3. Configure Hibernate using Annotations in Eclipse IDE.

Source Code:

//index.html

```
<!DOCTYPE html>
<html>
<head>
<meta charset="UTF-8">
<title>Hibernate Example</title>
</head>
<body>
<h3>Hibernate Annotations Demo </h3>
<a href="query-demo">Query  Here and get the data</a><br>
</body>
</html>
```

//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</pro
perty>
    <property name="connection.username">root</property>
    <property name="connection.password">Simplilearn</property>
    <mapping class="com.ecommerce.EProduct" />
  </session-factory>
</hibernate-configuration>
```

//HibernateUtil.java

```
package com.simpli;
import org.hibernate.SessionFactory;
import org.hibernate.boot.*;
import org.hibernate.boot.registry.*;

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;
        }

}
```

//HibernateInit.java

```
package com.simpli;

import javax.servlet.annotation.WebServlet;
import javax.servlet.http.HttpServlet;



import java.io.IOException;
import java.io.PrintWriter;

import javax.servlet.ServletException;
import javax.servlet.annotation.WebServlet;
import javax.servlet.http.*;

import org.hibernate.*;

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

```
   /**
    * @see HttpServlet#HttpServlet()
    */
   public HibernateInit() {
      super();
      // TODO Auto-generated constructor stub
   }
   protected void doGet(HttpServletRequest request,
HttpServletResponse response)
                  throws ServletException, IOException {
            PrintWriter out = response.getWriter();
            out.println("<html><body>");

            // STEP 1: Get a Session (connection) from the Session
Factory class
            SessionFactory factory = HibernateUtil.getSessionFactory();

            // STE2 Create the session object
            Session session = factory.openSession();

            out.println("Hibernate Session opened.<br>");

            session.close();
```

```
            out.println("Hibernate Session closed.<br>");

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

}
```

//EProduct.java

```
package com.ecommerce;
import java.math.BigDecimal;
import java.util.Date;


import javax.persistence.*;

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

```
public class EProduct {

	@Id
	@GeneratedValue
	@Column(name = "ID")
	private long ID;

	@Column(name = "name")
	private String name;

	@Column(name = "price")
	private BigDecimal price;

	@Column(name = "date_added")
	private Date dateAdded;




	public EProduct() {

	}
```

```
	public EProduct(long id, String name, BigDecimal price, Date
dateAdded) {
		this.ID = id;
		this.name = name;
		this.price = price;
		this.dateAdded = dateAdded;
	}

	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;
	}

}
```

//HibernateQueryDemo.java

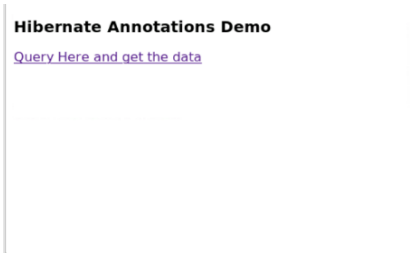
```
package com.simpli;
```

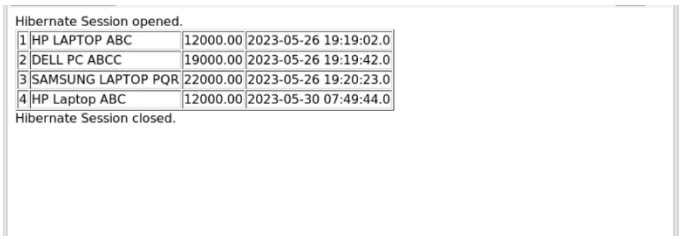
```
import java.io.IOException;
import java.io.PrintWriter;
import java.util.List;

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 org.hibernate.Session;
import org.hibernate.SessionFactory;

import com.ecommerce.EProduct;

/**
 * Servlet implementation class HibernateQueryDemo
 */
@WebServlet("/query-demo")
public class HibernateQueryDemo extends HttpServlet {
	private static final long serialVersionUID = 1L;
```

```
    /**
     * @see HttpServlet#HttpServlet()
     */
    public HibernateQueryDemo() {
        super();
        // TODO Auto-generated constructor stub
    }
    protected void doGet(HttpServletRequest request,
HttpServletResponse response)
                    throws ServletException, IOException {
                PrintWriter out = response.getWriter();
                out.println("<html><body>");

                // STEP 1: Get a Session (connection) from the Session
Factory class
                SessionFactory factory = HibernateUtil.getSessionFactory();

                // STE2 Create the session object
                Session session = factory.openSession();

                out.println("Hibernate Session opened.<br>");
                // 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());
                        }

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



                session.close();

                out.println("Hibernate Session closed.<br>");

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

}
```

Output:


Hibernate Annotations Demo
Query Here and get the data


Hibernate Session opened.
1 HP LAPTOP ABC 12000.00 2023-05-26 19:19:02.0
2 DELL PC ABCC 19000.00 2023-05-26 19:19:42.0
3 SAMSUNG LAPTOP PQR 22000.00 2023-05-26 19:20:23.0
4 HP Laptop ABC 12000.00 2023-05-30 07:49:44.0
Hibernate Session closed.