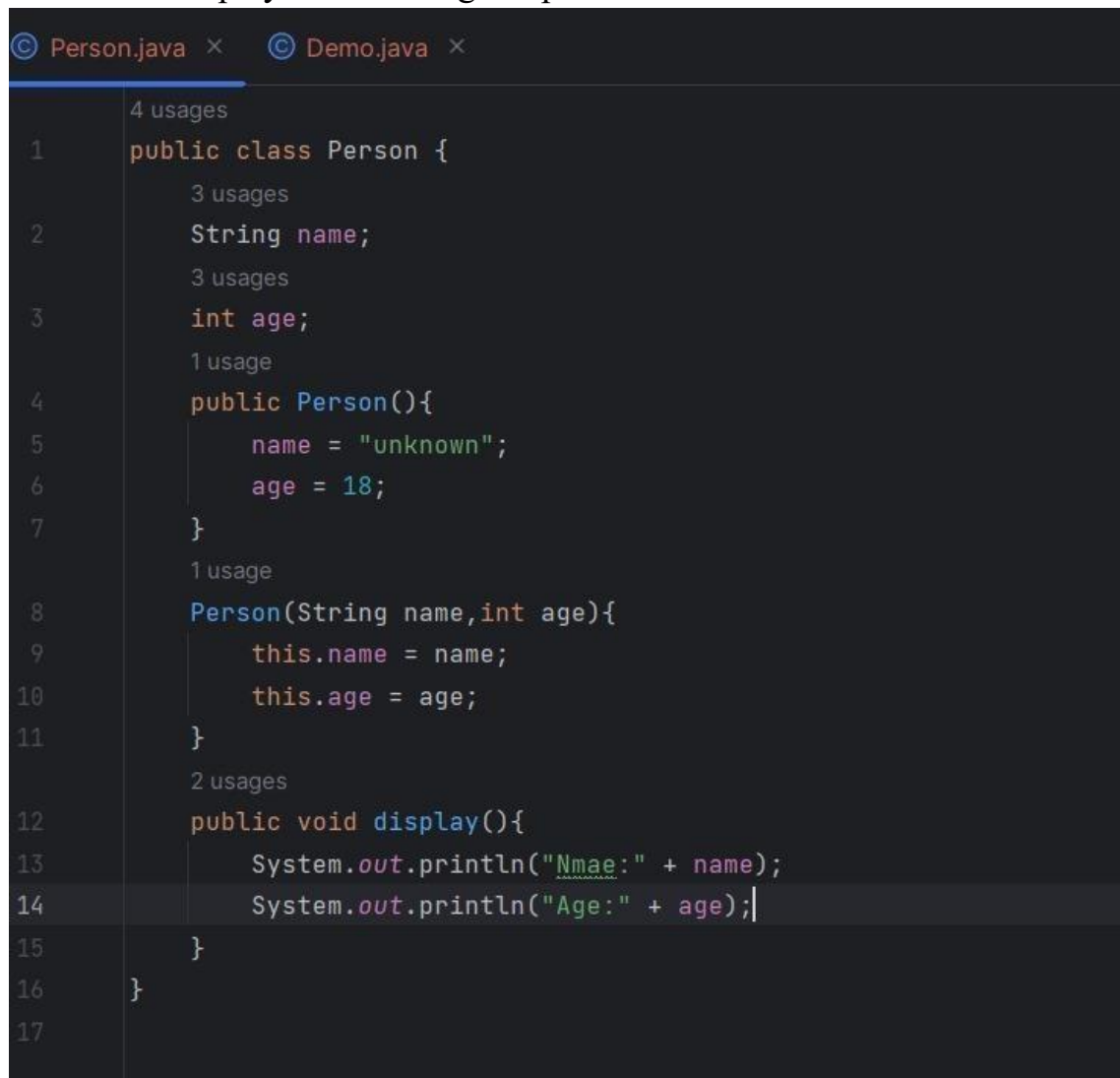


Task 2 Solutions

1.1) Create a class Person with properties (name and age) with following features.

- a. Default age of person should be 18;
- b. A person object can be initialized with name and age;
- c. Method to display name and age of person



```
Person.java x Demo.java x
4 usages
1 public class Person {
    3 usages
2     String name;
    3 usages
3     int age;
    1 usage
4     public Person(){
5         name = "unknown";
6         age = 18;
7     }
    1 usage
8     Person(String name,int age){
9         this.name = name;
10        this.age = age;
11    }
    2 usages
12    public void display(){
13        System.out.println("Name:" + name);
14        System.out.println("Age:" + age);
15    }
16 }
17
```

```
Person.java  Demo.java x
public class Demo{
    public static void main(String[] args) {
        Person p1 = new Person();
        p1.display();
        Person p2 = new Person( name: "sam", age: 23);
        p2.display();
    }
}
```

The output:

```

:
"C:\Program Files\Java\jdk-17.0.12\bin\java.exe" "-javaagent:C:\Users\manju\AppData
Nmae:unknown
Age:18
Nmae:sam
Age:23

Process finished with exit code 0
```

1.2). Create class Product (pid, price, quantity) with parameterized constructor.

Create a main function in different class (say ProductMain) and perform following task:

- Accept five product information from user and store in an array
- Find Pid of the product with the highest price.
- Create method (with array of product's object as argument) in ProductMain class to calculate and return the total amount spent on all products. (amount spent on single product price of product quantity of product)

```
Product.java x ProductMain.java
4 usages
1 class Product {
    3 usages
2     int pid;
    5 usages
3     double price;
    2 usages
4     int quantity;
    1 usage
5     Product(int pid, double price, int quantity) {
6         this.pid = pid;
7         this.price = price;
8         this.quantity = quantity;
9     }
10 }
1
```

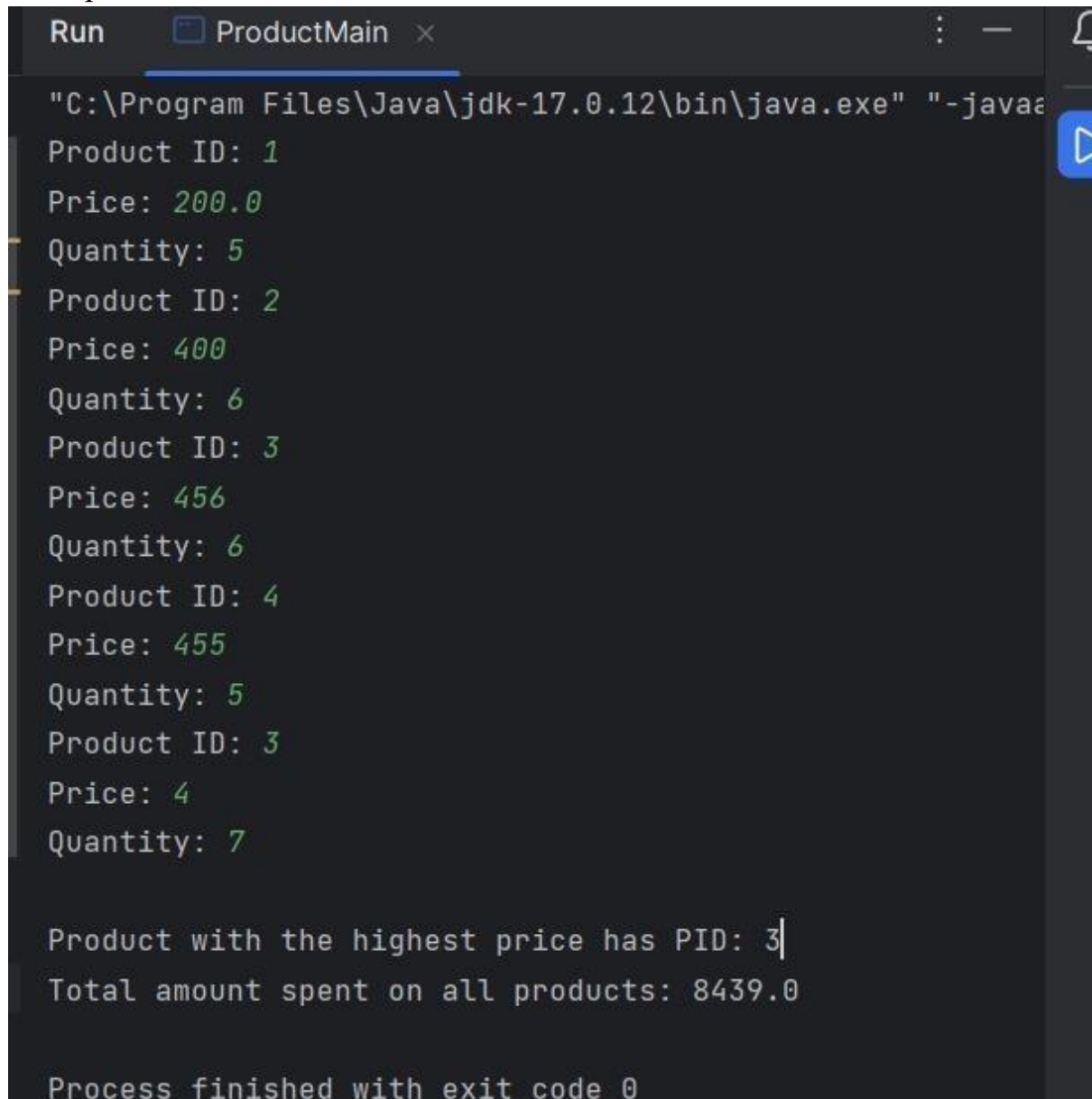
```
ProductMain.java
import java.util.Scanner;
public class ProductMain {
    1 usage
    @ public static double calculateTotalAmount(Product[] products) {
        double total = 0;
        for (int i = 0; i < products.length; i++) {
            total += products[i].price * products[i].quantity;
        }
        return total;
    }
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        Product[] products = new Product[5];
        for (int i = 0; i < 5; i++) {
            System.out.print("Product ID: ");
            int pid = sc.nextInt();
            System.out.print("Price: ");
            double price = sc.nextDouble();
            System.out.print("Quantity: ");
            int quantity = sc.nextInt();
            products[i] = new Product(pid, price, quantity);
        }
        double maxPrice = products[0].price;
        int maxPid = products[0].pid;
        for (int i = 1; i < products.length; i++) {
            if (products[i].price > maxPrice) {
                maxPrice = products[i].price;
                maxPid = products[i].pid;
            }
        }
    }
}
ution > src > ProductMain
Stories
for IPS Office...
```

```

    }
}
System.out.println("Product with the highest price has PID: " + maxPid);
double totalAmount = calculateTotalAmount(products);
System.out.println("Total amount spent on all products: " + totalAmount);
}
}

```

The output is:



```

Run ProductMain x
"C:\Program Files\Java\jdk-17.0.12\bin\java.exe" "-javaa
Product ID: 1
Price: 200.0
Quantity: 5
Product ID: 2
Price: 400
Quantity: 6
Product ID: 3
Price: 456
Quantity: 6
Product ID: 4
Price: 455
Quantity: 5
Product ID: 3
Price: 4
Quantity: 7

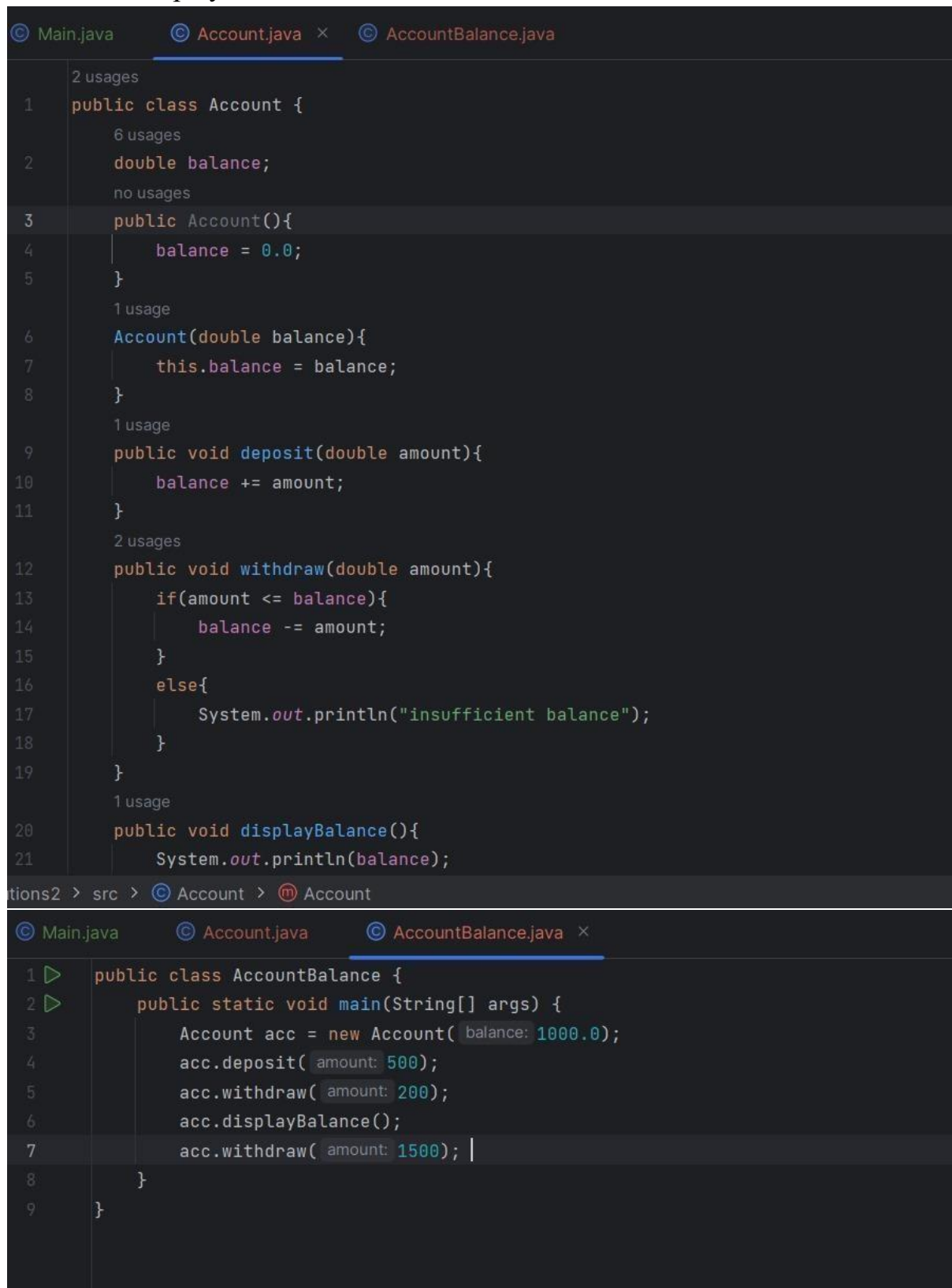
Product with the highest price has PID: 3
Total amount spent on all products: 8439.0

Process finished with exit code 0

```

1.3) Create Class Account with data member as Balance. Create two constructors (no argument, and with argument) and perform following task

- a. method to deposit the amount to the account.
- b. method to withdraw the amount from the account.
- c. method to display the Balance



```

Main.java  Account.java  AccountBalance.java
1  public class Account {
    double balance;
2  public Account(){
    balance = 0.0;
3  }
    Account(double balance){
    this.balance = balance;
4  }
    public void deposit(double amount){
    balance += amount;
5  }
    public void withdraw(double amount){
    if(amount <= balance){
    balance -= amount;
6  }
    else{
    System.out.println("insufficient balance");
7  }
    }
    public void displayBalance(){
    System.out.println(balance);
8  }
}

tions2 > src > Account > Account

Main.java  Account.java  AccountBalance.java
1  public class AccountBalance {
2  public static void main(String[] args) {
    Account acc = new Account( balance: 1000.0);
    acc.deposit( amount: 500);
    acc.withdraw( amount: 200);
    acc.displayBalance();
    acc.withdraw( amount: 1500); |
3  }
4  }
5  }
```

The output is:

```
"C:\Program Files\Java\jdk-17.0.12\bin\java.exe" "-javaagent:C:\Users\  
1300.0  
insufficient balance  
  
Process finished with exit code 0
```

1.4) Define a base class Person with attributes name and age.

Create a subclass Employee that inherits from Person and adds attributes like employeeID and salary.

Use the super keyword to initialize the Person attributes in the Employee constructor.

```
1 usage 1 inheritor  
1 @ public class Person {  
2     2 usages  
2     String name;  
3     2 usages  
3     int age;  
4     1 usage  
4     Person(String name,int age){  
5         this.name = name;  
6         this.age = age;  
7     }  
8  
9 }  
10
```

```
Solutions2 ▾ master ▾
Main.java Account.java Person.java Employee.java × EmployeeMain.java AccountBalance.java

2 usages
1 public class Employee extends Person {
    2 usages
2     int employeeID;
    2 usages
3     double salary;
4
    1 usage
5     Employee(String name, int age, int employeeID, double salary) {
6         super(name, age); // 🖱️ this calls Person(name, age)
7         this.employeeID = employeeID;
8         this.salary = salary;
9     }
10
    1 usage
11     public void display() {
12         System.out.println("Name: " + name);
13         System.out.println("Age: " + age);
14         System.out.println("Employee ID: " + employeeID);
15         System.out.println("Salary: " + salary);
16     }
17 }
18

Solutions2 > src > Employee
the best models ChatGPT can make mistakes. Check important info
```

```
public class EmployeeMain {
    💡 public static void main(String[] args) {
        Employee emp = new Employee( name: "Alice", age: 30, employeeID: 101, salary: 50000.0);
        emp.display();
    }
}
```

The output is:

```
Run EmployeeMain x
" C:\Program Files\Java\jdk-17.0.12\bin\java.exe" "-javaagent:C:\Users\manju\AppData\Local\Temp\jvarkit\jvarkit.jar" -Djvarkit.args=Name: Alice
Name: Alice
Age: 30
Employee ID: 101
Salary: 50000.0
Process finished with exit code 0
```