

Exercise 1: Create a Spring Web Project using Maven:

CODE:

EmployeeManagementSystemApplication.java:

```
package com.cognizant.spring_learn;

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

@SpringBootApplication

public class SpringLearnApplication {

    public static void main(String[] args) {

        SpringApplication.run(SpringLearnApplication.class, args);

    }

}
```

Output:

```
2025-07-13T11:42:45.805+05:30 INFO 5332 --- [spring-learn] [ restartedMain] o.a.c.c.C.[Tomcat].[localhost].[/] : Initializing Spring embedded WebApplicationContext
2025-07-13T11:42:45.807+05:30 INFO 5332 --- [spring-learn] [ restartedMain] w.s.c.ServletWebServerApplicationContext : Root WebApplicationContext: initialization completed in 1
044 ms
2025-07-13T11:42:46.187+05:30 INFO 5332 --- [spring-learn] [ restartedMain] o.s.b.d.a.OptionalLiveReloadServer : LiveReload server is running on port 35729
2025-07-13T11:42:46.224+05:30 INFO 5332 --- [spring-learn] [ restartedMain] o.s.b.w.embedded.tomcat.TomcatWebServer : Tomcat started on port 9090 (http) with context path '/'
2025-07-13T11:42:46.236+05:30 INFO 5332 --- [spring-learn] [ restartedMain] c.c.spring_learn.SpringLearnApplication : Started SpringLearnApplication in 2.053 seconds (process
running for 2.484)
2025-07-13T11:44:08.223+05:30 INFO 5332 --- [spring-learn] [ionShutdownHook] o.s.b.w.e.tomcat.GracefulShutdown : Commencing graceful shutdown. Waiting for active requests
to complete
2025-07-13T11:44:08.337+05:30 INFO 5332 --- [spring-learn] [tomcat-shutdown] o.s.b.w.e.tomcat.GracefulShutdown : Graceful shutdown complete
[INFO] -----
[INFO] BUILD SUCCESS
[INFO] -----
[INFO] Total time: 01:26 min
[INFO] Finished at: 2025-07-13T11:44:08+05:30
[INFO] -----
```

Exercise 4: Spring Core – Load Country from Spring Configuration XML:

CODE:

Country.java:

```
package com.cognizant.spring_learn;

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("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.boot.SpringApplication;
import org.springframework.boot.autoconfigure.SpringBootApplication;
import org.springframework.context.support.ClassPathXmlApplicationContext;

@SpringBootApplication
public class SpringLearnApplication {

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

    public static void main(String[] args) {
        SpringApplication.run(SpringLearnApplication.class, args);
    }
}

```

```

        displayCountry();
    }

    public static void displayCountry() {
        LOGGER.info("START");

        try (ClassPathXmlApplicationContext context = new
ClassPathXmlApplicationContext("country.xml")) {
            Country country = context.getBean("country", Country.class);
            LOGGER.debug("Country : {}", country.toString());
        }
        LOGGER.info("END");
    }
}

```

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>

```

application.properties:

```

spring.application.name=spring-learn
server.port=9090

```

logging.level.org.springframework=info

logging.level.com.cognizant.spring_learn=debug

Output:

```
running for 2.471)
2025-07-13T12:07:44.961+05:30 INFO 20680 --- [spring-learn] [ restartedMain] c.c.spring_learn.SpringLearnApplication : START
2025-07-13T12:07:45.065+05:30 DEBUG 20680 --- [spring-learn] [ restartedMain] com.cognizant.spring_learn.Country : Inside Country Constructor.
2025-07-13T12:07:45.066+05:30 DEBUG 20680 --- [spring-learn] [ restartedMain] com.cognizant.spring_learn.Country : Inside setCode.
2025-07-13T12:07:45.066+05:30 DEBUG 20680 --- [spring-learn] [ restartedMain] com.cognizant.spring_learn.Country : Inside setName.
2025-07-13T12:07:45.067+05:30 DEBUG 20680 --- [spring-learn] [ restartedMain] c.c.spring_learn.SpringLearnApplication : country : Country [code=IN, name=India]
2025-07-13T12:07:45.070+05:30 INFO 20680 --- [spring-learn] [ restartedMain] c.c.spring_learn.SpringLearnApplication : END
2025-07-13T12:07:53.356+05:30 INFO 20680 --- [spring-learn] [ ionShutdownHook] o.s.b.w.e.tomcat.GracefulShutdown : Commencing graceful shutdown. Waiting for active request
s to complete
2025-07-13T12:07:53.430+05:30 INFO 20680 --- [spring-learn] [ tomcat-shutdown] o.s.b.w.e.tomcat.GracefulShutdown : Graceful shutdown complete
[INFO] -----
[INFO] BUILD SUCCESS
[INFO] -----
[INFO] Total time: 12.957 s
[INFO] Finished at: 2025-07-13T12:07:53+05:30
[INFO] -----
Terminate batch job (Y/N)?
```

