**15-11-2022**

1. Take array value through keyboards as a integer and display sum of all number, sum even number, sum of odd numbers.
2. Take string array value through keyboards and display all names.
3. Take array value through keyboards as a integer and display those number in ascending order or descending order. (nested loop)
4. Create Employee class which contains three instance variable ie id,name,salary and two non static method calculateSalary() and dispalyEmployeeInfo() method.

In calcualteSalary method you need to declare three local variable as hra, da, and pf

And calculate the salary

Salary = salary+ 10% hra on salary + 5% da – 7% pf

In displayEmployeeInfo method display id,name,salary

In main class create two object and set the id,name,salary for one employee don’t call calculteSalary and second employee call both methods.

1. Create Employee class which contains three instance variable ie id,name,salary, write two constructor one is empty with default id,name,salary(123,unknown, 8000), parameterized constructor, and setValue Method, and two non static method calculateSalary() and dispalyEmployeeInfo() method.

In calcualteSalary method you need to declare three local variable as hra, da, and pf

And calculate the salary

Salary = salary+ 10% hra on salary + 5% da – 7% pf

In displayEmployeeInfo method display id,name,salary

In main class create 3 object

1st object empty constructor

2nd object parameterized constructor

3rd object set value through methods.

Then call display method for all objects.

**16-11-2022**

1. Take number of employee records using array as id,name,salary, designation. Then if desg is manager then give 5000, if designation is programmer then give 3000 else 1500 bonus

Then display all record ie id,name,salary,designation.

1. Make complete is and has relationship assignment.
2. Using method overloading do small example. Find the area of triangle, rectangle and circle and method name must be area and class may be Operation.
3. Please go through all example of method overriding, abstract and static example.

17-11-2022

1. create interface Bank which contains four abstract method ie

final int SIZE=10;

public String createAccount(int accno, String name, float amount);

public String withdrawAmount(int accno, float amount);

public String depositeAmount(int accno,float amount);

public String checkBalance(int accno);

createAccount, depositAmount, withdrawAmount, checkBalance

Create BankService class and that class implement Bank interface and it must be provide the body for all four methods.

This class contains three array variable of type accno, name and amount.

Static Count variable to keep the track how many account created.

public String createAccount(int accno, String name, float amount)

account number must be unique.

Min balance must be 500

public String withdrawAmount(int accno, float amount) {

if we give wrong account number then return account not exists

if correct we can display amount withdraw.

Maintain min 500.

}

public String depositAmount(int accno, float amount) {

if we give wrong account number then return account not exists

if correct we can display amount deposit.

We can’t deposit more than 50000

}

public String checkBalance(int accno) {

if we give wrong account number then return account not exists

return the balance.

}

Main class which contains main method

Do {

1: create accnount, 2:withdraw , 3: deposit 4: check balance

switch() {

1. pass the value for service class method

Break;

2

3

4

Default : wrong choice

}

Do you want to continue ?

}while();

18-11-2022

Inheritance, Interfaces and package.

|  |  |
| --- | --- |
| **Sr. No.** | **Assignment Question** |
| 1 | Australian Cricket is still the best team in the world and it has proved its supremacy in the last two World Cups held in the successive years 1999 and 2003.  The department handling the finance of the players is planning to develop software that would automatically calculate the income of the players based on their respective grade, the number of matches each player plays and also his performance in the tournament. Also, appropriate income tax should be applied on the income. To accomplish this, a team of experts have been chosen by the Australian Cricket Board to provide a solution for the same. Consider yourself to be a part of the team that implements the solution for designing the application.  Create an application using inheritance and interfaces to implement the Software. The application should consist of the following files.   1. Player.java 2. Tax.java bean package 3. PlayerIncome.java 4. GradeBonus.java 5. TournamentIncome.java service package 6. PlayerTest.java main package   Each file has a specific purpose and functionality. The descriptions of each file are as follows:  **Player.java**  The Player class is an abstract base class that contains an instance variable to store the name of the Player, and an abstract method.  The Booking class should contain an instance variable **name** to store the player’s name.  An abstract method named **displayDetails()** is created to display the name of the player.  **Tax.java**  An interface named **Tax** should be created that will act as an interface to calculate the tax of the players income.  Declare and initialize a constant variable **TAX\_PERCENT** to store the tax percentage. Also create an abstract method named **calculateTax()** to calculate the tax.  **PlayerIncome.java**  The PlayerIncome class should be a derived class of class Player and implement the interface Tax. This class should be used to store the Player details and display the same. |

|  |  |
| --- | --- |
|  | The PlayerIncome class should contain an instance variable **income** to store the player’s income.  Create a new instance of class PlayerIncome and store the salary in the instance variable and call the base constructor.  Now, create a method **calculateTax()** to calculate the tax on respective Player’s income.  Override the method **displayDetails()** to display the details about each player.  **GradeBonus.java**  Again an interface named **GradeBonus** is created to calculate the Bonus given to the player based on his grade.  Declare and initialize a constant variable **GRADE\_BONUS\_PERCENT** to store the grade bonus percentage. Also create an abstract method named **calculateGradeBonus()** to calculate the bonus to be given to a player on the basis of the grade he possess.  **TournamentIncome.java**  Again, create a derived class of the Player class and implement the GradeBonus interface. It should be used to store the details of income of each player for each tournament and display the same.  The TournamentIncome class should contain two instance variables **grade** and **rate** to store the player’s grade and the rate of each player per match he plays.  Create a new instance of class TournamentIncome and store the details in the three parameters **name**, **grade** and **rate**.  Now, create a method **calculateGradeBonus()** to calculate the bonus of each player based on his grade.  Create a method **displayDetails()** to display the details about each player. Here, calculate the income of the player using his income and the bonus amount he will be given as per his grade.  **PlayerTest.java**  Finally, create the main class PlayerTest that demonstrates the use of abstract classes and interfaces using the classes PlayerIncome and TournamentIncome. Create an object of PlayerIncome class and an object of TournamentIncome class and pass the argument. Now, display the details of  each employee of both the classes using the **displayDetails**() method. |

22-11-2022

1. Take array string value through keyboards and display all array names in ascending and descending order (hint take the help compareTo method of string class).
2. Create String reference and store one sentence. Convert each word in reverse word.

String msg = “Welcome to Java Training”;

Output must be

emocleW ot avaJ gniniarT;

(Hint check method in string and type of string classes) first convert string in array of string with delimiter(space) as word and with the of reverse method convert each word in reverse order and convert array string into string and display the output.

1. Create String variable and store the value as

String msg = “welcome to java trianing”;

While displaying the output we need as

Welcome To Java Training; each word first letter upper case.

Take the help of String and Character class methods.

23-11-2022

1. Find the number of words present in file
2. While copying from one file to another file convert all data
   1. Upper case
   2. Lower case
   3. Each word upper case.
3. While copying from one file to another file replace the word by example if file contains

Welcome to Java training. In another file we need to store as Welcome to Python training.

24- 11-2022

1. Create two classes one is StackService and another StackOperation

In StackService class create Stack object and write more than one business method

* 1. Check the size of stack if stack is empty display message as stack is underflow or display size.
  2. Add the elements in Stack. Make sure don’t add duplicate element.
  3. Display pop most elements from stack
  4. Remove top most elements form stack
  5. Search the elements from a stack
  6. Allow to store only five elements in stack if we are going to add more elements then display the message as overflow.

In main method using do while with switch statement all call all methods part of StackOperation class using case 1, 2,3,4,5 and 6.

**25-11-2022**

1. Create three classes

Product pid,pname,price

ProductService

List<Product> listOfProduct = new ArrayList<Product>();

storeProduct

but pid must be unique

price must > 500

then only store the product

search the product using pid then return product details.

Find the product price by passing pid

Retrieve all the products

Delete the product details using pid

Update the product price using pid but make sure price always increase.

Make sure don’t display any information in service class.

Those serialization and de serialization method

As well as static block or init must be part of service class.

ProductMain

Do while with switch statement

1: add,

2. search by pid and display all details.

3. find product price using pid

4. retrieve

5. delete

6. update

We want to store List of Product in external file using object serialization.

Product.ser

**Week 3**

28-11-2022

1. six operation in main class

1: Add Employee

2 : Delete Employee

3. update Employee salary

4. display all employee details

5. search employee using by id

6. find employee salary using id

In main class do while and switch statement.

**30-11-2022**

**Create maven project and add mysql connector dependency**

**Please follow classname, packagename, method name and variable properly.**

1. Create five classes

pms.bean, Product

pms.service ProductService

pms.dao ProductDao

pms.resource DbResouce

pms.main App

Product pid,pname,price

ProductService

List<Product> listOfProduct = new ArrayList<Product>();

storeProduct

but pid must be unique

price must > 500

then only store the product

search the product using pid then return product details.

Find the product price by passing pid

Retrieve all the products

Delete the product details using pid

Update the product price using pid but make sure price always increase.

All service class must be call dao method to do the operation with database.

ProductDao

Resource class

ProductMain

Do while with switch statement

1: add,

2. search by pid and display all details.

3. find product price using pid

4. retrieve all

5. delete

6. update

01-12-2022

**Servlet and JDBC**

1. Using Servlet and HTML page

Index.html

It contains form with action as Login Servlet with Get Method

In Login Servlet Program in doGet method we need to check value from database.

If you don’t have account then in index.html page you have to create hyper link to connect another page as createLogin.html that page contains form with action as Login Servlet with Post method which help to create the login details after login details creation you can check the login details from database.

Index.html

createLogin.html

Login Servlet

doGet : to check username and password form database.

doPost to create account details means store login details in database.

Home Servlet

MySQL Database you need to create the table as Login with emailId as PK and password columns.

Create dynamic web project with version 2.5 with tomcat 9.

Then create index.html , createLogin.html, Login Servlet and Home Servlt

**02-12-2022**

1. Develop MVC Style project using Servlet, JSP and JDBC

Create login table with emaild as pk and password column

In Maven with dynamic project create Login.java class with variable and setter and getter method to map to table.

Then create LoginController, LoginService, LoginDao, Resource class

In doGet method do SignIn operation

doPost method do SignUp operation

In Service method while storing username and password convert password into reverse order using StringBuffer class means if password is 1234 then you convert it into 4321 and store in database. While doing signIn operation convert reverse password in normal order format and check it.

Index.jsp contains form for signIn and hyperlink for singUp.jsp page if account present after verified it move to home server. If account not exits then we need to create account then verify the account and redirect to home page.

6/12/2022

1. Do CRUD Operation from main class for Employee or Customer Or Product etc.

Insert, Delete, Update and Retrieve using Spring framework with help of annotation as

@Component, @Service, @Repository and @Autowired and Database connectivity provided by DatatSource class by spring framework from xml file.

7/12/2022

1. Do CRUD Operation from main class for Employee or Customer Or Product etc.

Insert, Delete, Update and Retrieve using Spring framework with help of annotation as

@Component, @Service, @Repository and @Autowired and Database connectivity In DAO layer you need to use jdbcTemplate to do CRUD Operation.

8/12/2022

1. Project / Department and Employee (one to many relationship)

Project Entity class/Department entity class and Employee Entity class

Service classes

Dao classes

Main Class created

1: Add Project Or Department details

2. Add employee details

3. Delete employee

4. delete project / department

5. view project

6. view employee

7. view project and employee details.

9/12/2022

1. Using spring MVC with JPA do signIn and SignUp application. After successfully login move dashboard.jsp

@OneToMany(cascade = All)

13-12-2022

1. Create REST API for Customer (cid,cname,age)

GetAll customer

Find customer using customer id with path param or query

Store customer (only pass to controller )

15-12-2022

Create Rest API for the Student Management System and Test Those REST API Using Post man client and take screen shot for each rest api and then upload document with projects.

1. Admin can add student details like sid, sname, age, class, section
2. Admin can delete student details using student id
3. Admin can update student details.
4. Admin can add the student marks details like in marks table which contains mid, math, eng, marthi, social, science, total marks, grade.
5. Admin can update the marks, grade details.
6. Admin can view all student details.
7. Student view its own details using sid or roll number
8. Student can view its marks details and grade.
9. Student can request for admin to update marks details or grade
10. Admin can approved or reject base upon the conditions.
11. Student can view the request it approved or rejected as well as reason.