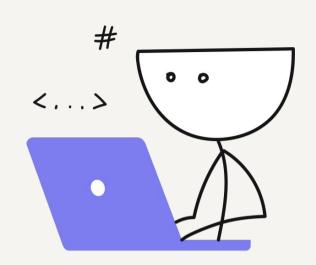


# Наследование



Наставник:

Полковников Дмитрий

#### Мотивация

- Наследование является одним из принципов ООП.
- Наследование позволяет удобно переиспользовать код и избегать дублирования.
- Так же как и в случае с интерфейсами, наследование важно знать для понимания некоторых полезных шаблонов проектирования.



#### План

- Проблематика
- Определение
- Синтаксис
- Работа с конструкторами
- Демо



## Проблематика

#### Задача:

Реализовать структуру персонала университета, включающую сотрудников и студентов

```
public class Employee {
    public String name;
    public String surname;
    public int age;
    public String title;
public class Student {
    public String name;
    public String surname;
    public int age;
    public int semesterNumber;
```

### Определение

Наследование (англ. inheritance) — концепция ООП, согласно которой тип данных может наследовать данные и функциональность некоторого существующего типа, способствуя повторному использованию компонентов программного обеспечения.

Наследование — механизм, позволяющий создавать классы на основе других классов.

ЯП Java позволяет наследоваться только от одного класса.



#### Синтаксис

```
// Класс родитель
public class Person {
    private String name;
    private String surname;
    private int age;
//Класс-наследник "Студент"
public class Student extends Person {
    public int semesterNumber;
//Класс-наследник "Сотрудник"
public class Employee extends Person {
    public String title;
```



## Работа с конструкторами

**Кличевохорскловой конструктора-**происходит первой строкой конструкторанаследника с помощью ключевого слова super

Если у класса-родителя используется конструктор без параметров или конструктор по умолчанию, то его вызов происходит неявно.

Ключевое слово super является ссылкой на объектродитель и позволяет использовать поля и методы класса-родителя.



#### Демо

- Среда разработки: Intellij Idea 2022.3
- SDK: OpenJDK 17.0.5



### Спасибо за внимание

