

# Component Mapping

## Create Database:

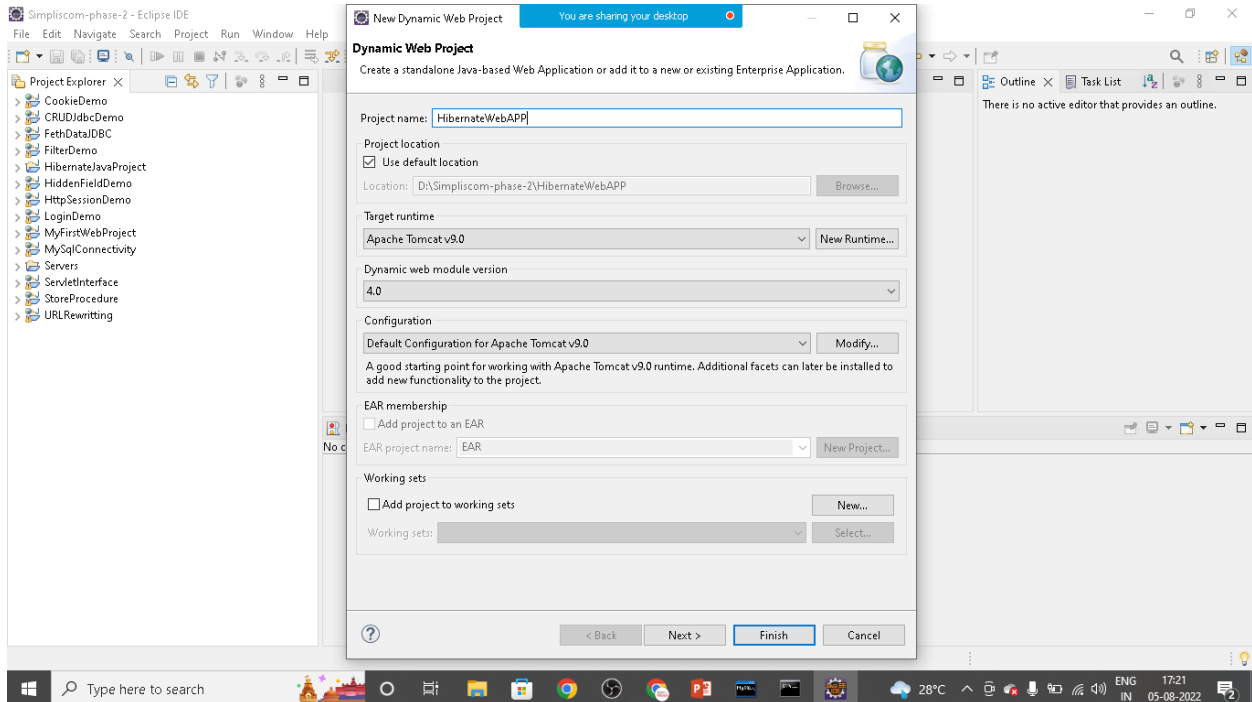
- create database hibernate\_mapping\_component;

## Use database :

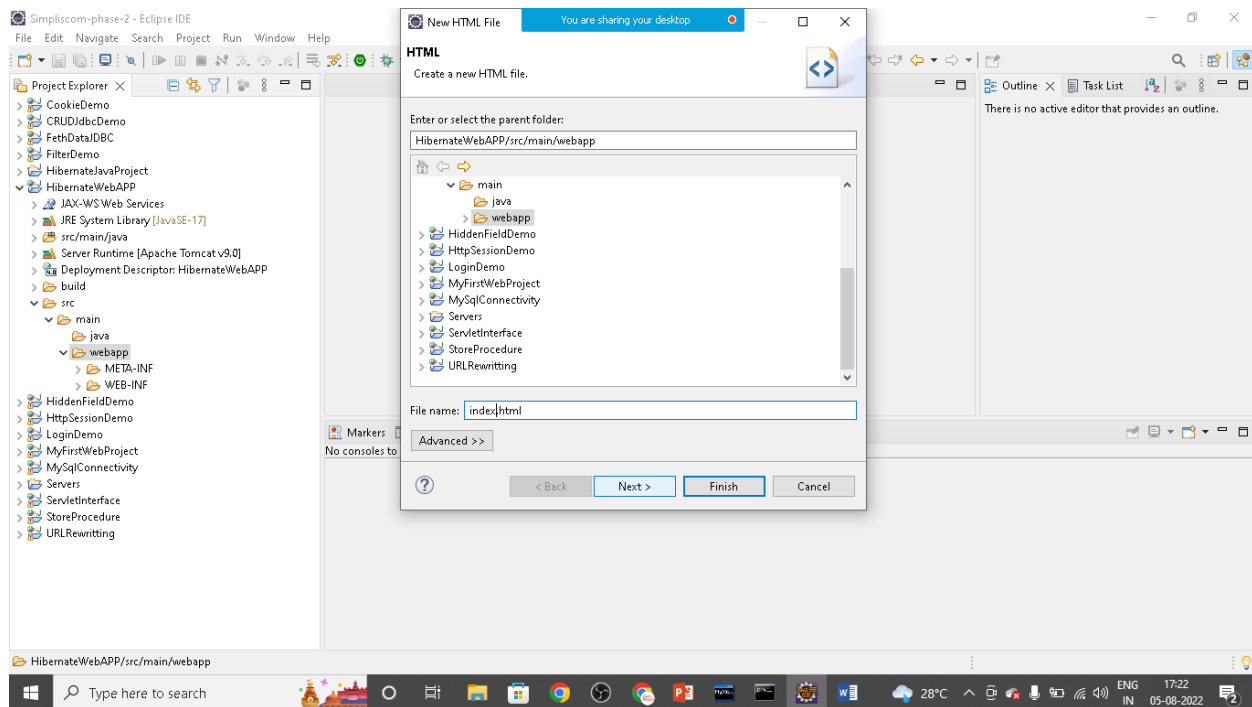
- use hibernate\_mapping\_component;

## Prepare dynamic web app

# Create dynamic webapp

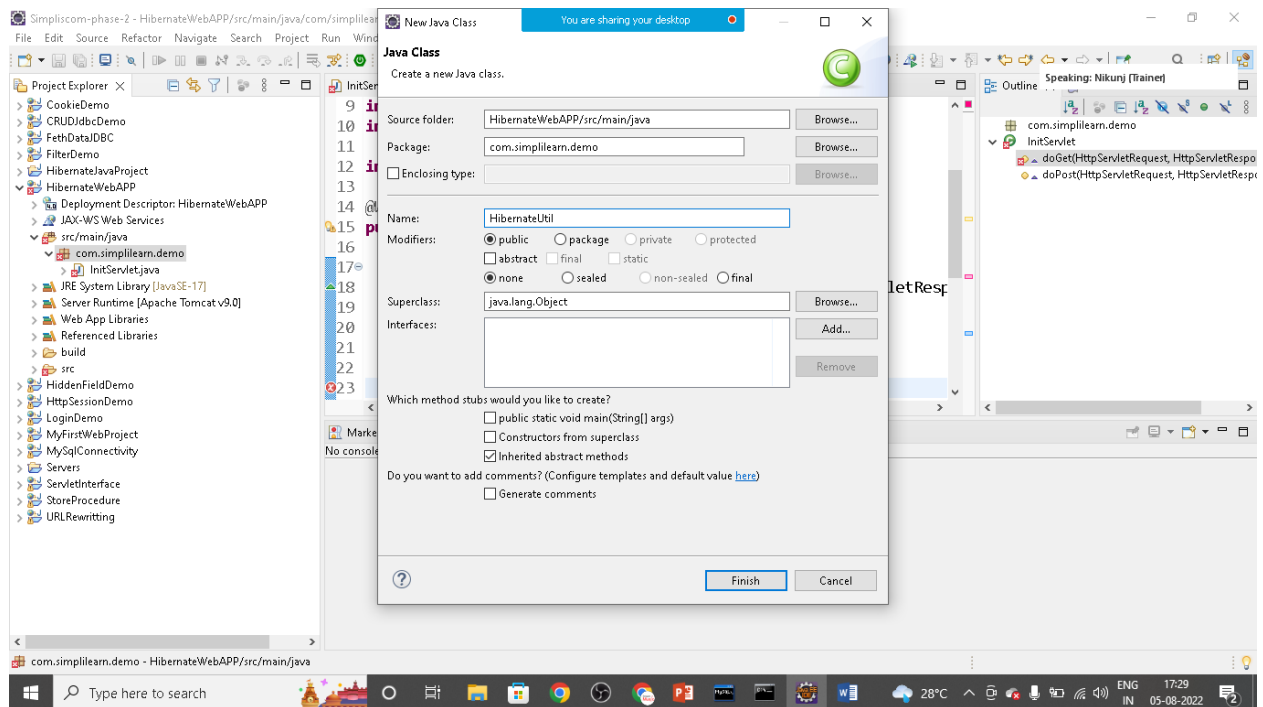


## Create index.html file



**Src>main>webapp>WEB-INF>lib>copy all hibernate jar file and mysql connector jar files**

Create **HibernateUtil** class



```
package com.simplilearn.demo;
```

```
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 HibenateUtil {
```

```
    private static SessionFactory sessionFactory;
```

```
    static {
```

```
        try {
```

```
            StandardServiceRegistry registry= new StandardServiceRegistryBuilder().
```

```

        configure("hibernate.cfg.xml").build();

        Metadata metadata= new MetadataSources(registry).getMetadataBuilder()

            .build();

        sessionFactory= metadata.getSessionFactoryBuilder().build();
    }

    catch(Exception e){

    }

}

public static SessionFactory getSessionFactory() {

    return sessionFactory;

}

}

```

Product

---

```

package com.simplilearn.demo;

import java.math.BigDecimal;

public class Product {

    private long id;
    private String name;
    private BigDecimal price;

```

```
private ProductParts parts;
```

```
public Product() {  
  
}
```

```
public Product( String name, BigDecimal price,  
ProductParts parts) {  
    super();  
    this.name = name;  
    this.price = price;  
    this.parts = parts;  
}
```

```
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 ProductParts getParts() {  
    return parts;  
}
```

```
        public void setParts(ProductParts parts) {
            this.parts = parts;
        }

    }
}
```

ProductParts

---

```
package com.simplilearn.demo;

public class ProductParts {

    private String hdd;
    private String cpu;
    private String ram;

    public ProductParts() {
        // TODO Auto-generated constructor stub
    }

    public ProductParts(String hdd, String cpu,
String ram) {
        super();
        this.hdd = hdd;
        this.cpu = cpu;
        this.ram = ram;
    }
}
```

```
public String getHdd() {  
    return hdd;  
}  
  
public void setHdd(String hdd) {  
    this.hdd = hdd;  
}  
  
public String getCpu() {  
    return cpu;  
}  
  
public void setCpu(String cpu) {  
    this.cpu = cpu;  
}  
  
public String getRam() {  
    return ram;  
}  
  
public void setRam(String ram) {  
    this.ram = ram;  
}  
  
@Override  
public String toString() {  
    return "ProductParts [hdd=" + hdd + ", cpu=" +  
+ cpu + ", ram=" + ram + "];"  
}
```



```
}
```

SaveDetailsServlet

---

```
package com.simplilearn.demo;

import java.io.IOException;
import java.io.PrintWriter;
import java.math.BigDecimal;

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 org.hibernate.Transaction;

@WebServlet("/save")
public class SaveDeatailServlet extends HttpServlet {

    @Override
    protected void doGet(HttpServletRequest req,
        HttpServletResponse resp) throws ServletException,
        IOException {
        // TODO Auto-generated method stub
        PrintWriter out=resp.getWriter();

        String name=req.getParameter("pname");
        String price=req.getParameter("pprice");
        String hdd=req.getParameter("hdd");
        String cpu=req.getParameter("cpu");
```

```

        String ram=req.getParameter("ram");

        ProductParts parts=new
ProductParts(hdd,cpu,ram);
        Product product=new Product(name,new
BigDecimal(price),parts);

        SessionFactory factory=
HibernateUtil.getSessionfactory();
        Session session= factory.openSession();

        Transaction tx=session.beginTransaction();
        session.save(product);

        tx.commit();

        session.close();
        out.print("Data Added Successfully");
        resp.sendRedirect("list");
    }

    @Override
    protected void doPost(HttpServletRequest req,
        HttpServletResponse resp) throws ServletException,
        IOException {
        // TODO Auto-generated method stub
        doGet(req, resp);
    }

}

```

## ListServlet

---

```
package com.simplilearn.demo;

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;

@WebServlet("/list")
public class ListServlet extends HttpServlet {

    @Override
    protected void doGet(HttpServletRequest req,
        HttpServletResponse resp) throws ServletException,
        IOException {
        // TODO Auto-generated method stub
        resp.setContentType("text/html");
        SessionFactory factory=
        HibernateUtil.getSessionfactory();
        Session session= factory.openSession();

        List<Product> list=session.createQuery("from
Product").list();
```

```

        PrintWriter out=resp.getWriter();
        out.print("<h1>Product List</h1>");

        for(Product p:list) {
            out.print("<b>" +p.getId()+" .
"+p.getName()+" " +p.getPrice()+"</b><br>");

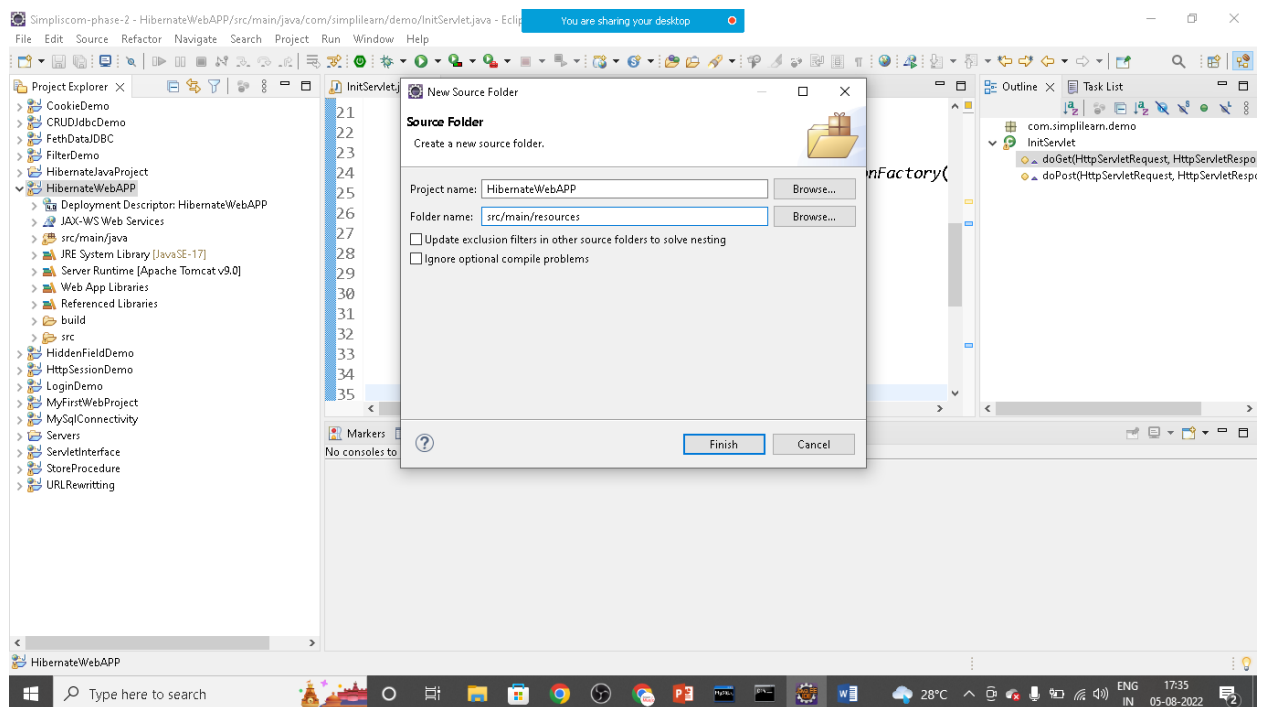
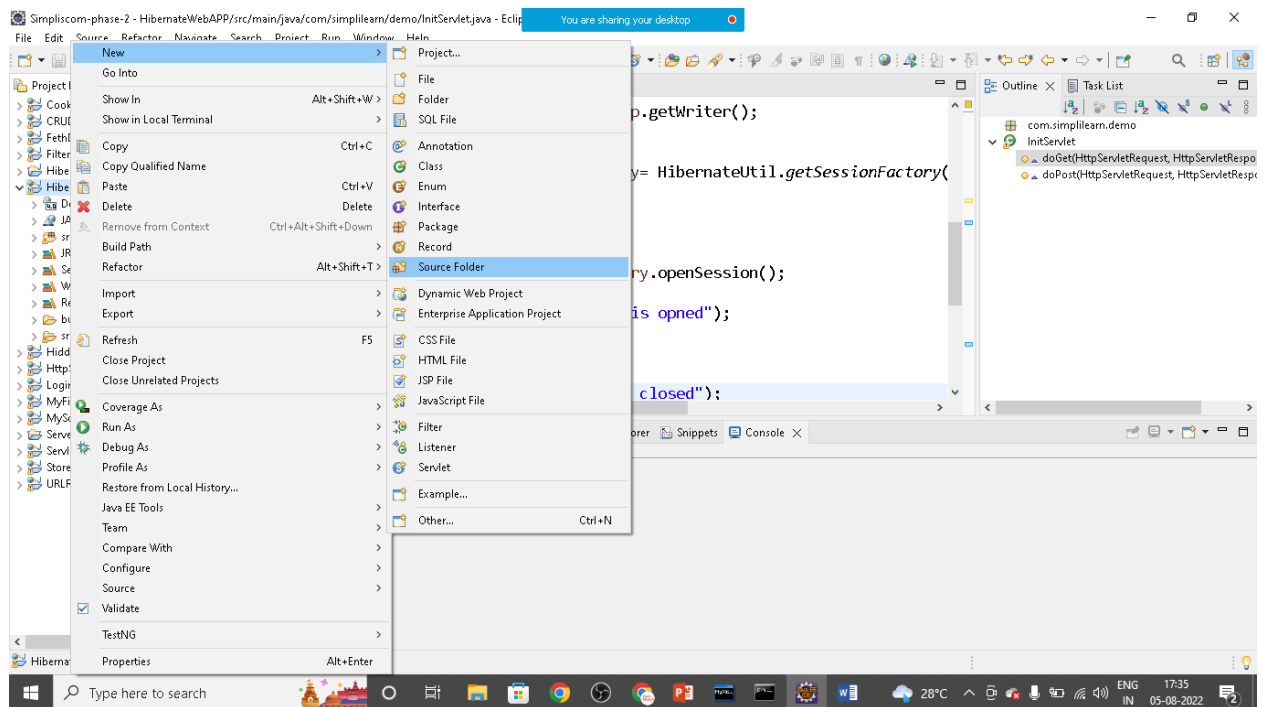
            ProductParts part=p.getParts();
            out.print("HDD: " +part.getHdd()+"<br>");
            out.print("CPU: " +part.getCpu()+"<br>");
            out.print("RAM: " +part.getRam()+"<br>");
            out.print("-----
-----<br>");
        }
    }

    @Override
    protected void doPost(HttpServletRequest req,
        HttpServletResponse resp) throws ServletException,
        IOException {
        // TODO Auto-generated method stub
        doGet(req, resp);
    }

}

```

Create **resource** folder



## Create

1. hibernate.cfg.xml
2. product.hbm.xml

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="connection.driver_class">com.mysql.cj.jdbc.Driv
er</property>
            <property
name="connection.url">jdbc:mysql://localhost:3306/hib
ernate_mapping_component</property>
            <property
name="connection.username">root</property>
            <property
name="connection.password">Nikunj@123</property>
            <property
name="hbm2ddl.auto">update</property>
            <property
name="dialect">org.hibernate.dialect.MySQL8Dialect</p
roperty>
            <property
name="show_sql">true</property>

            <mapping resource="product.hbm.xml"/>
        </session-factory>
    </hibernate-configuration>
```

product.hbm.xml

---

```
<?xml version="1.0" encoding="UTF-8"?>

<!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.simplilearn.demo" >

    <class name="Product" table="eproduct" >
        <id name="id" type="long" column="ID">
            <generator
class="increment"></generator>
        </id>

        <property name="name"
type="string"></property>
        <property name="price"
type="big_decimal"></property>

        <component name="parts"
class="com.simplilearn.demo.ProductParts">
            <property name="hdd"
column="parts_hdd"></property>
            <property name="cpu"
column="parts_cpu"></property>
            <property name="ram"
column="parts_ram"></property>
        </component>

    </class>
</hibernate-mapping>
```

## Index.html

---

```
<!DOCTYPE html>
<html>
<head>
<meta charset="ISO-8859-1">
<title>Insert title here</title>
</head>
<body>

<form action="save" method="post">
    Name:<input type="text" name="pname"><br><br>
    Price:<input type="text" name="pprice"><br><br>
    Hard Disc Details:<input type="text"
name="hdd"><br><br>
    CPU Details:<input type="text"
name="cpu"><br><br>
    RAM details:<input type="text"
name="ram"><br><br>

    <input type="submit" value="Save Data Using
Component"><br><br>

</form>

<br>

<br>

<a href="list">Product Details</a>

</body>
```



</html>