

08 - Creating Interface

In Spring, interfaces are often used to achieve loose coupling between components, allowing for flexible design and easy substitution of implementations.

- An interface defines a set of methods that implementing classes must provide.
- Spring can inject different implementations of the same interface based on the configuration.
- Different classes can implement the same interface.
- We can switch between implementations by specifying which one to inject in the Spring configuration.

Example:

Computer.java

This interface declares a method `compile()`, which both Laptop and Desktop classes will implement.

```
public interface Computer {  
    void compile();  
}
```

Laptop.java

Implements the Computer interface and provides the `compile()` method.

```
public class Laptop implements Computer {  
  
    public void compile() {  
        System.out.println("Compiling using Laptop");  
    }  
}
```

Desktop.java

Another implementation of the Computer interface with its own `compile()` method.

```
public class Desktop implements Computer {  
  
    public void compile() {  
        System.out.println("Compiling using Desktop");  
    }  
}
```

Alien.java

The Alien class has a dependency on the Computer interface, and it can work with any implementation of Computer.

```
public class Alien {  
  
    private int age;  
    private Computer comp;  
  
    public Alien() {  
        System.out.println("Object Created");  
    }  
  
    public int getAge() {  
        return age;  
    }  
  
    public void setAge(int age) { // Setter Injection  
        System.out.println("Setter called");  
        this.age = age;  
    }  
  
    public Computer getComp() {  
        return comp;  
    }  
  
    public void setComp(Computer comp) {
```

```
        this.comp = comp;
    }

    public void code() {
        System.out.println("Coding");
        comp.compile();
    }
}
```

Code Link:

<https://github.com/navinreddy20/spring6-course/tree/c6690e4f2c70d8f530d70623f13d14ff0ffd7e7d/2%20Exploring%20Spring%20Framework/2.8%20Creating%20Interface/Spring1>