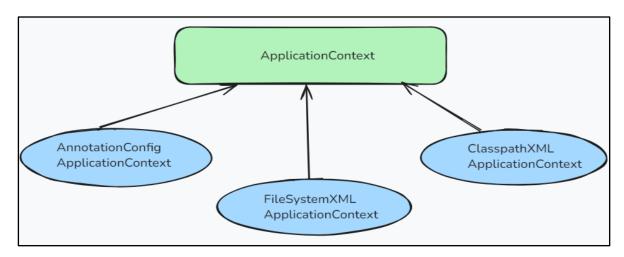
02 - Spring Based Xml Config

Spring Framework provides three main methods to create and manage objects within the application context:

- 1. **XML Configuration**: Configuring beans using XML files.
- 2. Java-Based Configuration: Configuring beans using Java classes.
- 3. **Annotations**: Configuring beans using annotations directly in the code



*** Steps for XML-Based Spring Configuration:**

1. Create XML Configuration file:

Create an XML configuration file in the **src/main/resources** directory. This file will define the beans that the Spring container will manage.

e.g., Spring.xml file in the src/main/resources

2. Define Beans in XML:

★ Beans are objects that form the backbone of a Spring application. They are managed by the Spring IoC (Inversion of Control) container.

Each bean is defined using the **<bean>** tag in the XML configuration file.

- id: A unique identifier for the bean, which is used to reference it.
- class: The fully qualified class name of the bean.

Example:

```
<br/>
```

3. Define XML Configuration Structure:

Use the <beans> root element. It specifies the XML namespace and schema definitions required for bean configuration.

Website Link: https://docs.spring.io/spring-framework/docs/4.2.x/spring-framework-reference/html/xsd-configuration.html

XML Schema:

```
<?xml version="1.0" encoding="UTF-8"?>
<beans xmlns="http://www.springframework.org/schema/beans"
    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xmlns:util="http://www.springframework.org/schema/util"
    xsi:schemaLocation="
        http://www.springframework.org/schema/beans
http://www.springframework.org/schema/beans/spring-beans.xsd
        http://www.springframework.org/schema/util
http://www.springframework.org/schema/util
http://www.springframework.org/schema/util/spring-util.xsd"> <!-- bean
definitions here -->
    </beans>
```



App.java

```
package com.telusko;
import org.springframework.context.ApplicationContext;
import org.springframework.context.support.ClassPathXmlApplicationContext;

public class App
{
    public static void main( String[] args )
    {
        Alien obj = new Alien();
        obj.code();

        ApplicationContext context = new ClassPathXmlApplicationContext("spring.xml");
        Alien obj = (Alien)context.getBean("alien");
        obj.code();
    }
}
```

Alien.java

```
package com.telusko;

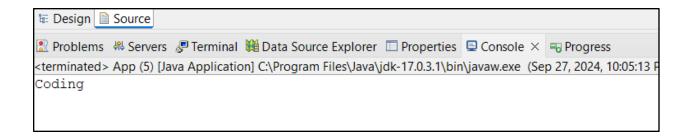
public class Alien {

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

Spring.xml

```
<
```

Output:



Code Link:

https://github.com/navinreddy20/spring6course/tree/c6690e4f2c70d8f530d70623f13d14ff0ffd7e7d/2%20Exploring%20Spring%20Framework/2.2%20Spring%20Bean%20Xml%20Config/Spring1