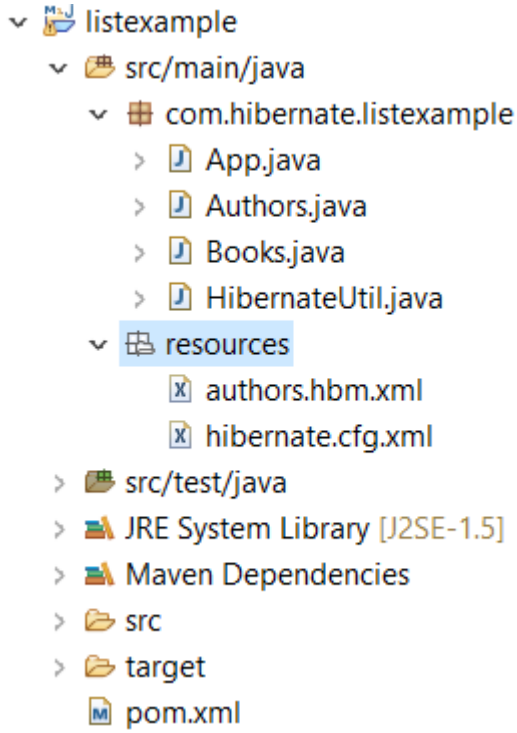
Project Structure



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: listexample

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 Book Class

1. Create the Authors class in the com.hibernate.listexample package.
2. Create the Books class in the com.hibernate.listexample package.

Class : Author

```
package com.hibernate.listexample;
import java.util.List;

import jakarta.persistence.Column;
import jakarta.persistence.Entity;
import jakarta.persistence.Table;

@Entity
@Table(name="Authors")
```

```
public class Authors {
	@Column
	private int authorID;

	@Column
	private String authorName;

	private List<String> books;

	public void setAuthorID(int authorID) { this.authorID = authorID; }
	public int getAuthorID() { return authorID; }

	public void setAuthorName(String authorName) { this.authorName = authorName; }
	public String getAuthorName() { return authorName; }

	public List<String> getBooks() { return books;}
	public void setBooks(List<String> books) { this.books = books; }
}
```

Class : Books

```
package com.hibernate.listexample;

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

@Entity
@Table(name = "Books")
public class Books {
	@Id
	@GeneratedValue
	private int bookID;

	@Column
	private String bookName;

	@Column
	private int authorID;

	public int getAuthorID() { return authorID; }
	public void setAuthorID(int authorID) { this.authorID = authorID; }
```

```
        public void setBookID(int bookID) { this.bookID = bookID; }
        public int getBookID() { return bookID; }

        public void setBookName(String bookName) { this.bookName = bookName; }
        public String getBookName() { return bookName; }

}
```

Step 5: Create the authors.hbm.xml File

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

```
<?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.listexample.Authors"
          table="AUTHORS">
      <id name="authorID" type="int" column="authorID">
         <generator class="assigned"/>
      </id>
      <property name="authorName" type="string">
         <column name="authorName"/>
      </property>

     <list name="books" table="BOOKS">
          <key column="authorID"></key>
          <index column="bookID"></index>
          <element column="bookName" type="string"></element>
    </list>
</class>
</hibernate-mapping>
```

Step 5: Create the hibernate.cfg.xml File

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

```
<?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/authors.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.listexample package:

```
package com.hibernate.listexample;
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.

```
package com.hibernate.listexample;

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

import java.util.*;

public class App
{
   public static void main( String[] args )
   {
        Session session = HibernateUtil.getSessionFactory().openSession();
        Transaction tran = null;
        try {
            tran = session.beginTransaction();

            Authors author1 = new Authors();
            author1.setAuthorID(2);
            author1.setAuthorName("Gotfield");

            ArrayList<String> bks = new ArrayList<String>();
            bks.add("Programming in C");
            bks.add("Programming in C++");

            author1.setBooks(bks);

            session.persist(author1);
            tran.commit();

            session.close();
        }
        catch (Exception e) {
            System.out.println(e.getMessage());
        }
   }
}
```

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