**Pseudo-code:**

Class Products:

Properties:

pid: int

pname: String

price: float

Getters and Setters for properties

Class HibernateConfig:

Static Properties:

sessionFactory: SessionFactory

Static Method getSessionFactory():

if sessionFactory is null:

Create a new Configuration object

Create a new Properties object

Set the properties for the database connection

Set the properties for Hibernate configuration

Add the Products class as an annotated class

Create a ServiceRegistry using the Configuration properties

Build the SessionFactory using the ServiceRegistry

Assign the SessionFactory to sessionFactory

return sessionFactory

Class ProOp:

Properties:

sfactory: SessionFactory

Constructor:

Initialize sfactory with HibernateConfig.getSessionFactory()

Method AddNewProduct(pro: Products):

result = "Error"

Open a new Session from sfactory

Begin a new Transaction

Save the pro object using the Session

Commit the Transaction

if the save operation is successful:

result = "Success"

return result

Method ShowAll():

Open a new Session from sfactory

Create a Query to fetch all Products

Execute the Query and assign the result to pro1

return pro1

prodPage.html:

HTML form to add a new product with inputs for product name and price

Class ProductServlet:

Method service(request, response):

Get the product name and price from the request parameters

Create a new Products object

Create a new ProOp object

Set the name and price of the Products object

Set the response content type to "text/html"

Get a PrintWriter for the response

Call AddNewProduct method of ProOp to add the product to the database

If the result is "Success":

Print a success message and a link to ShowProServlet

Close the PrintWriter

Class ShowProServlet:

Method service(request, response):

Create a new ProOp object

Call the ShowAll method of ProOp to get a list of all products

Set the response content type to "text/html"

Get a PrintWriter for the response

Print the product list in an HTML table

Close the PrintWriter