Assignment -02

COURSE: Full Stack Java Programming **Semester: III**

Class: SE IT Div-A

Date of Display:

Date of Submission:

Q.No	Questions	Bloom's level	CO/LO
Q.1	Perform database operations using JPA Repository. **Instructions-** **Implementation Guidelines** • Create a **Spring Boot project** using Spring Initializr with dependencies:Spring Web,Spring Data JPA, MySQL Driver • Define an **Entity class** with appropriate fields • Create a **JpaRepository interface** for CRUD operations. • Implement a **Controller** (and Service if required) for REST endpoints. • Use **Curl/Postman** to test API requests (POST, GET, DELETE). **Database Setup** • Create database and Configure application.properties with correct username, password, and database name. **Submission Rules** • Submit in **Word/PDF format as given below** with proper formatting.	Applying	**2343611.4**

Assignment Documentation Format

- Assignment title:
- Student Name:
- Roll Number:
- Submission Date:

1. Introduction

Briefly describe what **Spring Data JPA**.Explain what **JPA Repository** is and why it is used.

2. Tools and Technologies Used

List and describe the tools:
Java (version …)

Spring Boot (version …)

Spring Data JPA

MySQL / PostgreSQL (any RDBMS used)

Maven / Gradle

IDE (IntelliJ / Eclipse / STS)

EXCELSSIOR EDUCATION SOCIET
K.C. College of Engineering & Managem
MithBunder Road, Kopri, Thane (E
Department of Infor

3. Program Structure

The program structure for the Product Database Operations using JPA Repository is as follows.

```
product-database-jpa/
├── src/
│   ├── main/
│   │   ├── java/
│   │   │   └── com/
│   │   │       └── example/
│   │   │           └── productdemo/
│   │   │               ├── ProductDemoApplication.java  # Main application class
│   │   │               │
│   │   │               ├── config/
│   │   │               │   └── DataInitializer.java      # Database initialization
│   │   │               │
│   │   │               ├── controller/
│   │   │               │   └── ProductController.java    # REST endpoints
│   │   │               │
│   │   │               ├── entity/
│   │   │               │   └── Product.java              # JPA entity class
│   │   │               │
│   │   │               ├── repository/
│   │   │               │   └── ProductRepository.java    # JPA repository interface
```

```
│   │   │           │
│   │   │           ├── service/
│   │   │           │   └── ProductService.java       # Business logic
│   │   │           │
│   │   │           └── exception/
│   │   │               ├── ErrorResponse.java        # Error response DTO
│   │   │               ├── GlobalExceptionHandler.java # Global exception handler
│   │   │               └── ResourceNotFoundException.java # Custom exception
│   │   │
│   │   └── resources/
│   │       ├── application.properties              # Application configuration
│   │       └── data.sql                            # Optional SQL initialization script
│   │
│   └── test/
│       └── java/
│           └── com/
│               └── example/
│                   └── productdemo/
│                       ├── controller/
│                       │   └── ProductControllerTest.java # Controller tests
│                       ├── service/
│                       │   └── ProductServiceTest.java    # Service tests
│                       └── repository/
│                           └── ProductRepositoryTest.java  # Repository tests
│
├── pom.xml                                         # Maven dependencies
└── README.md                                       # Project documentation
```

4. **Implementation Code**

Include Code of few sections from the repository.

Put the link to the code repository if it is uploaded on Github.(optional)

5. **Screenshots**

Output Screenshots

- Screenshot of database table before operations.
- Screenshot of database table after operations.
  - Create Product
  - Get All Products
  - Get Product by ID
  - Update Product
  - Search Products
  - Price Range Filter
  - Delete Product

6. References (if any)

Spring Documentation: https://spring.io/projects/spring-data-jpa
Official MySQL Documentation

Solution including steps to perform the database connectivity.
https://chat.z.ai/s/015a75be-37a9-4120-836a-4475592d1b92