

```
// Hands-on 1 - Spring Boot Project Setup
```

```
// SpringLearnApplication.java
```

```
package com.cognizant.springlearn;
```

```
import org.slf4j.Logger;
```

```
import org.slf4j.LoggerFactory;
```

```
import org.springframework.boot.SpringApplication;
```

```
import org.springframework.boot.autoconfigure.SpringBootApplication;
```

```
@SpringBootApplication
```

```
public class SpringLearnApplication {
```

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

```
    public static void main(String[] args) {
```

```
        LOGGER.info("START main()");
```

```
        SpringApplication.run(SpringLearnApplication.class, args);
```

```
        displayDate();
```

```
        displayCountry();
```

```
        displayCountries();
```

```
        LOGGER.info("END main()");
```

```
    }
```

```
    public static void displayDate() {
```

```
        LOGGER.info("START displayDate()");
```

```
        ApplicationContext context = new ClassPathXmlApplicationContext("date-  
format.xml");
```

```
        SimpleDateFormat format = context.getBean("dateFormat",  
        SimpleDateFormat.class);
```

```
        try {
```

```

        Date date = format.parse("31/12/2018");
        LOGGER.debug("Parsed Date: {}", date);
    } catch (ParseException e) {
        LOGGER.error("Error parsing date", e);
    }
    LOGGER.info("END displayDate()");
}

```

```

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

```

```

public static void displayCountries() {
    LOGGER.info("START displayCountries()");
    ApplicationContext context = new
    ClassPathXmlApplicationContext("country.xml");
    List<Country> countryList = (List<Country>) context.getBean("countryList");
    for (Country c : countryList) {
        LOGGER.debug("Country : {}", c);
    }
    LOGGER.info("END displayCountries()");
}
}

```

```
// Country.java

package com.cognizant.springlearn;

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

public class Country {

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

    private String code;
    private String name;

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

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

    public void setCode(String code) {
        LOGGER.debug("Setting Code to {}", code);
        this.code = code;
    }

    public String getName() {
        LOGGER.debug("Getting Name");
        return name;
    }
}
```

```
}
```

```
public void setName(String name) {  
    LOGGER.debug("Setting Name to {}", name);  
    this.name = name;  
}
```

```
@Override
```

```
public String toString() {  
    return "Country [code=" + code + ", name=" + name + "];"  
}  
}
```

```
// application.properties (src/main/resources)
```

```
logging.level.org.springframework=info
```

```
logging.level.com.cognizant.springlearn=debug
```

```
logging.pattern.console=%d{yyMMdd}|%d{HH:mm:ss.SSS}|%-20.20thread|%5p|%-  
25.25logger{25}|%25M|%m%n
```

```
// date-format.xml (src/main/resources)
```

```
<?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
```

```
    https://www.springframework.org/schema/beans/spring-beans.xsd">
```

```
<bean id="dateFormat" class="java.text.SimpleDateFormat">
```

```
    <constructor-arg value="dd/MM/yyyy" />
```

```
</bean>
```

</beans>

// country.xml (src/main/resources)

<?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

https://www.springframework.org/schema/beans/spring-beans.xsd">

<bean id="country" class="com.cognizant.springlearn.Country">

<property name="code" value="IN" />

<property name="name" value="India" />

</bean>

<bean id="in" class="com.cognizant.springlearn.Country">

<property name="code" value="IN" />

<property name="name" value="India" />

</bean>

<bean id="us" class="com.cognizant.springlearn.Country">

<property name="code" value="US" />

<property name="name" value="United States" />

</bean>

<bean id="de" class="com.cognizant.springlearn.Country">

<property name="code" value="DE" />

<property name="name" value="Germany" />

</bean>

<bean id="jp" class="com.cognizant.springlearn.Country">

<property name="code" value="JP" />

<property name="name" value="Japan" />

```
</bean>
```

```
<bean id="countryList" class="java.util.ArrayList">
```

```
  <constructor-arg>
```

```
    <list>
```

```
      <ref bean="in" />
```

```
      <ref bean="us" />
```

```
      <ref bean="de" />
```

```
      <ref bean="jp" />
```

```
    </list>
```

```
  </constructor-arg>
```

```
</bean>
```

```
</beans>
```

```
Inside Country Constructor.  
Setting code: IN  
Setting name: India  
Getting code: IN  
Getting name: India  
Country : Country [code=IN, name=India]
```

```
Inside Country Constructor.  
Setting code: IN  
Setting name: India  
Inside Country Constructor.  
Setting code: IN  
Setting name: India
```

```
Country List: [Country [code=IN, name=India], Country [code=US, name=United States], Country [code=DE, name=Germany], Country [code=JP, name=Japan]]
```