

JAVA 编程进阶上机报告



学 院 智能与计算学部

专 业 软件工程

班 级 6 班

学 号 3018216298

姓 名 米思成

Lab 1：计算机销售系统的设计

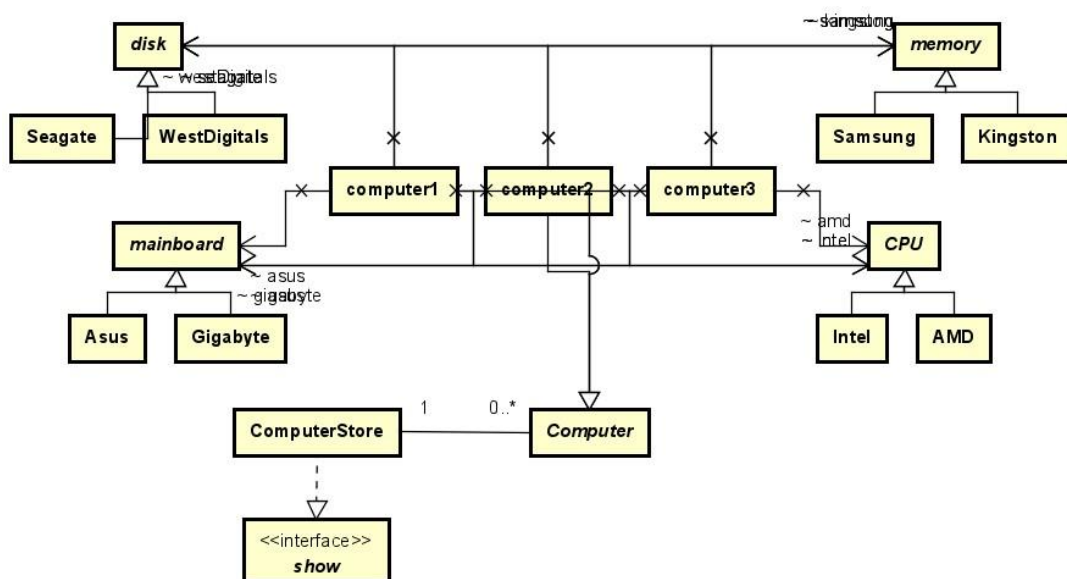
一、实验要求

某计算机组装公司主要销售各类组装计算机，计算机一般由 CPU、内存、主板、硬盘等组件构成。

具体要求：

- 1) 针对每个组件的每个品牌，设计一个类，并画成整体的类图
- 2) 设计计算机类 (Computer.java)，由上述四类组件组装而成，包括计算机的名称、计算机的描述（包括各个组件名）以及总价格等
- 3) 设计计算机销售主类 (ComputerStore.java)，包括 3 个由不同组件组装在一起的计算机实例，可实现计算机商品一览表，可展示每台计算机的描述、价格、工作等。
- 4) 设计时基于抽象类和接口，要尽可能的实现高内聚、低耦合。

二、整体类图



三、源代码

CPU.java

```
public abstract class CPU {  
    public String name;  
    public int coreNum;  
    public double price;  
  
    public CPU(String name, int coreNum, double price){  
        this.name = name;  
        this.coreNum = coreNum;  
        this.price = price;  
    }  
  
    abstract void work();  
  
}
```

```
class Intel extends CPU{  
  
    public Intel() {  
        super("Intel", 6, 2000);  
    }  
  
    @Override  
    void work() {  
        System.out.print("Intel_CPU work");  
    }  
  
}
```

```
class AMD extends CPU{  
  
    public AMD() {  
        super("AMD", 4, 1000);  
    }  
  
    @Override  
    void work() {  
        System.out.print("AMD_CPU work");  
    }  
  
}
```

```
}
```

memory.java

```
public abstract class memory {  
    public String name;  
    public String volume;  
    public double price;  
  
    public memory(String name, String volume, double price){  
        this.name = name;  
        this.volume = volume;  
        this.price = price;  
    }  
  
    abstract void work();  
}
```

```
class Samsung extends memory{  
  
    public Samsung() {  
        super("Samsung", "8G", 400);  
    }  
  
    @Override  
    void work() {  
        System.out.print("Samsung_memory work");  
    }  
}
```

```
class Kingston extends memory{  
  
    public Kingston() {  
        super("Kingston", "8G", 300);  
    }  
  
    @Override  
    void work() {  
        System.out.print("Kingston_memory work");  
    }  
}
```

```
}  
  
}
```

disk.java

```
public abstract class disk {  
    public String name;  
    public String volume;  
    public double price;  
  
    public disk(String name, String volume, double price){  
        this.name = name;  
        this.volume = volume;  
        this.price = price;  
    }  
  
    abstract void work();  
}
```

```
class Seagate extends disk{  
  
    public Seagate() {  
        super("Seagate", "1TB", 300);  
    }  
  
    @Override  
    void work() {  
        System.out.print("Seagate_disk work");  
    }  
  
}
```

```
class WestDigitals extends disk{  
  
    public WestDigitals() {  
        super("WestDigitals", "2TB", 400);  
    }  
  
    @Override  
    void work() {  
        System.out.print("WestDigitals_disk work");  
    }  
  
}
```

```
}  
  
}
```

mainboard.java

```
public abstract class mainboard {  
    public String name;  
    public String speed;  
    public double price;  
  
    public mainboard(String name, String speed, double price){  
        this.name = name;  
        this.speed = speed;  
        this.price = price;  
    }  
  
    abstract void work();  
}
```

```
class Asus extends mainboard{  
  
    public Asus() {  
        super("Asus", "3600GHZ", 700);  
    }  
  
    @Override  
    void work() {  
        System.out.print("Asus_mainboard work");  
    }  
  
}
```

```
class Gigabyte extends mainboard{  
  
    public Gigabyte() {  
        super("Gigabyte", "3600GHZ", 900);  
    }  
  
    @Override  
    void work() {  
        System.out.print("Gigabyte_mainboard work");  
    }  
}
```

```
}  
  
}
```

Computer.java

```
public abstract class Computer {  
    public String name;  
    public String information;  
    public double price;  
  
    public abstract String setName();  
  
    public abstract String setInformation();  
  
    public abstract double setPrice();  
  
    public abstract void work();  
}
```

```
class computer1 extends Computer{  
  
    CPU intel = new Intel();  
    memory samsung = new Samsung();  
    disk seagate = new Seagate();  
    mainboard asus = new Asus();  
  
    @Override  
    public String setName(){  
        return name = "COMPUTER1";  
    }  
  
    @Override  
    public String setInformation() {  
        return information = "CPU is " + intel.name + ", memory is " +  
samsung.name + ", disk is " + seagate.name + ", mainboard is " + asus.name  
+ ".";  
    }  
  
    @Override  
    public double setPrice() {  
        return price = intel.price + samsung.price + seagate.price +
```

```

asus.price;
    }

    @Override
    public void work() {
        intel.work();
        System.out.print(", ");
        samsung.work();
        System.out.print(", ");
        seagate.work();
        System.out.print(", ");
        asus.work();
        System.out.println("!");
    }
}

```

```

class computer2 extends Computer{

    CPU intel = new Intel();
    memory kingston = new Kingston();
    disk seagate = new Seagate();
    mainboard gigabyte = new Gigabyte();

    @Override
    public String setName(){
        return name = "COMPUTER2";
    }

    @Override
    public String setInformation() {
        return information = "CPU is " + intel.name + ", memory is " +
kingston.name + ", disk is " + seagate.name + ", mainboard is " + gigabyte.name
+ ".";
    }

    @Override
    public double setPrice() {
        return price = intel.price + kingston.price + seagate.price +
gigabyte.price;
    }

    @Override
    public void work() {

```



```

        intel.work();
        System.out.print(", ");
        kingston.work();
        System.out.print(", ");
        seagate.work();
        System.out.print(", ");
        gigabyte.work();
        System.out.println("!");
    }
}

```

```

class computer3 extends Computer{

    CPU amd = new AMD();
    memory samsung = new Samsung();
    disk westDigitals = new WestDigitals();
    mainboard asus = new Asus();

    @Override
    public String setName(){
        return name = "COMPUTER3";
    }

    @Override
    public String setInformation() {
        return information = "CPU is " + amd.name + ", memory is " +
samsung.name + ", disk is " + westDigitals.name + ", mainboard is " + asus.name
+ ".";
    }

    @Override
    public double setPrice() {
        return price = amd.price + samsung.price + westDigitals.price +
asus.price;
    }

    @Override
    public void work() {
        amd.work();
        System.out.print(", ");
        samsung.work();
        System.out.print(", ");
        westDigitals.work();
    }
}

```

```

        System.out.print(", ");
        asus.work();
        System.out.println("!");
    }
}

```

show.java

```

public interface show {

    void display();

}

```

ComputerStore.java

```

package lab1_computer;

public class ComputerStore implements show{

    @Override
    public void display() {
        System.out.println("Welcome to our computer store! We have 3 kinds
of computers now.");
        System.out.println();

        Computer computer1 = new computer1();
        System.out.println(computer1.setName() + ":");
        System.out.println("Information: " + computer1.setInformation());
        System.out.println("Price: " + computer1.setPrice() + " RMB");
        System.out.print("Does it work? ");
        computer1.work();
        System.out.println();

        Computer computer2 = new computer2();
        System.out.println(computer2.setName() + ":");
        System.out.println("Information: " + computer2.setInformation());
        System.out.println("Price: " + computer2.setPrice() + " RMB");
        System.out.print("Does it work? ");
        computer2.work();
        System.out.println();

        Computer computer3 = new computer3();
        System.out.println(computer3.setName() + ":");
    }
}

```

```

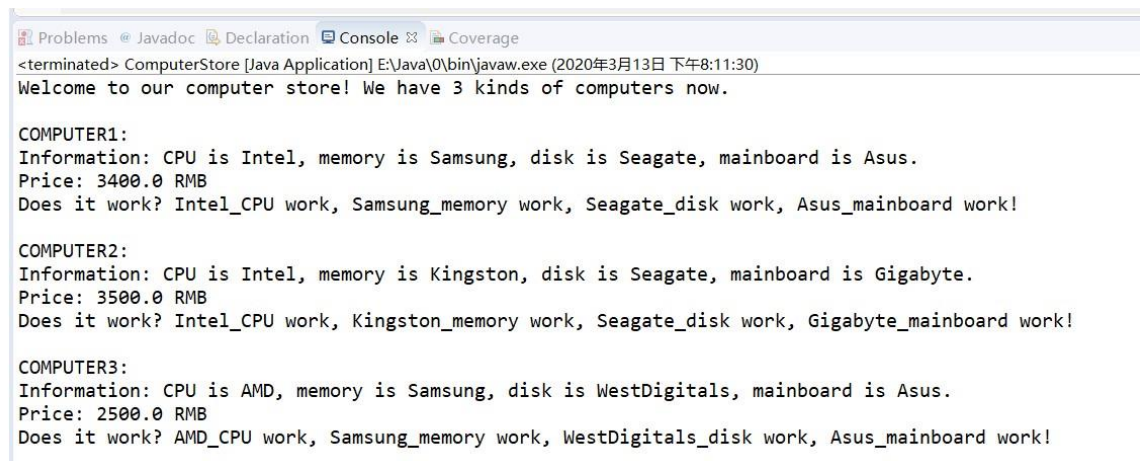
        System.out.println("Information: " + computer3.setInformation());
        System.out.println("Price: " + computer3.setPrice() + " RMB");
        System.out.print("Does it work? ");
        computer3.work();

    }

    public static void main(String[] args) {
        ComputerStore store = new ComputerStore();
        store.display();
    }
}

```

四、实验结果



The screenshot shows a Java IDE console window with the following output:

```

<terminated> ComputerStore [Java Application] E:\Java\0\bin\javaw.exe (2020年3月13日 下午8:11:30)
Welcome to our computer store! We have 3 kinds of computers now.

COMPUTER1:
Information: CPU is Intel, memory is Samsung, disk is Seagate, mainboard is Asus.
Price: 3400.0 RMB
Does it work? Intel_CPU work, Samsung_memory work, Seagate_disk work, Asus_mainboard work!

COMPUTER2:
Information: CPU is Intel, memory is Kingston, disk is Seagate, mainboard is Gigabyte.
Price: 3500.0 RMB
Does it work? Intel_CPU work, Kingston_memory work, Seagate_disk work, Gigabyte_mainboard work!

COMPUTER3:
Information: CPU is AMD, memory is Samsung, disk is WestDigitals, mainboard is Asus.
Price: 2500.0 RMB
Does it work? AMD_CPU work, Samsung_memory work, WestDigitals_disk work, Asus_mainboard work!

```