

Spring XML Based Configuration

Spring XML Based Configuration

In this topic, we will learn to create a Spring application and configure it using the XML code. Let's create a maven project and configure it using the XML file. If you are not familiar with the maven project, then you can read our detailed article.

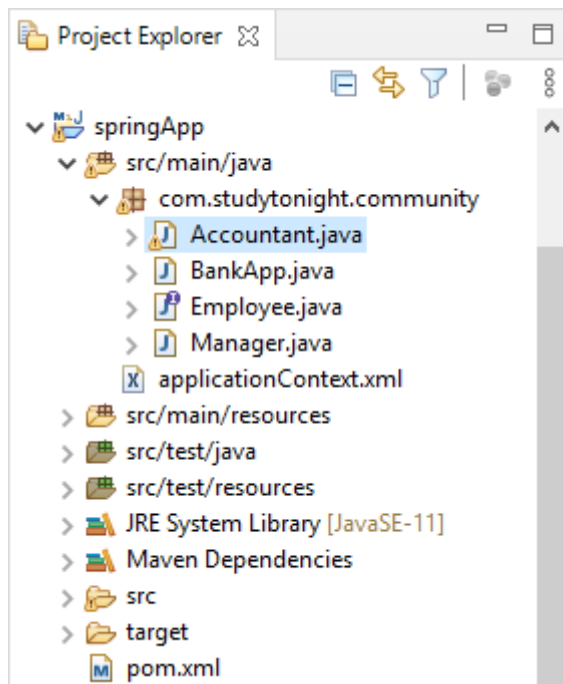
The following are the files created in our project. The source code of these files is given below.

- Employee.java
- Engineer.java
- Manager.java
- BankApp.java

And the following is our project structure after creating all the above files (java and xml).

Spring Project Structure:

Following is the structure of the project:



Now let's create the Java classes and the XML configuration file.

Spring XML Based Configuration

Employee.java

It is an interface that contains an abstract method `doWork()` which will be overridden by the implemented class.

```
package com.studytonight.community;
```

```
public interface Employee
```

```
{
```

```
    void doWork();
```

```
}
```

Accountant.java

It is our bean class that will be used to perform implementations. It implements the `Employee` interface and implements `doWork()` method.

```
package com.studytonight.community;
```

```
public class Accountant implements Employee
```

```
{
```

```
    public void doWork() {
```

```
        System.out.println("Audit the accounts...");
```

```
    }
```

```
}
```

Manager.java

This is another class that implements the `Employee` interface and overrides the `doWork()` method.

```
package com.studytonight.community;
```

```
public class Manager implements Employee
```

```
{
```

```
    public void doWork()
```

```
    {
```

```
        System.out.println("Manage the branch office");
```

```
    }
```

```
}
```

Spring XML Based Configuration

BankApp.java

It is a configuration file that reads the `applicationContext` file and get Bean using the `getBean()` method and then call method based on the retrieved object.

```
package com.studytonight.community;

import org.springframework.context.ApplicationContext;
import org.springframework.context.support.ClassPathXmlApplicationContext;

public class BankApp
{
    private static ApplicationContext context;
    public static void main(String[] args)
    {
        context = new ClassPathXmlApplicationContext("applicationContext.xml");
        Employee e = (Employee) context.getBean("manager");
        e.doWork();
    }
}
```

Spring XML Based Configuration

applicationContext.xml

It is an application context file that configures spring and registers the bean.

```
<?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:p="http://www.springframework.org/schema/p"
       xmlns:aop="http://www.springframework.org/schema/aop"
       xmlns:context="http://www.springframework.org/schema/context"
       xmlns:jee="http://www.springframework.org/schema/jee"
       xmlns:tx="http://www.springframework.org/schema/tx"
       xmlns:task="http://www.springframework.org/schema/task"
       xsi:schemaLocation="http://www.springframework.org/schema/aop http://www.springframework.org/schema/aop/spring-aop-3.2.xsd
http://www.springframework.org/schema/beans http://www.springframework.org/schema/beans/spring-beans-3.2.xsd
http://www.springframework.org/schema/context http://www.springframework.org/schema/context/spring-context-3.2.xsd
http://www.springframework.org/schema/jee http://www.springframework.org/schema/jee/spring-jee-3.2.xsd
http://www.springframework.org/schema/tx http://www.springframework.org/schema/tx/spring-tx-3.2.xsd
http://www.springframework.org/schema/task http://www.springframework.org/schema/task/spring-task-3.2.xsd">

    <bean id="manager"
          class="com.studytonight.community.Manager">

    </bean>

</beans>
```

Spring XML Based Configuration

pom.xml

This file contains all the dependencies of this project such as spring jars, servlet jars, etc. Put these dependencies into your project to run the application.

```
<project xmlns="http://maven.apache.org/POM/4.0.0"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://maven.apache.org/POM/4.0.0
http://maven.apache.org/xsd/maven-4.0.0.xsd">
  <modelVersion>4.0.0</modelVersion>
  <groupId>com.studytonight</groupId>
  <artifactId>springApp</artifactId>
  <version>0.0.1-SNAPSHOT</version>
  <dependencies>
    <!-- https://mvnrepository.com/artifact/org.springframework/spring-web -->
    <dependency>
      <groupId>org.springframework</groupId>
      <artifactId>spring-core</artifactId>
      <version>${spring.version}</version>
    </dependency>
    <dependency>
      <groupId>org.springframework</groupId>
      <artifactId>spring-context</artifactId>
      <version>${spring.version}</version>
    </dependency>
    <dependency>
      <groupId>javax.annotation</groupId>
      <artifactId>javax.annotation-api</artifactId>
      <version>1.3.2</version>
    </dependency>
  </dependencies>
  <properties>
    <spring.version>5.2.8.RELEASE</spring.version>
  </properties>
  <build>
    <sourceDirectory>src</sourceDirectory>
    <plugins>
      <plugin>
        <artifactId>maven-compiler-plugin</artifactId>
        <version>3.8.1</version>
      </plugin>
    </plugins>
  </build>
</project>
```

Spring XML Based Configuration

```
        <configuration>
            <source>1.8</source>
            <target>1.8</target>
        </configuration>
    </plugin>
</plugins>
</build>
</project>
```