**PROBLEM STATEMENT – 2**

**Create Spring Boot Microservice with Spring Data JPA**

**application.properties:**

server.port=8754

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

spring.datasource.url=jdbc:mysql://localhost:3306/Book\_Details

spring.datasource.username=root

spring.datasource.password=Password@12

spring.jpa.hibernate.ddl-auto=update

spring.jpa.show-sql=true

**Book.java:**

@Entity

**public** **class** Book {

@Id

@GeneratedValue(strategy = GenerationType.IDENTITY)

**private** Long id;

**private** String bookTitle;

**private** String bookPublisher;

**private** String bookIsbn;

**private** **int** bookNumberOfPages;

**private** **int** bookYear;

**public** Long getId() {

**return** id;

}

**public** **void** setId(Long id) {

**this**.id = id;

}

**public** String getBookTitle() {

**return** bookTitle;

}

**public** **void** setBookTitle(String bookTitle) {

**this**.bookTitle = bookTitle;

}

**public** String getBookPublisher() {

**return** bookPublisher;

}

**public** **void** setBookPublisher(String bookPublisher) {

**this**.bookPublisher = bookPublisher;

}

**public** String getBookIsbn() {

**return** bookIsbn;

}

**public** **void** setBookIsbn(String bookIsbn) {

**this**.bookIsbn = bookIsbn;

}

**public** **int** getBookNumberOfPages() {

**return** bookNumberOfPages;

}

**public** **void** setBookNumberOfPages(**int** bookNumberOfPages) {

**this**.bookNumberOfPages = bookNumberOfPages;

}

**public** **int** getBookYear() {

**return** bookYear;

}

**public** **void** setBookYear(**int** bookYear) {

**this**.bookYear = bookYear;

}

 }

**BookStoreRepository.java:**

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

import java.util.List;

public interface BookStoreRepository extends JpaRepository<Book, Long> {

    List<Book> findByBookTitle(String bookTitle);

    List<Book> findByBookPublisherLike(String bookPublisher);

    List<Book> findByBookYear(int bookYear);

}

**BookStoreService.java:**

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

import org.springframework.stereotype.Service;

import java.util.List;

@Service

public class BookStoreService {

    @Autowired

    private BookStoreRepository repository;

    public Book addBook(Book book) {

        return repository.save(book);

    }

    public Book updateBook(Book book) {

        return repository.save(book);

    }

    public List<Book> getAllBooks() {

        return repository.findAll();

    }

    public Book getBookById(Long id) {

        return repository.findById(id).orElse(null);

    }

    public void deleteBookById(Long id) {

        repository.deleteById(id);

    }

    public List<Book> getBooksByTitle(String title) {

        return repository.findByBookTitle(title);

    }

    public List<Book> getBooksByPublisher(String publisher) {

        return repository.findByBookPublisherLike(publisher);

    }

    public List<Book> getBooksByYear(int year) {

        return repository.findByBookYear(year);

    }

}

**BookStoreRestController.java:**

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

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

import java.util.List;

@RestController

@RequestMapping("/books")

public class BookStoreRestController {

    @Autowired

    private BookStoreService service;

    @PostMapping

    public Book addBook(@RequestBody Book book) {

        return service.addBook(book);

    }

    @PutMapping

    public Book updateBook(@RequestBody Book book) {

        return service.updateBook(book);

    }

    @GetMapping

    public List<Book> getAllBooks() {

        return service.getAllBooks();

    }

    @GetMapping("/{book\_id}")

    public Book getBookById(@PathVariable Long book\_id) {

        return service.getBookById(book\_id);

    }

    @DeleteMapping("/{book\_id}")

    public void deleteBookById(@PathVariable Long book\_id) {

        service.deleteBookById(book\_id);

    }

    @GetMapping("/title/{book\_title}")

    public List<Book> getBooksByTitle(@PathVariable String book\_title) {

        return service.getBooksByTitle(book\_title);

    }

    @GetMapping("/publisher/{book\_publisher}")

    public List<Book> getBooksByPublisher(@PathVariable String book\_publisher) {

        return service.getBooksByPublisher(book\_publisher);

    }

    @GetMapping("/year")

    public List<Book> getBooksByYear(@RequestParam int book\_year) {

        return service.getBooksByYear(book\_year);

    }

}