

1. Feedback.java

```
package com.bean;
```

```
import javax.persistence.Entity;
```

```
import javax.persistence.Id;
```

```
@Entity
```

```
public class Feedback {
```

```
    @Id
```

```
    private int id;
```

```
    private String fname;
```

```
    private String lname;
```

```
    private String feedback;
```

```
    public int getId() {
```

```
        return id;
```

```
    }
```

```
    public void setId(int id) {
```

```
        this.id = id;
```

```
    }
```

```
    public String getFname() {
```

```
        return fname;
```

```
    }
```

```
    public void setFname(String fname) {
```

```
        this.fname = fname;
```

```
    }
```

```
    public String getLname() {
```

```
        return lname;
```

```
    }
```

```
    public void setLname(String lname) {
```

```
        this.lname = lname;
```

```
    }
```

```
    public String getFeedback() {
```

```
        return feedback;
```

```
    }
```

```
    public void setFeedback(String feedback) {
```

```
        this.feedback = feedback;
```

```
    }
```

```
    @Override
```

```
    public String toString() {
```

```
        return "Feedback [id=" + id + ", fname=" + fname + ", lname=" + lname + ",  
feedback=" + feedback + "];"
```

```
    }
```

```
    public Feedback(int id, String fname, String lname, String feedback) {
```

```

        super();
        this.id = id;
        this.fname = fname;
        this.lname = lname;
        this.feedback = feedback;
    }
    public Feedback() {
        super();
        // TODO Auto-generated constructor stub
    }
}

```

2. FeedbackController.java

```

package com.controller;

import org.springframework.beans.factory.annotation.Autowired;
import org.springframework.http.MediaType;
import org.springframework.web.bind.annotation.RequestBody;
import org.springframework.web.bind.annotation.RequestMapping;
import org.springframework.web.bind.annotation.RequestMethod;
import org.springframework.web.bind.annotation.RestController;

import com.bean.Feedback;
import com.service.FeedbackService;

@RestController
public class FeedbackController {
    @Autowired
    FeedbackService feedbackService;

    @RequestMapping(value = "storeFeedback", consumes =
    MediaType.APPLICATION_JSON_VALUE, method = RequestMethod.POST)
    public String storeEmployee(@RequestBody Feedback feedback) { // scan
        the value from request body in the form json
        return feedbackService.storeFeedback(feedback);
    }
}

```

3. FeedbackDAO.java

```

package com.dao;

import javax.persistence.EntityManager;
import javax.persistence.EntityManagerFactory;
import javax.persistence.EntityTransaction;

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

import com.bean.Feedback;

@Repository
public class FeedbackDAO {
    @Autowired
    EntityManagerFactory emf;

    public int storeFeedback(Feedback feedback) {
        try {
            EntityManager manager = emf.createEntityManager();
            EntityTransaction tran = manager.getTransaction();
            tran.begin();
            manager.persist(feedback);
            // session.save(emp)
            tran.commit();
            return 1;
        } catch (Exception e) {
            System.out.println(e);
            return 0;
        }
    }
}

```

4. FeedbackService.java

```

package com.service;

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

import com.bean.Feedback;
import com.dao.FeedbackDAO;

```

```

@Service
public class FeedbackService {
    @Autowired
    FeedbackDAO feedbackDao;

    public String storeFeedback(Feedback feedback) {
        if(feedbackDao.storeFeedback(feedback)>0) {
            return "Record stored";
        }else {
            return "Record didn't store";
        }
    }
}

```

5. DemoTest.java

```

package com.main;

import org.springframework.boot.SpringApplication;
import org.springframework.boot.autoconfigure.SpringBootApplication;
import org.springframework.boot.autoconfigure.domain.EntityScan;

@SpringBootApplication(scanBasePackages = "com")
@EntityScan(basePackages = "com.bean")// mapping tag in hibernate.cfg.xml
public class DemoTest {

    public static void main(String[] args) {
        SpringApplication.run(DemoTest.class, args);
        System.out.println("Server up");
    }

}

```

6. pom.xml

```

<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>
    <groupId>MyProjects</groupId>
    <artifactId>UserFeedBackDatabase</artifactId>

```

```

<version>0.0.1-SNAPSHOT</version>

<properties>
  <project.build.sourceEncoding>UTF-8</project.build.sourceEncoding>
  <maven.compiler.source>1.8</maven.compiler.source>
  <maven.compiler.target>1.8</maven.compiler.target>
</properties>

<dependencies>
<dependency>
  <groupId>org.springframework.boot</groupId>
  <artifactId>spring-boot-starter-web</artifactId>
  <version>2.5.2</version>
</dependency>
<dependency>
  <groupId>org.springframework.boot</groupId>
  <artifactId>spring-boot-starter-data-jpa</artifactId>
  <version>2.5.2</version>
</dependency>
<dependency>
  <groupId>mysql</groupId>
  <artifactId>mysql-connector-java</artifactId>
  <version>8.0.16</version>
</dependency>
<dependency>
  <groupId>javax.xml.bind</groupId>
  <artifactId>jaxb-api</artifactId>
  <version>2.2.12</version>
</dependency>

</dependencies>
</project>

```

7. application.properties

```

#server.port 9090
spring.datasource.driver-class-name=com.mysql.cj.jdbc.Driver
spring.datasource.url=jdbc:mysql://localhost:3306/phase3
spring.datasource.username=root
spring.datasource.password=Saveri@12052000

```