

Addproduct.jsp

```
<%@ page language="java" contentType="text/html; charset=UTF-8"
pageEncoding="UTF-8"%>
<!DOCTYPE html>
<html>
<head>
    <title>Add Product</title>
</head>
<body>
    <h1>Add Product</h1>
    <br><br>
    <form action="AddProductServlet" method="post">
        Name: <input type="text" name="name" required><br><br>
        Price: <input type="text" name="price" required><br><br>
        HDD: <input type="text" name="partsHdd" required><br><br>
        CPU: <input type="text" name="partsCpu" required><br><br>
        RAM: <input type="text" name="partsRam" required><br><br>
        <input type="submit" value="Submit">
    </form>
</body>
</html>
```

AddProductServlet.java

```
import javax.servlet.*;
import javax.servlet.http.*;
import java.io.IOException;
import java.math.BigDecimal;
import java.util.Date;
import org.hibernate.Session;
import org.hibernate.SessionFactory;
import org.hibernate.Transaction;
import org.hibernate.cfg.Configuration;

import com.ecommerce.Product;

public class AddProductServlet extends HttpServlet {
    protected void doPost(HttpServletRequest request, HttpServletResponse response) throws
ServletException, IOException {
        // Retrieve form data
        String name = request.getParameter("name");
        BigDecimal price = new BigDecimal(request.getParameter("price"));
```

```

String partsHdd = request.getParameter("partsHdd");
String partsCpu = request.getParameter("partsCpu");
String partsRam = request.getParameter("partsRam");

// Create Product object
Product product = new Product();
product.setName(name);
product.setPrice(price);
product.setDateAdded(new Date());
product.setPartsHdd(partsHdd);
product.setPartsCpu(partsCpu);
product.setPartsRam(partsRam);

// Hibernate operations
Configuration configuration = new Configuration().configure();
SessionFactory sessionFactory = configuration.buildSessionFactory();
Session session = sessionFactory.openSession();
Transaction transaction = session.beginTransaction();

// Save product to database
session.save(product);

// Commit transaction and close session
transaction.commit();
session.close();

// Redirect back to addProduct.jsp
response.sendRedirect("success.html");
}
}

```

Product.java

```

package com.ecommerce;

import java.math.BigDecimal;
import java.util.Date;
import javax.persistence.*;

@Entity
@Table(name = "eproduct")
public class Product {

```

```
@Id
@GeneratedValue(strategy = GenerationType.IDENTITY)
@Column(name = "ID")
private Long id;
```

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

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

```
@Column(name = "date_added")
@Temporal(TemporalType.TIMESTAMP)
private Date dateAdded;
```

```
@Column(name = "parts_hdd")
private String partsHdd;
```

```
@Column(name = "parts_cpu")
private String partsCpu;
```

```
@Column(name = "parts_ram")
private String partsRam;
```

```
public Long getId() {
    return id;
}
```

```
public void setId(Long id) {
    this.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;
    }

    public String getPartsHdd() {
        return partsHdd;
    }

    public void setPartsHdd(String partsHdd) {
        this.partsHdd = partsHdd;
    }

    public String getPartsCpu() {
        return partsCpu;
    }

    public void setPartsCpu(String partsCpu) {
        this.partsCpu = partsCpu;
    }

    public String getPartsRam() {
        return partsRam;
    }

    public void setPartsRam(String partsRam) {
        this.partsRam = partsRam;
    }

}
```

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>
        <property name="hibernate.dialect">org.hibernate.dialect.MySQL5Dialect</property>
        <property
name="hibernate.connection.driver_class">com.mysql.cj.jdbc.Driver</property>
        <property
name="hibernate.connection.url">jdbc:mysql://localhost:3306/ecommerce</property>
        <property name="hibernate.connection.username">root</property>
        <property name="hibernate.connection.password">root</property>
        <property name="hibernate.hbm2ddl.auto">update</property>
        <property name="show_sql">true</property>
        <mapping class="com.ecommerce.Product"/>
    </session-factory>
</hibernate-configuration>
```

Product.hbm.xml

```
<?xml version="1.0"?>
<!DOCTYPE hibernate-mapping PUBLIC "-//Hibernate/Hibernate Mapping DTD 3.0//EN"
    "http://hibernate.sourceforge.net/hibernate-mapping-3.0.dtd">
<hibernate-mapping package="com.ecommerce">
    <class name="Product" table="eproduct">
        <id name="id" type="java.lang.Long">
            <column name="ID"/>
            <generator class="identity"/>
        </id>
        <property name="name" type="string">
            <column name="name"/>
        </property>
        <property name="price" type="big_decimal">
            <column name="price"/>
        </property>
        <property name="dateAdded" type="timestamp">
            <column name="date_added"/>
        </property>
        <property name="partsHdd" type="string">
            <column name="parts_hdd"/>
        </property>
    </class>
</hibernate-mapping>
```

```
</property>
<property name="partsCpu" type="string">
  <column name="parts_cpu"/>
</property>
<property name="partsRam" type="string">
  <column name="parts_ram"/>
</property>
</class>
</hibernate-mapping>
```

success.html

```
<!DOCTYPE html>
<html>
<head>
<meta charset="ISO-8859-1">
<title>Insert title here</title>
</head>
<body>
  <h1>Product added successfully !</h1>
  <br><br>
  <a href="addProduct.jsp">add another product</a>
</body>
</html>
```