Object Oriented Programming in Java

Abstract, Default, and Static Methods in Interfaces

Suhel Hammoud

Java Interfaces: Methods Overview

In Java, an **interface** is a contract that defines method signatures which classes agree to implement. Since Java 8, interfaces can also include **default** and **static** methods.

▒▒ 1. Abstract Methods

These are the original type of methods in interfaces.

- **No body**; must be implemented by the class.
- Declared without `abstract` keyword (implicitly abstract).

```
interface Drawable {
    void draw(); // abstract method
}
```

▒▒ 2. Default Methods

Introduced in **Java 8** to allow adding method implementations without breaking existing code.

- Marked with `default`
- Can be inherited or overridden by implementing classes

```
interface Drawable {
    default void print() {
        System.out.println("Drawing...");
    }
}
```

▒▒ 3. Static Methods

Also introduced in Java 8.

- Belong to the interface itself (not to instances)
- Not inherited by implementing classes

```
interface Drawable {
    static void info() {
        System.out.println("Interface for drawable objects.");
    }
}
```

Usage:

```
Drawable.info(); // Access via interface name
```

▒▒ Summary Table

Method Type	Keyword	Can Have Body	Can Be Overridden	Inherited
Abstract	none	❌	✅ (must be)	✅
Default	default	✅	✅ (optional)	✅
Static	static	✅	❌	❌