

1. <https://prog-cpp.ru/oop/>
<https://thecode.media/new-oop-inp/>
2. <https://skillbox.ru/media/base/klassy-i-obekty-v-java/>

В Java класс — это основная структура, используемая для создания объектов. Он представляет собой шаблон для создания экземпляров, обеспечивая их свойства (поля) и методы (функции), которые определяют их поведение. Класс объединяет данные и методологию обработки данных в одном логическом объединении.

-- Основные понятия о классах в Java:

1. Определение класса:

Класс определяется с помощью ключевого слова `class`, после которого следует имя класса. Имя класса должно начинаться с заглавной буквы по соглашению (например, `Car`, `Person`).

2. Поля класса (атрибуты):

Поля класса — это переменные, которые определяют состояние объекта. Они могут быть `private`, `protected`, или `public` в зависимости от уровня доступа.

3. Конструкторы:

Конструктор — это специальный метод, который вызывается при создании объекта. Он используется для инициализации полей объекта. Конструкторы имеют то же имя, что и класс, и не имеют возвращаемого типа, включая `void`.

4. Методы:

Методы — это функции, которые определяют поведение объекта. Они могут принимать параметры и возвращать значения. Методы могут быть определены как `static` (принадлежат классу, а не экземпляру) или экземпляльными.

5. Создание объектов:

Объекты создаются с использованием оператора `new`, за которым следует вызов конструктора.

6. Наследование:

Java поддерживает концепцию наследования, позволяющую создавать новые классы (производные классы) на основе существующих классов (базовых классов). Для этого используется ключевое слово `extends`.

7. Интерфейсы и абстрактные классы:

Java поддерживает интерфейсы и абстрактные классы, которые позволяют реализовать многоуровневую иерархию объектов и обеспечить полиморфизм.

3. <https://metanit.com/cpp/tutorial/5.7.php>

<https://studfile.net/preview/16555742/page:5/>

<https://metanit.com/java/tutorial/2.7.php>

В Java класс состоит из различных членов, которые могут включать поля (переменные) и методы (функции). Члены класса являются основными строительными блоками, которые определяют состояние и поведение объектов, создаваемых на основе этого класса. Давайте подробнее рассмотрим различные члены класса.

-- 1. Поля класса (атрибуты)

Поля класса, или атрибуты, представляют состояние объекта и определяются в теле класса. Они могут иметь разные модификаторы доступа (`private`, `protected`, `public` или `default`), что определяет, откуда доступ к ним может быть получен.

Пример:

```
public class Car {
    private String color; // Поле, доступное только внутри класса
    private String model; // Поле, доступное только внутри класса
    public int year;      // Публичное поле, доступное из других классов

    // Конструктор
    public Car(String color, String model, int year) {
        this.color = color;
        this.model = model;
        this.year = year;
    }
}
```

-- 2. Методы класса

Методы определяют поведение объекта и содержат инструкции, которые будут выполнены. Они могут возвращать значения или не возвращать ничего (в этом случае их возвращаемый тип будет void). Методы также могут принимать параметры.

Пример:

```
public class Car {
    // Поля
    private String color;
    private String model;

    // Конструктор
    public Car(String color, String model) {
        this.color = color;
        this.model = model;
    }

    // Метод для вывода информации об автомобиле
    public void displayInfo() {
        System.out.println("Model: " + model + ", Color: " + color);
    }

    // Метод для изменения цвета
    public void changeColor(String newColor) {
        this.color = newColor;
    }

    // Метод для получения цвета
    public String getColor() {
        return color;
    }
}
```

-- 3. Конструкторы

Конструктор — это специальный метод, который вызывается при создании объекта. Конструкторы имеют такое же имя, как и класс, и не имеют возвращаемого типа. С помощью конструкторов можно инициализировать поля класса.

Пример:

```
public Car(String color, String model) {
    this.color = color;
    this.model = model;
}
```

```
}
```

-- 4. Статические члены класса

Члены класса могут быть статическими, что означает, что они принадлежат самому классу, а не конкретным экземплярам. Статические члены могут быть доступны без создания экземпляра класса.

Пример:

```
public class Car {
    private String color;
    private String model;
    public static int count = 0; // Статическая переменная для подсчета автомобилей

    public Car(String color, String model) {
        this.color = color;
        this.model = model;
        count++; // Увеличение счетчика при создании нового объекта
    }

    public static void displayCount() {
        System.out.println("Total cars created: " + count);
    }
}
```

-- 5. Вложенные классы

Классы могут содержать другие классы, которые называются вложенными классами. Вложенные классы могут быть статическими или нестатическими и могут использовать поля и методы внешнего класса.

Пример:

```
public class OuterClass {
    private String outerField = "Outer";

    public class InnerClass {
        public void display() {
            System.out.println("Accessing: " + outerField); // Вложенный класс имеет доступ к
            полям внешнего класса
        }
    }
}
```

-- Заключение

Члены класса в Java формируют основу для построения объектов, и их правильное определение и использование позволяет реализовывать концепции объектно-ориентированного программирования, такие как инкапсуляция, наследование и полиморфизм. Внимательное проектирование членов класса способствует созданию удобного, поддерживаемого и расширяемого кода.

— — — — —

4. <https://javarush.com/groups/posts/2137-porjadok-deystviy-pri-sozdanii-obhhehta>
<https://gb.ru/blog/metody-java/>
https://skillbox.ru/media/base/sozдание_obektov_i_konstruktory/?ysclid=m42krmolpn333688806

5. https://skillbox.ru/media/base/oblast_vidimosti_peremennykh_v_java/

6. <https://comaqa.gitbook.io/java-automation/oop-v-java/modifikatory-static-i-final>
7. <https://habr.com/ru/articles/755654/>
<https://metanit.com/java/tutorial/3.2.php?ysclid=m35onnjsbs792574955>, sm