

03 - Object Creation

ApplicationContext is the central interface to the Spring IoC container.

- It is used to load the Spring XML configuration file and create objects (beans) defined within it.
- When the ApplicationContext is instantiated, it automatically reads the XML configuration file, creates the container, and instantiates all the beans defined in it.
- Every time a bean is created, the Spring container calls the constructor of the corresponding class, initializing the object.

Workflow Example:

- Use ClassPathXmlApplicationContext to load the XML configuration file, which sets up the Spring container and creates the beans defined within the <bean> tag.

```
ApplicationContext context = new  
ClassPathXmlApplicationContext("spring.xml");
```

- Use the getBean() method to retrieve an object from the container and get its reference. This method returns the object based on the bean's ID specified in the XML file.

```
Alien obj = (Alien) context.getBean("alien");
```

Here, the bean with ID alien is retrieved, and a reference to the Alien object is obtained.

- After retrieving the bean, we can call its methods with any object in Java.

```
obj.code();
```

👉 Multiple Beans Creation:

The number of beans defined in the XML file corresponds to the number of objects created when the ApplicationContext is initialized.

```
<beans xmlns="http://www.springframework.org/schema/beans"
        xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
        xsi:schemaLocation="http://www.springframework.org/schema/beans
http://www.springframework.org/schema/beans/spring-beans.xsd">

    <!-- Bean Definitions -->
    <bean id="alien1" class="com.telusko.Alien">
    </bean>

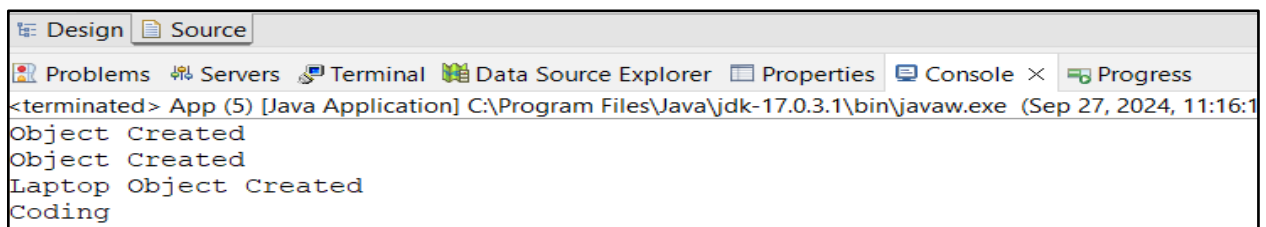
    <bean id="alien2" class="com.telusko.Alien">
    </bean>

    <bean id="lap" class="com.telusko.Laptop">
    </bean>

</beans>
```

- Two objects of the Alien class (alien1 and alien2) will be created.
- One object of the Laptop class (lap) will be created.
- Both alien1 and alien2 are **distinct objects**, allowing to work with multiple instances of the Alien class independently.
- By defining multiple beans in the XML file, we can create and manage multiple objects of the same class with different IDs.

Output:



```
<terminated> App (5) [Java Application] C:\Program Files\Java\jdk-17.0.3.1\bin\javaw.exe (Sep 27, 2024, 11:16:1)
Object Created
Object Created
Laptop Object Created
Coding
```

Code Link:

<https://github.com/navinreddy20/spring6-course/tree/c6690e4f2c70d8f530d70623f13d14ff0ffd7e7d/2%20Exploring%20Spring%20Framework/2.3%20Object%20Creation/Spring1>