GIRI's Tech Hub, Pune

Programming (Machine) Test

Batch: July-2024 **Date: 18/11/2024**

Time: 10:00 to 12:00 Pm

Instructions: **Total:- 05 Marks**

1. Solve any four questions.
2. Input should be from user.
3. Indentation and comments mandatory.
4. Each program 1 Marks and 1 Marks Comments.

Q1. Create a Product POJO with fields: productId, name, price, quantity, and category.

- **Task:**
  1. **Initialize details for 5 products using a constructor.**
  2. **Add methods to:**
     - **Update the stock (add or reduce quantity).**
     - **Display products in a specific category.**
     - **Find the product with the highest stock and display its details.**

Q2. Design a Employee POJO with fields: empId, name, designation, basicSalary, and hra.

- **Task:**
  1. **Create a constructor to initialize all fields.**
  2. **Add a method calculateGrossSalary() in the Employee class to calculate the gross salary as: grossSalary=basicSalary+hra+basicSalary×0.2(bonus).**
  3. **Write a program to create an array of 5 employees, display their details, and calculate gross salary for each employee.**

Q3. Write a java program to Create a class name as ArrayOperation with a parameterised constructor with function name as calSum() & one more function int[] getSumArray().

```
ArrayOperation(int [ ]) {
    // this constructor can accept the array.
    }
void calSum() {
    // implement the logic.
    }
```

Int[] getSumArray()

{

// this function can return result array.

}

Input :- 5732 8659 2534 9625 7354.

(Store the each digit sum at that index)

Output : - 17 28 14 22 19

Q4. Create a base class Employee with fields empId, name, and salary.
Create a subclass Manager with an additional field bonus.
Create another subclass Intern with an additional field internshipDuration.

- **Implement parameterized constructors.**
- **Write a method calculateFinalSalary() in each class to calculate the salary:**
  - **For Employee, return salary.**
  - **For Manager, add bonus to salary.**
  - **For Intern, divide salary by the internship duration in months.**
- **Use an array to store employee data for all types, and implement a method to display the final salary of each employee.**

Q5. Create a base class Person with fields id, name, and age.
Create subclasses:

- **Student with additional fields grade and marks.**
- **Teacher with additional fields subject and salary.**
- **Write parameterized constructors to initialize all fields.**
- **Implement a method displayInfo() in Person and override it in both subclasses.**
- **Write a program to calculate:**
  - **The average marks of all students.**
  - **The total salary of all teachers.**

Q5. Create a BankAccount POJO class with fields accountNumber, accountHolderName, and balance. Create a subclass SavingsAccount with an additional field interestRate. Write a program to:

1. **Initialize both types of accounts using parameterized constructors.**
2. **Add a method in SavingsAccount to calculate the annual interest earned (interest = balance * interestRate / 100).**
3. **Display account details along with the calculated interest.**

-----ALL THE BEST-----