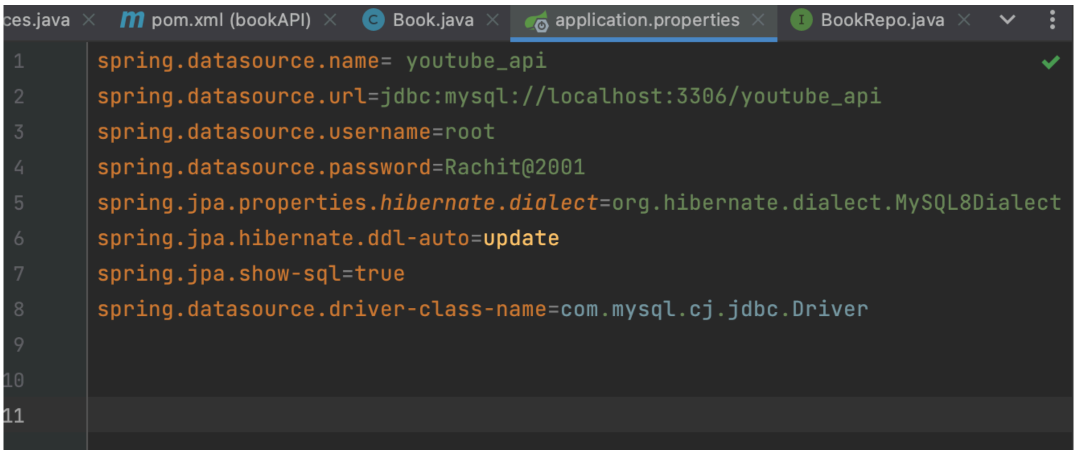
Name - Rachit Saxena
Reg Num - 20BCE2322
Campus - Vellore

Q1. Using Postman to access data from the database using Spring Boot -

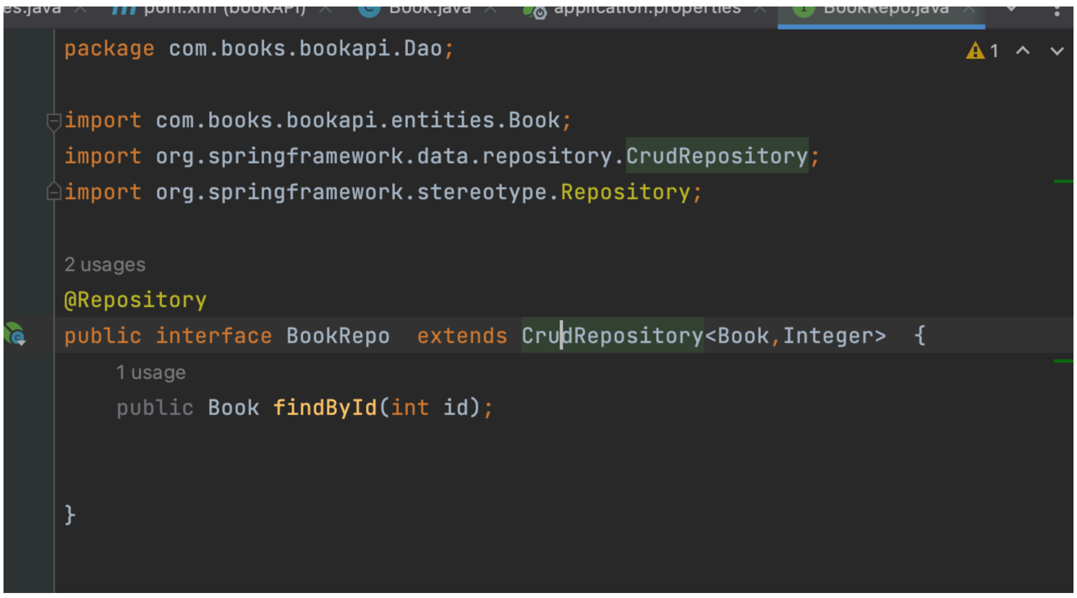
Application.properties -

```
spring.datasource.name= youtube_api
spring.datasource.url=jdbc:mysql://localhost:3306/youtube_api
spring.datasource.username=root
spring.datasource.password=XXXXXX
spring.jpa.properties.hibernate.dialect=org.hibernate.dialect.MySQL8Dialect
spring.jpa.hibernate.ddl-auto=update
spring.jpa.show-sql=true
spring.datasource.driver-class-name=com.mysql.cj.jdbc.Driver
```



```
spring.datasource.name= youtube_api
spring.datasource.url=jdbc:mysql://localhost:3306/youtube_api
spring.datasource.username=root
spring.datasource.password=Rachit@2001
spring.jpa.properties.hibernate.dialect=org.hibernate.dialect.MySQL8Dialect
spring.jpa.hibernate.ddl-auto=update
spring.jpa.show-sql=true
spring.datasource.driver-class-name=com.mysql.cj.jdbc.Driver
```

Repository -



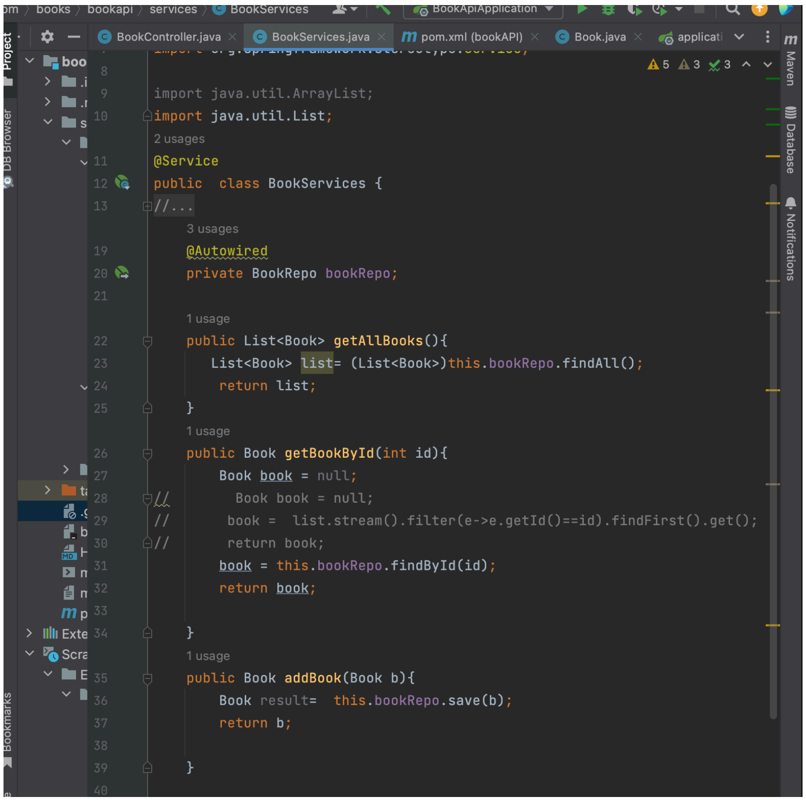
```
package com.books.bookapi.Dao;

import com.books.bookapi.entities.Book;
import org.springframework.data.repository.CrudRepository;
import org.springframework.stereotype.Repository;

2 usages
@Repository
public interface BookRepo  extends CrudRepository<Book,Integer>  {
    1 usage
    public Book findById(int id);


}
```

Service -



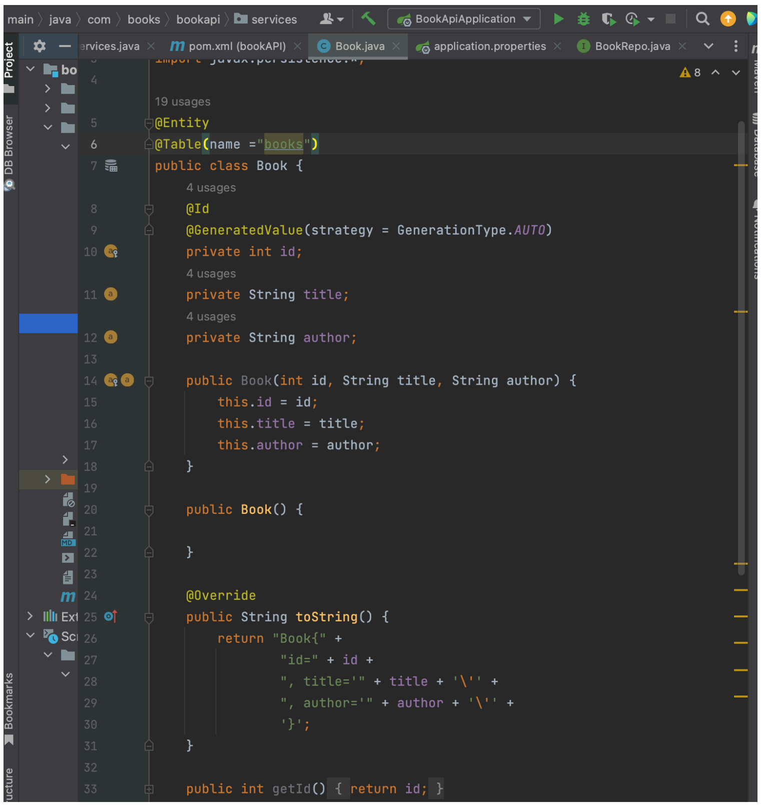
```java
import java.util.ArrayList;
import java.util.List;
2 usages
@Service
public  class BookServices {
//...
    3 usages
    @Autowired
    private BookRepo bookRepo;

    1 usage
    public List<Book> getAllBooks(){
       List<Book> list= (List<Book>)this.bookRepo.findAll();
        return list;
    }
    1 usage
    public Book getBookById(int id){
        Book book = null;
//        Book book = null;
//       book =  list.stream().filter(e->e.getId()==id).findFirst().get();
//       return book;
        book = this.bookRepo.findById(id);
        return book;

    }
    1 usage
    public Book addBook(Book b){
        Book result=  this.bookRepo.save(b);
        return b;

    }
```

Entity -



```java
19 usages
@Entity
@Table(name ="books")
public class Book {
    4 usages
    @Id
    @GeneratedValue(strategy = GenerationType.AUTO)
    private int id;
    4 usages
    private String title;
    4 usages
    private String author;

    public Book(int id, String title, String author) {
        this.id = id;
        this.title = title;
        this.author = author;
    }

    public Book() {

    }

    @Override
    public String toString() {
        return "Book{" +
                "id=" + id +
                ", title='" + title + '\'' +
                ", author='" + author + '\'' +
                '}';
    }

    public int getId() { return id; }
```

Demonstration - https://drive.google.com/drive/folders/11tgXHhwyqQ3Sr5jpSovGMb1AcP3WvnOa?usp=sharing