Assisted project6: Demonstrate lazy collection in Hibernate.

Source Code:

```
DROP TABLE IF EXISTS `colors`;
/*!40101 SET @saved_cs_client     = @@character_set_client */;
/*!40101 SET character_set_client = utf8 */;
CREATE TABLE `colors` (
  `ID` bigint(20) NOT NULL AUTO_INCREMENT,
  `color_name` varchar(40) DEFAULT NULL,
  `idx` int(11) DEFAULT NULL,
  `product_id` bigint(20) DEFAULT NULL,
  PRIMARY KEY (`ID`)
) ENGINE=InnoDB AUTO_INCREMENT=6 DEFAULT CHARSET=latin1;
/*!40101 SET character_set_client = @saved_cs_client */;


LOCK TABLES `colors` WRITE;
/*!40000 ALTER TABLE `colors` DISABLE KEYS */;
INSERT INTO `colors` VALUES
(1,'Red',0,1),(2,'Silver',1,1),(3,'Gray',0,2),(4,'White',1,2),(5,'Maroon',0,3);•/*!40000 ALTER
TABLE `colors` ENABLE KEYS */;
UNLOCK TABLES;

--
-- Table structure for table `eproduct`
--

DROP TABLE IF EXISTS `eproduct`;
/*!40101 SET @saved_cs_client     = @@character_set_client */;
/*!40101 SET character_set_client = utf8 */;
CREATE TABLE `eproduct` (
  `ID` bigint(20) NOT NULL AUTO_INCREMENT,
  `name` varchar(100) DEFAULT NULL,
  `price` decimal(10,2) DEFAULT NULL,
  `date_added` timestamp NOT NULL DEFAULT CURRENT_TIMESTAMP,
  PRIMARY KEY (`ID`)
) ENGINE=InnoDB AUTO_INCREMENT=4 DEFAULT CHARSET=latin1;
/*!40101 SET character_set_client = @saved_cs_client */;


LOCK TABLES `eproduct` WRITE;
/*!40000 ALTER TABLE `eproduct` DISABLE KEYS */;
```

```
INSERT INTO `eproduct` VALUES (1,'HP Laptop ABC',21900.00,'2019-06-04
07:18:57'),(2,'Acer Laptop ABC',23300.00,'2019-06-04 07:19:07'),(3,'Lenovo Laptop
ABC',33322.00,'2019-06-04 07:19:19');
/*!40000 ALTER TABLE `eproduct` ENABLE KEYS */;
UNLOCK TABLES;


DROP TABLE IF EXISTS `finance`;
/*!40101 SET @saved_cs_client     = @@character_set_client */;
/*!40101 SET character_set_client = utf8 */;
CREATE TABLE `finance` (
  `ID` bigint(20) NOT NULL AUTO_INCREMENT,
  `ftype` varchar(10) DEFAULT NULL,
  `name` varchar(30) DEFAULT NULL,
  `product_id` bigint(20) DEFAULT NULL,
  PRIMARY KEY (`ID`)
) ENGINE=InnoDB AUTO_INCREMENT=6 DEFAULT CHARSET=latin1;
/*!40101 SET character_set_client = @saved_cs_client */;



LOCK TABLES `finance` WRITE;
/*!40000 ALTER TABLE `finance` DISABLE KEYS */;
INSERT INTO `finance` VALUES (1,'CREDITCARD','EMI on Citibank Card',1),(3,'BANK','40%
finance from SBI',2),(4,'BANK','60% finance from ICICI',3),(5,'BANK','20% finance from ICICI',1);
/*!40000 ALTER TABLE `finance` ENABLE KEYS */;
UNLOCK TABLES;


DROP TABLE IF EXISTS `os`;
/*!40101 SET @saved_cs_client     = @@character_set_client */;
/*!40101 SET character_set_client = utf8 */;
CREATE TABLE `os` (
  `ID` bigint(20) NOT NULL AUTO_INCREMENT,
  `name` varchar(30) DEFAULT NULL,
  `product_id` bigint(20) DEFAULT NULL,
  PRIMARY KEY (`ID`)
) ENGINE=InnoDB AUTO_INCREMENT=6 DEFAULT CHARSET=latin1;
/*!40101 SET character_set_client = @saved_cs_client */;



LOCK TABLES `os` WRITE;
/*!40000 ALTER TABLE `os` DISABLE KEYS */;
```

```
INSERT INTO `os` VALUES (1,'Windows 10',1),(2,'Windows 10',2),(3,'FreeDOS',2),(4,'RedHat
Linux',2),(5,'Windows 10',3);
/*!40000 ALTER TABLE `os` ENABLE KEYS */;
UNLOCK TABLES;



DROP TABLE IF EXISTS `screensizes`;
/*!40101 SET @saved_cs_client     = @@character_set_client */;
/*!40101 SET character_set_client = utf8 */;
CREATE TABLE `screensizes` (
  `ID` bigint(20) NOT NULL AUTO_INCREMENT,
  `size` varchar(10) DEFAULT NULL,
  `product_id` bigint(20) DEFAULT NULL,
  PRIMARY KEY (`ID`)
) ENGINE=InnoDB AUTO_INCREMENT=5 DEFAULT CHARSET=latin1;
/*!40101 SET character_set_client = @saved_cs_client */;



LOCK TABLES `screensizes` WRITE;
/*!40000 ALTER TABLE `screensizes` DISABLE KEYS */;
INSERT INTO `screensizes` VALUES (1,'12 in',1),(2,'14.5 in',2),(3,'14.9 in',2),(4,'15.5 in',3);
/*!40000 ALTER TABLE `screensizes` ENABLE KEYS */;
UNLOCK TABLES;
```

```
package com.ecommerce;


public class Color {
    private long COLORID;
    private String name;

    public Color() {

    }
    public Color(String name) {
        this.COLORID = 0;
        this.name = name;
    }

    public long getCOLORID() {return this.COLORID; }
    public String getName() { return this.name;}
    public void setCOLORID(long id) { this.COLORID = id;}
    public void setName(String name) { this.name = name;}


}



package com.ecommerce;

import java.math.BigDecimal;
import java.util.Collection;
import java.util.Date;
import java.util.List;
import java.util.Set;
import java.util.Map;


public class EProduct {
    private long ID;
    private String name;
    private BigDecimal price;
    private Date dateAdded;
    private List<Color> colors;
    private Set<Finance> finance;
    private PDescription pdescrip;

    public EProduct() {

    }

    public long getID() {return this.ID; }
    public String getName() { return this.name;}
    public BigDecimal getPrice() { return this.price;}
    public Date getDateAdded() { return this.dateAdded;}
    public List<Color> getColors() { return this.colors;}
    public Set<Finance> getFinance() { return this.finance;}
    public PDescription getPdescrip() { return this.pdescrip;}

    public void setID(long id) { this.ID = id;}
    public void setName(String name) { this.name = name;}
    public void setPrice(BigDecimal price) { this.price = price;}
    public void setDateAdded(Date date) { this.dateAdded = date;}
    public void setColors(List<Color> colors) { this.colors = colors;}
    public void setFinance(Set<Finance> finance) { this.finance = finance;}
    public void setPdescrip(PDescription pdescrip) { this.pdescrip = pdescrip;}
}



package com.ecommerce;


public class Finance {
    private long FINANCEID;
    private String name;
    private String ftype;

    public Finance() {

    }
    public Finance(String name, String ftype) {
        this.FINANCEID = 0;
        this.name = name;
        this.ftype = ftype;
    }

    public long getFINANCEID() {return this.FINANCEID; }
    public String getName() { return this.name;}
    public String getFtype() { return this.ftype;}
    public void setFINANCEID(long id) { this.FINANCEID = id;}
    public void setName(String name) { this.name = name;}
    public void setFtype(String ftype) { this.ftype= ftype;}


}



package com.ecommerce;


public class OS {

    private long OSID;
    private String name;

    public OS() {
	}
	public OS(String name) {
		this.OSID = 0;
		this.name = name;
	}

	public long getOSID() {return this.OSID; }
	public String getName() { return this.name;}
	public void setOSID(long id) { this.OSID = id;}
	public void setName(String name) { this.name= name;}


}



package com.ecommerce;


public class ScreenSizes {
	private long SCREENID;
	private String size;

	public ScreenSizes() {

	}
	public ScreenSizes(String size) {
		this.SCREENID = 0;
		this.size = size;
	}

	public long getSCREENID() {return this.SCREENID; }
	public String getSize() { return this.size;}
	public void setSCREENID(long id) { this.SCREENID = id;}
	public void setSize(String size) { this.size = size;}


}



package com.ecommerce;
```

```
import org.hibernate.SessionFactory;
import org.hibernate.boot.Metadata;
import org.hibernate.boot.MetadataSources;
import org.hibernate.boot.registry.StandardServiceRegistry;
import org.hibernate.boot.registry.StandardServiceRegistryBuilder;

public class HibernateUtil {

    private static final SessionFactory sessionFactory;

    static {
        try {
            StandardServiceRegistry standardRegistry = new
StandardServiceRegistryBuilder()
                    .configure("hibernate.cfg.xml").build();
            Metadata metaData = new
MetadataSources(standardRegistry).getMetadataBuilder().build();
            sessionFactory = metaData.getSessionFactoryBuilder().build();
        } catch (Throwable th) {
            throw new ExceptionInInitializerError(th);
        }
    }

    public static SessionFactory getSessionFactory() {
        return sessionFactory;
    }
}
```

```
<?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="Color" table="colors">
       <id name="COLORID" type="long" column="ID">
           <generator class="identity"/>
       </id>
       <property name="name" type="string" column="COLOR_NAME"/>
   </class>
</hibernate-mapping>
```

```
<?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" type="long" column="ID">
      <generator class="identity"/>
    </id>
    <property name="name" type="string" column="NAME"/>
    <property name="price" type="big_decimal" column="PRICE"/>
    <property name="dateAdded" type="timestamp" column="DATE_ADDED"/>

          <list name="colors" cascade="all" lazy="true">
               <key column="product_id" />
               <list-index column="idx" />
               <one-to-many class="com.ecommerce.Color" />
          </list>

          <bag name="screensizes" cascade="all" lazy="true">
          <key column="product_id"></key>
           <one-to-many class="com.ecommerce.ScreenSizes"/>
          </bag>

          <set name = "os" cascade="all" lazy=”true”>
     <key column = "product_id"/>
     <one-to-many class="OS"/>
   </set>

   <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>


<?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="Finance" table="finance">
    <id name="FINANCEID" type="long" column="ID">
```

```
      <generator class="identity"/>
    </id>
    <property name="name" type="string" column="NAME"/>
    <property name="ftype" type="string" column="FTYPE"/>
  </class>
</hibernate-mapping>


<?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="OS" table="os">
    <id name="OSID" type="long" column="ID">
      <generator class="identity"/>
    </id>
    <property name="name" type="string" column="NAME"/>
  </class>
</hibernate-mapping>


<?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="ScreenSizes" table="screensizes">
    <id name="SCREENID" type="long" column="ID">
      <generator class="identity"/>
    </id>
    <property name="size" type="string" column="SIZE"/>
  </class>
</hibernate-mapping>

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

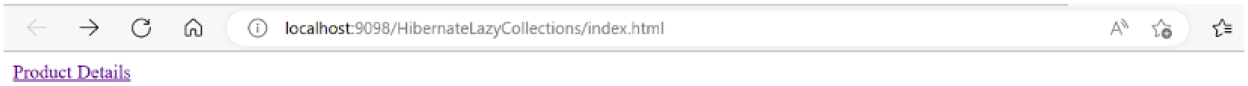
```
   <property name="connection.username">root</property>
   <property name="connection.password">vamsi</property>
   <mapping resource="com/ecommerce/EProduct.hbm.xml"/>
   <mapping resource="com/ecommerce/Color.hbm.xml"/>
   <mapping resource="com/ecommerce/ScreenSizes.hbm.xml"/>
   <mapping resource="com/ecommerce/Os.hbm.xml"/>
   <mapping resource="com/ecommerce/Finance.hbm.xml"/>
 </session-factory>
</hibernate-configuration>
```

```
<!DOCTYPE html>
<html>
<head>
<meta charset="UTF-8">
<title>Hibernate Collection Mapping With XML</title>
</head>
<body>
<a href="details">Product Details</a><br>

</body>
</html>
```

```
import java.io.IOException;
import java.io.PrintWriter;
import javax.servlet.ServletConfig;
import javax.servlet.ServletException;
import javax.servlet.annotation.WebServlet;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;
import javax.transaction.*;
import javax.xml.bind.*;
import java.io.Serializable;
import java.math.BigDecimal;
import java.util.ArrayList;
import java.util.Calendar;
import java.util.Collection;
import java.util.List;
import java.util.Map;
import java.util.Set;

import org.hibernate.Session;
import org.hibernate.SessionFactory;
import org.hibernate.Transaction;
import org.hibernate.cfg.Configuration;

import com.ecommerce.Color;
import com.ecommerce.EProduct;
import com.ecommerce.Finance;
import com.ecommerce.HibernateUtil;
import com.ecommerce.PDescription;

/**
* Servlet implementation class ProductDetails
*/
@WebServlet("/ProductDetails")
public class ProductDetails extends HttpServlet {
      private static final long serialVersionUID = 1L;
   /**
* @see HttpServlet#HttpServlet()
*/
   public ProductDetails() {
      super();
```

```
    // TODO Auto-generated constructor stub
  }

    /**
     * @see HttpServlet#doGet(HttpServletRequest request, HttpServletResponse response)
     */
    protected void doGet(HttpServletRequest request, HttpServletResponse response) throws
ServletException, IOException {
        // TODO Auto-generated method stub
         try {
             SessionFactory factory = HibernateUtil.getSessionFactory();
             Session session = factory.openSession();
             List<EProduct> list = session.createQuery("from EProduct").list();

              PrintWriter out = response.getWriter();
              out.println("<html><body>");

              out.println("<b>One to One Mapping</b><br>");
              for(EProduct p: list) {
                  out.println("ID: " + String.valueOf(p.getID()) + ", Name: " + p.getName() +
                          ", Price: " + String.valueOf(p.getPrice()) + ", Date Added: " +
p.getDateAdded().toString());
                  PDescription descrip = p.getPdescrip();
                  out.println("<br>Description:" + descrip.getDescrip());
                  out.println("<hr>");
              }
              out.println("<b>One to Many and Many to One Mapping</b><br>");
              for(EProduct p: list) {
                  out.println("ID: " + String.valueOf(p.getID()) + ", Name: " + p.getName() +
                          ", Price: " + String.valueOf(p.getPrice()) + ", Date Added: " +
p.getDateAdded().toString());
                  List<Color> colors = p.getColors();
                  out.println("<br>Colors: <ul>");
                  for(Color c: colors) {
                      out.print("<li>" + c.getName() + "</li>");
                  }
                  out.println("</ul>");
                  out.println("<hr>");

              }
              out.println("<b>Many to Many Mapping</b><br>");
              for(EProduct p: list) {
                  out.println("ID: " + String.valueOf(p.getID()) + ", Name: " + p.getName() +
                                        ", Price: " + String.valueOf(p.getPrice()) + ", Date Added: " +
p.getDateAdded().toString());
                           Set<Finance> finances= p.getFinance();
                           out.println("<br>Finance Options : <ul>");
                           for(Finance f: finances) {
                                  out.print("<li>" + f.getFtype() + "</li>");
                           }
                           out.println("</ul>");
                           out.println("<hr>");
                    }

                          session.close();

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


             } catch (Exception ex) {
                    throw ex;
             }

      }

      /**
       * @see HttpServlet#doPost(HttpServletRequest request, HttpServletResponse response)
       */
      protected void doPost(HttpServletRequest request, HttpServletResponse response) throws
ServletException, IOException {
             // TODO Auto-generated method stub
             doGet(request, response);
      }

}
```

```
<?xml version="1.0" encoding="UTF-8"?>
<web-app xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns="http://xmlns.jcp.org/xml/ns/javaee"
xsi:schemaLocation="http://xmlns.jcp.org/xml/ns/javaee http://xmlns.jcp.org/xml/ns/javaee/web-
app_4_0.xsd" id="WebApp_ID" version="4.0">
 <display-name>HibernateLazyCollections</display-name>
 <welcome-file-list>
   <welcome-file>index.html</welcome-file>
   <welcome-file>index.htm</welcome-file>
   <welcome-file>index.jsp</welcome-file>
   <welcome-file>default.html</welcome-file>
   <welcome-file>default.htm</welcome-file>
   <welcome-file>default.jsp</welcome-file>
 </welcome-file-list>
 <servlet>
   <servlet-name>ProductDetails</servlet-name>
   <servlet-class>ProductDetails</servlet-class>
 </servlet>
 <servlet-mapping>
   <servlet-name>ProductDetails</servlet-name>
   <url-pattern>/details</url-pattern>
 </servlet-mapping>
</web-app>
```



localhost:9098/HibernateLazyCollections/index.html

Product Details