

Create a Spring Web Project using Maven

1. Go to <https://start.spring.io/>
2. Change Group as “com.cognizant”
3. Change Artifact Id as “spring-learn”
4. Select Spring Boot DevTools and Spring Web
5. Create and download the project as zip
6. Extract the zip in root folder to Eclipse Workspace
7. Build the project using ‘mvn clean package -Dhttp.proxyHost=proxy.cognizant.com -Dhttp.proxyPort=6050 -Dhttps.proxyHost=proxy.cognizant.com -Dhttps.proxyPort=6050 -Dhttp.proxyUser=123456’ command in command line
8. Import the project in Eclipse "File > Import > Maven > Existing Maven Projects > Click Browse and select extracted folder > Finish"
9. Include logs to verify if main() method of SpringLearnApplication.
10. Run the SpringLearnApplication class.

SME to walk through the following aspects related to the project created:

1. src/main/java - Folder with application code
2. src/main/resources - Folder for application configuration
3. src/test/java - Folder with code for testing the application
4. SpringLearnApplication.java - Walkthrough the main() method.
5. Purpose of @SpringBootApplication annotation
6. pom.xml
 1. Walkthrough all the configuration defined in XML file
 2. Open 'Dependency Hierarchy' and show the dependency tree.

OUTPUT:

```
SpringLearnApplication.java ×  
1 package com.cognizant.spring_learn;  
2  
3 import org.springframework.boot.SpringApplication;  
4 import org.springframework.boot.autoconfigure.SpringBootApplication;  
5  
6 @SpringBootApplication  
7 public class SpringLearnApplication {  
8  
9     public static void main(String[] args) {  
10         SpringApplication.run(SpringLearnApplication.class, args);  
11     }  
12  
13 }  
14
```

Problems Javadoc Declaration Console ×

springLearnApplication [Java Application] C:\Users\SANDHYA\p2\pool\plugins\org.eclipse.justj.openjdk.hotspot.jre.full.win32.x86_64_21.0.7.v20250502-0

```
:: Spring Boot ::                (v3.5.3)  
  
2025-07-13T17:47:12.693+05:30 INFO 197084 --- [spring-learn] [ restartedMain] c.c.spring_learn.SpringLearnAppl  
2025-07-13T17:47:12.698+05:30 INFO 197084 --- [spring-learn] [ restartedMain] c.c.spring_learn.SpringLearnAppl  
2025-07-13T17:47:12.819+05:30 INFO 197084 --- [spring-learn] [ restartedMain] .e.DevToolsPropertyDefaultsPostP
```

Spring Core – Load Country from Spring Configuration XML

Country.java

```
package com.cognizant.spring_learn;

import org.slf4j.Logger;
import org.slf4j.LoggerFactory;

public class Country {

    private String code;

    private String name;

    private static final Logger LOGGER = LoggerFactory.getLogger(Country.class);

    public Country() {
        LOGGER.debug("Inside Country Constructor.");
    }

    public String getCode() {
        LOGGER.debug("Inside getCode()");
        return code;
    }

    public void setCode(String code) {
        LOGGER.debug("Inside setCode()");
        this.code = code;
    }

    public String getName() {
        LOGGER.debug("Inside getName()");
        return name;
    }

    public void setName(String name) {
        LOGGER.debug("Inside setName()");
        this.name = name;
    }

    @Override
    public String toString() {
        return "Country{" +
```

```

        "code=" + code + "\" +
        ", name=" + name + "\" +
        '}'
    }
}

```

SpringLearnApplication.java

```

package com.cognizant.spring_learn;

import org.slf4j.Logger;
import org.slf4j.LoggerFactory;
import org.springframework.context.ApplicationContext;
import org.springframework.context.support.ClassPathXmlApplicationContext;

public class SpringLearnApplication {

    private static final Logger LOGGER =
    LoggerFactory.getLogger(SpringLearnApplication.class);

    public static void main(String[] args) {
        displayCountry();
    }

    public static void displayCountry() {
        ApplicationContext context = new ClassPathXmlApplicationContext("country.xml");
        Country country = context.getBean("country", Country.class);
        LOGGER.debug("Country : {}", country.toString());
    }
}

```

LogBack.xml

```

<configuration>

    <appender name="STDOUT" class="ch.qos.logback.core.ConsoleAppender">

        <encoder>

            <pattern>%d{HH:mm:ss.SSS}    [%thread]    %-5level    %logger{36}    -
            %msg%n</pattern>

        </encoder>
    </appender>

```

```

</appender>
<root level="debug">
    <appender-ref ref="STDOUT" />
</root>
</configuration>

```

Country.xml

```

<?xml version="1.0" encoding="UTF-8"?>
<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 id="country" class="com.cognizant.spring_learn.Country">
        <property name="code" value="IN"/>
        <property name="name" value="India"/>
    </bean>
</beans>

```

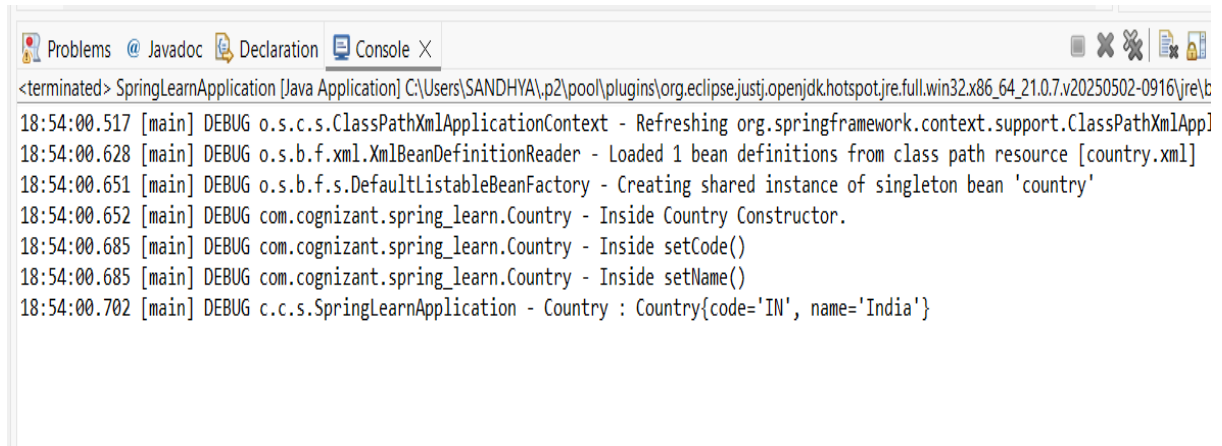
Log4j.xml

```

<?xml version="1.0" encoding="UTF-8" ?>
<!DOCTYPE log4j:configuration SYSTEM "log4j.dtd">
<log4j:configuration>
    <appender name="console" class="org.apache.log4j.ConsoleAppender">
        <layout class="org.apache.log4j.PatternLayout">
            <param name="ConversionPattern" value="%d [%t] %-5p %c - %m%n"/>
        </layout>
    </appender>
    <root>
        <priority value="debug"/>
        <appender-ref ref="console"/>
    </root>
</log4j:configuration>

```

OUTPUT:



The screenshot shows the Eclipse IDE's console window. The title bar includes tabs for 'Problems', 'Javadoc', 'Declaration', and 'Console'. The console output shows the following sequence of events:

```
<terminated> SpringLearnApplication [Java Application] C:\Users\SANDHYA\p2\pool\plugins\org.eclipse.justj.openjdk.hotspot.jre.full.win32.x86_64_21.0.7.v20250502-0916\jre\bin\java.exe
18:54:00.517 [main] DEBUG o.s.c.s.ClassPathXmlApplicationContext - Refreshing org.springframework.context.support.ClassPathXmlApp
18:54:00.628 [main] DEBUG o.s.b.f.xml.XmlBeanDefinitionReader - Loaded 1 bean definitions from class path resource [country.xml]
18:54:00.651 [main] DEBUG o.s.b.f.s.DefaultListableBeanFactory - Creating shared instance of singleton bean 'country'
18:54:00.652 [main] DEBUG com.cognizant.spring_learn.Country - Inside Country Constructor.
18:54:00.685 [main] DEBUG com.cognizant.spring_learn.Country - Inside setCode()
18:54:00.685 [main] DEBUG com.cognizant.spring_learn.Country - Inside setName()
18:54:00.702 [main] DEBUG c.c.s.SpringLearnApplication - Country : Country{code='IN', name='India'}
```