**Hands On 1:**

**Create a Spring Web Project using Maven**

package com.cognizant.spring\_learn;

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");

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

*LOGGER*.info("END");

}

}

**SME Walkthrough :**

**1. src/main/java**

* Contains **application source code**.
* All the controllers, services, configs, and main class are here.
* Example: SpringLearnApplication.java is the entry point.

**2. src/main/resources**

* Stores **application configuration files**:
  + application.properties for Spring Boot settings.
  + XML files (country.xml, date-format.xml) for Spring Core beans.
  + static/ or templates/ for web projects.

**3. src/test/java**

* Contains **unit and integration tests**.
* Follows the same package structure as src/main/java.

**4. SpringLearnApplication.java**

* Has the main() method that:
  1. Starts the Spring Boot app using SpringApplication.run().
  2. Initializes **Spring Boot auto-configuration**.
  3. Starts the embedded Tomcat server.
* @SpringBootApplication combines:
  1. @Configuration → defines configuration class.
  2. @EnableAutoConfiguration → enables Spring Boot’s auto-config.
  3. @ComponentScan → scans for components in the package.

**5. Purpose of @SpringBootApplication**

* Shortcut for the three key annotations mentioned above.
* Makes your app auto-configure itself and discover beans automatically.

**6. pom.xml**

* **Contains:**
  + Dependencies: spring-boot-starter-web, spring-boot-devtools
  + Spring Boot version
  + Maven build plugins

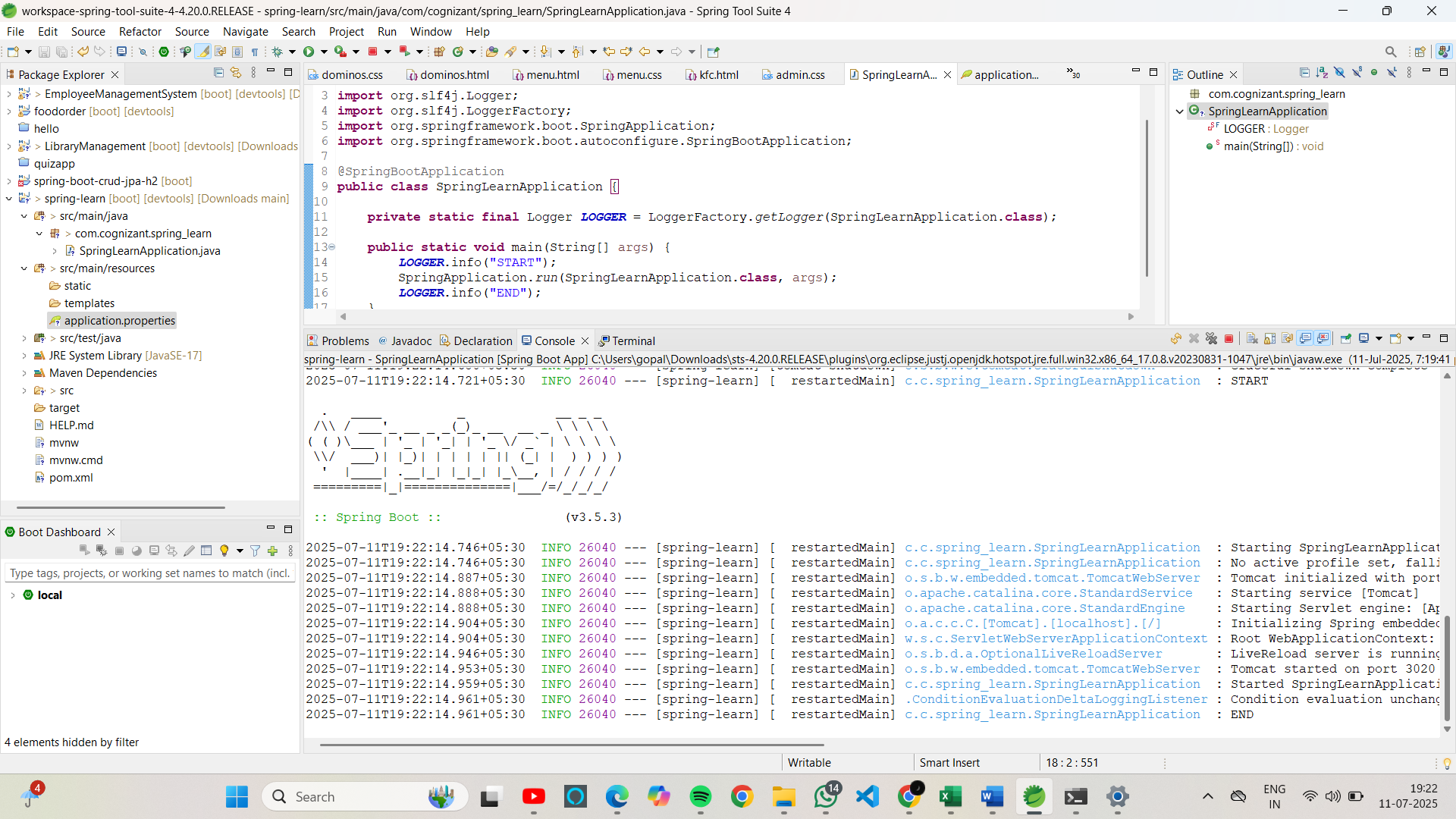
**7. Configuration defined in XML files**

* Can add XML files (date-format.xml, country.xml)
* They declare beans for Spring Core.
* They’re stored in src/main/resources.

**8. Dependency Tree**

* Use the Dependency Hierarchy view in STS to show:
  + Parent POM: spring-boot-starter-parent
  + Starters: spring-boot-starter-web, etc.
  + Transitive dependencies: log frameworks, Tomcat, Jackson.

**Output:**



**Hands-On 2:**

**Spring Core – Load SimpleDateFormat from Spring Configuration XML**

**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.ApplicationContext;

import org.springframework.context.support.ClassPathXmlApplicationContext;

import java.text.SimpleDateFormat;

import java.util.Date;

@SpringBootApplication

public class SpringLearnApplication {

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

public static void main(String[] args) throws Exception {

*LOGGER*.info("START");

// Bootstraps Spring Boot

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

// Call displayDate()

SpringLearnApplication app = new SpringLearnApplication();

app.displayDate();

*LOGGER*.info("END");

}

public void displayDate() throws Exception {

*LOGGER*.info("START");

// Load bean from Spring Core XML config

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

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

Date date = format.parse("31/12/2018");

*LOGGER*.debug("Parsed Date: {}", date);

*LOGGER*.info("END");

}

}

**Application.properties:**

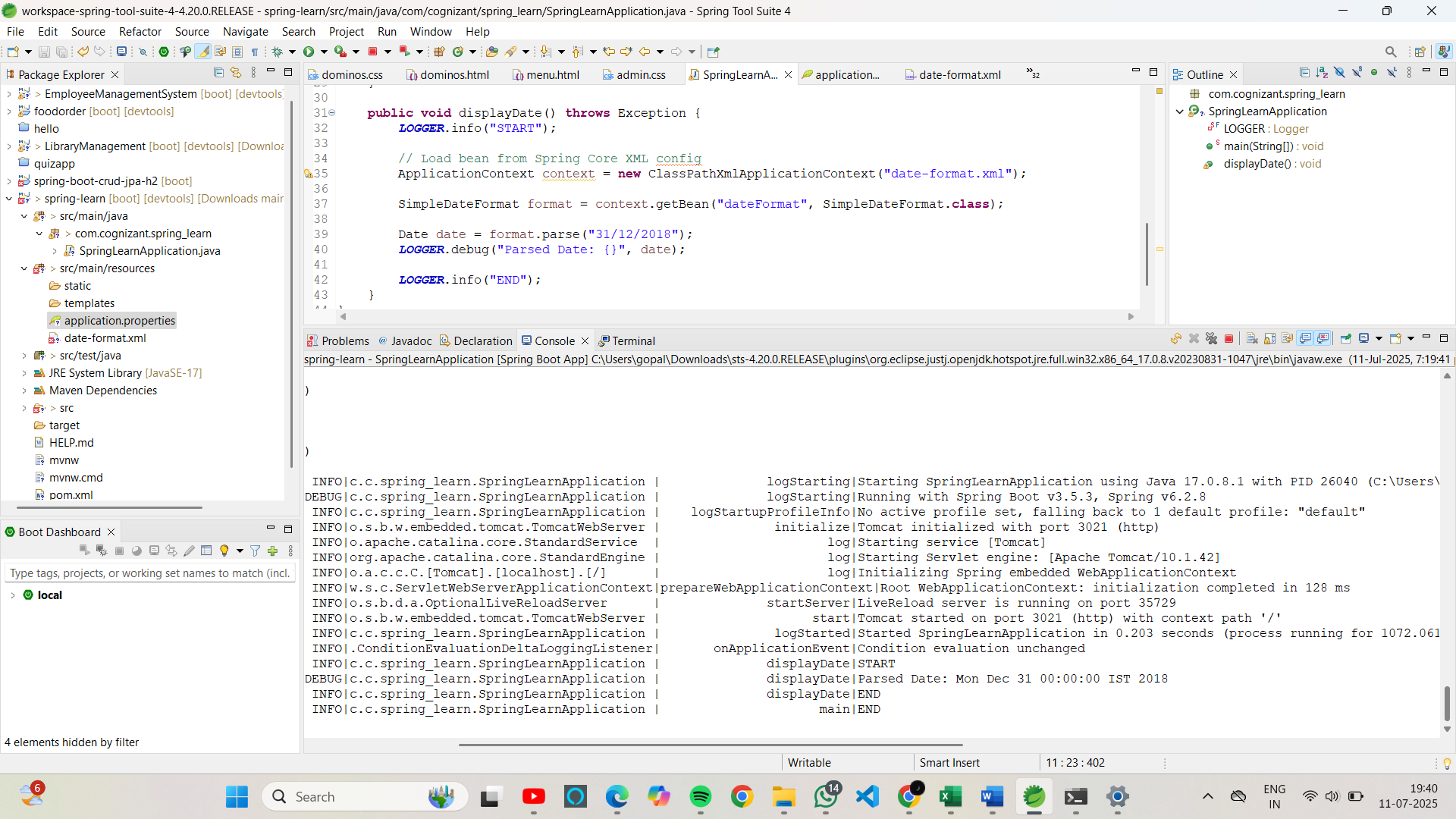
spring.application.name=spring-learn

server.port=3021

logging.level.com.cognizant.spring\_learn=DEBUG

logging.pattern.console=%d{yyMMdd}|%d{HH:mm:ss.SSS}|%-20.20thread|%5p|%-40.40logger{40}|%25M|%m%n

**Output:**

****

**Hands on 3**

**Spring Core - Incorporate Logging**

**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.ApplicationContext;

import org.springframework.context.support.ClassPathXmlApplicationContext;

import java.text.SimpleDateFormat;

import java.util.Date;

@SpringBootApplication

public class SpringLearnApplication {

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

public static void main(String[] args) throws Exception {

*LOGGER*.info("START");

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

SpringLearnApplication app = new SpringLearnApplication();

app.displayDate();

*LOGGER*.info("END");

}

public void displayDate() throws Exception {

*LOGGER*.info("START");

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

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

Date date = format.parse("31/12/2018");

*LOGGER*.debug("Parsed Date: {}", date);

*LOGGER*.info("END");

}

}

**Application.properties:**

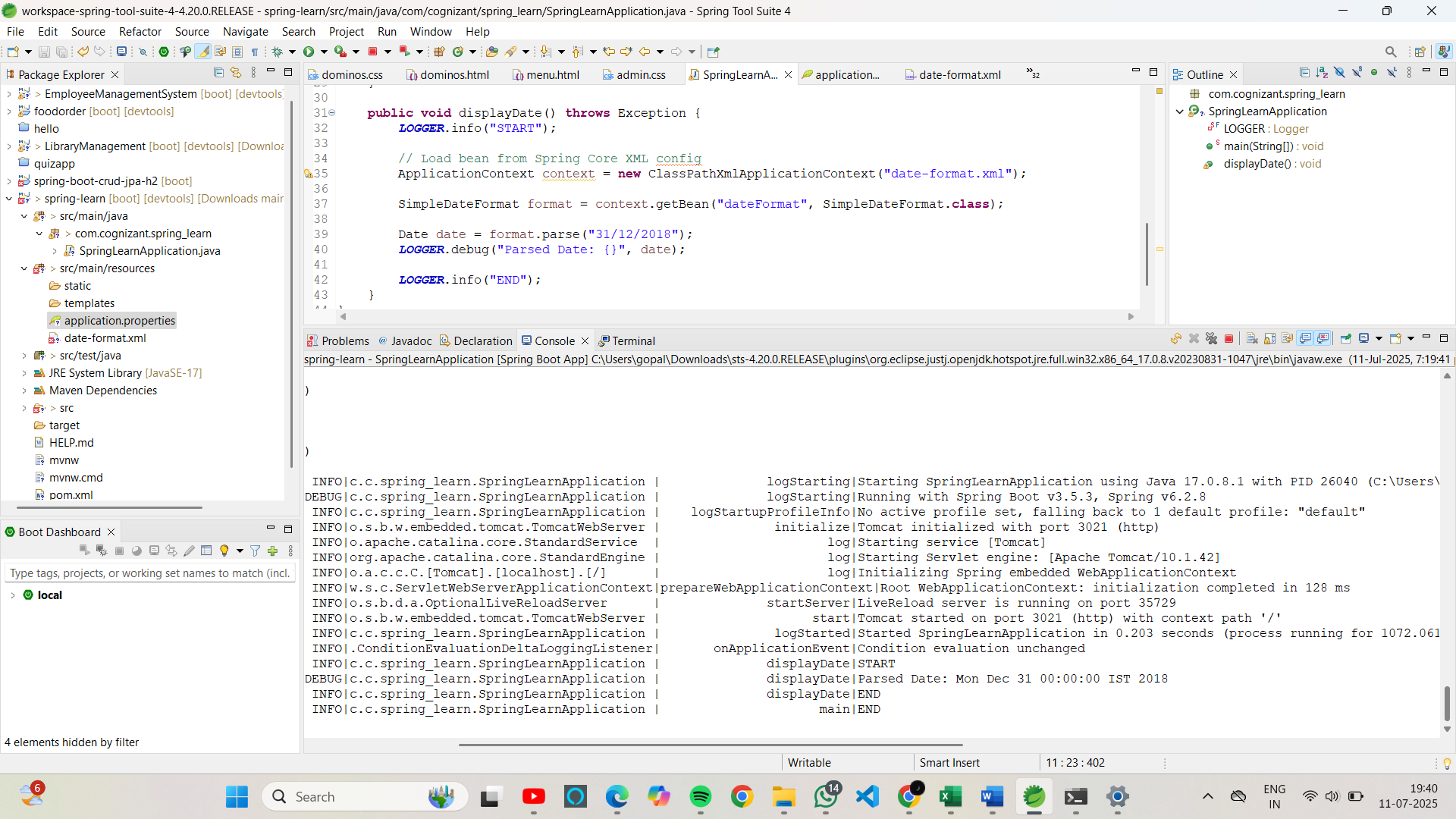
spring.application.name=spring-learn

server.port=3021

logging.level.com.cognizant.spring\_learn=DEBUG

logging.pattern.console=%d{yyMMdd}|%d{HH:mm:ss.SSS}|%-20.20thread|%5p|%-40.40logger{40}|%25M|%m%n

**Output:**

****

**Hands on 4**

**Spring Core – Load Country from Spring Configuration XML**