

# 2023 년 Java 프로그래밍 중간고사

학번:

이름 :

★ 주의 사항: 주어진 문제를 분석하고, 클래스명은 주어진 클래스 명 앞에 본인이름이니셜을 붙여서 만들어 주셔야 하며, 문제에서 주어진 멤버 함수는 주어진 이름 그대로 만들어야 합니다.

★ 프로젝트 명은 자유롭게 정하시면 되고, 각 클래스는 midtest.class0410 라는 패키지에 생성하세요.

\* 다음은 쇼핑몰 프로그램의 일부이며. 쇼핑몰에서 판매하는 제품은 가전제품(HA), 옷(Clothing), 스낵(Snack)을 판매하고 있다고 가정하였으며, 각 제품은 일반 주문과 회원전용 할인 주문이 가능하다고 가정하였다. 아래 클래스를 만들어 주세요. 각 클래스는 별도의 java 파일로 존재해야 합니다. 다음의 모든 클래스 이름 앞에는 **본인이름이니셜**이 와야 합니다 (예, JhchiProduct, JhchiHA, JhchiTestMain 등). **이 조건들을 지키지 않은 클래스는 각 5 점씩 감점함.**

## Product, HA, Clothing, Snack, MBSProduct, Kupang, Client, TestMain 클래스

\* TestMain 클래스를 실행클래스로 만들고, 기타 클래스는 라이브러리 클래스로 만들면 됨. 채점시 입력되는 제품 개수나 순서는 바뀔 수 있으며, 입력제품의 개수나 순서가 변경되도 실행가능해야 함. 각 클래스에는 주어진 멤버이외의 멤버변수나 함수를 추가하여 작성 가능함. 사용하지 않는 멤버 추가시 1 점씩 감점하며, 배열이나 정렬관련 라이브러리 클래스는 사용할 수 없음.

1) 학번과 이름을 출력하세요(5 점).

2) Product 라는 추상 클래스를 만들고, (제품이름, 제품가격)를 가지는 생성자를 만들어주세요 (2.5 점)

```
System.out.println("2) Product 클래스 만들기");  
// JhchiProduct product = new JhchiProduct("TV", 1000);
```

3) 다음 같이 HA 클래스를 만들고, (제품이름, 제품가격, 에너지등급)을 가지고 객체를 생성 및 초기화하고, 출력하세요(7.5 점). 단, 객체의 제품번호는 제품 생성시 1 부터 연속으로 자동부여 한다 (4 번, 5 번 문제의 실행결과를 참고) (2.5 점). 에너지 등급이 1 등급인 제품은 구매시 10% 할인가로 제공하며, 그 이외 등급인 제품은 10% 할증된 가격으로 구매하게 한다(2.5 점).

```
System.out.println("3) HA 클래스 만들기");  
JhchiProduct ha1 = new JhchiHA("건국TV1", 1000, "1등급");  
System.out.println(ha1);  
JhchiProduct ha2 = new JhchiHA("건국TV2", 1000, "3등급");  
System.out.println(ha2);
```

3) HA 클래스 만들기  
제품번호 : 1  
제품이름 : 건국TV1  
제품가격 : 1000원  
에너지 등급 : 1등급  
구매개수 : 0개  
구매금액 : 0원

제품번호 : 2  
제품이름 : 건국TV2  
제품가격 : 1000원  
에너지 등급 : 3등급  
구매개수 : 0개  
구매금액 : 0원

4) 다음 같이 Snack 클래스를 만들고, (제품이름, 제품가격, 제조업체)을 가지고 객체를 생성하고, 출력하세요 (7.5 점).

<pre>System.out.println("4) Snack 클래스 만들기"); JhchiProduct snack = new JhchiSnack("새우깡", 500, "농심"); System.out.println(snack);</pre>	4) Snack 클래스 만들기 제품번호 : 3 제품이름 : 새우깡 제품가격 : 500원 제조업체 : 농심 구매개수 : 0개 구매금액 : 0원
--	--

5) 다음 같이 Clothing 클래스를 만들고, (제품이름, 제품가격, 계절)을 가지고 객체를 생성하고, 출력하세요 (7.5 점).

<pre>System.out.println("5) Clothing 클래스 만들기"); JhchiProduct clothing = new JhchiClothing("청바지", 1000, "여름"); System.out.println(clothing);</pre>	5) Clothing 클래스 만들기 제품번호 : 4 제품이름 : 청바지 제품가격 : 1000원 계절 : 여름 구매개수 : 0개 구매금액 : 0원
---	--

6) 다음과 같이 Kupang 클래스를 만들고, (저장제품개수)가지고 객체를 생성해 주세요(2.5 점). 제품입고를 담당하는 inBound 메소드 (오버로딩 불가 : -5 점)를 만들어 주세요(10 점). 이 메소드를 통해 입고제품을 저장하며, 입고시 상태를 출력하며, 모든 제품은 중복 저장될 수 있으며, 입고가능 개수만큼 저장할 수 있다.

<pre>System.out.println("6) Kupang 클래스 만들기 및 제품 입고하기"); JhchiKupang shop = new JhchiKupang(5); shop.inBound(new JhchiHA("건국TV1", 1000, "1등급")); shop.inBound(new JhchiHA("건국TV2", 1000, "3등급")); shop.inBound(new JhchiSnack("새우깡", 500, "농심")); shop.inBound(new JhchiClothing("청바지", 1000, "여름")); shop.inBound(new JhchiClothing("청바지", 1000, "가을")); shop.inBound(new JhchiClothing("청바지", 1000, "가을"));</pre>	6) Kupang 클래스 만들기 및 제품 입고하기 건국TV1 제품 입고를 완료하였습니다. 건국TV2 제품 입고를 완료하였습니다. 새우깡 제품 입고를 완료하였습니다. 청바지 제품 입고를 완료하였습니다. 청바지 제품 입고를 완료하였습니다. 더이상 입고할 수 없습니다.
--	---

7) 다음과 같이 제품을 검색하는 searchItem 메소드를 완성해 주세요 (10 점). 이 메소드는 제품이름으로 검색하는 기능을 제공하며, 제품이름이 같은 모든 제품을 반환해야 하며, 반환된 결과 출력시 실행결과와 같이 출력되어야 한다.

<pre>System.out.println("7) 제품 검색하기"); JhchiProduct[] sResult1 = shop.searchItem("건국TV2"); System.out.println("=====TV2 검색 결과====="); if (sResult1 != null) {     for (JhchiProduct item : sResult1) {         System.out.println(item);     } }  JhchiProduct[] sResult2 = shop.searchItem("청바지"); System.out.println("=====청바지 검색 결과====="); if (sResult2 != null) {     for (JhchiProduct item : sResult2) {         System.out.println(item);     } }</pre>	7) 제품 검색하기 =====TV2 검색 결과===== 제품번호 : 6 제품이름 : 건국TV2 제품가격 : 1000원 에너지 등급 : 3등급 구매개수 : 0개 구매금액 : 0원  =====청바지 검색 결과===== 제품번호 : 8 제품이름 : 청바지 제품가격 : 1000원 계절 : 여름 구매개수 : 0개 구매금액 : 0원  제품번호 : 9 제품이름 : 청바지 제품가격 : 1000원 계절 : 가을 구매개수 : 0개 구매금액 : 0원
---	--

8) 다음과 같이 회원전용 할인 제품을 검색하는 searchMBSProduct 메소드의 기능을 완성하세요 (15 점).  
Snack 과 Clothing 클래스의 제품은 회원에게만 20% 할인가에 제공되는 제품입니다.

<pre>System.out.println("8) 회원전용 할인제품 검색하기"); JhchiMBSProduct[] sResult3 = shop.searchMBSProduct(); if(sResult3!=null) {     for(JhchiMBSProduct item : sResult3) {         System.out.println(item);     } }</pre>	<p>8) 회원전용 할인제품 검색하기</p> <p>제품번호 : 7 제품이름 : 새우깡 제품가격 : 500원 제조업체 : 농심 구매개수 : 0개 구매금액 : 0원</p> <p>제품번호 : 8 제품이름 : 청바지 제품가격 : 1000원 계절 : 여름 구매개수 : 0개 구매금액 : 0원</p> <p>제품번호 : 9 제품이름 : 청바지 제품가격 : 1000원 계절 : 가을 구매개수 : 0개 구매금액 : 0원</p>
---	---

9) 다음과 같이 Client 클래스를 만들고, (이름, 구매가능개수, 회원여부)를 가지고 객체를 생성하고, 출력하세요 (5 점). 구매가능개수는 주문해서 저장할 수 있는 제품의 수를 의미하며, 회원여부로는 회원과 비회원을 구분한다.

<pre>System.out.println("9) Client 클래스 만들기, 객체생성 및 출력하기"); JhchiClient gdhong = new JhchiClient("홍길동", 3, true); System.out.println(gdhong); JhchiClient gdkim = new JhchiClient("김길동", 4, false);</pre>	<p>9) Client 클래스 만들기, 객체생성 및 출력하기</p> <p>홍길동님의 주문내역</p> <p>회원여부 : true</p> <p>-----</p> <p>주문 내역이 없습니다.</p>
--	---

10) 다음과 같이 제품을 주문 저장하는 order 메소드를 완성하세요 (17.5 점). 회원은 일반제품과 회원전용 할인제품을 모두 주문할 수 있지만, 회원이 아닌 경우 회원전용 할인 제품은 주문할 수 없다. 회원전용 할인제품은 searchMBSProduct 로 검색된 제품으로 20% 할인가를 적용 받는다. 회원이 같은 제품을 searchItem 으로 검색해서 주문한 경우는 할인가를 적용받지 못한다. 제품번호가 같은 제품은 중복 주문할 수 있으며, 이미 주문한 경우, 제품의 구매 개수 카운트를 증가시킨다. 구매 금액도 주문한 제품의 가격과 할인 및 할증 가격을 고려하여 계산한다. 중복 주문하는 제품은 구매가능개수에서 제외되며 11 번의 출력결과를 참고하여 작성한다.

<pre>System.out.println("10) 주문하기"); System.out.println("=== 홍길동 주문 ==="); gdhong.order(sResult1[0]); gdhong.order(sResult1[0]); gdhong.order(sResult2[1]); gdhong.order(sResult3[0]); gdhong.order(sResult2[1]); gdhong.order(sResult3[1]); gdhong.order(sResult3[2]);  System.out.println("=== 김길동 주문 ==="); gdkim.order(sResult1[0]); gdkim.order(sResult2[0]); gdkim.order(sResult3[0]); gdkim.order(sResult2[0]);</pre>	<p>10) 주문하기</p> <p>=== 홍길동 주문 ===</p> <p>주문이 완료되었습니다. 주문이 완료되었습니다. 주문이 완료되었습니다. 주문이 완료되었습니다. 주문이 완료되었습니다. 더이상 주문 할 수 없습니다. 주문이 완료되었습니다.</p> <p>=== 김길동 주문 ===</p> <p>주문이 완료되었습니다. 주문이 완료되었습니다. 회원전용 할인제품은 주문할 수 없습니다. 주문이 완료되었습니다.</p>
--	--

11) 주문 후 고객정보를 다음과 같이 출력합니다 (5 점). 출력시 선택정렬 알고리즘을 이용하여 구매금액이 큰 순서로 정렬하여 출력하며, 고객이 주문한 구매금액들을 합하여 총결제금액도 같이 출력한다.

<pre>System.out.println("11) 주문 고객정보 출력하기"); System.out.println(gdhong); System.out.println(gdkim);</pre>	
11) 주문 고객정보 출력하기 홍길동님의 주문내역 회원여부 : true ----- 제품번호 : 9 제품이름 : 청바지 제품가격 : 1000원 계절 : 가을 구매개수 : 3개 구매금액 : 2800원 ----- 제품번호 : 6 제품이름 : 건국TV2 제품가격 : 1000원 에너지 등급 : 3등급 구매개수 : 2개 구매금액 : 2200원 ----- 제품번호 : 7 제품이름 : 새우깡 제품가격 : 500원 제조업체 : 농심 구매개수 : 1개 구매금액 : 400원 ----- 총결제금액 : 5400원 -----	김길동님의 주문내역 회원여부 : false ----- 제품번호 : 8 제품이름 : 청바지 제품가격 : 1000원 계절 : 여름 구매개수 : 2개 구매금액 : 2000원 ----- 제품번호 : 6 제품이름 : 건국TV2 제품가격 : 1000원 에너지 등급 : 3등급 구매개수 : 1개 구매금액 : 1100원 ----- 총결제금액 : 3100원 -----