Project Structure

- setexample
  - src/main/java
    - com.hibernate.setexample
      - Answers.java
      - App.java
      - HibernateUtil.java
      - Questions.java
    - resources
      - hibernate.cfg.xml
      - questions.hbm.xml
  - src/test/java
  - JRE System Library [J2SE-1.5]
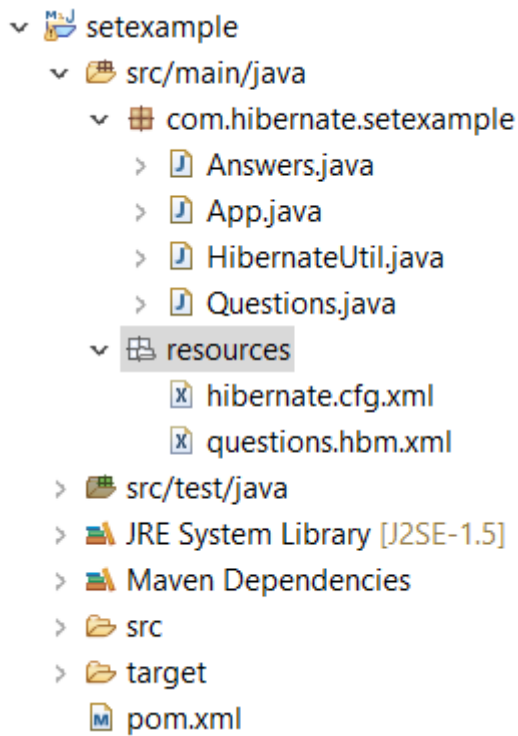 - Maven Dependencies
  - src
  - target
  - pom.xml

Step 1: Create Maven Project

1. Open your IDE (e.g., IntelliJ IDEA, Eclipse).
2. Create a new Maven project.
   - Group ID: com.hibernate
   - Artifact ID: setexample

Step 2: Add the below Dependencies to pom.xml

1. mysql-connector-j
2. hibernate-core (hibernate-core)
3. jakarta.persistence-api (jakarta.persistence)

Step-3 Create the below Classes

1. Create the Questions class in the com.hibernate.setexample package.
2. Create the Answers class in the com.hibernate.setexample package.

Class : Questions

package com.hibernate.setexample;

import java.util.Set;

import jakarta.persistence.Column;
import jakarta.persistence.Entity;
import jakarta.persistence.Id;

@Entity
public class Questions {

```java
    @Id
    private int qid;

    @Column
    private String qname;

    private Set<String> answers;

    public int getQid() { return qid; }
    public void setQid(int qid) { this.qid = qid; }

    public String getQname() { return qname; }
    public void setQname(String qname) { this.qname = qname; }

    public Set<String> getAnswers() { return answers; }
    public void setAnswers(Set<String> answers) {
            this.answers = answers; }
}
```

Class : Answers
package com.hibernate.setexample;

```java
import java.util.Set;
import jakarta.persistence.Column;
import jakarta.persistence.Entity;

@Entity
public class Answers {
        @Column
        private int aid;

        @Column
        private Set<String> questionAns;

        public int getAid() { return aid; }
        public void setAid(int aid) { this.aid = aid; }

        public Set<String> getQuestionAns() { return questionAns; }
        public void setQuestionAns(Set<String> questionAns) {
                this.questionAns = questionAns;
        }
}
```

Step 4: Create the questions.hbm.xml File

Create a resources package under the src/main/java folder and then create an XML file named **questions.hbm.xml** with the following content:

```xml
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE hibernate-mapping PUBLIC
    "-//Hibernate/Hibernate Mapping DTD 3.0//EN"
    "http://hibernate.sourceforge.net/hibernate-mapping-3.0.dtd">
<hibernate-mapping>
  <class name="com.hibernate.setexample.Questions" table="QUESTIONS">
    <id name="qid" type="int" column="qid">
      <generator class="assigned"/>
    </id>
    <property name="qname" type="string">
      <column name="questionName"/>
    </property>


                <set name="Answers" table="ANSWERS">
                <key column="qid"></key>
                <element column="questionAns" type="string"></element>
                </set>
          </class>
</hibernate-mapping>
```

Step 5: Create the hibernate.cfg.xml File

Create an XML file named hibernate.cfg.xml with the following content inside the resources folder.

```xml
<?xml version="1.0" encoding="UTF-8"?>
<hibernate-configuration>
        <session-factory>
                <property name="hibernate.connection.driver_class"> com.mysql.cj.jdbc.Driver </property>
                <property name="hibernate.connection.url">
                        jdbc:mysql://localhost:3306/hibernateDB
                </property>
                <property name="hibernate.connection.username">root</property>
                <property name="hibernate.connection.password">password</property>
                <property name="hibernate.dialect">org.hibernate.dialect.MySQLDialect</property>
                <property name="show_sql">true</property>
                <property name="format_sql">true</property>
                <property name="hbm2ddl.auto">create </property>
```

```
                    <mapping resource="resources/questions.hbm.xml" />
        </session-factory>
</hibernate-configuration>
```

Note: Ensure the MySQL database hibernateDB exists, and replace the username and password with your actual MySQL credentials.

Step 5: Create the HibernateUtil Class

Create the HibernateUtil class in the com.hibernate.setexample package:

```java
package com.hibernate.setexample;
import org.hibernate.SessionFactory;
import org.hibernate.cfg.Configuration;

public class HibernateUtil {
        private static final SessionFactory sessionFactory = buildSessionFactory();
        private static SessionFactory buildSessionFactory() {
                SessionFactory sessionFactory = null;
                try {
                        Configuration configuration = new Configuration();
                        configuration.configure("resources/hibernate.cfg.xml");
                        sessionFactory = configuration.buildSessionFactory();
                }
                catch (Exception e) { e.printStackTrace(); }
                return sessionFactory;
        }

        public static SessionFactory getSessionFactory()  {
                return sessionFactory;
        }
}
```

Step-6 Now add the below lines of code into **App.java** class file.

```java
package com.hibernate.setexample;

import org.hibernate.HibernateException;
import org.hibernate.Session;
import org.hibernate.Transaction;

import java.util.*;

public class App
```

```java
{
    public static void main( String[] args )
    {
        Session session = HibernateUtil.getSessionFactory().openSession();
        Transaction tran = null;

        try {
            tran = session.beginTransaction();
            Questions question1=new Questions();
            question1.setQid(1);
            question1.setQname("What is Java?");

            HashSet<String> qans1 = new HashSet<String>();
            qans1.add("Java is a programming language");
            qans1.add("Java is a platform");
            qans1.add("Java is a low-level language");
            qans1.add("Java is a software");

            question1.setAnswers(qans1);

            session.persist(question1);
            tran.commit();

            session.close();
            System.out.println("success");
        }
        catch(HibernateException ex) {
            System.out.println("Error :" + ex.getMessage());
        }
    }
}
```

Step-7 Right click on App.java file and Run the application