**Pom.xml**

<?xml version="1.0" encoding="UTF-8"?>

<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 https://maven.apache.org/xsd/maven-4.0.0.xsd">

<modelVersion>4.0.0</modelVersion>

<parent>

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

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

<version>2.7.10</version>

<relativePath /> <!-- lookup parent from repository -->

</parent>

<groupId>com.cdac</groupId>

<artifactId>SB-Many2One-Mapping-Project</artifactId>

<version>0.0.1-SNAPSHOT</version>

<name>SB-One2One-Mapping-Project</name>

<description>Spring Boot with one to one mapping</description>

<properties>

<java.version>11</java.version>

</properties>

<dependencies>

<dependency>

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

<artifactId>spring-boot-starter-data-jdbc</artifactId>

</dependency>

<dependency>

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

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

</dependency>

<dependency>

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

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

</dependency>

<dependency>

<groupId>com.mysql</groupId>

<artifactId>mysql-connector-j</artifactId>

<scope>runtime</scope>

</dependency>

<dependency>

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

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

<scope>test</scope>

</dependency>

<dependency>

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

<artifactId>spring-boot-starter-data-jpa</artifactId>

</dependency>

</dependencies>

<build>

<plugins>

<plugin>

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

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

</plugin>

</plugins>

</build>

</project>

**Application.properties**

# JDBC PROPERTIES

spring.datasource.url=jdbc:mysql://localhost:3306/pune?useSSL=false&allowPublicKeyRetrieval=true

spring.datasource.username=root

spring.datasource.password=cdac

# HIBERNATE PROPERTIES

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

spring.jpa.hibernate.ddl-auto=update

## Show sql query.

spring.jpa.show-sql=true

**Movie.java**

package com.cdac.model;

import javax.persistence.Entity;

import javax.persistence.GeneratedValue;

import javax.persistence.GenerationType;

import javax.persistence.Id;

import javax.persistence.Table;

@Entity

@Table(name="movie\_table")

public class Movie {

@Id

@GeneratedValue(strategy = GenerationType.IDENTITY)

private int movie\_Id;

private String movie\_name;

private String movie\_copyrights;

public Movie() {

}

public Movie(int movie\_Id) {

this.movie\_Id = movie\_Id;

}

public Movie(int movie\_Id, String movie\_name, String movie\_copyrights) {

this.movie\_Id = movie\_Id;

this.movie\_name = movie\_name;

this.movie\_copyrights = movie\_copyrights;

}

public int getMovie\_Id() {

return movie\_Id;

}

public void setMovie\_Id(int movie\_Id) {

this.movie\_Id = movie\_Id;

}

public String getMovie\_name() {

return movie\_name;

}

public void setMovie\_name(String movie\_name) {

this.movie\_name = movie\_name;

}

public String getMovie\_copyrights() {

return movie\_copyrights;

}

public void setMovie\_copyrights(String movie\_copyrights) {

this.movie\_copyrights = movie\_copyrights;

}

@Override

public String toString() {

return movie\_Id + " " + movie\_name + " " + movie\_copyrights;

}

}

**Songs.java**

package com.cdac.model;

import javax.persistence.CascadeType;

import javax.persistence.Entity;

import javax.persistence.GeneratedValue;

import javax.persistence.GenerationType;

import javax.persistence.Id;

import javax.persistence.JoinColumn;

import javax.persistence.ManyToOne;

import javax.persistence.Table;

@Entity

@Table(name="songs\_table")

public class Songs {

@Id

@GeneratedValue(strategy = GenerationType.IDENTITY)

private int song\_Id;

private String song\_name;

private String song\_singers;

@ManyToOne(cascade = CascadeType.ALL)

@JoinColumn(name = "movie\_Id")

private Movie movie;

public Songs() {

}

public Songs(int song\_Id) {

super();

this.song\_Id = song\_Id;

}

public Songs(int song\_Id, String song\_name, String song\_singers, Movie movie) {

super();

this.song\_Id = song\_Id;

this.song\_name = song\_name;

this.song\_singers = song\_singers;

this.movie = movie;

}

public int getSong\_Id() {

return song\_Id;

}

public void setSong\_Id(int song\_Id) {

this.song\_Id = song\_Id;

}

public String getSong\_name() {

return song\_name;

}

public void setSong\_name(String song\_name) {

this.song\_name = song\_name;

}

public String getSong\_singers() {

return song\_singers;

}

public void setSong\_singers(String song\_singers) {

this.song\_singers = song\_singers;

}

public Movie getMovie() {

return movie;

}

public void setMovie(Movie movie) {

this.movie = movie;

}

@Override

public String toString() {

return song\_Id + " " + song\_name + " " + song\_singers + " " + movie;

}

}

**MovieDao.java**

package com.cdac.dao;

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

import org.springframework.stereotype.Repository;

import com.cdac.model.Post;

@Repository

public interface MovieDao extends JpaRepository<Movie, Integer>{

}

**CommentDao.java**

package com.cdac.dao;

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

import org.springframework.stereotype.Repository;

import com.cdac.model.Comment;

@Repository

public interface SongsDao extends JpaRepository<Songs, Integer>{

}

**MovieService.java**

package com.cdac.service;

import java.util.List;

import com.cdac.model.Movie;

public interface MovieServies {

void add\_Movie(Movie movie);

void update\_Movie(Movie movie);

void delete\_Movie\_By\_Id(int id);

Movie get\_Movie\_By\_Id(int id);

List<Movie> get\_All\_Movies();

}

**MovieServiceImpl.java**

package com.cdac.service;

import java.util.ArrayList;

import java.util.List;

import java.util.Optional;

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

import org.springframework.stereotype.Service;

import com.cdac.dao.MovieDao;

import com.cdac.model.Movie;

@Service

public class MovieServiceImpl implements MovieServies {

@Autowired

private MovieDao movieDao;

@Override

public void add\_Movie(Movie movie) {

movieDao.save(movie);

}

@Override

public void update\_Movie(Movie movie) {

movieDao.save(movie);

}

@Override

public void delete\_Movie\_By\_Id(int id) {

movieDao.deleteById(id);

}

@Override

public Movie get\_Movie\_By\_Id(int id) {

Optional<Movie> opt = movieDao.findById(id);

if (opt.isPresent()) {

return opt.get();

}

return null;

}

@Override

public List<Movie> get\_All\_Movies() {

Iterable<Movie> itr = movieDao.findAll();

List<Movie> lst = new ArrayList<Movie>();

itr.forEach(ele -> lst.add(ele));

return lst;

}

}

**SongService.java**

package com.cdac.service;

import java.util.List;

import com.cdac.model.Songs;

public interface SongsService {

void add\_Songs(Songs song);

void update\_Songs(Songs song);

void delete\_Songs\_By\_Id(int id);

Songs get\_Songs\_By\_Id(int id);

List<Songs> get\_All\_Songs();

}

**SongServiceImpl.java**

package com.cdac.service;

import java.util.ArrayList;

import java.util.List;

import java.util.Optional;

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

import org.springframework.stereotype.Service;

import com.cdac.dao.SongsDao;

import com.cdac.model.Songs;

@Service

public class SongsServiceImpl implements SongsService {

@Autowired

private SongsDao songsDao;

@Override

public void add\_Songs(Songs song) {

songsDao.save(song);

}

@Override

public void update\_Songs(Songs song) {

songsDao.save(song);

}

@Override

public void delete\_Songs\_By\_Id(int id) {

songsDao.deleteById(id);

}

@Override

public Songs get\_Songs\_By\_Id(int id) {

Optional<Songs> opt = songsDao.findById(id);

if (opt.isPresent()) {

return opt.get();

}

return null;

}

@Override

public List<Songs> get\_All\_Songs() {

Iterable<Songs> itr = songsDao.findAll();

List<Songs> lst = new ArrayList<Songs>();

itr.forEach(ele -> lst.add(ele));

return lst;

}

}

**SongsController.java**

package com.cdac.controller;

import java.util.List;

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

import org.springframework.web.bind.annotation.DeleteMapping;

import org.springframework.web.bind.annotation.GetMapping;

import org.springframework.web.bind.annotation.PathVariable;

import org.springframework.web.bind.annotation.PostMapping;

import org.springframework.web.bind.annotation.PutMapping;

import org.springframework.web.bind.annotation.RequestBody;

import org.springframework.web.bind.annotation.RequestParam;

import org.springframework.web.bind.annotation.RestController;

import com.cdac.model.Songs;

import com.cdac.service.SongsService;

@RestController

public class SongsController {

@Autowired

private SongsService songsService;

@PostMapping(value = { "/song" })

public String add\_Songs(@RequestBody Songs song) {

songsService.add\_Songs(song);

return "Songs added successfully";

}

@PutMapping(value = { "/song" })

public String update\_Songs(@RequestBody Songs song) {

songsService.update\_Songs(song);

return "Songs updated successfully";

}

@DeleteMapping(value = { "/song" })

public String delete\_Songs(@RequestParam int id) {

System.out.println("first delete");

songsService.delete\_Songs\_By\_Id(id);

return "Songs deleted successfully";

}

@DeleteMapping(value = { "/song/{id}" })

public String delete\_Songs1(@PathVariable int id) {

System.out.println("second delete");

songsService.delete\_Songs\_By\_Id(id);

return "Songs deleted successfully";

}

@GetMapping(value = { "/song/{id}" })

public Songs get\_Songs\_By\_Id(@PathVariable int id) {

return songsService.get\_Songs\_By\_Id(id);

}

@GetMapping(value = { "/song" })

public List<Songs> get\_All\_Songs() {

return songsService get\_All\_Songs();

}

}

**SBMany2OneMappingProject.java**

package com.cdac;

import org.springframework.boot.SpringApplication;

import org.springframework.boot.autoconfigure.SpringBootApplication;

@SpringBootApplication

public class SBOne2ManyMappingProjectApplication {

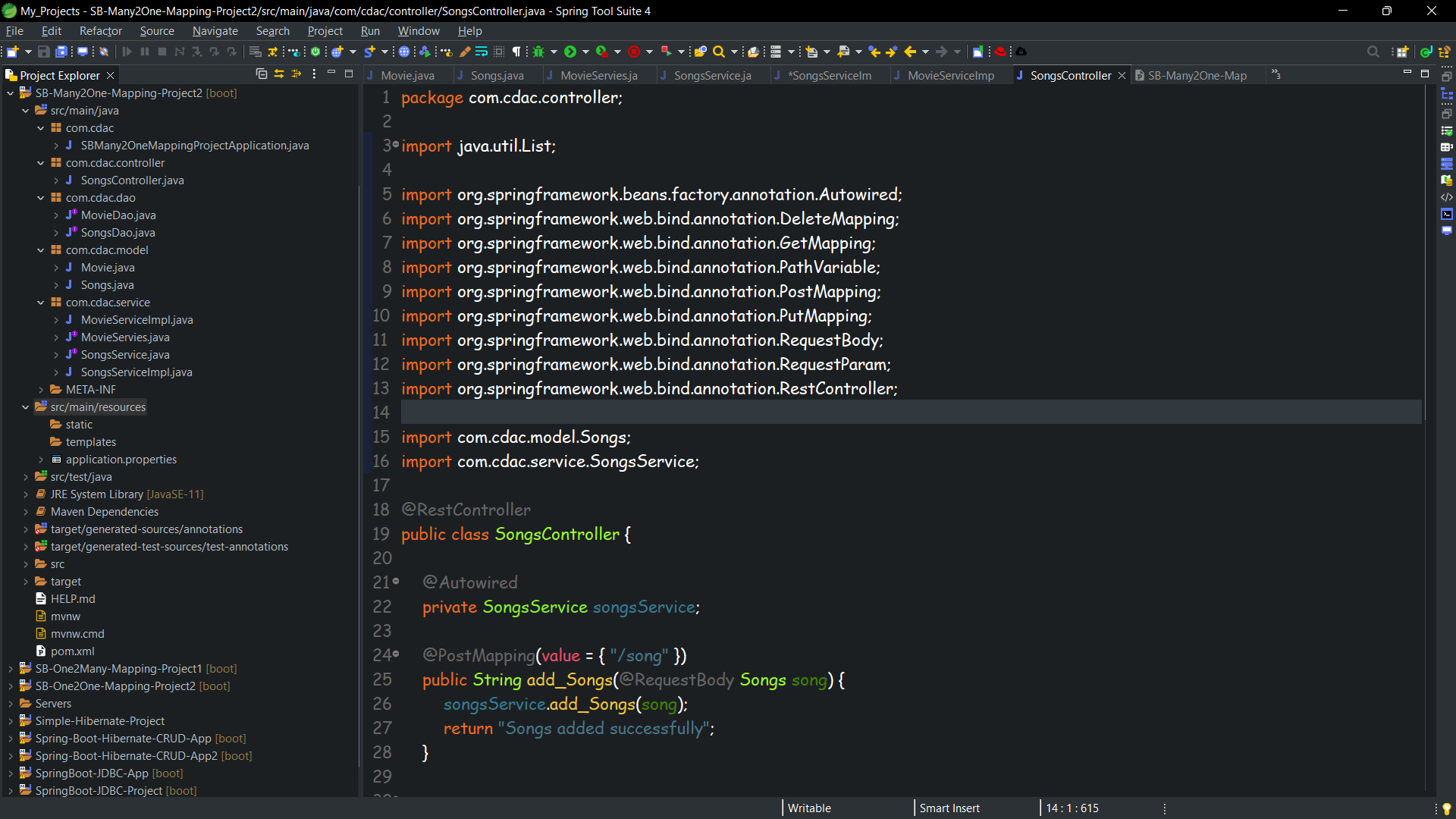
public static void main(String[] args) {

SpringApplication.run(SBMany2OneMappingProjectApplication.class, args);

}

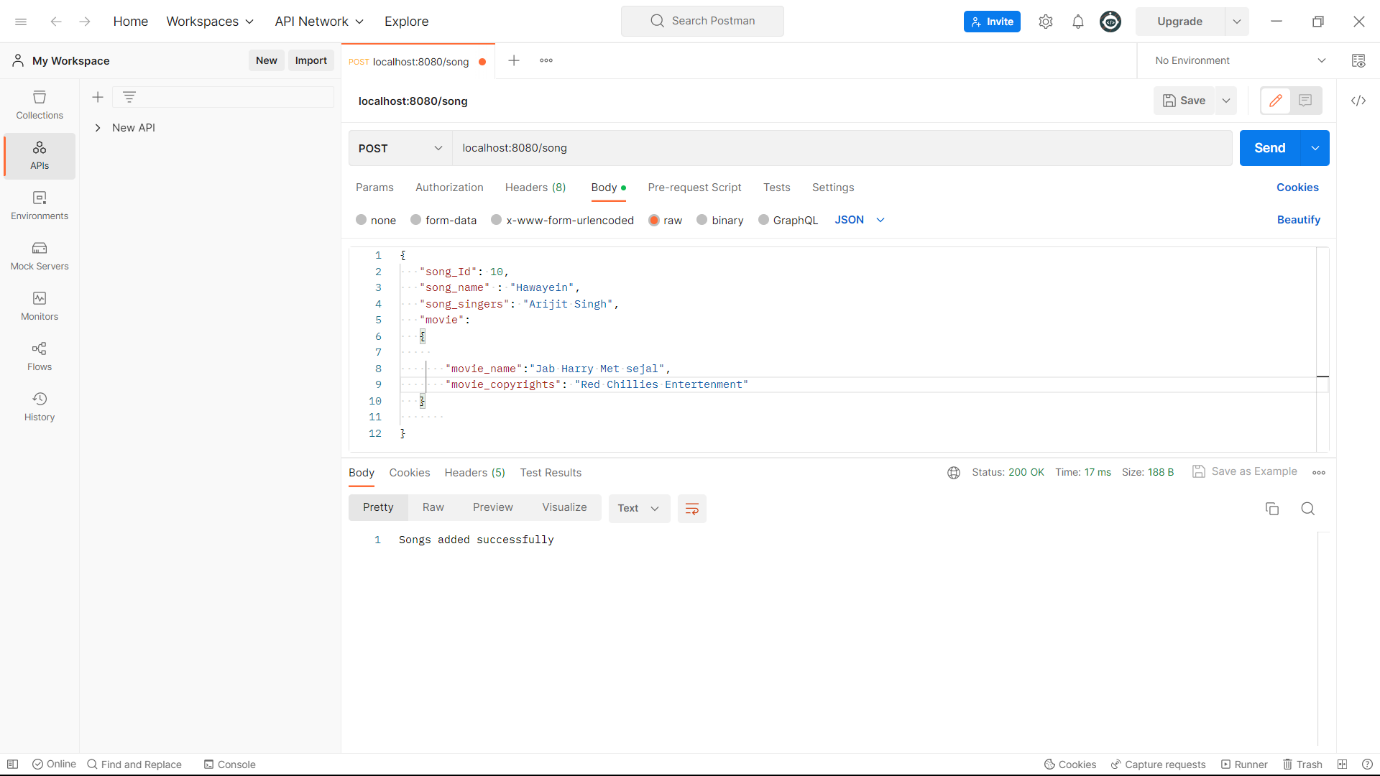
}

**Project Structrure**

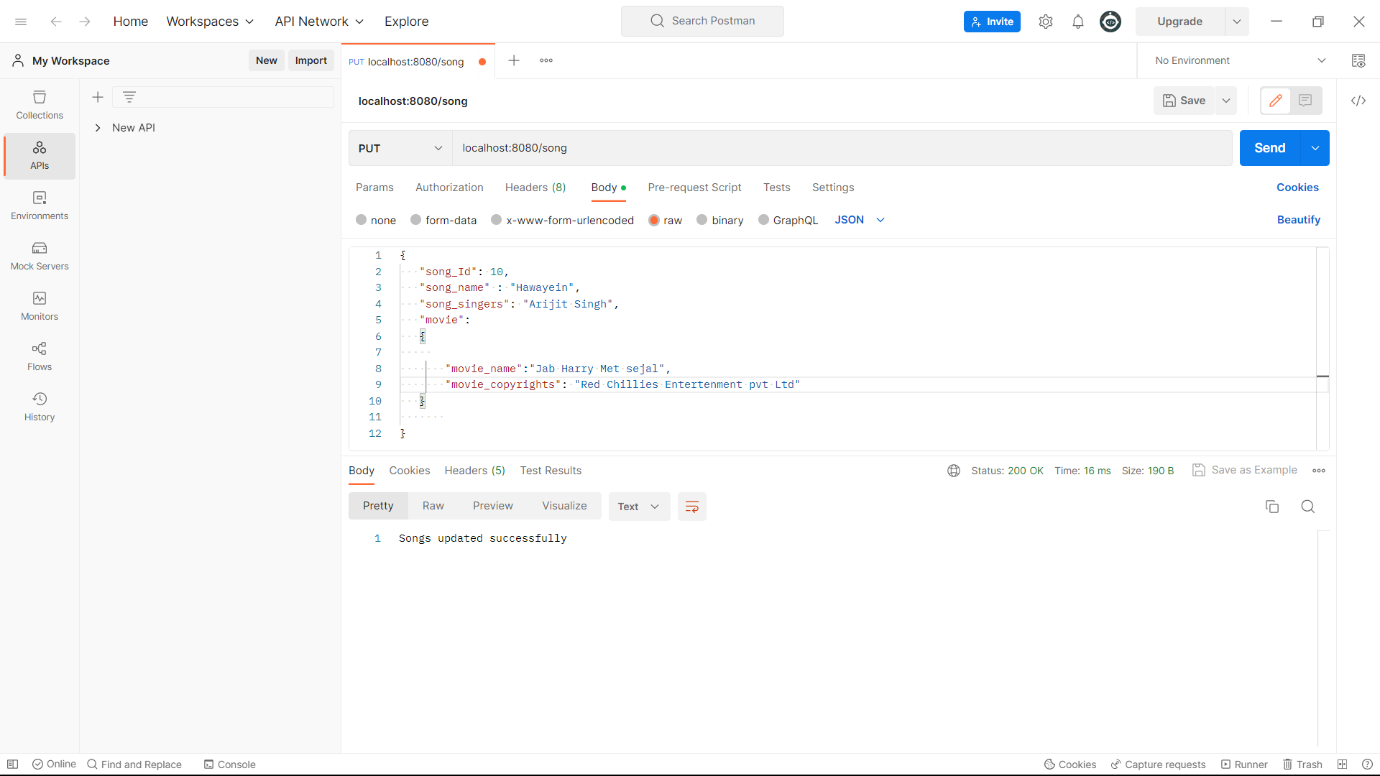


**PostMan and Database ScreenShots**

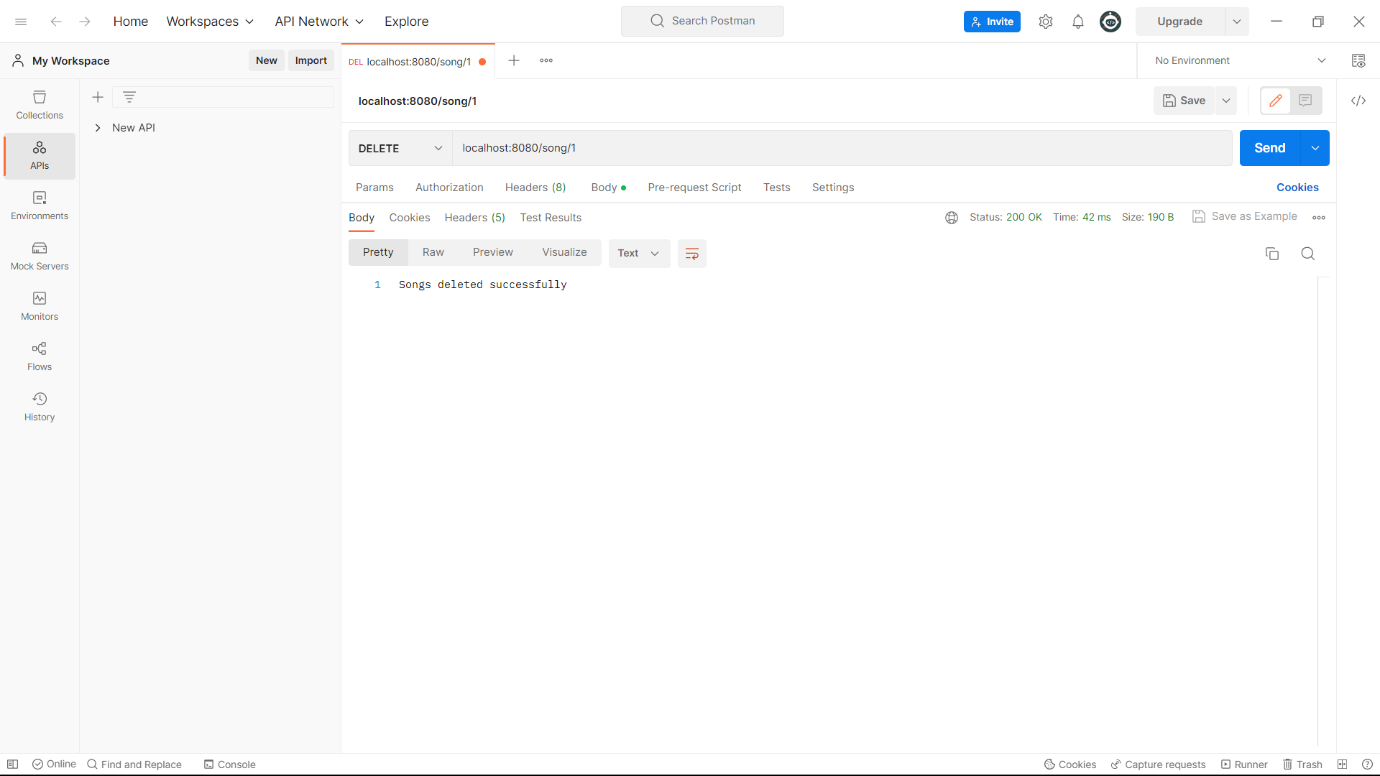
**Insert**



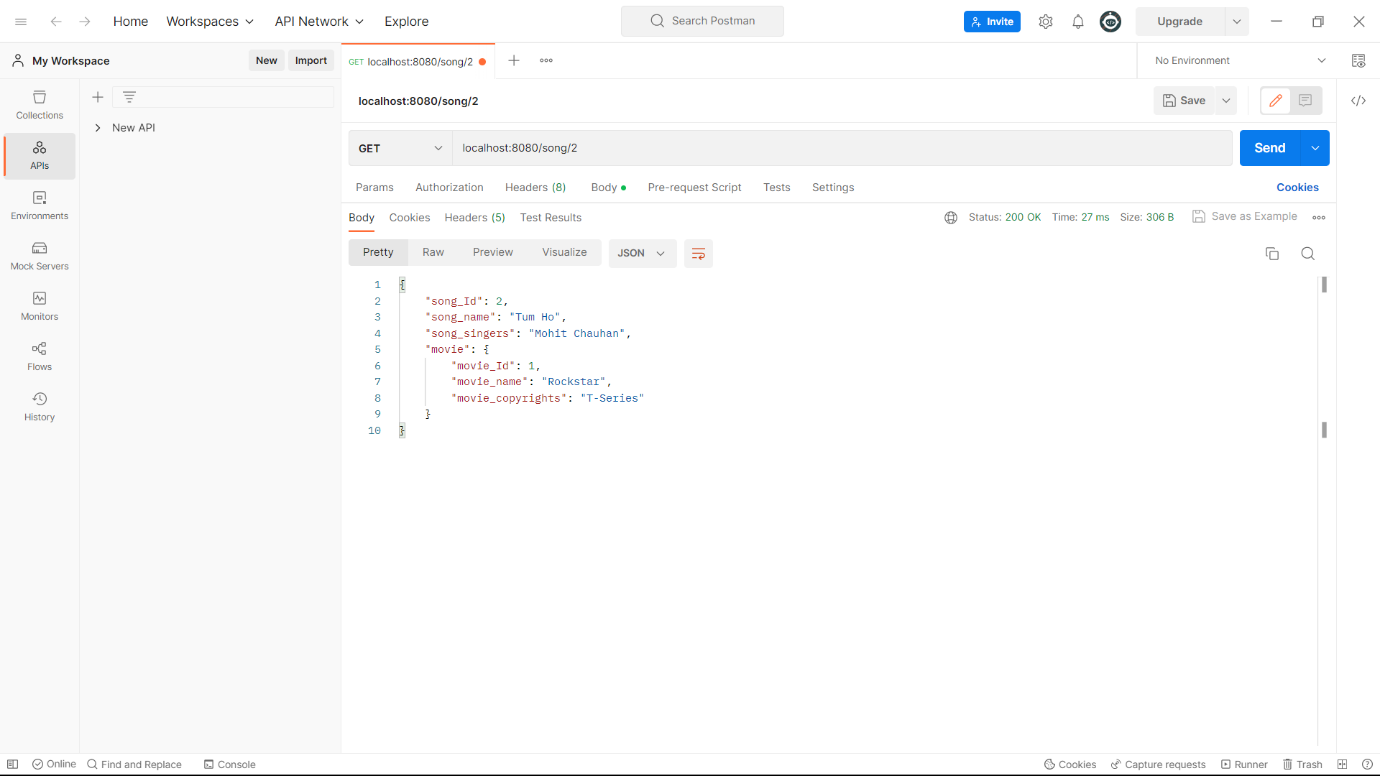
**Update**

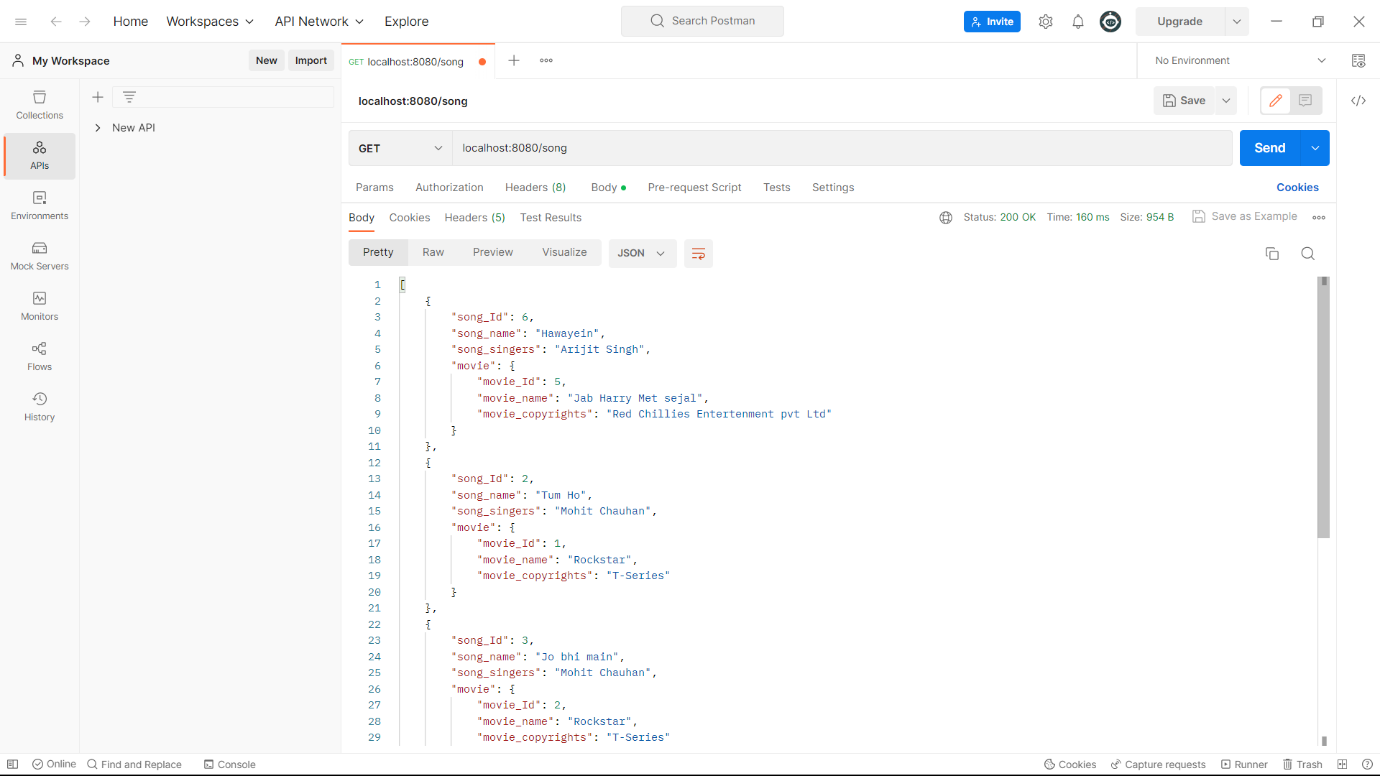


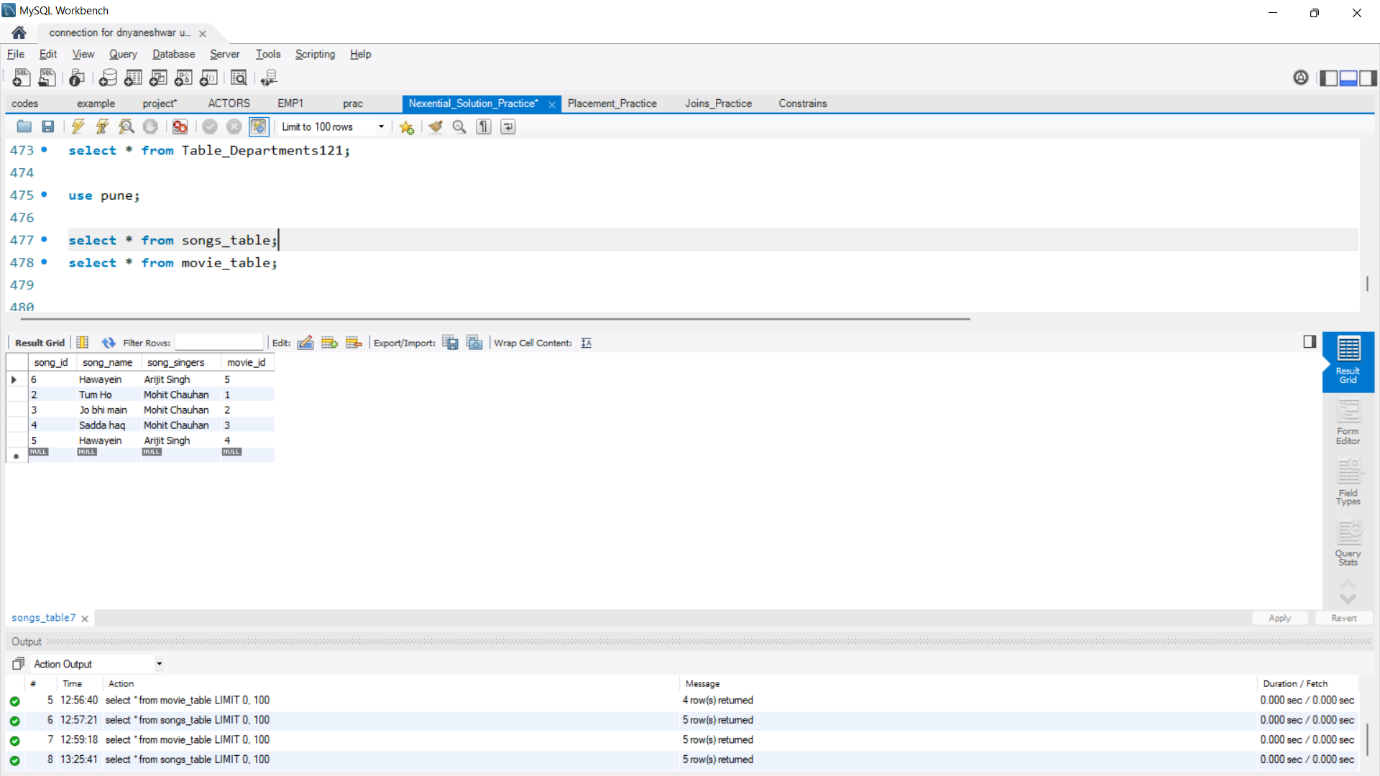
**delete**



**select by id**



**select all** 

**Table Songs** 

**Table Movie** 