

Solution:1

```
package task2;
```

```
class PersonTask1 {  
    int age=25;  
    String name;  
  
    public void displayDetails(){  
        System.out.println("Person age is"+" "+this.age);  
        System.out.println("Person name is"+" "+name);  
    }  
    public PersonTask1(int age,String name){  
        this.age=age;  
        this.name=name;  
    }  
    public static void main(String[] args) {  
        PersonTask1 person=new PersonTask1(25,"Vamsi");  
        person.displayDetails();  
  
    }  
}
```

Output:

Person age is 25

Person name is Vamsi

Solution 2:-

```
package task2;
```

```
import java.util.Scanner;
```

```

public class XYZ{
    // Method to calculate total amount spent on all products
    public static double calculateTotalAmount(Product[] products) {
        double totalAmount = 0;
        for (Product product : products) {
            totalAmount += product.getPrice() * product.getQuantity();
        }
        return totalAmount;
    }

    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);

        // Array to store 5 products
        Product[] products = new Product[5];

        // Accept 5 product details from the user
        for (int i = 0; i < 5; i++) {
            System.out.println("Enter details for product " + (i + 1) + " (pid, price, quantity):");
            int pid = scanner.nextInt();
            double price = scanner.nextDouble();
            int quantity = scanner.nextInt();

            // Create a product and add it to the array
            products[i] = new Product(pid, price, quantity);
        }

        // Find the product ID (pid) with the highest price
        int maxPricePid = products[0].getPid();
        double maxPrice = products[0].getPrice();

        for (Product product : products) {
            if (product.getPrice() > maxPrice) {
                maxPrice = product.getPrice();
                maxPricePid = product.getPid();
            }
        }
    }
}

```

```

        System.out.println("Product ID with the highest price: " + maxPricePid);

        // Calculate and display the total amount spent
        double totalAmount = calculateTotalAmount(products);
        System.out.println("Total amount spent on all products: " + totalAmount);

    }
}

```

```

package task2;
import java.util.Scanner;

class Product {
    // Data members
    private int pid;
    private double price;
    private int quantity;

    // Parameterized constructor
    public Product(int pid, double price, int quantity) {
        this.pid = pid;
        this.price = price;
        this.quantity = quantity;
    }

    // Getters
    public int getPid() {
        return pid;
    }

    public double getPrice() {
        return price;
    }

    public int getQuantity() {
        return quantity;
    }

    public static class Xyz {
        // Method to calculate total amount spent on all products
        public static double calculateTotalAmount(Product[] products) {
            double totalAmount = 0;
            for (Product product : products) {

```

```

        totalAmount += product.getPrice() * product.getQuantity();
    }
    return totalAmount;
}

public static void main(String[] args) {
    Scanner scanner = new Scanner(System.in);

    // Array to store 5 products
    Product[] products = new Product[5];

    // Accept 5 product details from the user
    for (int i = 0; i < 5; i++) {
        System.out.println("Enter details for product " + (i + 1) + " (pid, price, quantity):");
        int pid = scanner.nextInt();
        double price = scanner.nextDouble();
        int quantity = scanner.nextInt();

        // Create a product and add it to the array
        products[i] = new Product(pid, price, quantity);
    }

    // Find the product ID (pid) with the highest price
    int maxPricePid = products[0].getPid();
    double maxPrice = products[0].getPrice();

    for (Product product : products) {
        if (product.getPrice() > maxPrice) {
            maxPrice = product.getPrice();
            maxPricePid = product.getPid();
        }
    }

    System.out.println("Product ID with the highest price: " + maxPricePid);

    // Calculate and display the total amount spent
    double totalAmount = calculateTotalAmount(products);
    System.out.println("Total amount spent on all products: " + totalAmount);

}
}
}

```

Output:

Enter details for product 1 (pid, price, quantity):

1

10

2

Enter details for product 2 (pid, price, quantity):

2

20

3

Enter details for product 3 (pid, price, quantity):

3

30

4

Enter details for product 4 (pid, price, quantity):

4

40

5

Enter details for product 5 (pid, price, quantity):

5

50

6

Product ID with the highest price: 5

Total amount spent on all products: 700.0

Solution 3:

```
package task2;
import java.util.Scanner;

class Account {
    private double balance;

    // Constructor with no argument
    public Account() {
        this.balance = 0;
    }

    // Constructor with two arguments
    public Account(double balance) {
        this.balance = balance;
    }

    // Method to deposit the amount to the account
    public void deposit(double amount) {
        this.balance += amount;
    }

    // Method to withdraw the amount from the account
    public void withdraw(double amount) {
        if (amount > this.balance) {
            System.out.println("Insufficient balance.");
        } else {
            this.balance -= amount;
        }
    }

    // Method to display the Balance
    public void displayBalance() {
        System.out.println("Balance: " + this.balance);
    }
}

public class AccountMain {
    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);
        Account account = new Account();

        while (true) {
            System.out.println("\nChoose an option:");
            System.out.println("1. Deposit");
            System.out.println("2. Withdraw");
            System.out.println("3. Display Balance");
            System.out.println("4. Exit");
        }
    }
}
```

```

        int choice = scanner.nextInt();

        switch (choice) {
            case 1:
                System.out.print("Enter deposit amount: ");
                double depositAmount = scanner.nextDouble();
                account.deposit(depositAmount);
                break;
            case 2:
                System.out.print("Enter withdraw amount: ");
                double withdrawAmount = scanner.nextDouble();
                account.withdraw(withdrawAmount);
                break;
            case 3:
                account.displayBalance();
                break;
            case 4:
                System.out.println("Exiting...");
                System.exit(0);
            default:
                System.out.println("Invalid option.");
        }
    }
}

```

Output:

Choose an option:

1. Deposit
2. Withdraw
3. Display Balance
4. Exit

1

Enter deposit amount: 1000

Choose an option:

1. Deposit
2. Withdraw
3. Display Balance
4. Exit

2

Enter withdraw amount: 200

Choose an option:

1. Deposit
2. Withdraw
3. Display Balance
4. Exit

3

Balance: 800.0

Choose an option:

1. Deposit
2. Withdraw
3. Display Balance
4. Exit

4

Exiting...

Process finished with exit code 0

Solution 4:


```
package task2;

public class Person {
    String name;
    int age;
}
```

```
package task2;

public class Employee extends Person{
    int employeeID;
    double salary;
    public Employee(String name,int age){
        System.out.println("Employee name is"+(super.name=name));
        System.out.println("Employee age is"+(super.age=age));
    }
    public static void main(String[] args) {
        Employee em=new Employee("Vamsi",25);
    }
}
```

Output:

Employee name isVamsi

Employee age is25