2. Configure Hibernate using XML in Eclipse IDE.

**Index.html**

<br><br>

<h3>Hibernate Query Demo </h3>

<a href=*"query-demo"*>Query Here and get the data</a><br>

**EProduct.java**

**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 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;**

**}**

**}**

**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"*/>

</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"*>Annu!a@123456</property>

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

</session-factory>

</hibernate-configuration>

HibernateQueryDemo

**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;

@WebServlet("/query-demo")

**public** **class** HibernateQueryDemo **extends** HttpServlet {

**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>");

}

}



