Al Imam Mohammad Ibn Saud Islamic University
College of Computer and Information Sciences
Computer Science Department

CS151 Final Project

CS151 - Object-Oriented Programming

Second semester 1440-1441

Section: 372

Group: 2	
Group members' names	**Students IDs**
Sadeem Faisal Alqahtani	440021429
Sarah Khalid Aloraidi	440023365
Aljowharah Turki Aldawood	440022805

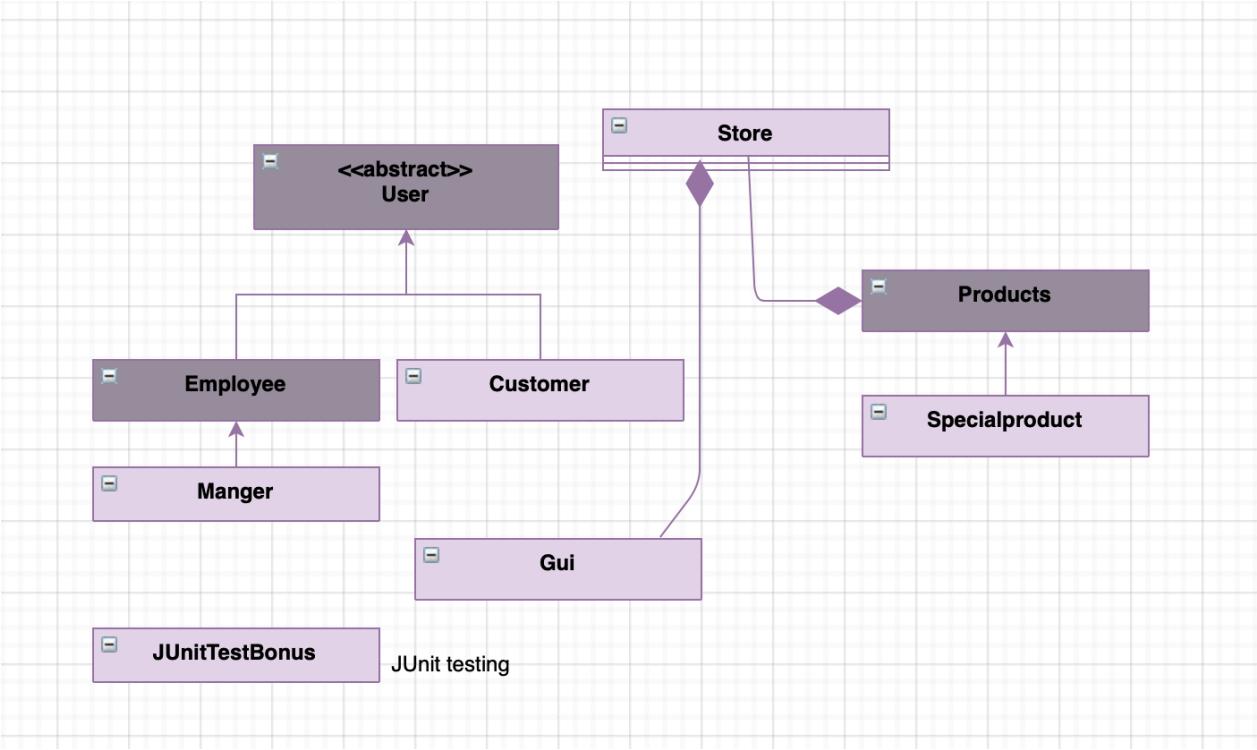
Contents

UML 3
Inheritance issues 4
Output 5
Screenshots 5
Additional Features (if any) 11
Screenshots 11

UML

Note:

- Put your correct UML for each class in your work.
- Your UML shows classes name with all relations only, without showing attribute and method sections.
- You can use MS Word directly or any tool for drawing.



Inheritance issues

One of the inheritance issues we encountered was that in the class user, we defined an attribute called ID which starts at 1000. Since ID is inherited into class employee and customer it was similar to all objects created as shown below :

```
New Customer : Sara(1000) visits 0 time(s).
New Customer : Sadeem(1000) visits 0 time(s).
(Manger)Employee : Aljohra(1000) has $15000.0
Employee : Noura(1000) has $10000.0
```

To prevent this repetition from happening we created an attribute in class user called ID counter which starts at 0. We made this counter static and is incremented and added to the ID once and object is created. The correct ID is shown below :

```
New Customer : Sara(1000) visits 0 time(s).
New Customer : Sadeem(1001) visits 0 time(s).
(Manger)Employee : Aljohra(1002) has $15000.0
Employee : Noura(1003) has $10000.0
```

Output

Put the outputs for each operation in your classes, e.g., Test1: add salary for an employee.

- Test1 – Test 7

Screenshots

Test 1 : Create objects

```
Welcome to SAS Store

It contains 15 product(s)
Product: Milk(1) costs $2.5
Product: Egg(2) costs $10.75
Product: Cake(3) costs $30.0
Product: Apple(4) costs $4.99
Product: Banana(5) costs $6.65
Product: Candy(6) costs $8.7
Product: Backpack(7) costs $10.65
Product: Pencil(8) costs $2.0
Product: Pen(9) costs $1.7
Product: Orange(10) costs $5.0
Product: Watermelon(11) costs $3.99
Product: Eraser(12) costs $0.5
Product: Headphone(13) costs $15.0
Product: Phone(14) costs $1300.0
Product: Ipad(15) costs $1499.99

The users are :
New Customer : Sara(1000) visits 0 time(s).
New Customer : Sadeem(1001) visits 0 time(s).
(Manger)Employee : Aljohra(1002) has $15000.0
Employee : Noura(1003) has $10000.0
```

Test 2 : Update the employee salary.

```
Increase all salaries with 15%
(Manger)Employee : Aljohra(1002) has $17250.0
Employee : Noura(1003) has $11500.0
```

Test 3 : Three correct buy operation

```
Store before 3 correct buy opeartion :SAS Store
It contains 15 product(s)
Product: Milk(1) costs $2.5
Product: Egg(2) costs $10.75
Product: Cake(3) costs $30.0
Product: Apple(4) costs $4.99
Product: Banana(5) costs $6.65
Product: Candy(6) costs $8.7
Product: Backpack(7) costs $10.65
Product: Pencil(8) costs $2.0
Product: Pen(9) costs $1.7
Product: Orange(10) costs $5.0
Product: Watermelon(11) costs $3.99
Product: Eraser(12) costs $0.5
Product: Headphone(13) costs $15.0
Product: Phone(14) costs $1300.0
Product: Ipad(15) costs $1499.99


WELCOME TO * SAS Store *
Receipt Number is : 1
Products Purchased :
Milk(1) costs $2.5
Banana(5) costs $6.65
Cake(3) costs $30.0
Total Price : $39.15
New Customer : Sara(1000) visits 1 time(s).

Store after the first correct buy opeartion :SAS Store

It contains 12 product(s)
Product: Egg(2) costs $10.75
Product: Apple(4) costs $4.99
Product: Candy(6) costs $8.7
Product: Backpack(7) costs $10.65
Product: Pencil(8) costs $2.0
Product: Pen(9) costs $1.7
Product: Orange(10) costs $5.0
Product: Watermelon(11) costs $3.99
Product: Eraser(12) costs $0.5
Product: Headphone(13) costs $15.0
Product: Phone(14) costs $1300.0
Product: Ipad(15) costs $1499.99


WELCOME TO * SAS Store *
Receipt Number is : 2
Products Purchased :
Apple(4) costs $4.99
Watermelon(11) costs $3.99
Candy(6) costs $8.7
Total Price : $17.68
New Customer : Sara(1000) visits 2 time(s).
```

```
Store after the second correct buy opeartion :SAS Store

It contains 9 product(s)
Product: Egg(2) costs $10.75
Product: Backpack(7) costs $10.65
Product: Pencil(8) costs $2.0
Product: Pen(9) costs $1.7
Product: Orange(10) costs $5.0
Product: Eraser(12) costs $0.5
Product: Headphone(13) costs $15.0
Product: Phone(14) costs $1300.0
Product: Ipad(15) costs $1499.99

WELCOME TO * SAS Store *
Receipt Number is : 3
Products Purchased :
Egg(2) costs $10.75
Total Price : (after discount)$8.06
Special Customer : Sara(1000) visits 3 time(s).
------------------------------------------------

Store after the third correct buy opeartion :SAS Store

It contains 8 product(s)
Product: Backpack(7) costs $10.65
Product: Pencil(8) costs $2.0
Product: Pen(9) costs $1.7
Product: Orange(10) costs $5.0
Product: Eraser(12) costs $0.5
Product: Headphone(13) costs $15.0
Product: Phone(14) costs $1300.0
Product: Ipad(15) costs $1499.99

The customers are :
Special Customer : Sara(1000) visits 3 time(s).
New Customer : Sadeem(1001) visits 0 time(s).
```

Test 4 : A wrong buy operation.

```
Store before a wrong buy opeartion :SAS Store

It contains 8 product(s)
Product: Backpack(7) costs $10.65
Product: Pencil(8) costs $2.0
Product: Pen(9) costs $1.7
Product: Orange(10) costs $5.0
Product: Eraser(12) costs $0.5
Product: Headphone(13) costs $15.0
Product: Phone(14) costs $1300.0
Product: Ipad(15) costs $1499.99

The customer needs :
Apple(4) costs $4.99
Orange(10) costs $5.0
Sorry the product Apple doesn't exist in the store
---
Store after a wrong buy opeartion :SAS Store

It contains 8 product(s)
Product: Backpack(7) costs $10.65
Product: Pencil(8) costs $2.0
Product: Pen(9) costs $1.7
Product: Orange(10) costs $5.0
Product: Eraser(12) costs $0.5
Product: Headphone(13) costs $15.0
Product: Phone(14) costs $1300.0
Product: Ipad(15) costs $1499.99

The users are :
Special Customer : Sara(1000) visits 3 time(s).
New Customer : Sadeem(1001) visits 0 time(s).
(Manger)Employee : Aljohra(1002) has $17250.0
Employee : Noura(1003) has $11500.0
```

Test 5 : Add product to the store

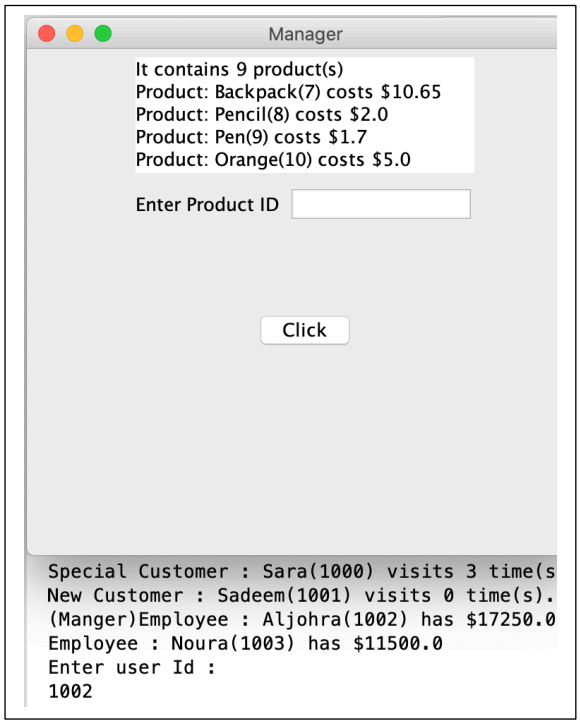
```
store before add product:SAS Store
It contains 8 product(s)
Product: Backpack(7) costs $10.65
Product: Pencil(8) costs $2.0
Product: Pen(9) costs $1.7
Product: Orange(10) costs $5.0
Product: Eraser(12) costs $0.5
Product: Headphone(13) costs $15.0
Product: Phone(14) costs $1300.0
Product: Ipad(15) costs $1499.99
```

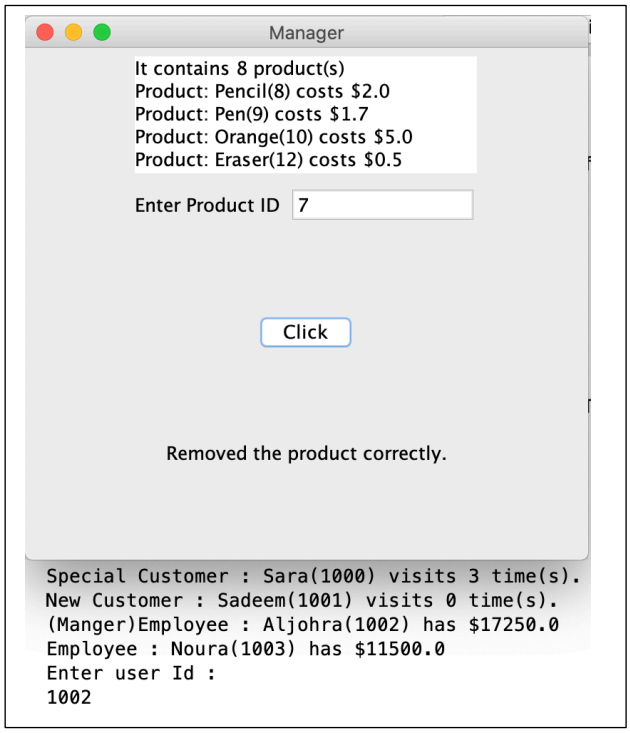
Test 6 : Two lists of special products.

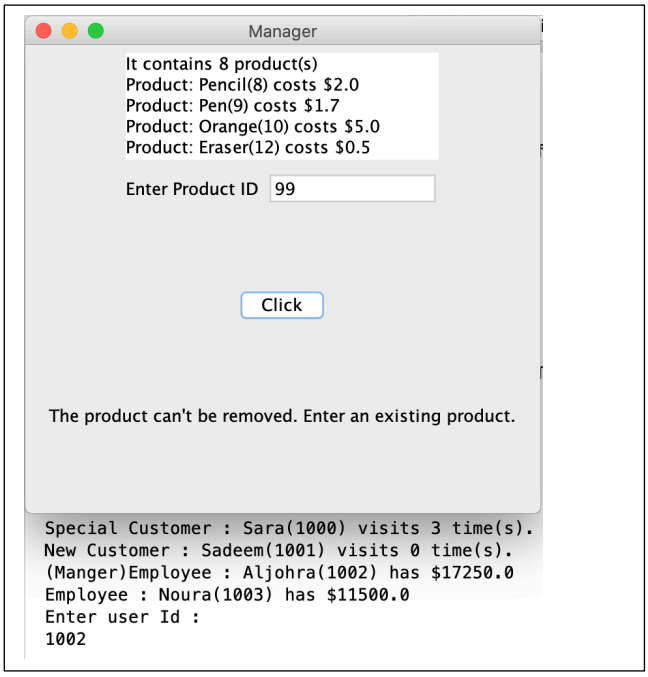
```
The list of the first special products :
Products in the offer Back to School:
- Backpack(7) costs $10.65
- Pencil(8) costs $2.0
- Pen(9) costs $1.7
- Eraser(12) costs $0.5


( in Backpack ) the relation Products in the offer Back to School:
- Backpack(7) costs $10.65
- Pencil(8) costs $2.0
- Pen(9) costs $1.7
- Eraser(12) costs $0.5

( in Pencil ) the relation Products in the offer Back to School:
- Backpack(7) costs $10.65
- Pencil(8) costs $2.0
- Pen(9) costs $1.7
- Eraser(12) costs $0.5

( in Pen ) the relation Products in the offer Back to School:
- Backpack(7) costs $10.65
- Pencil(8) costs $2.0
- Pen(9) costs $1.7
- Eraser(12) costs $0.5

( in Eraser ) the relation Products in the offer Back to School:
- Backpack(7) costs $10.65
- Pencil(8) costs $2.0
- Pen(9) costs $1.7
- Eraser(12) costs $0.5

The list of the second special products :

Products in the offer Technology SALE:
- Headphone(13) costs $15.0
- Phone(14) costs $1300.0
- Ipad(15) costs $1499.99

( in Headphone ) the relation Products in the offer Technology SALE:
- Headphone(13) costs $15.0
- Phone(14) costs $1300.0
- Ipad(15) costs $1499.99

( in Phone ) the relation Products in the offer Technology SALE:
- Headphone(13) costs $15.0
- Phone(14) costs $1300.0
- Ipad(15) costs $1499.99

( in Ipad ) the relation Products in the offer Technology SALE:
- Headphone(13) costs $15.0
- Phone(14) costs $1300.0
- Ipad(15) costs $1499.99
```

Test 7 : Remove an existing product for manager only.

```
Special Customer : Sara(1000) visits 3 time(s).
New Customer : Sadeem(1001) visits 0 time(s).
(Manger)Employee : Aljohra(1002) has $17250.0
Employee : Noura(1003) has $11500.0
Enter user Id :
1000
Can't remove a product. Only Mangers can remove the product
```







Additional Features (if any)

Describe the features that you chose. Put the output for each feature. Make sure, if your description is not clear, you cannot get the mark.

- **Feature 1:** File storage, using a text file (.txt) to initialize all objects instead of initializing manually in the main.
- **Feature 2:** Using input validation for users' ID.
- **Feature 3:** Using Junit to test isExist method of output.

Screenshots

Feature one :

Code :

```
public static void main(String[] args) throws Exception
{



   Scanner input = new Scanner(System.in);
     input = new Scanner(Paths.get("Products.txt"));
     while (input.hasNext())
     {  Store store = new Store(input.nextLine());
        Products p1=new Products (input.next(),input.nextDouble());
        Products p2=new Products (input.next(),input.nextDouble());
        Products p3=new Products (input.next(),input.nextDouble());
        Products p4=new Products (input.next(),input.nextDouble());
        Products p5=new Specialproduct (input.next(),input.nextDouble());
        Products p6=new Products (input.next(),input.nextDouble());
        Products p7=new Products (input.next(),input.nextDouble());
        Products p8=new Products (input.next(),input.nextDouble());
        Products p9=new Products (input.next(),input.nextDouble());
        Products p10=new Products (input.next(),input.nextDouble());
        Products p11=new Products (input.next(),input.nextDouble());
        Products p12=new Products (input.next(),input.nextDouble());
        Products p13=new Products (input.next(),input.nextDouble());
        Products p14=new Products (input.next(),input.nextDouble());
        Products p15=new Products (input.next(),input.nextDouble());
        Customer C1=new Customer(input.next());
        Customer C2=new Customer(input.next());
        Employee E1 = new Manger(input.next(),input.nextDouble());
        Employee E2 = new Employee(input.next(),input.nextDouble());
         closeFile1();

}}
 public static void closeFile1() {
     if (input != null) {
         input.close();
```

Feature one continued.

Output :

```
It contains 15 product(s)
Product: Milk(1) costs $2.5
Product: Egg(2) costs $10.75
Product: Cake(3) costs $30.0
Product: Apple(4) costs $4.99
Product: Banana(5) costs $6.65
Product: Candy(6) costs $8.7
Product: Backpack(7) costs $10.65
Product: Pencil(8) costs $2.0
Product: Pen(9) costs $1.7
Product: Orange(10) costs $5.0
Product: Watermelon(11) costs $3.99
Product: Eraser(12) costs $0.5
Product: Headphone(13) costs $15.0
Product: Phone(14) costs $1300.0
Product: Ipad(15) costs $1499.99
```

Feature 2 :

Code :

```
System.out.println("Enter user Number Of Id :");
  int ID=input.nextInt();
  while (ID<1000){
      System.out.println("This can't be Number of ID");
       ID=input.nextInt();}


  if(Manger.Check(ID))
    new Gui(store);
  else
     System.out.println("Can't remove a product. Only Mangers can "
             + "remove the product");
```

Output :

```
Enter user Number Of Id :
102
Incorrect ID, Please try again :
1004
Can't remove a product. Only Mangers can remove the product
```

Feature three :

```
Source  History
 8  import org.junit.Before;
 9  import org.junit.Test;
10  import static org.junit.Assert.*;
11  import project.Store;
12
13  public class JunitTestBonus {
14      Store a;
15
16      public JunitTestBonus() {
17      }
18
19
20      @Before
21      public void setUp() {
22          a=new Store("SAS");
23      }
24
25      @Test
26      public void test() {
27
28
29       boolean b=a.isExist(99);
30       assertEquals(true,b);
31
32
33
34      }
35
36  }
37
```

JunitTestBonus > test

Notifications | Test Results | Output - projectFinalEdit (test)

Ant suite

Tests passed: 0.00 %

No test passed, 1 test failed. (0.022 s)

JunitTestBonus Failed

```
[TestNG] Running:
  Command line suite
===============================================
Command line suite
Total tests run: 1, Failures: 1, Skips: 0
===============================================
```

```
Source  History
 8  import org.junit.Before;
 9  import org.junit.Test;
10  import static org.junit.Assert.*;
11  import project.Store;
12
13  public class JunitTestBonus {
14      Store a;
15
16      public JunitTestBonus() {
17      }
18
19
20      @Before
21      public void setUp() {
22          a=new Store("SAS");
23      }
24
25      @Test
26      public void test() {
27
28
29       boolean b=a.isExist(99);
30       assertEquals(false,b);
31
32
33
34      }
35
36  }
37
```

JunitTestBonus > test

Notifications | Test Results | Output - projectFinalEdit (test)

Ant suite

Tests passed: 100.00 %

The test passed. (0.03 s)

```
[TestNG] Running:
  Command line suite
===============================================
Command line suite
Total tests run: 1, Failures: 0, Skips: 0
===============================================
```