

# Java Professional

lecture #2. Inheritance. Abstract classes. Polymorphism.  
Mentor: Iurii Avramenko

## lecture #2. Inheritance. Abstract classes. Polymorphism.

- Inheritance
  - Важная терминология
  - Использование наследования
  - Типы наследования
  - Факты и только факты
- Abstract classes
  - Understanding. Syntax
  - Important Conclusions
  - Abstract class VS Interface
- Polymorphism
  - overloading
  - overriding

## Наследование в Java (Inheritance)

- Это аспект в java, с помощью которого одному классу разрешено наследовать (перенять) свойства (поля и методы) другого класса.
- С помощью наследования можно расширить функционал уже имеющихся классов

Ключевое слово, используемое для наследования, — **extends**

## Терминология и свойства

- Super Class: класс, функции которого наследуются, называется **суперклассом**.
- Sub Class: класс, который наследует другой класс, называется **подкласс**.
- Reusability: Наследование поддерживает концепцию «**повторного использования**»

### Свойства:

1. Наследуем только один класс. Один класс - один родитель.
2. Наследуется все кроме private переменных и private методов.
3. Переопределить @Override метод класса-родителя.
4. Вызываем методы родителя через ключевое слово super.
5. Запрещаем наследование – **final**.



Как использовать

Ключевое слово, используемое для наследования, — **extends**.

```
class Ananas extends Fruit
```

Класс Fruit является базовым классом, класс Ananas является производным классом, который расширяет класс Fruit.

При создании объекта класса Ananas для копия всех полей суперкласса выделяется память в этом объекте.


??? вопрос

Сколько объектов создается?



## Типы наследования

- 1. Одиночное наследование.
- 1. Многоуровневое наследование.
- 1. Иерархическое наследование.
- 1. Множественное наследование



Факты! Коллеги, только факты о наследовании!

**!Суперкласс по умолчанию:** кроме класса Object, у которого нет суперкласса, каждый класс имеет один и только один прямой суперкласс (одионое наследование).

**!Суперкласс может быть только один:** Суперкласс может иметь любое количество подклассов. Но подкласс может иметь только один суперкласс.

**!Наследование конструкторов:** подкласс наследует все члены (поля, методы и вложенные классы) от своего суперкласса.

Конструкторы не являются членами, поэтому они **НЕ** наследуются подклассами.

**!Наследование private:** подкласс не наследует закрытые члены своего родительского класса. (доступ может быть только через геттеры и сеттеры)

