**Hello World RESTful Web Service**

**Question:**

**Hello World RESTful Web Service**   
  
Write a REST service in the spring learn application created earlier, that returns the text "Hello World!!" using Spring Web Framework. Refer details below:  
  
**Method:** GET  
**URL:** /hello  
**Controller:** com.cognizant.spring-learn.controller.HelloController  
**Method Signature:** public String sayHello()  
**Method Implementation:** return hard coded string "Hello World!!"  
**Sample Request**: http://localhost:8083/hello  
**Sample Response:** Hello World!!   
  
**IMPORTANT NOTE**: Don't forget to include start and end log in the sayHello() method.  
  
Try the URL http://localhost:8083/hello in both chrome browser and postman.  
  
SME to explain the following aspects:

* In network tab of developer tools show the HTTP header details received
* In postman click on "Headers" tab to view the HTTP header details received

**Solution:**

**Pom.xml**

<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.example</groupId>

    <artifactId>spring-rest-demo</artifactId>

    <version>0.0.1-SNAPSHOT</version>

    <packaging>jar</packaging>

    <name>spring-rest-demo</name>

    <properties>

        <java.version>17</java.version>

        <spring.boot.version>3.2.4</spring.boot.version>

    </properties>

    <dependencies>

*<!-- ✅ Spring Boot Web Starter -->*

        <dependency>

            <groupId>org.springframework.boot</groupId>

            <artifactId>spring-boot-starter-web</artifactId>

            <version>${spring.boot.version}</version>

        </dependency>

*<!-- ✅ Spring Boot Starter Test (optional) -->*

        <dependency>

            <groupId>org.springframework.boot</groupId>

            <artifactId>spring-boot-starter-test</artifactId>

            <version>${spring.boot.version}</version>

            <scope>test</scope>

        </dependency>

    </dependencies>

    <build>

        <plugins>

*<!-- ✅ Spring Boot Maven Plugin -->*

            <plugin>

                <groupId>org.springframework.boot</groupId>

                <artifactId>spring-boot-maven-plugin</artifactId>

                <version>${spring.boot.version}</version>

            </plugin>

        </plugins>

    </build>

</project>

**SpringRestDemoApplication.java**

**package** **com.example**;

**import** **org.springframework.boot.SpringApplication**;

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

@**SpringBootApplication**

**public** **class** SpringRestDemoApplication {

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

        SpringApplication.run(SpringRestDemoApplication.class, args);

    }

}

**HelloController.java**

**package** **com.example.controller**;

**import** **org.slf4j.Logger**;

**import** **org.slf4j.LoggerFactory**;

**import** **org.springframework.web.bind.annotation.\***;

@**RestController**

**public** **class** HelloController {

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

    @**GetMapping**("/hello")

**public** **String** sayHello() {

        LOGGER.info("START sayHello()");

**String** message **=** "Hello World!!";

        LOGGER.info("END sayHello()");

**return** message;

    }

}

**Application.properties**

*# Optional: Run on specific port*

server.port=8083

**Output:**

