3.Configure Hibernate using Annotations in Eclipse IDE.

Configuring hibernate with hibernate.cfg.xml:

```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.jdbc.Driver</property>
    <property name="connection.url">jdbc:mysql://localhost:3306/ecommerce</property>
    <property name="connection.username">mysql user id</property>
    <property name="connection.password">mysql password</property>
  </session-factory>
</hibernate-configuration>
```

Creating an hibernateutil class to initate hibernate in code:

```java
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()
```

```java
                    .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;

    }

}
```

Creating an HTML page index.html:

```html
<!DOCTYPE html>

<html>

<head>

<meta charset="UTF-8">

<title>Hibernate Annotation Example</title>

</head>

<body>

<a href="list">List Products</a><br>


</body>
```

Creating a java classs Eproduct to act as the Data Access Object(DAO)for the table:

```java
package com.ecommerce;


import java.math.BigDecimal;

import java.util.Date;


import javax.persistence.Column;
```

```java
import javax.persistence.Entity;

import javax.persistence.GeneratedValue;

import javax.persistence.Id;

import javax.persistence.Table;


@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 long getID() {return this.ID; }
    public String getName() { return this.name;}
    public BigDecimal getPrice() { return this.price;}
    public Date getDateAdded() { return this.dateAdded;}


    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;}
```

}

Creating a ListProducts servlet:

```java
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.util.List;


import org.hibernate.Session;

import org.hibernate.SessionFactory;

import org.hibernate.Transaction;

import org.hibernate.cfg.Configuration;



import com.ecommerce.EProduct;

import com.ecommerce.HibernateUtil;


/**
 * Servlet implementation class ListProducts
 */
@WebServlet("/ListProducts")
public class ListProducts extends HttpServlet {
```

```java
    private static final long serialVersionUID = 1L;




    /**
     * @see HttpServlet#HttpServlet()
     */
    public ListProducts() {
        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();
            // using HQL
            List<EProduct> list = session.createQuery("from EProduct").list();

            session.close();

            PrintWriter out = response.getWriter();
            out.println("<html><body>");
            out.println("<b>Product Listing</b><br>");
            for(EProduct p: list) {
```

```java
                out.println("ID: " + String.valueOf(p.getID()) + ", Name: " + p.getName() +

                        ", Price: " + String.valueOf(p.getPrice()) + ", Date Added: " +
p.getDateAdded().toString() + "<br>");

            }


        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);

}


}
```

Configuring web.xml:

```xml
<?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>HibernateAnnotations</display-name>

 <welcome-file-list>

  <welcome-file>index.html</welcome-file>
```

```xml
        <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>ListProducts</servlet-name>

        <servlet-class>ListProducts</servlet-class>

    </servlet>


    <servlet-mapping>

        <servlet-name>ListProducts</servlet-name>

        <url-pattern>/list</url-pattern>

    </servlet-mapping>



</web-app>
```