

### **Вопросы по теме Интерфейс:**

1. Что такое интерфейс?
2. Что такое default-метод интерфейса?
3. Как вызвать default-метод интерфейса в реализующем классе?
4. Что такое static метод интерфейса?
5. Как вызвать static метод интерфейса?
6. Как разрешить проблему с двумя одинаковыми методами в интерфейсе?
7. Почему в некоторых интерфейсах не определены методы?
8. Почему нельзя объявить метод интерфейса final?

**Класс** – структура в языке программирования Java, которая является шаблоном для создания объекта и на основе которой создаются объекты.

### **Интерфейс.**

**Интерфейс** – это ссылочный тип данных, аналогичный классу, который может содержать только:

1. константы,
2. сигнатуры методов,
3. методы по умолчанию,
4. статические методы
5. вложенные типы.

Интерфейс нужен для описания поведения, которым должен обладать класс.

#### **Особенности интерфейса:**

- можно реализовывать методы по умолчанию (default-методы) и статические методы;
- может содержать поля, они автоматически становятся public static final;
- все методы в интерфейсе abstract (кроме default-методов) и public;
- на основе интерфейса нельзя создавать объекты, его методы могут быть реализованы только классами или расширены другими интерфейсами;
- классы, которые реализуют интерфейс, должны предоставить реализацию всех методов, которые есть в интерфейсе (кроме методов по умолчанию).

#### **Метод по умолчанию (default-метод) интерфейса:**

- не обязательно реализовывать, можно использовать реализацию по умолчанию;
- если некий класс реализует несколько интерфейсов, которые имеют одинаковый метод по умолчанию, то класс должен реализовать метод с совпадающей сигнатурой самостоятельно. Ситуация аналогична, если один интерфейс имеет метод по умолчанию, а в другом этот же метод является абстрактным;
- создан в Java 8 для реализации новой функциональности: Stream API и лямбда-выражения;
- также создан для решения проблемы повторения кода (если реализация метода во многих классах, реализующих интерфейс, одинаковая);

— может быть вызван в классе, реализующем интерфейс:  
`ИмяИнтерфейса.super.имяМетода()`.

**Статический метод интерфейса:**

— являются частью интерфейса и не может быть переопределен в классе, реализующем интерфейс;

— может иметь тело метода;

— в основном используются как вспомогательные методы (например, проверка на null);

— может быть вызван: `ИмяИнтерфейса.имяМетода()`.

Маркерный интерфейс – интерфейс, в котором нету методов и констант. Служит маркером, обозначающим, что класс, реализующий такой интерфейс, обладает определенными свойствами. Например, интерфейс `Cloneable` означает, что класс поддерживает технологию клонирования.

В интерфейсах нельзя объявить метод `final`, т.к. это бессмысленно, ведь такой метод не может быть переопределен.