# 7) Demonstrate Component mapping in Hibernate

package com.simpli;

import java.io.\*;

import java.io.PrintWriter;

import java.util.List;

import javax.servlet.ServletException;

import javax.servlet.annotation.WebServlet;

import javax.servlet.http.\*;

import org.hibernate.\*;

import com.ecommerce.EProduct;

import com.ecommerce.ProductParts;

@WebServlet("/component\_mapping-demo")

public class ComponentMappingDemo extends HttpServlet {

protected void doGet(HttpServletRequest request, HttpServletResponse response)

throws ServletException, IOException {

PrintWriter out = response.getWriter();

out.println("<html><body>");

SessionFactory factory = HibernateUtil.getSessionFactory();

Session session = factory.openSession();

// STEP 3 Query the DB and get the data

List<EProduct> eproducts = session.createQuery("from EProduct").list();

out.println("<table border=1>");

for (EProduct prod : eproducts) {

out.println("<tr>" + "<td>" + prod.getID() + "<td>" + prod.getName() + "<td>" + prod.getPrice() + "<td>"

+ prod.getDateAdded());

ProductParts parts = prod.getParts();

out.println("<td> Parts =" + parts.getCpu() + ", " + parts.getHdd() + ", " + parts.getRam());

}

out.println("</table>");

session.close();

out.println("</body></html>");

}

}

package com.simpli;

import java.io.\*;

import java.util.\*;

import javax.servlet.ServletException;

import javax.servlet.annotation.WebServlet;

import javax.servlet.http.\*;

import org.hibernate.\*;

import com.ecommerce.\*;

@WebServlet("/product-details")

public class ProductDetailsServlet extends HttpServlet {

protected void doGet(HttpServletRequest request, HttpServletResponse response)

throws ServletException, IOException {

PrintWriter out = response.getWriter();

out.println("<html><body>");

SessionFactory factory = HibernateUtil.getSessionFactory();

Session session = factory.openSession();

// STEP 3 Query the DB and get the data

List<EProduct> eproducts = session.createQuery("from EProduct").list();

out.println(

"<table border=1><th>ID<th>NAME<th>PRICE<th>DATE\_ADDED<th>COLORS<th>OS<th>SCREENSIZES<th>FINANCE</th>");

for (EProduct prod : eproducts) {

out.println("<tr>" + "<td>" + prod.getID() + "<td>" + prod.getName() + "<td>" + prod.getPrice() + "<td>"

+ prod.getDateAdded());

// Display the colors the product is available in

List<Color> colors = prod.getColors();

out.println("<td>");

for (Color color : colors)

out.println(color.getName());

// Display the OS the product is available in

Set<OS> Oses = prod.getOs();

out.println("<td>");

for (OS os : Oses)

out.println(os.getName());

// Display the Screen Sizes the product is available in

Collection<ScreenSizes> screenSizes = prod.getScreenSizes();

out.println("<td>");

for (ScreenSizes ss : screenSizes)

out.println(ss.getSize());

// Map finances = prod.getFinance();

// out.println("<td>");

// if (finances.get("CREDITCARD") != null) {

// Finance f = (Finance) finances .get("CREDITCARD");

// out.println(f.getName() + " &nbsp;");

// }

// if (finances .get("BANK") != null) {

// Finance f = (Finance) finances .get("BANK");

// out.println(f.getName() + " &nbsp;");

// }

}

out.println("</table>");

session.close();

out.println("</body></html>");

}

}

EProduct.hbm.xml

<?xml version="1.0"?>

<!DOCTYPE hibernate-mapping PUBLIC

"-//Hibernate/Hibernate Mapping DTD 3.0//EN"

"http://www.hibernate.org/dtd/hibernate-mapping-3.0.dtd">

<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" />

<component name="parts" class="com.ecommerce.ProductParts">

<property name="hdd" column="parts\_hdd" type="string" />

<property name="cpu" column="parts\_cpu" type="string" />

<property name="ram" column="parts\_ram" type="string" />

</component>

<list name="colors" cascade="all">

<key column="product\_id" />

<list-index column="idx" />

<one-to-many class="com.ecommerce.Color" />

</list>

<set name="os" cascade="all">

<key column="product\_id" />

<one-to-many class="com.ecommerce.OS" />

</set>

<bag name="screenSizes" cascade="all">

<key column="product\_id"></key>

<one-to-many class="com.ecommerce.ScreenSizes" />

</bag>

<map name = "finance" cascade="all">

<key column = "product\_id"/>

<index column = "ftype" type = "string"/>

<one-to-many class="com.ecommerce.Finance"/>

</map>

</class>

</hibernate-mapping>

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">com.mysql.cj.jdbc.Driver</property>

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

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

<property name="connection.password">Simplilearn</property>

<property name="dialect">org.hibernate.dialect.MySQL8Dialect</property>

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

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

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

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

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

</session-factory>

</hibernate-configuration>

HTML Code:

<br><br>

<h3>Hibernate Component Mapping Demo </h3>

<a href="component\_mapping-demo">Click here</a><br>

OUTPUT:

1 HP Laptop ABC 21900.00 2019-06-04 07:18:57.0 Parts =AMD Phenom, 2 Gb HDD, 4 Gb

2 Acer Laptop ABC 23300.00 2019-06-04 07:19:07.0 Parts =Core-i7, 500 Gb HDD, 4 Gb

3 Lenovo Laptop ABC 33322.00 2019-06-04 07:19:19.0 Parts =Core-i7, 1 Tb HDD, 8 Gb