

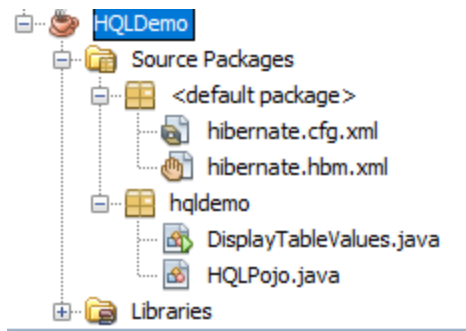
## Ex 12: Implement HQL to list all the records from the table

### Step 1:

Use the same table created for Ex 11

At least one record must be available.

### Step 2:



### HQLPojo.java

```
package hqldemo;

public class HQLPojo {

    public int id;

    public String firstName;

    public String lastName;

    public int salary;

    public int getId() {

        return id;

    }

    public void setId(int id) {

        this.id = id;

    }

    public String getFirstName() {

        return firstName;

    }

    public void setFirstName(String firstName) {
```

```

        this.firstName = firstName;
    }

    public String getLastName() {
        return lastName;
    }

    public void setLastName(String lastName) {
        this.lastName = lastName;
    }

    public int getSalary() {
        return salary;
    }

    public void setSalary(int salary) {
        this.salary = salary;
    }
}

```

### Step 3:

**Create hibernate.cfg.xml file**

### Step 4:

Create hibernate.hbm.xml file and map the variable objects with table column name.

```

<hibernate-mapping>
    <class name="hqldemo.HQLPojo" table="employee">
        <id column="id" name="id" type="int"></id>
        <property column="first_name" name="firstName" />
        <property column="last_name" name="lastName" />
        <property column="salary" name="salary" />
    </class>
</hibernate-mapping>

```

**Step 5:**

DisplayTableValues.java

```
package hqldemo;

// Import required Hibernate classes
import java.util.List;
import org.hibernate.Session;
import org.hibernate.SessionFactory;
import org.hibernate.cfg.Configuration;
import org.hibernate.Query;

public class DisplayTableValues {

    public static void main(String[] args) {

        // Create a Hibernate configuration object
        Configuration cfg = new Configuration();
        cfg.configure("hibernate.cfg.xml");

        // Create a Hibernate session factory object
        SessionFactory factory = cfg.buildSessionFactory();

        // Open a Hibernate session
        Session session = factory.openSession();

        // Define an HQL query to retrieve all records from example_table
        String hql = "FROM HQLPojo";

        // Create the HQL query object
        Query query = session.createQuery(hql);

        // Execute the query and retrieve the results
```

```

List<HQLPojo> results = query.list();

// Print out the table values with columns(id, name city)
for (HQLPojo example : results) {

    System.out.println("ID: " + example.getId() +

        ", First Name: " + example.getFirstName() +

        ", Last Name: " + example.getLastName() +

        ", Salary: " + example.getSalary());

}

// Close the Hibernate session
session.close();

// Close the Hibernate session factory
factory.close();

}
}

```

## Output:

```

INFO: HHH000039: Using default transaction strategy (direct JDBC transactions)
Apr 02, 2023 10:33:07 PM org.hibernate.hql.internal.ast.ASTQueryTranslatorFactory <init>
INFO: HHH000397: Using ASTQueryTranslatorFactory
ID: 101, First Name: Abc, Last Name: Xyz, Salary: 12345
ID: 111, First Name: RK, Last Name: Keynotes, Salary: 1000
ID: 113, First Name: Aarav, Last Name: Natarajan, Salary: 10000
Apr 02, 2023 10:33:09 PM org.hibernate.engine.jdbc.connections.internal.DriverManagerConnectionProviderImpl stop
INFO: HHH000030: Cleaning up connection pool [jdbc:mysql://localhost:3306/retailer?zeroDateTimeBehavior=convertToNull]
BUILD SUCCESSFUL (total time: 6 seconds)

```

Activate Wir