**Oops Concept Tasks**

**Q1.Write a java program on below questions[Oop’s].**

**1.1) Person class creation with properties of age and name with default age is 18**

1. **Default age of person should be 18**

**//Class creation**

**public class Person {**

**// Properties name and age=default-18**

**String name;**

**final int age = 18;**

**b. Person object initialized with name and age**

**// Person object initialized with name and age**

**Person person=new Person("john",8);**

**c.Method to display output of name and age**

**// Method to display output of name and age**

**public void display(){**

**System.*out*.println("Name: "+this.name);**

**System.*out*.println("Age: "+this.age);**

**}**

**Output:**

**Name: john**

**Age: 18**

**1.2) Product class creation:**

**// Product class creation**

**public class Product {**

**public int pid;**

**public int price;**

**public int quantity;**

**Constructor with parameters:**

**public Product(int pid,int price,int quantity){**

**this.pid =pid;**

**this.price =price;**

**this.quantity =quantity;**

**}**

**Create main function in different class ProductMain:**

**//ProductMain class creation**

**class ProductMain {**

**// Main method**

**public static void main(String[]args){**

**// Scanner method for get product info from the user**

**Scanner se=new Scanner(System.*in*);**

**Product[]products=new Product[5];**

1. **Accept five product and store in array**

**Scanner se=new Scanner(System.*in*);**

**Product[]products=new Product[5];**

**for (int i = 0; i < products.length; i++) {**

**System.*out*.println("Enter product "+(i+1)+" Information!");**

**System.*out*.println("Enter the Product ID: ");**

**int pid=se.nextInt();**

**System.*out*.println("Enter the Price: ");**

**int price=se.nextInt();**

**System.*out*.println("Enter Product Quantity: ");**

**int quantity=se.nextInt();**

**// Store the product info in the arrays**

**products[i]=new Product(pid,price,quantity);**

**}**

1. **Fid pid of the product with high price**

**Scanner se=new Scanner(System.*in*);**

**Product[]products=new Product[5];**

**for (int i = 0; i < products.length; i++) {**

**System.*out*.println("Enter product "+(i+1)+" Information!");**

**System.*out*.println("Enter the Product ID: ");**

**int pid=se.nextInt();**

**System.*out*.println("Enter the Price: ");**

**int price=se.nextInt();**

**System.*out*.println("Enter Product Quantity: ");**

**int quantity=se.nextInt();**

**// Store the product info in the arrays**

**products[i]=new Product(pid,price,quantity);**

**}**

**// finding the high price product pid**

**int highestPricePID=*findHighPriceProduct*(products);**

**System.*out*.println("Highest price Product's PID: "+highestPricePID);**

1. **Calculate total amount spend on all products**

**// Method to calculate total amount spend on all products**

**public static int totalAmount(Product[] products){**

**int totalAmount=0;**

**for (Product product:products){**

**totalAmount=totalAmount+ product.price\* product.quantity;**

**}**

**return totalAmount;**

**}**

**// finding the total amount spend on the all products**

**int totalSpentAmount=*totalAmount*(products);**

**System.*out*.println("Total amount spend on all products: "+totalSpentAmount);**

**Output:**

**Enter product 1 Information!**

**Enter the Product ID:**

**1**

**Enter the Price:**

**10**

**Enter Product Quantity:**

**1**

**Enter product 2 Information!**

**Enter the Product ID:**

**2**

**Enter the Price:**

**20**

**Enter Product Quantity:**

**2**

**Enter product 3 Information!**

**Enter the Product ID:**

**3**

**Enter the Price:**

**30**

**Enter Product Quantity:**

**3**

**Enter product 4 Information!**

**Enter the Product ID:**

**4**

**Enter the Price:**

**40**

**Enter Product Quantity:**

**4**

**Enter product 5 Information!**

**Enter the Product ID:**

**5**

**Enter the Price:**

**50**

**Enter Product Quantity:**

**5**

**Highest price Product's PID: 5**

**Total amount spend on all products: 550**

**1.3) create class account with data member as balance**

**// Class account Creation**

**public class Account {**

**double balance;**

**Constructors With no arguments and with arguments**

**// Constructors with argument**

**public Account(double balance){**

**this.balance=balance;**

**}**

**// Constructors without argument**

**public Account(){**

**}**

1. **Method to deposit the amount**

**// method to deposit amount**

**public void depositAmount(){**

**Scanner se=new Scanner(System.*in*);**

**System.*out*.println("Enter Deposit Amount: ");**

**double deposit = se.nextDouble();**

**balance=balance+deposit;**

**System.*out*.println("The Deposited Amount: "+deposit);**

**System.*out*.println("The Balance Amount: "+balance);**

**}**

1. **Method to withdraw the amount**

**// method to withdraw amount**

**public void withdrawAmount(){**

**Scanner se=new Scanner(System.*in*);**

**System.*out*.println("Enter withdraw Amount: ");**

**double withdraw=se.nextDouble();**

**balance=balance-withdraw;**

**System.*out*.println("the withdraw Amount: "+withdraw);**

**System.*out*.println("the balance Amount: "+balance);**

**}**

1. **Method to display balance**

**// method to display amount**

**public void displayAmount(){**

**System.*out*.println("The Account balance: "+balance);**

**}**

**Output:**

**The Account balance: 1000.0**

**Enter Deposit Amount:**

**200**

**The Deposited Amount: 200.0**

**The Balance Amount: 1200.0**

**Enter withdraw Amount:**

**500**

**the withdraw Amount: 500.0**

**the balance Amount: 700.0**

**The Account balance: 700.0**

**1.4) Person class creation with properties of age and name**

**//Class creation**

**public class persons {**

**// Properties name and age=default-18**

**String name;**

**int age;**

**Subclass employee creation with employeeid and salary**

**class Employee extends persons{**

**int employeeId;**

**double salary;**

**Super keyword to initialized person attributes**

**class Employee extends persons{**

**int employeeId;**

**double salary;**

**public Employee(String name, int age,int employeeId,double salary) {**

**super(name, age);**

**this.employeeId=employeeId;**

**this.salary=salary;**

**}**

**Output:**

**Name: naveen**

**Age: 20**

**Name: john**

**Age: 10**

**EmployeeID: 1**

**Salary: 20000.0**