JAVA 과제 #3

201810754 조윤직

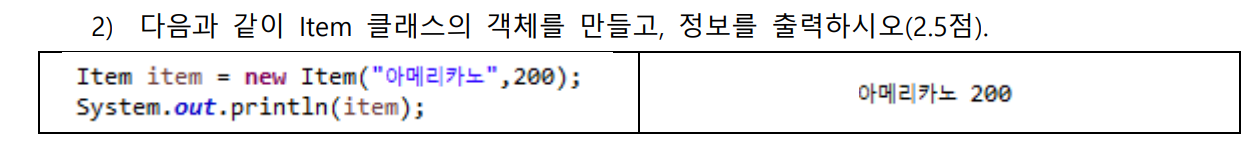
제출일: 2019. 11. 04

카페의 주문 관리 시스템

문제정의 : Item, Coffee, Beer, Client, Manager 클래스 완성,

문제 분석 및 해결책 :

-단순히 멤버 변수만을 set하거나 get하는 메소드들은 보고서에 포함 하지 않았습니다.



소스코드:

Item.java

public class Item {

private String Iname;

private int price;

private int sellcount;

private int totalprice=0;

public Item() {

}

public Item(String iname, int price) {

super();

Iname = iname;

this.price = price;

}

*@Override*

public String toString() {

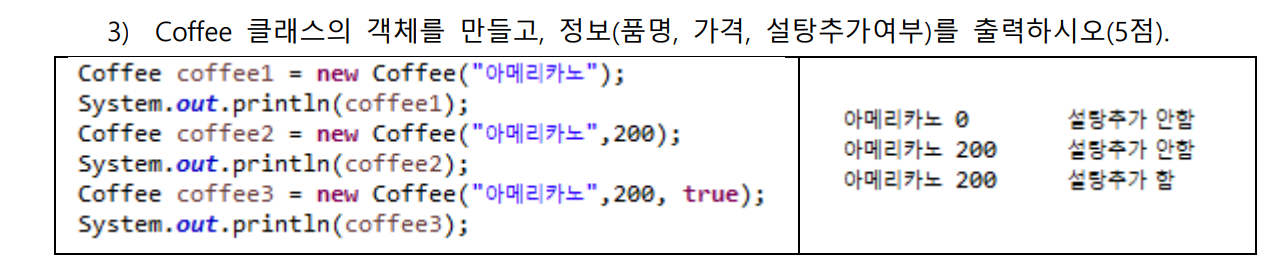
return Iname + "\t" + price;

}

}

필요한 생성자 (string, int)정의, 기본생성자 정의 , toString 오버라이드

Item.java



Coffee 클래스,- 생성자, item의 자식클래스, 멤버변수 sul(설탕 추가 여부)

Coffee.java

package konkuk.yjcho;

public class Coffee extends Item {

private boolean sul;

public Coffee(String iname) {

this(iname,0,false);

}

public Coffee(String iname,boolean sul) {

this(iname,0,false);

this.sul=sul;

}

public Coffee(String iname, int price, boolean sul) {

super(iname, price);

this.sul = sul;

}

public Coffee(String iname, int price) {

this(iname,price,false);

}

*@Override*

public String toString() {

String ss;

if(sul)

ss="설탕추가 함";

else

ss="설탕추가 안함";

return super.toString()+"\t"+ss;

}

public boolean isSul() {

return sul;

}

public void setSul(boolean sul) {

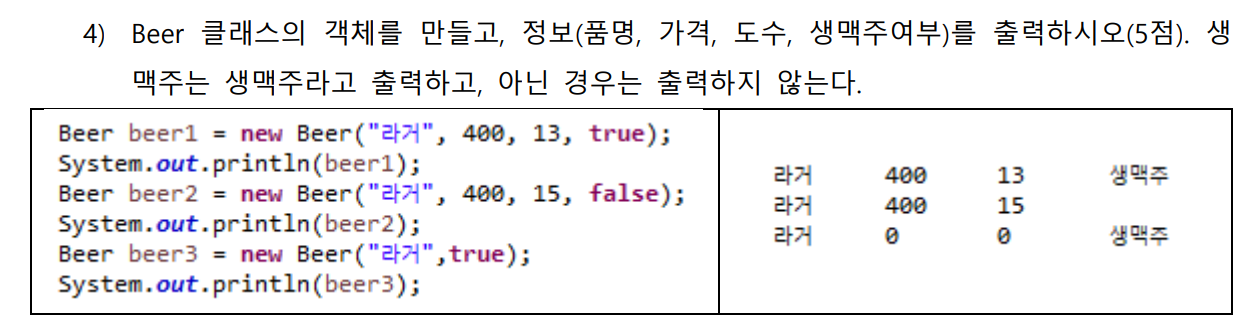
this.sul = sul;

}

}

예제에 있는 생성자 (String,int ,boolean),(String,int),(String) 추가

toString 함수 - item의 tostring 함수에 이어서 설탕 추가 여부 출력 –(String, Boolean)은 2)에서는 나오지 않으나 8)에서 나와 추가



Beer클래스- item에서 도수, 생맥주여부 멤버 변수 추가, 출력함수 재정의

Beer.java

package konkuk.yjcho;

public class Beer extends Item {

private int dosoo;

private boolean fresh;

public Beer(String iname, int price,int dosoo,boolean fresh) {

super(iname, price);

this.dosoo=dosoo;

this.fresh = fresh;

}

public Beer(String iname, boolean fresh) {

// **TODO** Auto-generated constructor stub

this(iname, 0, 0, fresh);

}

*@Override*

public String toString() {

// **TODO** Auto-generated method stub

String ss;

if(fresh)

ss="생맥주";

else

ss="";

return super.toString()+"\t"+this.dosoo+"\t"+ss;

}

public boolean isFresh() {

return fresh;

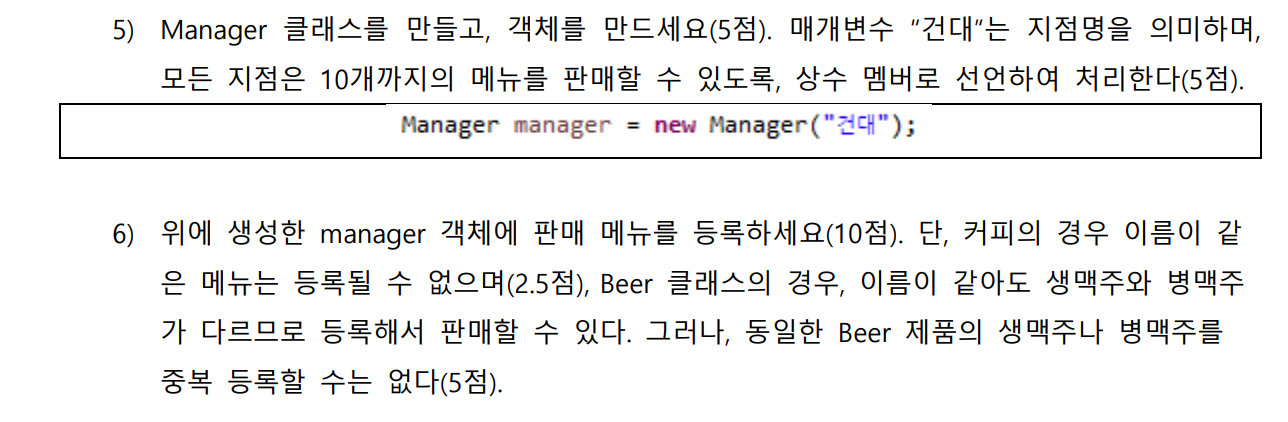
}

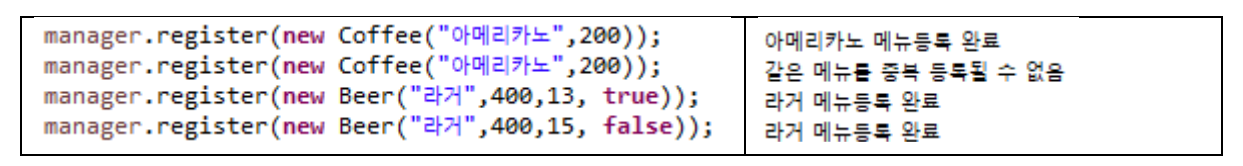
public void setFresh(boolean fresh) {

this.fresh = fresh;

}

}





Manager.java

package konkuk.yjcho;

public class Manager {

private String Mname;

static final int ***maxcount***=10;//최대 치

private int count;

private Item[] Ilist;

private int total;

public Manager(String mname) {//String 하나를 받아 객체 생성

super();

total=0; //총 매출 0;

Mname = mname;

Ilist=new Item[***maxcount***]; //최대로 팔수 있는 물품 만큼 체크

this.count=0; //팔고있는 종류 0개

}

public void register(Item item) {

if(count>=***maxcount***) {//혹시 등뢱된 갯수가 최대 갯수이상이라면

System.***out***.println("더 이상 등록 할수 없음..");

return;

}

if(item instanceof Coffee) { //아이템이 커피이면, 커피이면 중복 아니면 등록

for(int i=0;i<count;i++) {

if(Ilist[i] instanceof Coffee) {

if(Ilist[i].getIname().equals(item.getIname())) {

System.***out***.println("같은 메뉴를 중복 등록될 수 없음");

return;

}

}

}

Ilist[count++]=item;//아이템 등록

System.***out***.println(item.getIname()+" 메뉴등록 완료");

}

else if(item instanceof Beer) {//만약에 맥주라면

for(int i=0;i<count;i++) {

if(Ilist[i] instanceof Beer) {

if(Ilist[i].getIname().equals(item.getIname())) {//이름이 같다면

Beer b1 =(Beer)Ilist[i];

Beer b2 =(Beer)item;

if(b1.isFresh()==b2.isFresh()) {//생맥주 여부가 같다면 등록 x

System.***out***.println("같은 메뉴를 등록될수 없음");

return;

}

}

}

}

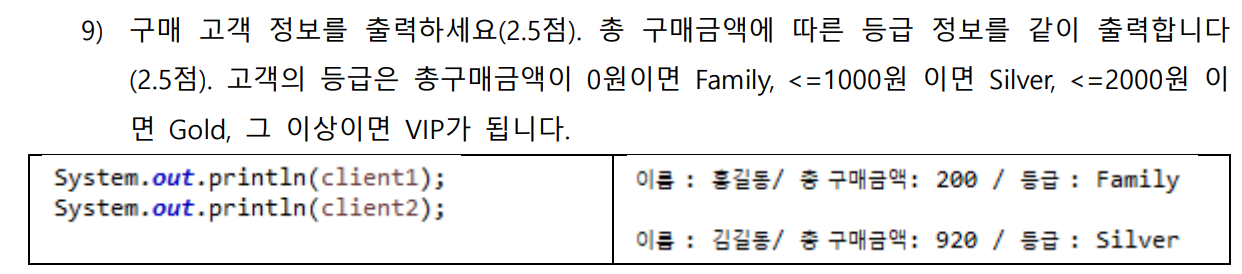
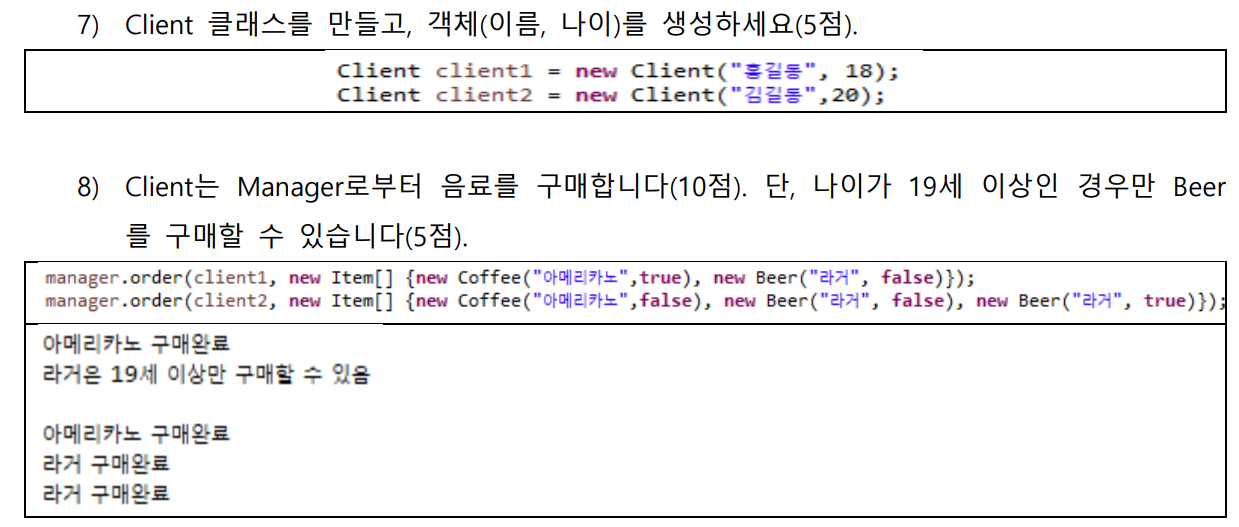
Ilist[count++]=item;

System.***out***.println(item.getIname()+" 메뉴등록 완료");

}

}

}



Client클래스-이름 ,나이 멤버 변수,생성자(String, int)

Client.java

package konkuk.yjcho;

public class Client {

private String Cname;

private int age;

private int total;

public Client(String cname, int age) {

super();

this.Cname = cname;

this.age = age;

total=0;

}

public String getgrade() {

String ss;

if(total==0)

ss="Family";

else if(total<=1000)

ss = "Silver";

else if(total <=2000)

ss = "Gold";

else

ss="VIP";

return ss;

}

*@Override*

public String toString() {

return "이름 : " + Cname + "/ 총 구매금액 : " + total +"/ 등급 : "+ getgrade();

}

public int getAge() {

return age;

}

public void setAge(int age) {

this.age = age;

}

public int getTotal() {

return total;

}

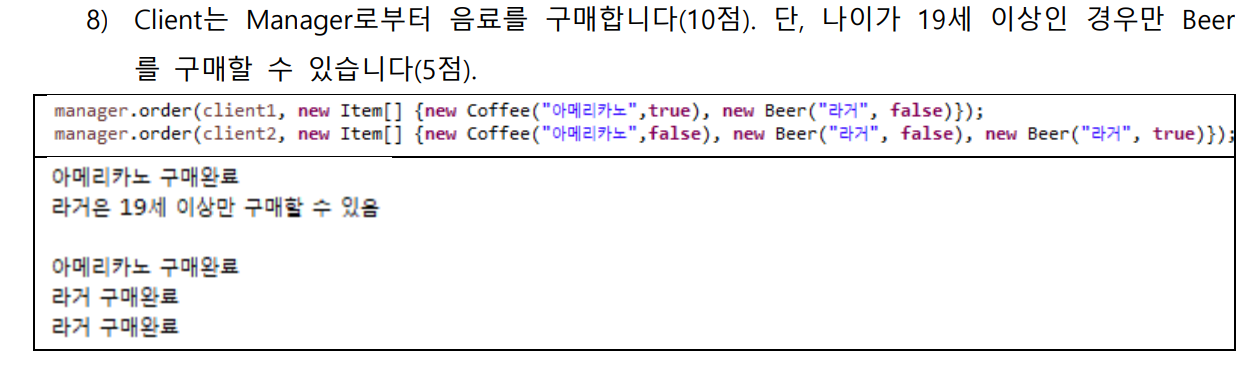
public void setTotal(int total) {

this.total = total;

}

}

명시 되지않은 고객의 총 구매금액변수 total을 멤버 변수로 선언, total에 따른 등급을 리턴해주는 getgrade함수 정의 , 유저 정보 출력을 위한 toString 오버라이드



Item에 함수추가-이 아이템이 주문되어 팔린갯수를 +1하고 가격 또한 업데이트

public void sellcountup() {

this.sellcount++;

totalprice+=price;

}

Manager에 함수 추가 order(주문 고객, 주문할 아이템배열)

public void order(Client c,Item[] item) {

for(int i=0;i<item.length;i++) { //아이템 하니씩 for문 실행

if(item[i] instanceof Coffee) { //커피라면

for(int j=0;j<count;j++) {//메뉴중 일치하는 메뉴 찾기

if(Ilist[j] instanceof Coffee) {//메뉴중 커피에대해

if(item[i].getIname().equals(Ilist[j].getIname())) {

//커피니까 이름만 같다면

Ilist[j].sellcountup();

c.setTotal(c.getTotal()+Ilist[j].getPrice());

//고객 정보의 총구매 금액 업데이트

total+=Ilist[j].getPrice();

// total 이 아이템의 가격 추가

System.***out***.println(item[i].getIname()+ " 구매완료");

continue;

}

}

}

}

else if(item[i] instanceof Beer) {

if(c.getAge()<19) {//19살 미만 맥주 구매불가

System.***out***.println(item[i].getIname()+"는 19세 이상만 구매할 수 있음");

continue;//다음 아이템(반복)으로

}

for(int j=0;j<count;j++) {

if(Ilist[j] instanceof Beer) {

if(item[i].getIname().equals(Ilist[j].getIname())) {

//이름이 같을 시에는

Beer b1=(Beer)item[i];

Beer b2=(Beer)Ilist[j];

if(b1.isFresh() != b2.isFresh())

continue;//생맥여부가 다르면 다른메뉴

Ilist[j].sellcountup();

c.setTotal(c.getTotal()+Ilist[j].getPrice());

total+=Ilist[j].getPrice();

System.***out***.println(item[i].getIname()+ " 구매완료");

}

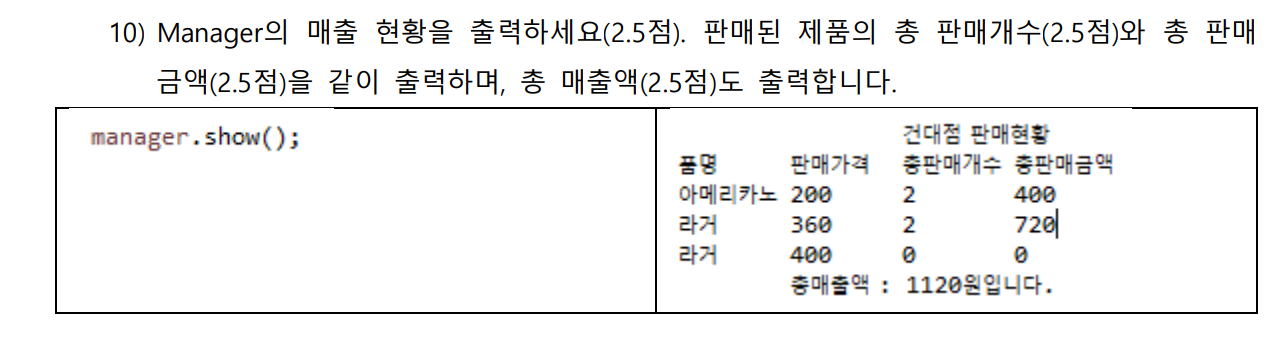
}

}

}

}

}



manager에 show함수 추가- for문으로 아이템리스트의 아이템을 하나씩 출력후 총판매 금액 출력

Manager.java

public void show() {

System.***out***.println(Mname+"점 판매현황"); //출력부

System.***out***.println("품명\t판매가격\t총판매개수\t총판매금액");

for(int i=0;i<count;i++) {

String ss="";

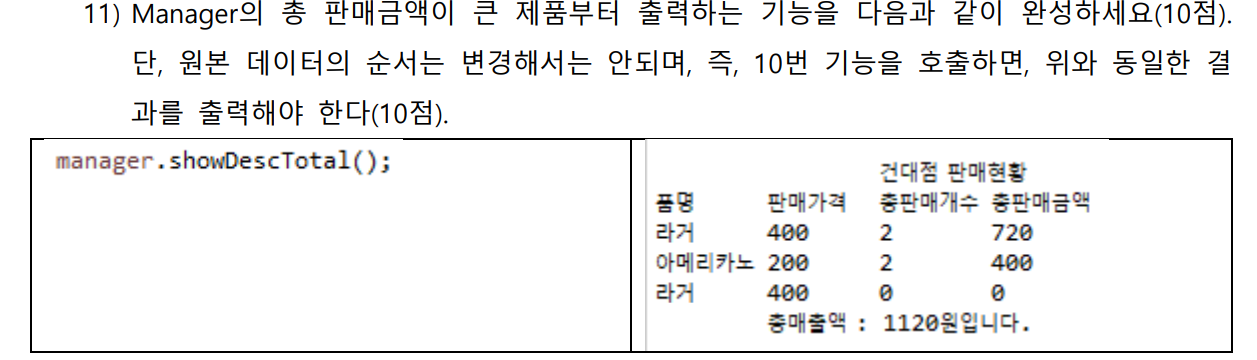
ss += Ilist[i].getIname()+"\t"+Ilist[i].getPrice()+"\t"+Ilist[i].getSellcount()+"\t"+(Ilist[i].getSellcount()\*Ilist[i].getPrice());

System.***out***.println(ss);

}

System.***out***.println("총 매출액 : "+total);

}



public void showDescTotal(){

//총 매출량이 같을 때 값을 잡지 못함

System.***out***.println(Mname+"점 판매현황");

System.***out***.println("품명\t판매가격\t총판매개수\t총판매금액");

int row=0,high=0;

int aa[]=new int[count];//랭크 지정을 위해 메뉴 갯수만큼 배열 할당

for(int j=0;j<count;j++) {//j+1등 정하기 1~count등 정하기

high=0;

for(int i=0;i<count;i++) {

//순위가 정해지지 않은 것들중 1등 찾기

if (aa[i] != 0) continue;

int top=Ilist[i].getTotalprice();

//매출액 받아오기

if(high<= top) {//i번째 객체의 매출액이더크면

row=i; //인덱스 지정

high=top;//최대값 설정

}

}

aa[row]=j+1;//row를 j+1등 지정

//total += high;

}

for(int j=0;j<count;j++) {

for(int i=0;i<count;i++) {

if(aa[i]==j+1) {// 랭크 순위부터 순서대로 출력

String ss="";

ss += Ilist[i].getIname()+"\t"+Ilist[i].getPrice()+"\t"+Ilist[i].getSellcount()+"\t"+Ilist[i].getTotalprice();

System.***out***.println(ss);

}

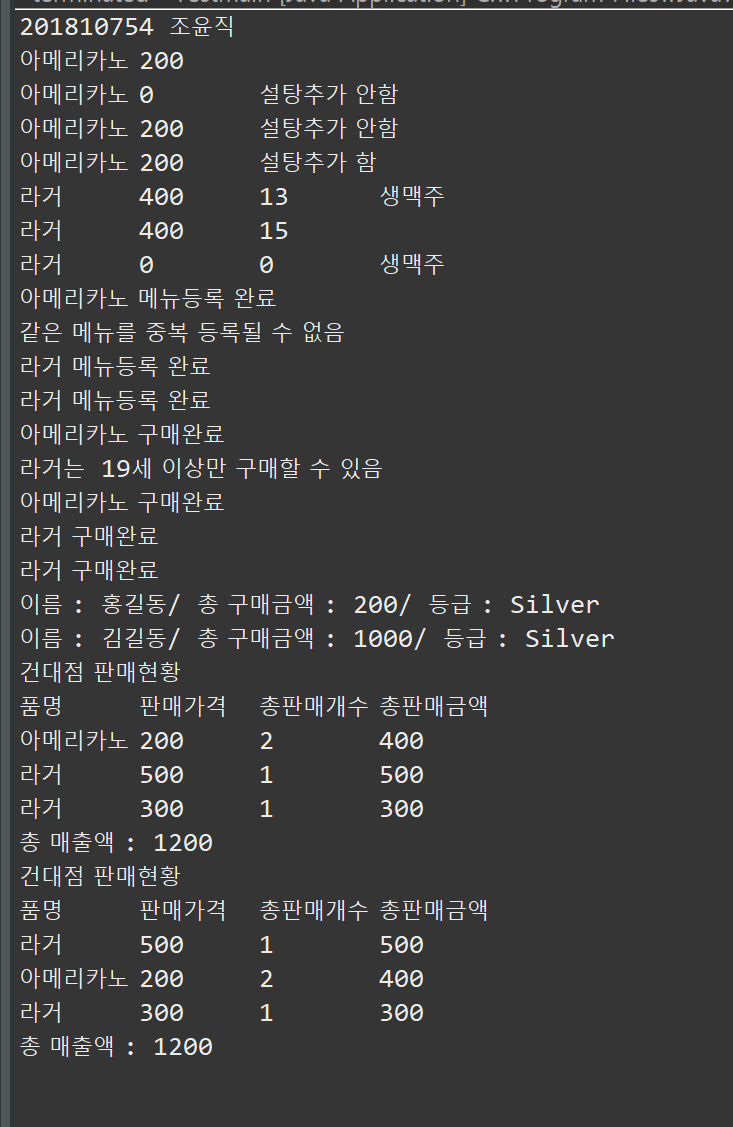
}

}

System.***out***.println("총 매출액 : "+total);

}

실행결과



느낀점

정렬알고리즘 이해 부족,