

# Прекрасный мир Java

Часть 2.

Объекты, классы, наследование...

Объекты, классы, наследование...

Зачем?

# Как создать объект?

Формальное описание структуры объекта (набор методов, полей и их типы) называется классом.

Соответственно, каждый объект относится к какому-то классу (то есть конкретный объект является экземпляром некоторого класса).

Для создания объекта требуется:

- Описать класс (дать формальное описание полей и методов класса)
- Создать (в памяти машины) экземпляр объекта

Часто формальное описание класса делят на две части: прототип (фиксирует набор полей и сигнатуры методов) и реализацию (описание кода методов).

```
public class Message {  
    private String strMsg;  
  
    public Message(String currentStrMsg) {  
        setMessage(currentStrMsg);  
    }  
  
    public String getMessage() {  
        return strMsg;  
    }  
  
    public void setMessage(String currentStrMsg) {  
        strMsg = currentStrMsg;  
    }  
  
    public static void main(String[] args) {  
        Message helloWorld = new Message("Hello, World!");  
        System.out.println(helloWorld.getMessage());  
    }  
}
```

```
int myInt =
```

```
int myInt = 5;  
Integer myInteger =
```

```
int myInt = 5;
```

```
Integer myInteger = new Integer(5);
```



# Размещение объекта (сущности) класса в памяти

- Заголовок объекта
- Память для примитивных типов
- Память для ссылочных типов
- Байты выравнивания

Размер любого объекта кратен 8 байтам.

# Модификаторы доступа

- public
- protected
- private

# static

Статические члены класса используются совместно всеми экземплярами класса.

Статические методы – это методы, которые не связаны с конкретным экземпляром класса.

# Константы в Java

## Ключевое слово `final`

Константы в Java реализуются как члены класса с модификаторами `static` и `final`.

Модификатор `static` позволяет использовать константу без создания экземпляра класса.

Модификатор `final` запрещает изменение значения члена класса (то есть, собственно, и формирует константу).

```
public class ConstTest {  
    public static final double PI = 3.1415;  
}
```

Константы единственный случай когда нарушается правило Camel Case.

# Экземпляры как контейнеры данных

Экземпляр класса = поля + методы

Как в методе добраться до полей  
экземпляра?

```
public class Test {  
  
    private int iValue;  
  
    public void setValue(int iValue) {  
        this.iValue = iValue;  
    }  
  
    public int getValue(){  
        return iValue;  
    }  
}
```

```
public static void main(String[] args) {  
  
    Test t1 = new Test();  
    t1.setValue(123);  
  
    Test t2 = new Test();  
    t2.setValue(321);  
}
```



