

SimpliLearn :

Practise Project : 3

- Create a servlet-based web application that shows a form to add new products. A MySQL database will be created to store product data. The form data will be validated, and a row will be added to the database. All database processing will be done using Hibernate.

Source Code:

MainProduct.java

```
package com;

import org.hibernate.Session;
import org.hibernate.SessionFactory;
import org.hibernate.Transaction;
import org.hibernate.boot.registry.StandardServiceRegistryBuilder;
import org.hibernate.cfg.Configuration;

public class MainProduct {

    public static void main(String[] args) {
        Configuration configuration=new Configuration().configure();
        configuration.addAnnotatedClass(DataBaseProducts.class);
        StandardServiceRegistryBuilder builder=new
StandardServiceRegistryBuilder().applySettings(configuration.getProperties ());
        SessionFactory factory= configuration.buildSessionFactory(builder.build());
        Session session=factory.openSession();
        Transaction transaction=session.beginTransaction();
        DataBaseProducts p1=new DataBaseProducts("Asus-Laptop",101,120000);
        session.save(p1);
        System.out.println("deatils"+p1);
        transaction.commit();
        session.close();
    }
}
```

```
    }  
}
```

DataBaseProduct.java

```
package com;
```

```
import javax.persistence.Entity;  
import javax.persistence.Id;  
import javax.persistence.Table;
```

```
@Table
```

```
@Entity
```

```
public class DataBaseProducts {  
    @Id  
    private String ProductName;  
    private int ProductId;  
    private int productPrice;
```

```
public DataBaseProducts(String productName, int productId, int productPrice) {  
    super();  
    ProductName = productName;  
    ProductId = productId;  
    this.productPrice = productPrice;  
}
```

```
@Override
```

```
public String toString() {  
    return "DataBaseProducts [ProductName=" + ProductName + ", ProductId=" + ProductId  
+ ", productPrice=" + productPrice + "];"  
}
```

```
public String getProductName() {  
    return ProductName;  
}
```

```
public void setProductName(String productName) {  
    ProductName = productName;  
}
```

```
public int getProductId() {  
    return ProductId;  
}
```

```

public void setProductId(int productId) {
    ProductId = productId;
}
public int getProductPrice() {
    return productPrice;
}
public void setProductPrice(int productPrice) {
    this.productPrice = productPrice;
}

}

```

hibernate.cfg.xml

```

<?xml version='1.0' encoding='utf-8'?>
<!DOCTYPE hibernate-configuration PUBLIC
    "-//Hibernate/Hibernate Configuration DTD 3.0//EN"
    "http://www.hibernate.org/dtd/hibernate-configuration-3.0.dtd">
<hibernate-configuration>
    <session-factory>
        <!-- Database connection settings -->
        <property
name="connection.driver_class">oracle.jdbc.driver.OracleDriver</property>
        <property
name="connection.url">jdbc:mysql://localhost:3306/addproduct:xe</property>
        <property name="connection.username">root</property>
        <property name="connection.password">ramani44</property>
        <!-- JDBC connection pool (use the built-in) -->
        <property name="connection.pool_size">1</property>
        <!-- SQL dialect -->
        <property
name="dialect">org.hibernate.dialect.Oracle10gDialect</property>
        <!-- Disable the second-level cache -->
        <property
name="cache.provider_class">org.hibernate.cache.NoCacheProvider</property>
        <!-- Echo all executed SQL to stdout -->
        <property name="show"></property>

    <!-- Echo all executed SQL to stdout -->

```

```

<property name="show_sql">true</property>
<!-- Drop and re-create the database schema on startup -->
<property name="hbm2ddl.auto">create</property>
<!-- Names the annotated entity class -->
<mapping class="com.DataBaseProducts" />
</session-factory>
</hibernate-configuration>

```

OutPut

The screenshot shows the Eclipse IDE with the Package Explorer on the left, the Editor in the center, and the Console on the right. The Package Explorer shows a project named 'ADDProduct' with a source folder 'src' containing a package 'com'. Inside the 'com' package, there is a file 'hibernate.cfg.xml' which is currently open in the Editor. The Editor shows the XML configuration for Hibernate, which is the same as the one provided in the first block. The Console on the right shows the output of the application, which is a log of Hibernate startup messages. The messages include the Hibernate version (5.6.5.Final), the connection pool configuration (using the built-in pool), the driver (Oracle JDBC driver), the connection URL (jdbc:oracle:thin:@localhost:1521:xe), the connection properties (password=****, user=skv), the connection pool size (1), the dialect (org.hibernate.dialect.Oracle10gDialect), and the creation of the 'DataBaseProducts' table. The console output ends with the message 'Hibernate: insert into DataBaseProducts (ProductId, productPrice, ProductName) values (?, ?, ?)'.

```

eclipse-workspace - ADDProduct/src/hibernate.cfg.xml - Eclipse IDE
File Edit Source Navigate Search Project Run Window Help

Package Explorer | h... | Console
ADDProduct [eclipse-workspace master]
  JRE System Library [JavaSE-16]
  src
    com
      DataBaseProducts.java
      MainProduct.java
      hibernate.cfg.xml
  Referenced Libraries
  CookiesSessTracking [eclipse-workspace master]
  FirstServlet [eclipse-workspace master]
  GenericServlet [eclipse-workspace master]
  Get_Post [eclipse-workspace master]
  Hibernate config [eclipse-workspace master]
  Hibernate demo [eclipse-workspace master]
  JDBC_Insertion_upadation_deletion [eclipse-workspace master]
  JDBC_Select_Drop [eclipse-workspace master]
  JDBC Connection [eclipse-workspace master]
  JDBCSetup [eclipse-workspace master]
  JDBCStoreProcedure [eclipse-workspace master]
  JSPActionTag [eclipse-workspace master]
  JSPDirectives [eclipse-workspace master]
  JSPImplicitObject [eclipse-workspace master]
  JSPProject [eclipse-workspace master]
  JSPSession [eclipse-workspace master]
  Login_Logout [eclipse-workspace master]
  Product [eclipse-workspace master]
  Servers [eclipse-workspace master]
  ServletClassInterface [eclipse-workspace master]
  ServletFilter [eclipse-workspace master]
  SessionTracking_http [eclipse-workspace master]
  SessionTrackingHiddenForm [eclipse-workspace master]
  SessionTrackingURL [eclipse-workspace master]
  ValidatedUserLogin [eclipse-workspace master]

1 <?xml ver
2 <!DOCTYPE
3
4
5 <hibernat
6 <sess
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21 <!-- Echo
22
23
24
25
26
27 </ses
28 </hiberna

<terminated> MainProduct [Java Application] C:\Program Files\Java\jdk-16.0.2\bin\javaw.exe (Feb 25, 2022, 11:36:43 AM - 11:36:47 AM)
Feb 25, 2022 11:36:44 AM org.hibernate.Version logVersion
INFO: HHH000412: Hibernate ORM core version 5.6.5.Final
Feb 25, 2022 11:36:44 AM org.hibernate.annotations.common.reflection.java.JavaReflectionManager <clinit>
INFO: HCAN000001: Hibernate Commons Annotations {5.1.2.Final}
Feb 25, 2022 11:36:45 AM org.hibernate.engine.jdbc.connections.internal.DriverManagerConnectionProviderImpl configure
WARN: HHH10001002: Using Hibernate built-in connection pool (not for production use!)
Feb 25, 2022 11:36:45 AM org.hibernate.engine.jdbc.connections.internal.DriverManagerConnectionProviderImpl buildCreator
INFO: HHH10001005: using driver [oracle.jdbc.driver.OracleDriver] at URL [jdbc:oracle:thin:@localhost:1521:xe]
Feb 25, 2022 11:36:45 AM org.hibernate.engine.jdbc.connections.internal.DriverManagerConnectionProviderImpl buildCreator
INFO: HHH10001001: Connection properties: {password=****, user=skv}
Feb 25, 2022 11:36:45 AM org.hibernate.engine.jdbc.connections.internal.DriverManagerConnectionProviderImpl buildCreator
INFO: HHH10001003: Autocommit mode: false
Feb 25, 2022 11:36:45 AM org.hibernate.engine.jdbc.connections.internal.DriverManagerConnectionProviderImpl$PooledConnections <init>
INFO: HHH000115: Hibernate connection pool size: 1 (min=1)
Feb 25, 2022 11:36:45 AM org.hibernate.dialect.Dialect <init>
INFO: HHH000400: Using dialect: org.hibernate.dialect.Oracle10gDialect
Feb 25, 2022 11:36:45 AM org.hibernate.tuple.PojoInstantiator <init>
INFO: HHH000182: No default (no-argument) constructor for class: com.DataBaseProducts (class must be instantiated by Interceptor)
Hibernate: drop table DataBaseProducts cascade constraints
Feb 25, 2022 11:36:46 AM org.hibernate.resource.transaction.backend.jdbc.internal.DdlTransactionIsolatorNonJtaImpl getIsolatedConnection
INFO: HHH10001501: Connection obtained from JdbcConnectionAccess [org.hibernate.engine.jdbc.env.internal.JdbcEnvironmentInitiator$ConnectionProviderJdbcConne
Hibernate: create table DataBaseProducts (ProductName varchar2(255 char) not null, ProductId number(10,0) not null, productPrice number(10,0) not null, prima
Feb 25, 2022 11:36:47 AM org.hibernate.resource.transaction.backend.jdbc.internal.DdlTransactionIsolatorNonJtaImpl getIsolatedConnection
INFO: HHH10001501: Connection obtained from JdbcConnectionAccess [org.hibernate.engine.jdbc.env.internal.JdbcEnvironmentInitiator$ConnectionProviderJdbcConne
Feb 25, 2022 11:36:47 AM org.hibernate.engine.transaction.jta.platform.internal.JtaPlatformInitiator initiateService
INFO: HHH000490: Using JtaPlatform implementation: [org.hibernate.engine.transaction.jta.platform.internal.NoJtaPlatform]
deatilsDataBaseProducts [ProductName=Asus-Laptop, ProductId=101, productPrice=120000]
Hibernate: insert into DataBaseProducts (ProductId, productPrice, ProductName) values (?, ?, ?)

```