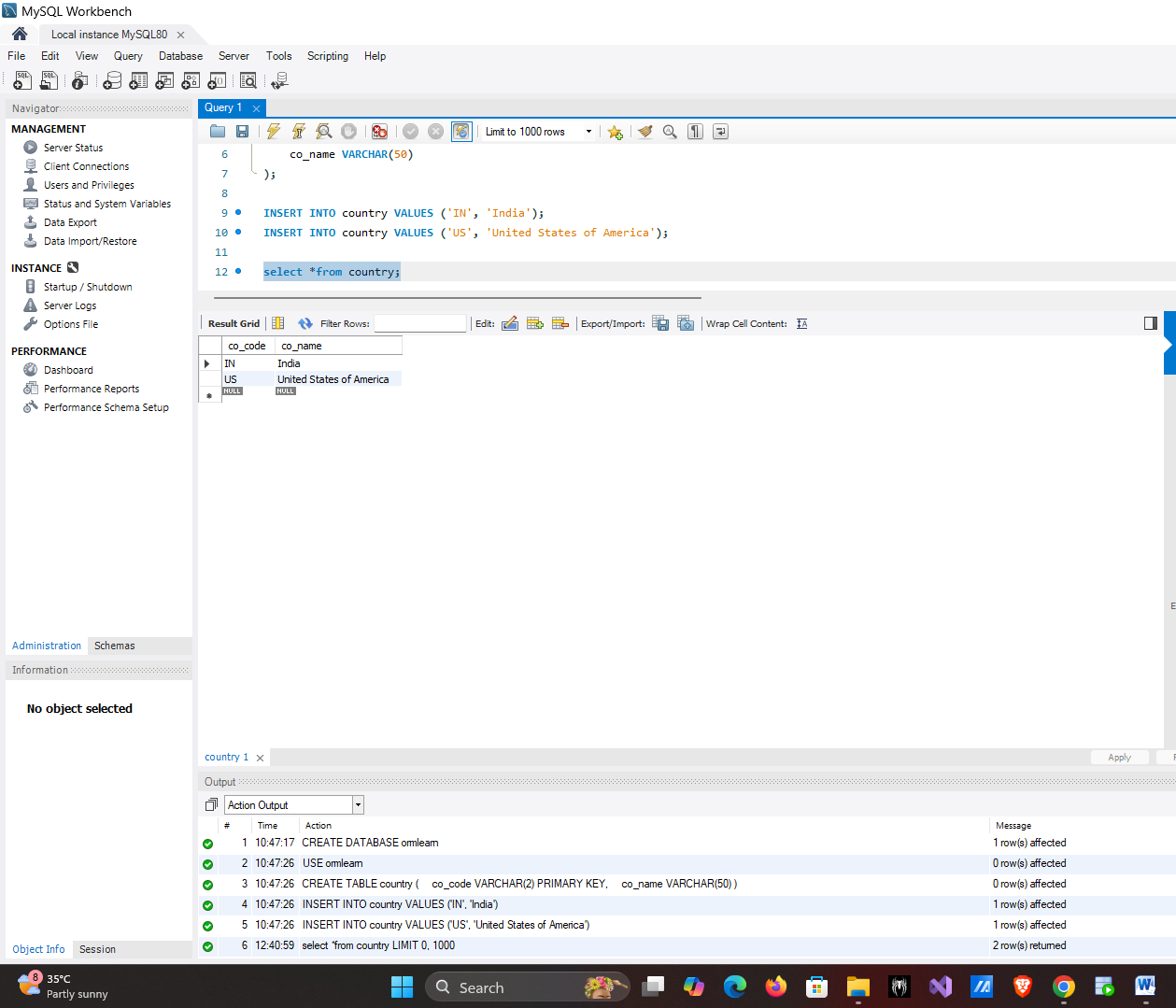
1.Spring-data-jpa-handson

## HandsOn 1

### Spring Data JPA - Quick Example

**Creating database :**

****

CREATE DATABASE ormlearn;

USE ormlearn;

CREATE TABLE country (

co\_code VARCHAR(2) PRIMARY KEY,

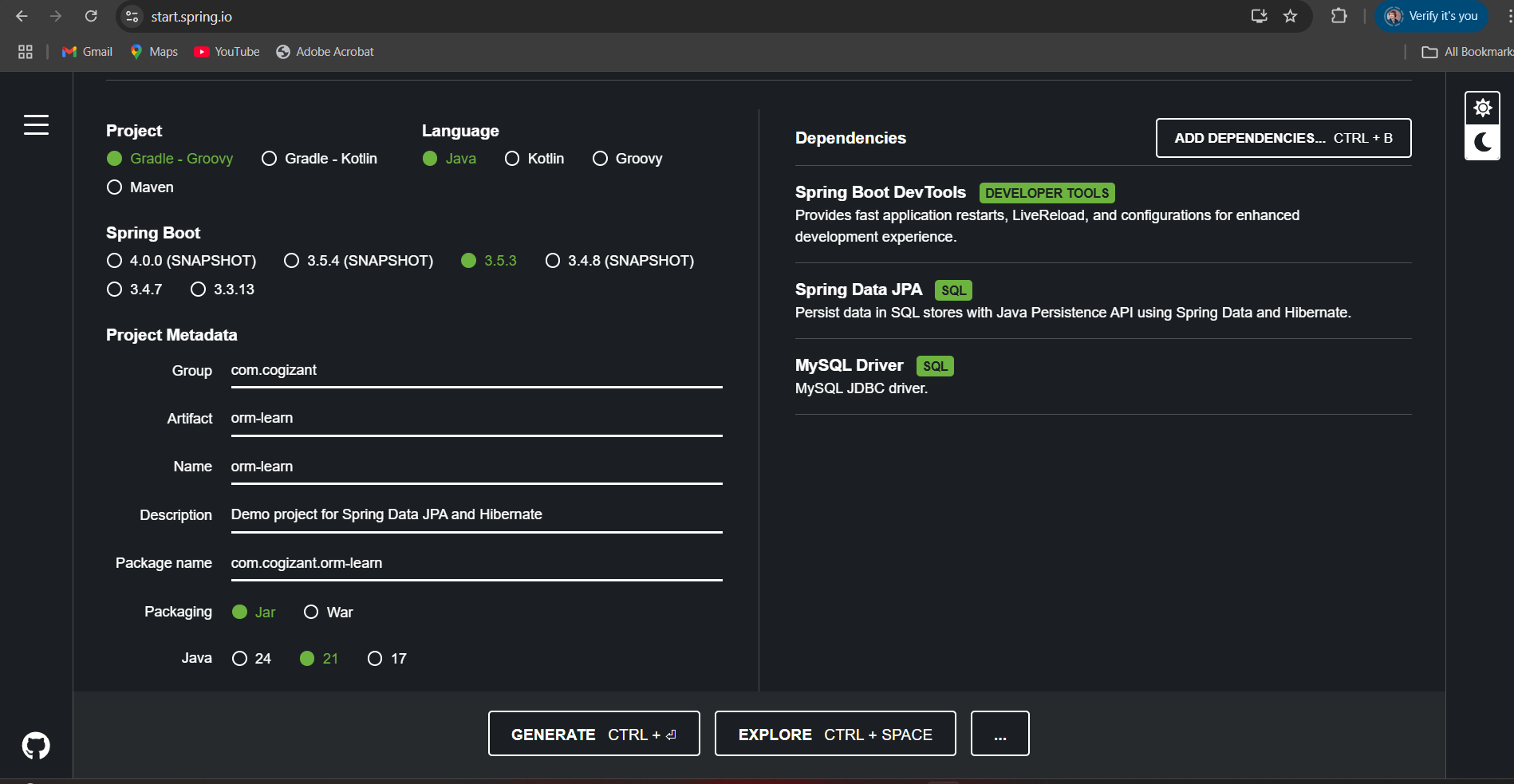
co\_name VARCHAR(50)

);

INSERT INTO country VALUES ('IN', 'India');

INSERT INTO country VALUES ('US', 'United States of America');

**Creating Project in Spring Initializr :**



**application.properties :**

# Logging Configuration

logging.level.org.springframework=INFO

logging.level.com.cognizant=DEBUG

logging.level.org.hibernate.SQL=TRACE

logging.level.org.hibernate.type.descriptor.sql=TRACE

logging.pattern.console=%d{dd-MM-yy} %d{HH:mm:ss.SSS} %-20.20thread %5p %-25.25logger{25} %25M %4L %m%n

# Database Configuration

spring.datasource.driver-class-name=com.mysql.cj.jdbc.Driver

spring.datasource.url=jdbc:mysql://localhost:3306/ormlearn

spring.datasource.username=root

spring.datasource.password=root

# Hibernate Configuration

spring.jpa.hibernate.ddl-auto=validate

spring.jpa.properties.hibernate.dialect=org.hibernate.dialect.MySQL5Dialect

**Country.java :**

package com.cognizant.orm\_learn.model;

import javax.persistence.\*;

@Entity

@Table(name = "country")

public class Country {

@Id

@Column(name = "co\_code")

private String code;

@Column(name = "co\_name")

private String name;

public String getCode() {

return code;

}

public void setCode(String code) {

this.code = code;

}

public String getName() {

return name;

}

public void setName(String name) {

this.name = name;

}

@Override

public String toString() {

return "Country [code=" + code + ", name=" + name + "]";

}

}

**CountryRepository.java :**

package com.cognizant.orm\_learn.repository;

import com.cognizant.orm\_learn.model.Country;

import org.springframework.data.jpa.repository.JpaRepository;

import org.springframework.stereotype.Repository;

@Repository

public interface CountryRepository extends JpaRepository<Country, String> {

}

**CountryService :**

package com.cognizant.orm\_learn.service;

import com.cognizant.orm\_learn.model.Country;

import com.cognizant.orm\_learn.repository.CountryRepository;

import org.springframework.beans.factory.annotation.Autowired;

import org.springframework.stereotype.Service;

import javax.transaction.Transactional;

import java.util.List;

@Service

public class CountryService {

@Autowired

private CountryRepository countryRepository;

@Transactional

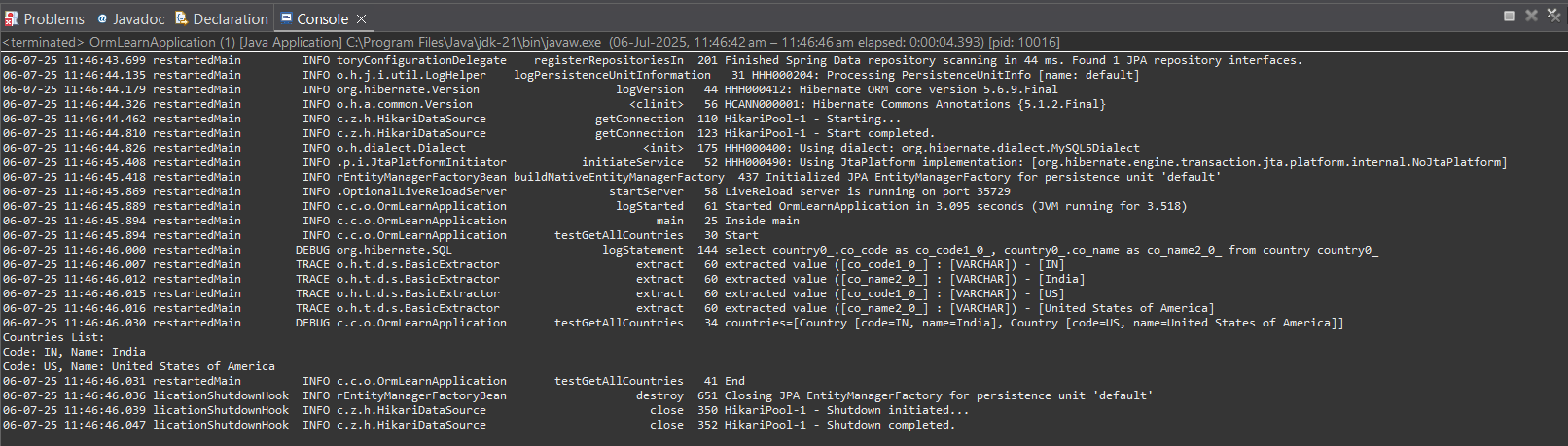
public List<Country> getAllCountries() {

return countryRepository.findAll();

}

}

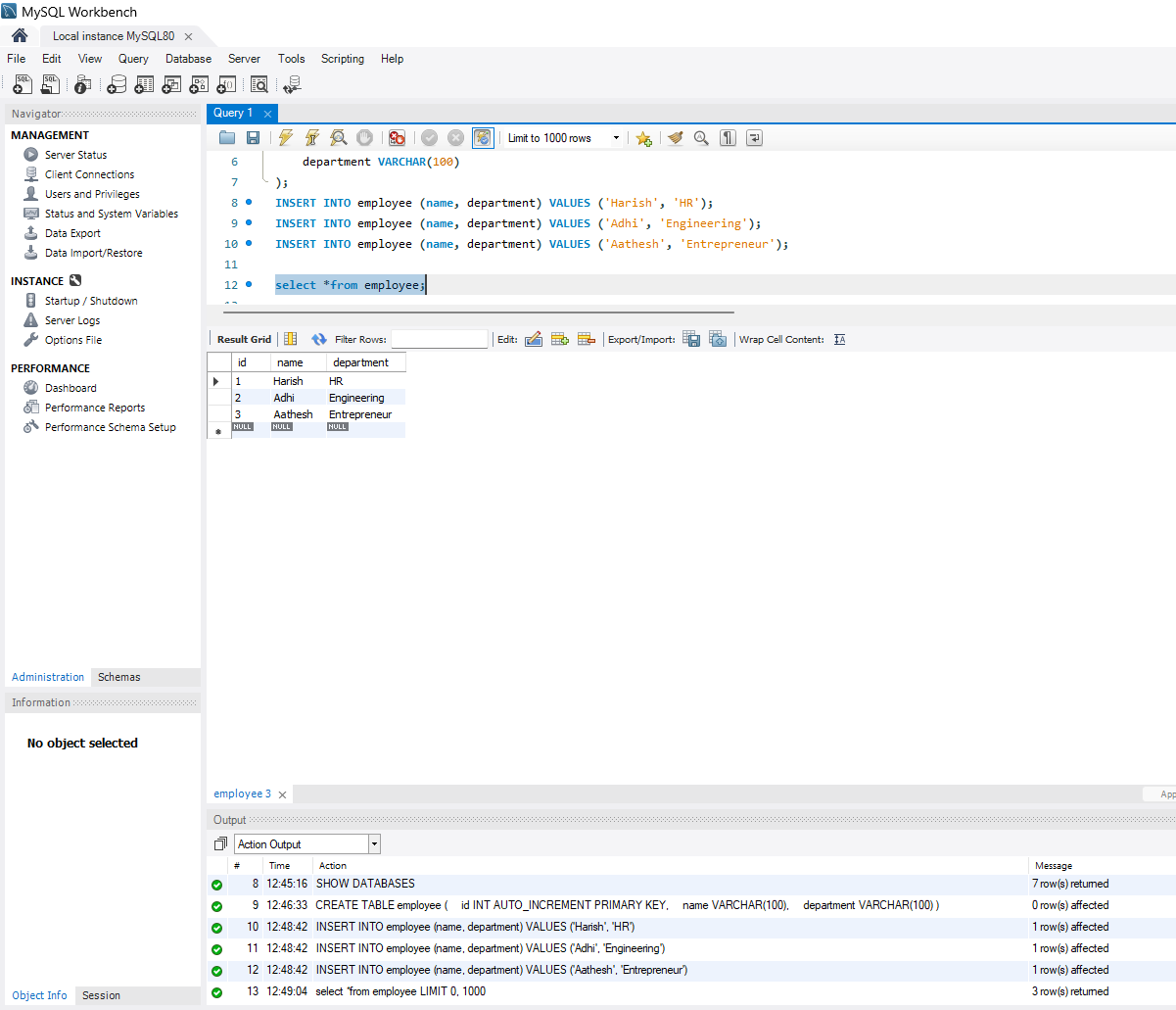
**OUTPUT :**



## Hands On 4

## Difference between JPA, Hibernate and Spring Data JPA

**create database :**

****

CREATE DATABASE newemployeedb;

CREATE TABLE employee (

id INT AUTO\_INCREMENT PRIMARY KEY,

name VARCHAR(100),

department VARCHAR(100)

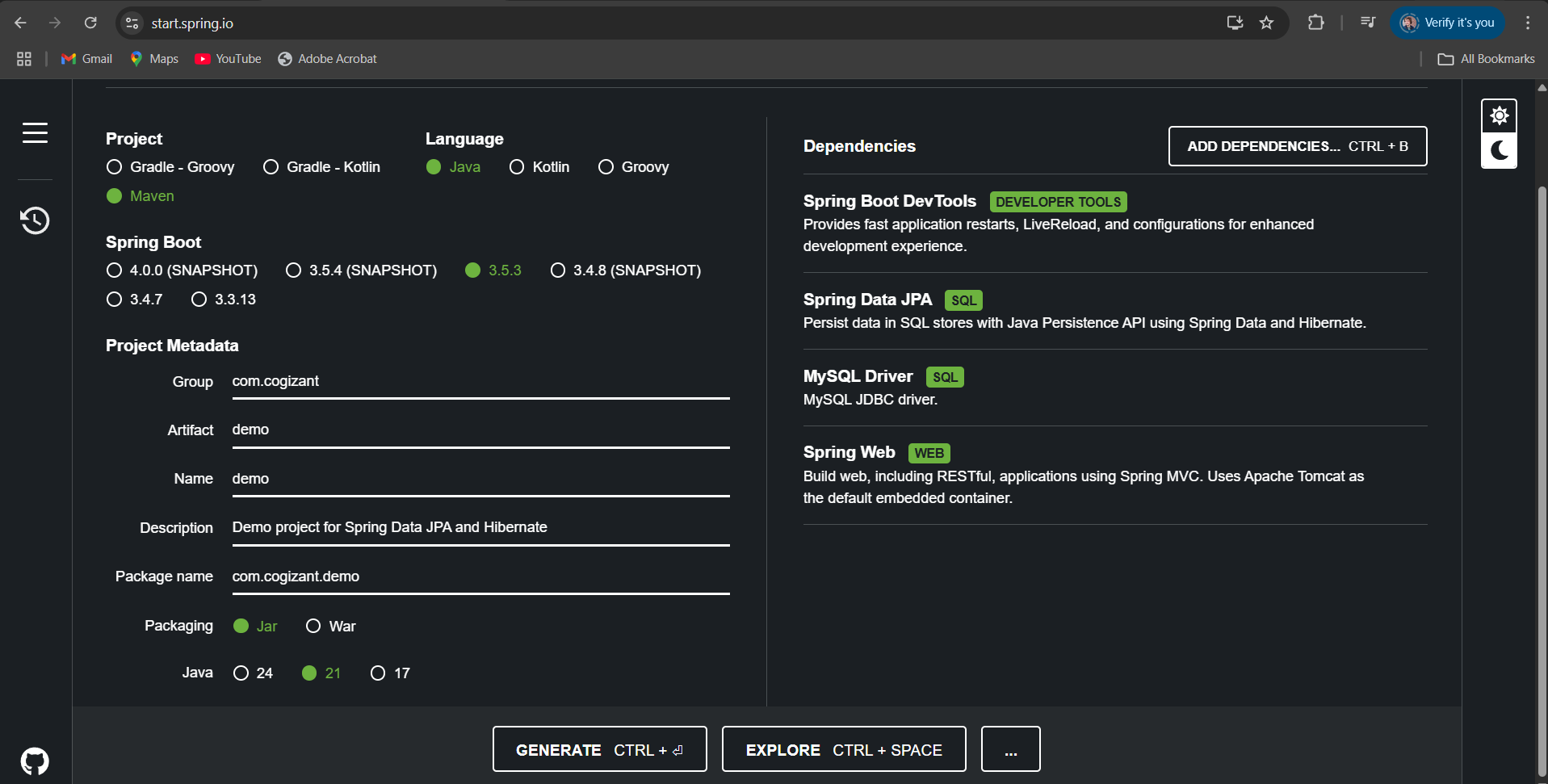
);

INSERT INTO employee (name, department) VALUES ('Harish', 'HR');

INSERT INTO employee (name, department) VALUES ('Adhi', 'Engineering');

INSERT INTO employee (name, department) VALUES ('Aathesh', 'Entrepreneur');

**Creating Project in Spring Initializr :**

****

**Application.properties :**

spring.datasource.url=jdbc:mysql://localhost:3306/newemployeedb

spring.datasource.username=root

spring.datasource.password=root

spring.datasource.initialization-mode=always

spring.sql.init.mode=always

spring.jpa.hibernate.ddl-auto=update

spring.jpa.show-sql=true

spring.jpa.properties.hibernate.dialect=org.hibernate.dialect.MySQL8Dialect

**Employee.java :**

package com.cogizant.demo.model;

import jakarta.persistence.\*;

@Entity

@Table(name = "employee")

public class Employee {

@Id

@GeneratedValue(strategy = GenerationType.IDENTITY)

private int id;

private String name;

private String department;

// Getters and Setters

public int getId() {

return id;

}

public void setId(int id) {

this.id = id;

}

public String getName() {

return name;

}

public void setName(String name) {

this.name = name;

}

public String getDepartment() {

return department;

}

public void setDepartment(String department) {

this.department = department;

}

@Override

public String toString() {

return "Employee [id=" + id + ", name=" + name + ", department=" + department + "]";

}

}

**EmployeeRepository.java :**

package com.cogizant.demo.repository;

import com.cogizant.demo.model.Employee;

import org.springframework.data.jpa.repository.JpaRepository;

import org.springframework.stereotype.Repository;

@Repository

public interface EmployeeRepository extends JpaRepository<Employee, Integer> {

}

**EmpolyeeService.java :**

package com.cogizant.demo.service;

import com.cogizant.demo.model.Employee;

import com.cogizant.demo.repository.EmployeeRepository;

import org.springframework.beans.factory.annotation.Autowired;

import org.springframework.stereotype.Service;

import jakarta.transaction.Transactional;

import java.util.List;

@Service

public class EmployeeService {

@Autowired

private EmployeeRepository employeeRepository;

@Transactional

public void addEmployee(Employee employee) {

employeeRepository.save(employee);

}

public List<Employee> getAllEmployees() {

return employeeRepository.findAll();

}

}

**EmployeeController.java :**

package com.cogizant.demo.controller;

import com.cogizant.demo.model.Employee;

import com.cogizant.demo.service.EmployeeService;

import org.springframework.beans.factory.annotation.Autowired;

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

import java.util.List;

@RestController

@RequestMapping("/employees")

public class EmployeeController {

@Autowired

private EmployeeService employeeService;

@PostMapping("/add")

public String addEmployee(@RequestBody Employee employee) {

employeeService.addEmployee(employee);

return "Employee added successfully!";

}

@GetMapping("/all")

public List<Employee> getAllEmployees() {

return employeeService.getAllEmployees();

}

@GetMapping("/{id}")

public Employee getEmployee(@PathVariable int id) {

return employeeService.getEmployeeById(id);

}

@PutMapping("/update/{id}")

public Employee updateEmployee(@PathVariable int id, @RequestBody Employee updated) {

return employeeService.updateEmployee(id, updated);

}

@DeleteMapping("/delete/{id}")

public String deleteEmployee(@PathVariable int id) {

employeeService.deleteEmployeeById(id);

return "Employee deleted successfully!";

}

}

**DemoApp.java :**

package com.cogizant.demo;

import com.cogizant.demo.model.Employee;

import com.cogizant.demo.service.EmployeeService;

import org.springframework.beans.factory.annotation.Autowired;

import org.springframework.boot.CommandLineRunner;

import org.springframework.boot.SpringApplication;

import org.springframework.boot.autoconfigure.SpringBootApplication;

import java.util.List;

@SpringBootApplication

public class DemoApp implements CommandLineRunner {

@Autowired

private EmployeeService employeeService;

public static void main(String[] args) {

SpringApplication.run(DemoApp.class, args);

}

@Override

public void run(String... args) throws Exception {

System.out.println("Starting demo...");

// Add employee

Employee emp = new Employee();

emp.setName("Mathi");

emp.setDepartment("Full Stack");

employeeService.addEmployee(emp);

System.out.println("Employee saved: " + emp);

// Fetch all employees

List<Employee> employees = employeeService.getAllEmployees();

System.out.println("All Employees:");

employees.forEach(System.out::println);

}

}

**OUTPUT :**

