E Product:

package com.ecommerce;

import java.math.BigDecimal;

import java.util.Date;

public class EProduct {

private long ID;

private String name;

private BigDecimal price;

private Date dateAdded;

public EProduct() {

}

public long getID() {

return ID;

}

public void setID(long iD) {

ID = iD;

}

public String getName() {

return name;

}

public void setName(String name) {

this.name = name;

}

public BigDecimal getPrice() {

return price;

}

public void setPrice(BigDecimal price) {

this.price = price;

}

public Date getDateAdded() {

return dateAdded;

}

public void setDateAdded(Date dateAdded) {

this.dateAdded = dateAdded;

}

}

Hibernate.util

package com.ecommerce;

import java.math.BigDecimal;

import java.util.Date;

public class EProduct {

private long ID;

private String name;

private BigDecimal price;

private Date dateAdded;

public EProduct() {

}

public long getID() {

return ID;

}

public void setID(long iD) {

ID = iD;

}

public String getName() {

return name;

}

public void setName(String name) {

this.name = name;

}

public BigDecimal getPrice() {

return price;

}

public void setPrice(BigDecimal price) {

this.price = price;

}

public Date getDateAdded() {

return dateAdded;

}

public void setDateAdded(Date dateAdded) {

this.dateAdded = dateAdded;

}

}

Web xml

Hibernate\_Demo\_Project

index.html

index.htm

index.jsp

default.html

default.htm

default.jsp

HibernateQueryDemo

HibernateQueryDemo

com.simply.HibernateQueryDemo

HibernateQueryDemo

/HibernateQueryDemo

Hibernate query demo:

String name = request.getParameter("name");

double price = Double.parseDouble(request.getParameter("price"));

// STEP 1: Get a Session (connection) from the Session Factory class

SessionFactory factory = HibernateUtil.getSessionFactory();

Transaction transaction = null; Session session = factory.openSession(); out.println("Hibernate Session opened.");

// STEP 2 execute the HQL commands // for now we will only test if the connection is establised with MySQL

server. try {

transaction = session.beginTransaction();

//session.save(product);

String hql = "INSERT INTO EProduct (name,price) VALUES (:name,:price)";

Query query = session.createQuery(hql);

// Set query parameters query.setParameter("name",name);

query.setParameter("price", price);

// Execute the query and get the number of affected rows int rowCount = query.executeUpdate();

System.out.println("Rows affected: " + rowCount); transaction.commit(); System.out.println("row added"); response.sendRedirect("success.jsp");

}

catch (Exception e) {

if (transaction != null) {

transaction.rollback(); }

e.printStackTrace(); }

finally { session.close();

System.out.println("Session Closed"); }

out.println("Hibernate Session closed.");

out.println("”);

Eproduct.hbm:

hibernate-mapping package="com.ecommerce">

<class name="EProduct" table="eproduct">

<id name="ID" column="ID">

<generator class="increment"/>

</id>

<property name="name" type="string" column="NAME"/>

<property name="price" type="big\_decimal" column="PRICE"/>

<property name="dateAdded" type="timestamp" column="DATE\_ADDED"/>

</class>

</hibernate-mapping>

Hibernate cfg:

<hibernate-configuration>

<session-factory>

<!-- Database connection settings -->

<property name="connection.driver\_class">com.mysql.cj.jdbc.Driver</property>

<property name="connection.url">jdbc:mysql://127.0.0.1:3306/ecommerce</property>

<property name="connection.username">root</property>

<property name="connection.password">@Classicaldancer23</property>

<mapping resource="com/ecommerce/EProduct.hbm.xml"/>

</session-factory>

</hibernate-configuration>