#AIM: Define a RESTful WebService that accepts the details to be stored in a "student" table(id, sname, sclass) and perform CRUD(CREATE, READ, UPDATE AND DELETE) operations.

STEPS(MYSQL Command Line Client):

1) Create a database and use it:
   - **>create database p6;**
   - **>use p6;**

STEPS(NetBeans IDE):

1) Create a **Web Application**:
   File > New Project > Java Web > Web Application > Name: (*p6*) **>**
   Add GlassFish Server **> Finish >** (you'll get index.jsp, delete this file)
2) Create a database connection:
   (NOTE THAT: here we are connecting our created database with NetBeans IDE, so make sure you add your *database_name* and *password* correctly)
   Navigate to **Services** tab **> Databases > Drivers >** right-click over **MySQL(Connector/J driver) >** click **Connect Using… >** Database: (*change "mysql" to "p6"*) & Password: (*root*) & click **Test Connection > Finish.**

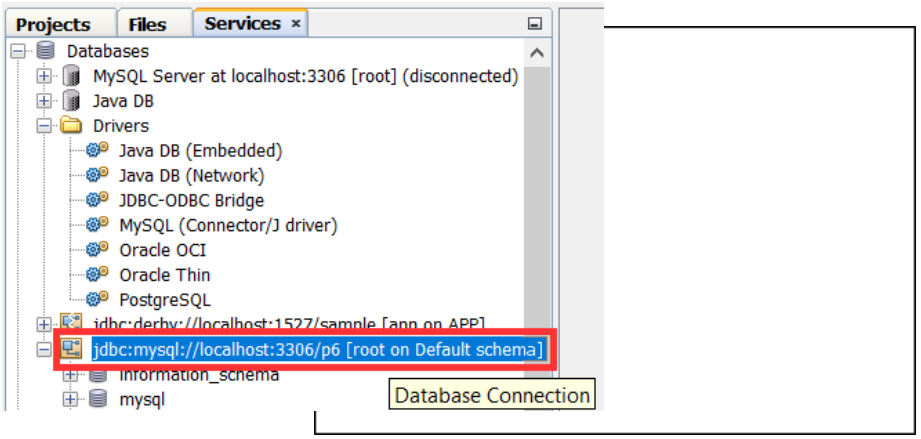


Figure1: Connecting with your database will give you this db connection string

3) Create an **Entity Class**:
   (NOTE THAT: this entity class will create a table with the class name we are providing now.)
   Navigate to **Projec**
   ts tab **> right-click over Project Name(p6) > New > Entity Class… >**
   Class Name: (*student*) & Package: (*tycs*) **> Next >**
   Data Source: (*select **New Data Source…** from dropdown*) **>**
   JNDI Name: (*ANYNAME*) &
   Database Connection: (*select **your db connection string** from dropdown*) **> OK > Finish >**
   (you'll get student.java file).
4) Adding some code under class student in "student.java"(entity class) file:
   (NOTE THAT: these properties will be created as columns in the student table
   We'll not create "id" property as it is AUTO-GENERATED)
   - **Right-click > Insert Code… > Add Property… >** Name: (*sname*) **> OK.**
   - **Right-click > Insert Code… > Add Property… >** Name: (*sclass*) **> OK.**

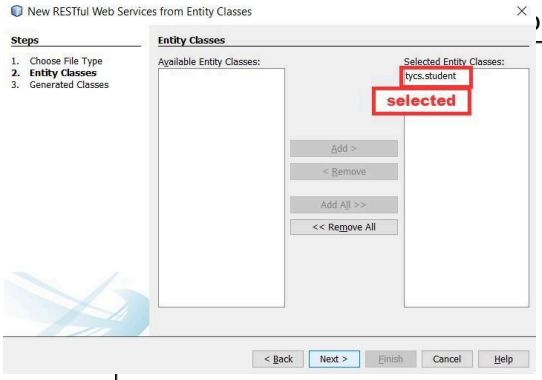
CODE(student.java):

```
package tycs;

import         java.io.Serializable;
import javax.persistence.Entity;
import  javax.persistence.GeneratedValue;
import  javax.persistence.GenerationType;
import javax.persistence.Id;

@Entity
public class student implements Serializable {
   private static final long serialVersionUID = 1L;
   @Id
   @GeneratedValue(strategy = GenerationType.AUTO)
   private Long id;

   public Long getId() {
      return id;
   }

   public void setId(Long id) {
      this.id = id;
   }

   //OUR INSERTED CODE for sname
   private String sname;

   /**
    * Get the value of sname
    *
    * @return the value of sname
    */
   public String getSname() {
      return sname;
   }
```

```
/**
 * Set the value of sname
 *
 * @param sname new value of sname
 */
public void setSname(String sname)
   { this.sname = sname;
}

//OUR INSERTED CODE for sclass
private String sclass;

/**
 * Get the value of sclass
 *
 * @return the value of sclass
 */
public String getSclass() {
   return sclass;
}

/**
 * Set the value of sclass
 *
 * @param sclass new value of sclass
 */
public void setSclass(String sclass) {
   this.sclass = sclass;
}

@Override
public int hashCode() {
   int hash = 0;
   hash += (id != null ? id.hashCode() : 0);
   return hash;
}
```

```
    @Override
    public boolean equals(Object object) {
        // TODO: Warning - this method won't work in the case the id fields are not set
        if (!(object instanceof student)) {
            return false;
        }
        student other = (student) object;
        if ((this.id == null && other.id != null) || (this.id != null && !this.id.equals(other.id))) {
            return false;
        }
        return true;
    }

    @Override
    public String toString() {
        return "tycs.student[ id=" + id + " ]";
    }
}
```

5) Create a **RESTful Web Services from Entity Classes**:
Right-click over Project Name > New > RESTful Web Services from Entity Classes… >



om Available to Selected

> Next > Resource Package: (*select **tycs** from dropdown*) **> Finish > OK.**

6) Create a **JSF Pages from Entity Classes:**
Right-click over Project Name > New > JSF Pages from Entity Classes… >
Add your entity class(tycs.student) from Available to Selected **> Next > Next > Finish.**

7) Perform CRUD(CREATE, READ, UPDATE, DELETE) operations.

- **OUTPUTS:**

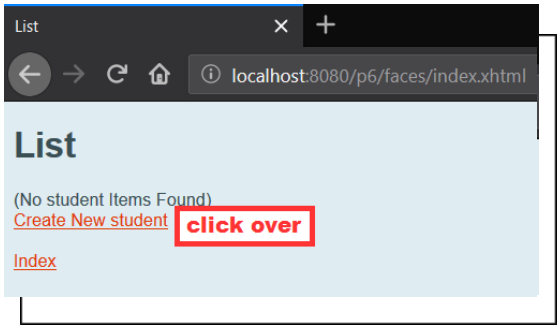


Figure 1

Figure 2

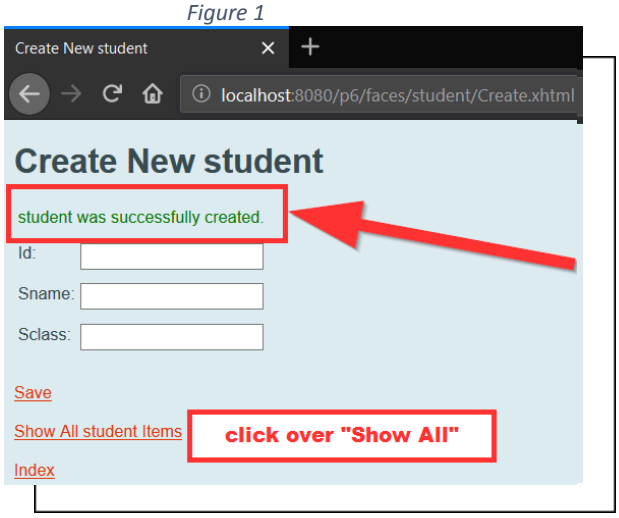


Figure 3

Figure 4

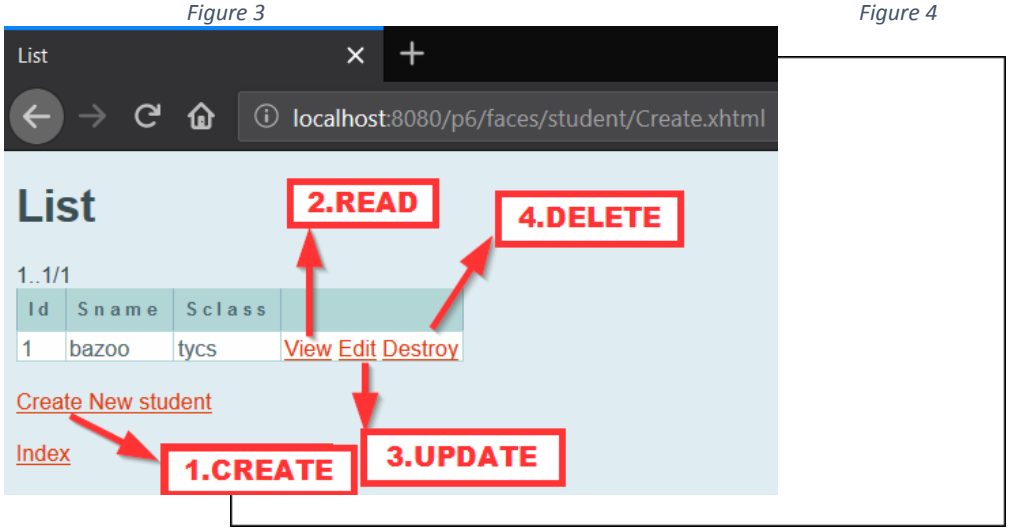


Figure 5

```
mysql> select * from student;
+----+--------+-------+
| ID | SCLASS | SNAME |
+----+--------+-------+
|  1 | tycs   | bazoo |
+----+--------+-------+
1 row in set (0.00 sec)
```

Figure 6