Autowiring

Autowiring is a feature in the Spring Framework that allows the Spring container to automatically inject dependencies into a bean. It simplifies the process of wiring together beans by eliminating the need for explicit setter or constructor calls. When a bean requires a dependency, Spring resolves and injects it into the bean automatically.

1. Main Application Class (SpringBootDemoApplication)

This class starts the Spring Boot application and retrieves the Alien bean from the Spring context.

```
package com.telusko.app;

import org.springframework.boot.SpringApplication;
import org.springframework.boot.autoconfigure.SpringBootApplication;
import org.springframework.context.ApplicationContext;

@SpringBootApplication
public class SpringBootDemoApplication {
   public static void main(String[] args) {
      ApplicationContext context =
SpringApplication.run(SpringBootDemoApplication.class, args);

      Alien obj = context.getBean(Alien.class); // Retrieve Alien bean
      obj.code();  // Call method on Alien bean
   }
}
```

2. Alien Class (Alien.java)

This class represents a component that depends on a Laptop bean, which is injected automatically using the @Autowired annotation.

```
package com.telusko.app;

import org.springframework.beans.factory.annotation.Autowired;
import org.springframework.stereotype.Component;

@Component
public class Alien {

    @Autowired
    Laptop laptop;  // Autowiring of Laptop bean

    public void code() {
        laptop.compile();  // Calling Laptop's compile method
    }
}
```

3. Laptop Class (Laptop.java)

This is the class whose instance (bean) is injected into the Alien class. The compile method is called when the Alien object invokes its code() method.

```
package com.telusko.app;

import org.springframework.stereotype.Component;

@Component
```

```
public class Laptop {

    public void compile() {
        System.out.println("Compiling...");  // Output message
    }
}
```

Output

When you run the SpringBootDemoApplication class, the Alien bean's code() method is called, which in turn calls the Laptop bean's compile() method. The output will be:

```
Compiling...
```