**CSE2035(C프로그래밍)**

**(설계 프로젝트 : 쇼핑몰 시스템 구현)**

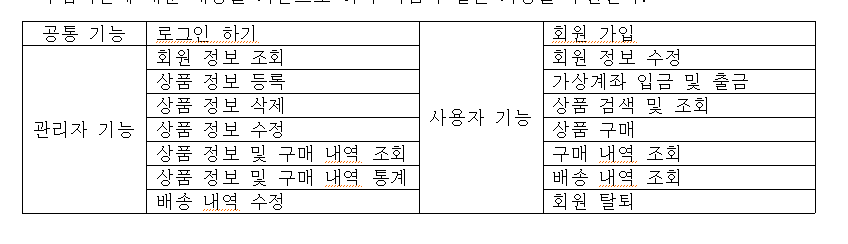
**담당교수 : 서강대학교 컴퓨터공학과 김지환**

**1. 설계 문제 및 목표**

본 프로젝트에서는 C언어를 활용하여 사용자와 관리자의 관점에서 쇼핑몰 시스템을 구현한다.

**1.1 현실적 제한조건**

수업시간에 배운 내용을 기반으로 하여 다음과 같은 기능을 구현한다.



**2. 요구사항**

**2.1 합성**

본 쇼핑몰 시스템 구현에는 크게 관리자 기능과 사용자 기능, 공통기능으로 나누어져 있다. 먼저 공통기능에는 로그인 기능이 있다. 각 로그인이 성공하면 사용자 페이지와 관리자 페이지로 들어가게 된다. 관리자 기능에는 회원 정보를 조회하는 기능이 있다. 회원 정보는 회원 번호 순대로 정렬되어있으며 회원번호, 회원 이름, 아이디, 주소, 잔액 순으로 정보가 나열된다. 두 번째로 상품 정보 등록 및 삭제 기능이 있다. 상품 정보 등록은 상품 리스트에 상품을 등록해주고 상품 정보 삭제는 상품의 상태를 삭제상태(-1)로 만들어준다. 또한, 상품의 상태가 -1일 경우 구매가 불가능하도록 구현하였다. 다음 관리자 기능은 상품 정보 수정이다. 상품 정보 수정에서는 상품명, 카테고리, 가격을 수정할 수 있으며 이 때 상품명은 다른 상품과 겹치지 않도록 하였다. 네 번째로 상품 정보 및 구매 내역 조회 기능은 상품 정보는 상품 번호 순대로 정렬하였고, 상품번호, 상품명, 카테고리, 가격, 상태 순으로 정보가 나오도록 구현하였다. 구매 내역 또한 구매 번호 순대로 정렬하였고, 주문번호, 상품번호, 회원번호, 배송현황 순으로 정보가 나오도록 하였다. 다음 관리자 기능은 상품 정보 및 구매 내역 통계이다. 상품 정보 통계는 상품 가격 순으로 정렬하여 상품번호, 상품명, 가격이 나오도록 하였다. 구매 내역 통계는 판매량 순으로 정렬하여 상품 번호, 상품명, 판매량, 가격이 나오도록 하였다. 마지막 관리자 기능은 배송 내역 수정이다. 배송 내역 수정 기능에서는 주문번호를 선택하여 배송준비, 배송중, 배송완료 중 하나로 배송현황을 변경할 수 있다. 다음은 사용자 기능이다. 사용자 기능에는 먼저 회원 가입을 할 수 있는 기능이 있다. 회원 리스트에 새로운 회원을 추가해주는 기능을 지원한다. 다음으로 회원 정보 수정 기능이 있다. 회원 정보 수정은 이름 수정, 아이디 수정, 비밀번호 수정, 주소 수정, 회원 탈퇴 기능이 있고 아이디 수정은 아이디 중복체크 기능을 포함하고 있으며 회원탈퇴는 회원상태를 탈퇴 상태(-1)로 만들어 준다. 다음 사용자 기능은 가상계좌 입금 및 출금 기능이다. 사용자의 가상계좌에 입금과 출금이 가능하다. 출금에서는 출금할 양이 가상계좌보다 더 많거나 가상계좌가 0원일 경우 출금이 불가능하도록 구현하였다. 다음은 상품 검색 기능이다. 이 기능은 main에서도 수행가능하다. 상품명 또는 카테고리로 검색하여 상품 정보를 보여주도록 하는 기능이다. 이어서 사용자 기능에서는 상품 구매도 진행된다. 상품 구매 기능은 가상계좌와 연결되어 구매하기를 진행할 수 있고 구매내역에 상품번호, 회원번호 등이 기록된다. 구매 내역 조회 기능은 사용자가 구매한 상품 내역과 배송현황을 볼 수 있다. 마지막으로 사용자 기능에서 배송 내역 조회는 구매 내역에서 배송중인 상품들만 볼 수 있도록 구현하였다.

**2.2 분석**

모든 함수는 project.c에 들어있다.

1. void create\_Node(Node\*) : 노드 새로 만들기 함수

2. void get\_user\_data(Node\*) : customer 리스트 만드는 함수

3. void get\_prod\_data(Node\*) : product 리스트 만드는 함수

4. void get\_admin\_data(Node\*) : admin 리스트 만드는 함수

5. void get\_hist\_data(Node\*) : 구매내역 리스트 만드는 함수

---------------------------------------------------------------------1~5는 수시로 사용되는 함수들이다.

6. void free\_func(Node\*,Node\*,Node\*,Node\*) : 마지막 종료하기 전 리스트 전부 free시켜주는 함수

7. int main () : main page겸 main 함수

8. int user\_login(Node\*,Node\*,Node\*) : 공통 기능의 회원 로그인 함수

9. void user\_register(Node\*) : 사용자 기능의 회원 가입을 하는 함수

10. int user\_page(Node\*,Node\*,Node\*,Node\*) : 회원 페이지 함수

11. view\_delivery\_hist(Node\*,Node\*,Node\*,Node\*) : 사용자 기능의 배송 내역을 조회하는 함수

12. view\_purchase\_hist(Node\*, Node\*,Node\*,Node\*) : 사용자 기능의 구매 내역을 조회하는 함수

13. virtual\_account(Node\*, Node\*) : 사용자 기능의 회원 가상 계좌 관련 함수

14. int search\_product(Node\*) : 사용자 기능의 상품을 찾는 함수

15. buy\_product(Node\*, Node\*, Node\*, Node\*) : 사용자 기능의 상품을 구매하는 함수

16. user\_data\_edit(Node \*, Node\*) : 사용자 기능의 회원 정보를 수정하는 함수

17. int admin\_login(Node\* , Node\* ,Node\* ,Node\* ) : 공통 기능의 관리자 로그인 함수

18. int admin\_page(Node\*, Node\* , Node\* , Node\*) : 관리자 페이지 함수

19. int delivery\_data\_edit(Node\*, Node\*) : 관리자 기능의 배송 내역을 수정하는 함수

20. int view\_purchase\_statistics(Node\*, Node\*) : 관리자 기능의 상품 및 구매 내역 통계를 보여주는 함수

21. int view\_prod\_purchase\_info(Node\*, Node\*) : 관리자 기능의 상품 및 구매 내역 조회를 하는 함수

22. int prod\_data\_edit(Node\* ) : 관리자 기능의 상품 정보를 수정하는 함수

23. int prod\_regist\_del(Node\*) : 관리자 기능의 상품 등록 및 삭제를 하는 함수

24. int view\_user\_data(Node\*) : 관리자 기능의 회원 정보 조회를 하는 함수

**2.3 제작**

모든 함수는 project.c에 들어있다.

1. void create\_Node(Node\* cur) : 새로운 노드를 생성하여 맨 뒤에 추가해주는 함수이다.

2. void get\_user\_data(Node\* head) : customer.csv 파일에서 한줄씩 읽을 때마다 노드에 정보를 넣어주고 새로운 노드를 생성하는 과정을 파일의 끝까지 반복해준다. customer 리스트를 만들어 주는 과정이다.

3. void get\_prod\_data(Node\* head) : product.csv 파일에서 한줄씩 읽을 때마다 노드에 정보를 넣어주고 새로운 노드를 생성하는 과정을 파일의 끝까지 반복해준다. product 리스트를 만들어 주는 과정이다.

4. void get\_admin\_data(Node\* head) : admin.csv 파일에서 한줄씩 읽을 때마다 노드에 정보를 넣어주고 새로운 노드를 생성하는 과정을 파일의 끝까지 반복해준다. admin 리스트를 만들어 주는 과정이다.

5. void get\_hist\_data(Node\* head) : history.csv 파일에서 한줄씩 읽을 때마다 노드에 정보를 넣어주고 새로운 노드를 생성하는 과정을 파일의 끝까지 반복해준다. 구매 내역 리스트를 만들어 주는 과정이다.

6. void free\_func(Node\* head,Node\* head\_pr,Node\* head\_ad,Node\* head\_hs) : 모든 리스트를 프리 시켜주는 함수(customer 리스트, product 리스트,admin 리스트,구매내역 리스트)

7. int main () : main page를 의미한다. 각각의 리스트의 head를 선언해주고 위의 get\_\*\*\*\*\_data(Node \* head)함수를 통해서 리스트를 생성해준다. main page에서는 1이 입력되면 회원 로그인 페이지, 2를 입력하면 회원 가입 페이지, 3을 입력하면 상품 검색 페이지, 4를 입력하면 관리자 로그인 페이지로 넘어가고, 5를 입력하면 전체 코드가 종료된다. 잘못된 숫자를 입력했을 경우 다시 시도하게 된다.

8. int user\_login(Node\* head,Node\* head\_pr,Node\* head\_hs) : main page에서 1을 입력했을 경우 나오는 회원 로그인 페이지이다. 아이디와 비밀번호를 입력하고 아이디,비밀번호가 모두 일치하는 회원이 있을 경우 user page로 넘어가게 된다. 단, 회원 상태가 -1 즉 탈퇴상태일 경우 "탈퇴한 회원입니다."와 함께 main page로 돌아가게 된다. 로그인은 3회까지 가능하며 3회 실패할 경우 "확인 후 다시 로그인을 시도하여 주십시오"와 함께 enter키를 누르면 main page로 다시 돌아가게 된다. 이 함수에서 head\_pr와 head\_hs가 사용된 이유는 user\_page에서 필요하기 때문이다.

9. void user\_register(Node\* head) : main page에서 2를 입력했을 때 나오는 회원 가입 페이지이다. customer.csv파일을 a+로 열어서 맨 뒤에 추가로 회원정보를 입력해준다. 아이디는 회원이 중복될 수 없으므로 중복된 id를 입력할 경우 다시 다른 아이디를 입력하도록 되어있다. 비밀번호를 입력한 후 비밀번호 확인을 다시 입력하여 비밀번호가 비밀번호 확인과 일치하지 않으면 다시 비밀번호를 입력해야한다. 비밀번호와 비밀번호 확인이 일치할 경우에는 주소 입력으로 넘어간다. 주소 입력까지 완료되면 회원가입이 완료되었다. customer.csv에는 새로운 회원정보가 생겼을 것이다.

10. int user\_page(Node\* cur,Node\* head\_pr,Node \*head,Node \*head\_hs) : 회원 로그인이 성공했을 시 넘어오게 되는 user page이다. cur에는 현재 로그인이 성공한 회원의 node정보가 담겨있으며 나머지는 각 리스트의 head들이다. user page에서는 1을 입력하면 회원 정보 수정, 2를 입력하면 가상계좌 입금 및 출금, 3을 입력하면 상품 검색 및 구매, 4를 입력하면 구매 내역 조회, 5를 입력하면 배송내역 조회( 배송 중인 상품만), 그리고 6을 입력할 경우 다시 main page로 돌아가게 된다. 잘못된 숫자를 입력했을 경우 다시 시도하게 된다.

11. view\_delivery\_hist(Node\* cur,Node \*head,Node\* head\_pr,Node\* head\_hs) : user page에서 5를 입력하면 나오는 배송 내역 조회 페이지이다. 배송 상태가 배송중인 상품만 나오게 된다. cur에 담긴 회원을 전체 회원에서 찾은 뒤 구매 내역 리스트에서 회원 번호가 일치하는 구매내역을 찾는다. 그리고 그 중 배송중인 상품만 보여준다. enter키를 누르면 다시 user page로 돌아가게 된다. 배송 중인 내역이 없으면 아무것도 뜨지 않고 enter키를 누르면 user page로 돌아간다.

12. view\_purchase\_hist(Node\* cur, Node\* head,Node\* head\_pr,Node\* head\_hs) : user page에서 4를 입력하면 나오는 구매 내역 조회 페이지 이다. 배송 내역 조회와 비슷하게 cur에 담긴 회원을 전체 회원에서 찾은 뒤 구매 내역 리스트에서 회원 번호가 일치하는 구매내역을 찾는다. 상품명, 상품번호, 배송상태가 뜨도록한다. 배송상태는 배송준비, 배송중, 배송완료로 나뉜다. enter키를 누르면 다시 user page로 돌아가게 된다. 구매 내역이 없으면 아무것도 뜨지 않고 enter키를 누르면 user page로 돌아간다.

13. virtual\_account(Node\* cur, Node\* head) : user page에서 2를 입력하면 나오는 가상계좌 입금 및 출력 페이지이다. 이번에도 마찬가지로 cur에 담긴 회원을 전체 회원에서 찾아준다. 1을 입력할 경우 입금하기, 2을 입력할 경우 출금하기, 3을 입력하면 다시 user page로 돌아간다. 입금하기 기능에서는 입금 금액을 입력하면 cur를 이용해서 찾은 회원의 가상계좌에 입력한 숫자만큼 더해준다. 출금하기 기능에서는 현재 회원의 가상게좌에 잔액이 0일 경우 "잔액이 부족합니다. 입금 후 사용해주십시오."와 함께 enter키를 누르면 다시 user page로 돌아간다. 계좌에 돈이 있을 경우에는 출금 금액이 계좌에 있는 돈보다 많지 않으면 출금이 가능하고 그렇지 않을 경우는 잔액이 0일 경우와 동일하게 enter키를 누르면 user page로 돌아간다. 회원의 계좌 정보가 바뀌었으므로 customer.csv를 새로 작성하여 준다.

14. int search\_product(Node\* head\_pr) : 상품 검색이 가능한 함수이다. 이 함수는 두 군데에 사용되는데 첫 번째는 main page에서 3을 입력했을 경우이고 두 번째는 user page에서 3을 입력했을 경우이다. 상품 검색 페이지에서는 1을 입력하면 이름으로 검색이 가능하고, 2를 입력하면 카테고리로 검색이 가능하다. 이름 검색의 경우 product 리스트를 쭉 지나가면서 상품 이름이 입력값과 같을 경우 상품코드,상품 이름, 카테고리, 가격을 프린트 해준다. 카테고리 검색의 경우 이름 검색과 동일한 방식으로 입력한 카테고리에 해당하는 모든 상품들을 프린트해준다. 0을 입력할 경우 main page의 상품검색에서는 main page로 돌아가고 user page의 상품검색에서는 user page로 돌아간다.

15. buy\_product(Node\* head\_pr, Node\*cur, Node\*head, Node\* head\_hs) : user page에서 3을 입력했을 경우 상품을 검색한 뒤 바로 이어지는 상품 구매 페이지이다. 구매할 상품코드를 입력하고 product 리스트를 쭉 지나가면서 입력한 상품 코드와 상품번호가 일치하는 상품을 찾아준다. 상품을 찾으면 그 상품 정보가 프린트 되고 1을 입력하면 구매하기 2를 입력하면 이전페이지로 돌아가기를 할 수 있다. 사실상 그 이전 페이지는 user page로 돌아가게 된다. 구매하기 기능에서는 삭제된 상품일 경우에는 구매할 수 없다. 등록되어있는 상품일 경우에는 상품 가격이 가상계좌 잔액보다 큰 경우 필요금액을 보여주고 추가 입금을 해주고 구매를 진행한다. 아닐 경우에는 그냥 구매를 진행한다. 구매를 진행한 후에는 구매 후 잔액을 보여준다. 잔액 상태가 바뀌었으므로 customer.csv를 다시 작성해준다. 구매내역도 history.csv에 새로운 정보를 입력해준다. enter키를 누르면 user page로 돌아간다.

16. user\_data\_edit(Node \*cur, Node\* head) : user page에서 1을 입력할 경우 나오는 회원 정보 수정 페이지이다. 1을 입력하면 이름 수정, 2를 입력하면 아이디 수정, 3을 입력하면 비밀번호 수정, 4를 입력하면 주소 수정, 5를 입력하면 회원 탈퇴가 가능하다. 여기서 아이디는 다른 회원과 중복되면 안되므로 중복체크를 해준다. 회원탈퇴는 회원 상태를 -1로 바꾸는 형태로 하였다. 회원 정보가 변경되었으므로 customer.csv를 새로 다시 작성해준다. enter키를 누르면 user page로 돌아간다.

17. int admin\_login(Node\* head, Node\* head\_pr,Node\* head\_ad,Node\* head\_hs) : main page에서 4를 누르면 나오는 관리자 로그인 페이지이다. 이미 저장되어있는 admin.csv에 등록되어있는 아이디 비밀번호로만 로그인이 가능하다. 아이디와 비밀번호가 모두 일치하는 관리자가 있을 경우 관리자 페이지로 넘어간다. 로그인은 3회까지 가능하다.

18. int admin\_page(Node\* head, Node\* head\_pr, Node\* head\_ad, Node\* head\_hs) : 관리자 로그인에 성공하면 들어오게 되는 관리자 페이지이다. 1을 입력할 경우 회원 정보 조회, 2를 입력할 경우 상품 등록 및 삭제 ,3을 입력할 경우 상품 정보 수정, 4를 입력할 경우 상품 정보 조회 및 구매 내역 조회, 5를 입력할 경우 상품 정보 통계 및 구매 내역 통계, 6을 입력할 경우 배송내역 수정, 7을 입력할 경우에는 main page로 돌아간다.

19. int delivery\_data\_edit(Node\* head\_hs, Node\* head) : 관리자 페이지에서 6을 입력할 경우 나오는 배송 내역 수정 페이지이다. 수정할 주문 번호를 입력해준다. 입력받은 주문번호와 일치하는 배송내역을 찾아준다. 배송내역에 대한 정보(주문번호, 상품 번호, 회원번호, 아이디, 주소, 배송현황)을 프린트 해준다. 배송 현황을 변경하는데 1을 입력하면 배송준비, 2를 입력하면 배송중, 3을 입력하면 배송완료로 변경해준다. 배송준비일 경우 구매 내역 node의 배송상태를 0으로, 배송 중일 경우는 1, 배송 완료일 경우 2로 수정해준다. 배송내역이 수정되었으므로 history.csv를 새로 작성해준다.

20. int view\_purchase\_statistics(Node\* head\_pr, Node\* head\_hs) : 관리자 페이지에서 5를 입력할 경우 나오는 상품 정보 통계 및 구매 내역 통계 페이지이다. 구매 내역 통계는 구매 내역 리스트를 읽으면서 상품 번호가 일치할 때마다 해당 상품의 판매량을 1씩 증가시켜준다. 판매량 순대로 상품번호, 상품명, 판매량, 가격을 정렬시켜주고 구매 내역 통계를 보여준다. 그 후 리스트를 다시 원상복구 시키기 위해 상품 번호 순대로 다시 상품번호, 상품명, 판매량, 가격을 정렬시켜준다. 상품 정보 통계는 상품 가격 순대로 상품 번호, 상품명, 가격을 정렬시켜주고 상품 통계를 보여준다. 그 후 똑같이 다시 원상복구 해주기 위해 상품 번호 순대로 상품번호, 상품명, 가격을 정렬 시켜주었다. enter키를 누르면 다시 관리자 페이지로 돌아간다.

21. int view\_prod\_purchase\_info(Node\* head\_pr, Node \*head\_hs) : 관리자 페이지에서 4를 입력할 경우 나오는 상품 정보 조회 및 구매 내역 조회 페이지이다. 상품 정보 조회는 상품 리스트를 쭉 읽어서 정보를 보여주고 구매 내역 조회는 구매내역 리스트를 쭉 읽어서 정보를 보여준다. enter키를 누르면 다시 관리자 페이지로 돌아간다.

22. int prod\_data\_edit(Node\* head\_pr) : 관리자 페이지에서 3을 입력할 경우 나오는 상품 정보 수정 페이지이다. 수정할 상품명을 입력한다. 만약 상품이 존재하지 않으면 "상품이 존재하지 않습니다"와 함께 다시 관리자 페이지로 넘어간다. 상품이 존재할 경우 1을 입력하면 상품명 수정, 2를 입력하면 카테고리 수정, 3을 입력하면 가격 수정이 가능하다. 여기서 상품명은 다른 상품과 겹치면 안되므로 중복 체크를 해준다. 상품 정보가 수정되었으므로 product.csv를 다시 작성해준다.

23. int prod\_regist\_del(Node\* head\_pr) : 관리자 페이지에서 2를 입력할 경우 나오는 상품 등록 및 삭제 페이지이다. 1을 입력하면 상품 등록, 2를 입력하면 상품 삭제가 가능하다. 상품 등록의 경우 상품 이름을 입력해주고 이름 중복체크를 해준다. 중복체크가 완료되면 카테고리와 가격까지 입력해준다. 상태는 등록인 1로 기본적으로 등록해준다. 상품 삭제의 경우 상품명을 입력하고 없으면 "상품이 존재하지 않습니다"와 함께 관리자 페이지로 돌아간다. 그렇지 않을 경우, y를 입력하면 상품상태가 -1로 바뀌고, n을 입력하면 상품 상태가 1로 바뀌므로 재등록도 가능하다. 상품 정보가 바뀌었으므로 product.csv를 새로 다시 작성한다.

24. int view\_user\_data(Node\* head) : 관리자 페이지에서 1을 입력할 경우 나오는 회원 정보 조회 페이지이다. 회원 리스트를 쭉 읽어서 정보를 보여준다. enter키를 누르면 다시 관리자 페이지로 돌아간다.

**2.4 시험**

**<**사용자 기능>

1. 회원가입 id: testid pw: testpw 이름: tester 주소: 띄어쓰기 가능

이름 입력 -> 아이디 입력(ididid는 기존 회원임->중복체크,testid는 중복체크에 안걸림) -> 비밀번호 입력(비밀번호 확인과 다르게 입력하면 안넘어감) -> 주소입력 -> 메인화면으로 돌아감

2. 로그인 하기(공통 기능)

위에서 회원 가입 한 아이디로 로그인 가능( 3회 이상 잘못 입력할 경우 다시 메인화면으로 돌아감) -> 로그인 성공시 회원 페이지로 넘어감

<회원페이지>

3. 가상계좌 입금 및 출금

회원 페이지에서 2 선택 -> 현재 새로 회원가입한 tester이기 때문에 가상계좌 잔액:0 -> 출금불가능 -> 엔터키 누르면 회원 페이지로 넘어감 다시 2선택 -> 입금 1000원 -> 엔터키 누르면 회원페이지로 넘어감 다시 2선택 -> 출금 1200원 (가상계좌보다 큰 돈 출금 불가능) -> 엔터키 누르면 회원페이지로 넘어감 다시 2선택 -> 출금 700원 -> 현재 잔액 보여주고 끝 -> 엔터키 누르면 회원페이지로 돌아감

4. 상품 검색 및 조회 + 상품 구매

1) 회원 페이지에서 3 선택 -> 1선택(이름 검색) -> 귤이 이미 등록되어있음 귤 검색 -> 구매할 상품 코드 입력(귤은 상품코드가 9이므로 9입력) -> 구매하기 (현재 잔고가 더 크면 그냥 구매됨) -> 엔터키 입력하면 회원 페이지로 넘어감 다시 3선택 -> 상품 카테고리 검색 -> 상품 코드 이번엔 10 입력 (초콜릿을 살거임) -> 구매하기 (잔고 부족하므로 입금하고 다시 구매하는 방식) -> 엔터키 입력하면 회원 페이지로 넘어감 다시 3선택 -> 0 누르면 다시 회원 페이지로 넘어감

2) 회원 페이지에서 3 선택 -> 1선택(이름 검색) -> 없는 이름일 경우 상품이 존재하지 않습니다 와 함께 엔터키 누르면 회원페이지로 넘어감

3) 회원 페이지에서 3 선택 -> 2선택(카테고리 검색) -> 없는 카테고리 이름일 경우 press enter와 함께 엔터키 누르면 회원페이지로 넘어감

4) 회원 페이지에서 3 선택 -> 0선택 -> 회원페이지로 돌아감

5) 구매할 상품코드 입력 후 -> 2선택 -> 회원페이지로 돌아감

5. 구매 내역 조회

1) 회원페이지에서 4 입력 -> 방금 구매한 상품은 배송 준비 상태이기 때문에 tester가 방금 구매한 귤과 초콜릿이 둘다 배송준비 상태로 상품 정보와 함께 뜸 ->엔터키를 누르면 회원 페이지로 돌아감

6. 배송 내역 조회 (배송 중)

1) 회원 페이지에서 5 입력 -> 회원 페이지에서 배송 내역 조회는 배송 중인 상품만 뜨므로 아직 아무것도 안뜸(귤이랑 초콜릿 둘 다 배송준비임) -> 엔터키를 누르면 회원 페이지로 돌아감

7. 회원 정보 수정 (회원 정보 수정된 거는 관리자에서 확인)

1) 회원 페이지에서 1 입력 -> 1입력(이름 수정) -> 테스터로 변경 -> 엔터키를 누르면 회원 페이지로 돌아감

2) 회원 페이지에서 1 입력 -> 2입력(아이디 수정) -> ididid로 변경 -> 아이디 중복 체크 -> testidd로 변경 -> 엔터키를 누르면 회원 페이지로 돌아감

3) 회원 페이지에서 1 입력 -> 3입력(비밀번호 수정) -> testpww로 변경 -> 엔터키를 누르면 회원 페이지로 돌아감

4) 회원 페이지에서 1 입력 -> 4입력(주소 수정) -> 이거슨 테스터의 주소입니다. 로 변경 -> 엔터키를 누르면 회원 페이지로 돌아감

--->관리자 기능에서 확인 가능

<관리자 기능>

8. 관리자 로그인 하기

아이디:admin, 비밀번호: password로 로그인( 3회 이상 잘못 입력할 경우 다시 메인화면으로 돌아감) -> 로그인 성공시 관리자 페이지로 넘어감 (관리자 이름 뜸)

\*\*로그인 가능한 관리자 정보는 admin.csv에 있는 관리자 모두 로그인 가능

9. 회원 정보 조회(아까 회원 정보 수정된 거 보여줌)

관리자 페이지에서 1 선택 -> 회원 정보가 (회원번호, 이름, 아이디, 주소, 잔액, 상태) 순으로 회원번호로 정렬되어 뜸 -> 엔터키를 누르면 관리자 페이지로 돌아감

10. 상품 등록 및 삭제

1) 관리자 페이지에서 2 선택 -> 1 선택( 상품 등록) -> 상품 이름 입력(중복체크를 위한 귤 입력) ->다시 상품 이름 입력 : "헤어 롤"(띄어쓰기 가능) ->카테고리 입력: "생활용품"(띄어쓰기 불가능) -> 가격 :"1500" 입력(숫자만 입력가능) -> 엔터키를 누르면 관리자 페이지로 돌아감

2) 관리자 페이지에서 2 선택 -> 2 선택 (상품 삭제) -> 상품 이름 입력 아무거나 -> 상품이 존재하지 않습니다 와 함께 엔터키를 누르면 관리자 페이지로 돌아감 다시 2선택 -> 2선택( 상품 삭제) -> 다시 상품 이름 입력 : "헤어 롤"(띄어쓰기 가능) -> y 입력하면 상태 -1, n입력하면 상태 1로 바뀌기 때문에 재등록도 가능 (바뀐 것은 나둥에 상품 정보 조회에서 확인가능) -> y입력 후 엔터키를 누르면 관리자 페이지로 돌아감

11. 상품 정보 수정

1) 관리자 페이지에서 3선택 -> 상품 이름 입력 아무거나 -> 상품이 존재하지 않습니다 와 함께 엔터키를 누르면 관리자 페이지로 돌아감 다시 3선택 -> 상품명 입력 "헤어 롤" -> 1선택( 상품명 수정)-> 중복 체크를 위해 귤 입력 -> 다시 수정할 상품명 입력: "헤어롤" -> 엔터키를 누르면 관리자 페이지로 돌아감 다시 3선택 -> 상품명 입력 "헤어롤"(수정된 상품명) -> 2선택 (카테고리 수정) -> 카테고리 입력 "생필품"(띄어쓰기 불가능) -> 엔터키를 누르면 관리자 페이지로 돌아감 다시 3선택 -> 상품명 입력 "헤어롤"(수정된 상품명) -> 3선택 (가격 수정) : "2000" 입력(숫자만 입력 가능) -> 엔터키를 누르면 관리자 페이지로 돌아감

12. 상품 정보 및 구매 내역 조회(아까 tester가 잘 샀는지 검사)

관리자 페이지에서 4선택 -> 상품 정보 및 구매 내역 조회 (상품번호, 주문번호 순대로 정렬) -> 엔터키를 누르면 관리자 페이지로 돌아감

13. 상품 정보 및 구매 내역 통계(구매 내역 조회 했으므로 구매 내역 통계 맞는지 확인/상품 정보도)

관리자 페이지에서 5선택 -> 상품 정보 및 구매 내역 통계 (가격, 판매량 순대로 정렬) -> 엔터키를 누르면 관리자 페이지로 돌아감

14. 배송 내역 수정(tester의 구매내역을 배송중으로 변경해줌-> 배송 내역 조회 다시 검사)

관리자 페이지에서 6선택 -> 수정할 주문 번호 입력( 아까 구매 내역 조회에서 tester가 구매한 주문번호가 14,15 였음//14는 배송중으로 15는 배송완료로 바꿔보겠음) -> 주문 번호 14 입력 -> 배송정보 뜸 ->2 선택(배송중으로 변경) -> 엔터키를 누르면 관리자 페이지로 돌아감 다시 6 선택 -> 주문 번호 15 입력 -> 배송정보 뜸 ->3 선택(배송완료로 변경) -> 엔터키를 누르면 관리자 페이지로 돌아감

-----> 이후 확인 하기 위해 배송내역 조회에서 0배송준비,1배송중,2배송완료로 확인가능

-----> 또는 testidd/testpww로 회원로그인 후 구매내역 조회 배송내역 조회 가능

15. 회원 탈퇴

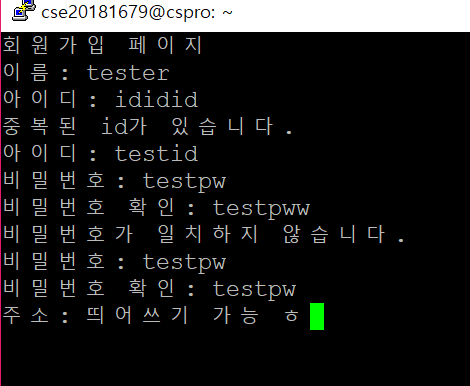
메인페이지에서 회원로그인( testidd/testpww로) -> 1선택(회원정보 수정) -> y선택하면 탈퇴 됨 -> 엔터키 누르면 메인페이지로 돌아감

-----> 이후 확인 하기 위해 관리자 로그인 해서 관리자 페이지에서 회원 정보 조회

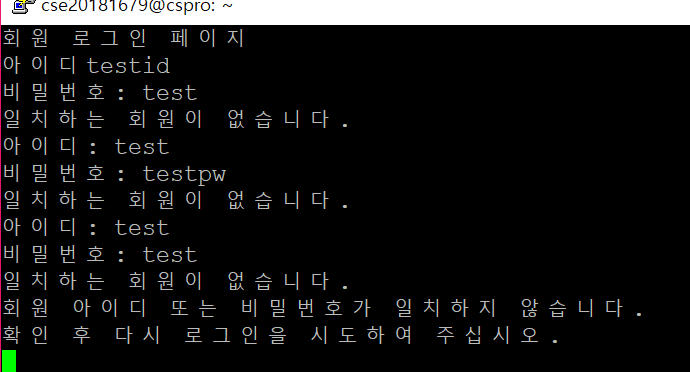
-----> 또는 testidd/testpww로 회원로그인 시도 ->엔터키 누르면 메인페이지로 돌아감

**2.5 평가**

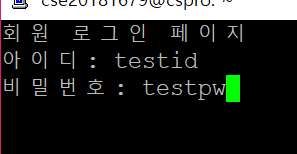
1. 회원 가입 페이지



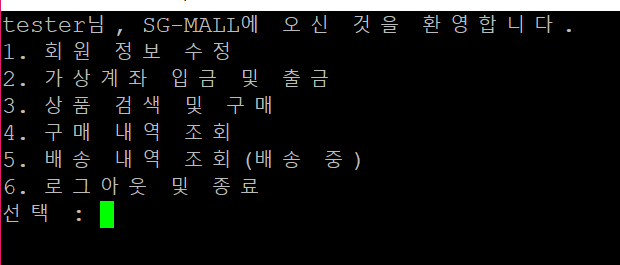
2. 회원 로그인 페이지



3회이상 잘못 시도할 경우 이런 화면이 뜨고 엔터키를 누르면 메인 화면으로 돌아감

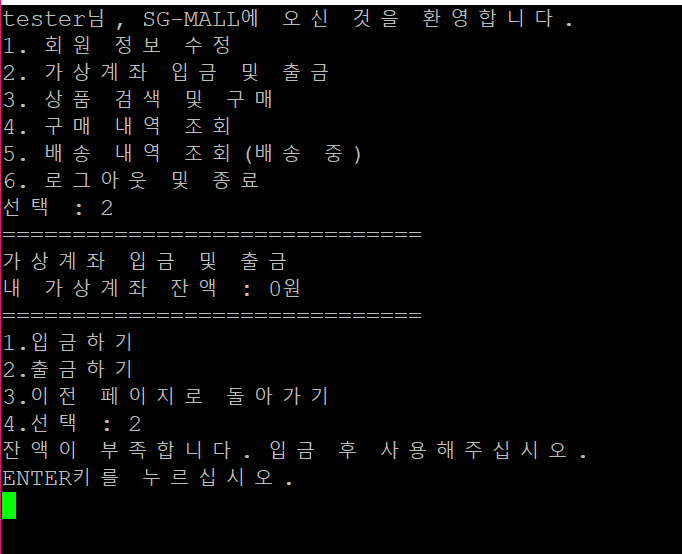


1번에서 회원 가입한 아이디와 비밀번호로 로그인

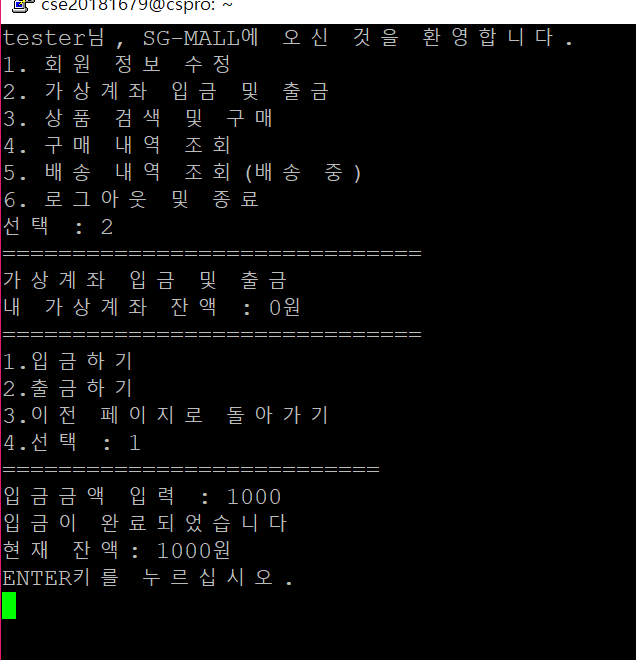


“tester님,”이라고 사용자 이름이 뜨고 로그인 성공 시 회원페이지로 넘어옴

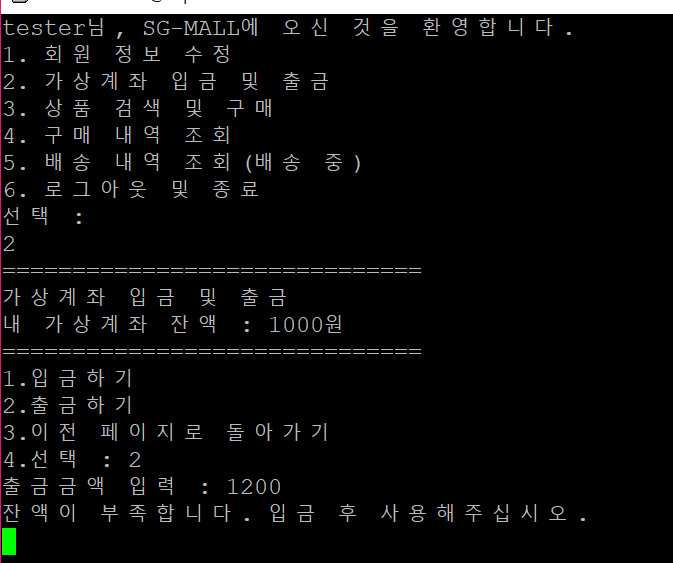
3. 가상계좌 입금 및 출금



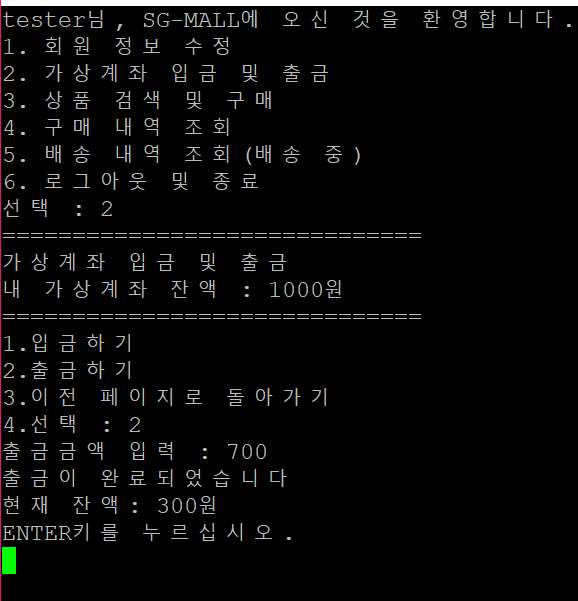
잔고 0에서 출금하기 불가능 -> 엔터 누르면 회원페이지로 넘어감



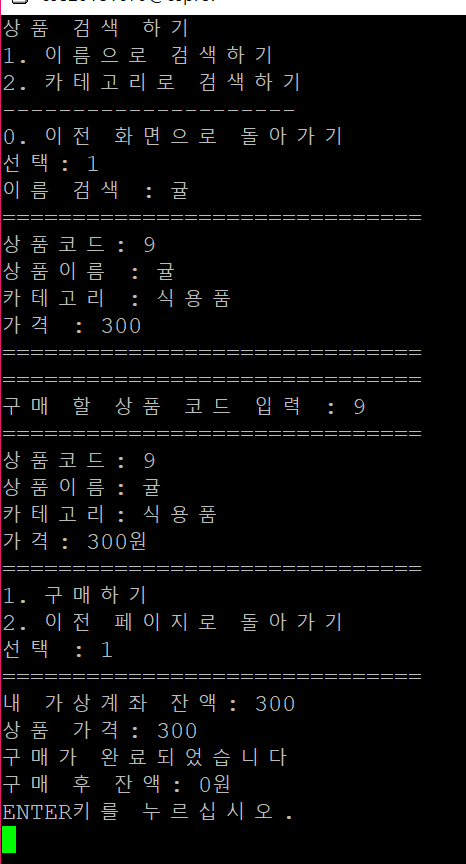
입금 1000원 -> 엔터키 누르면 회원페이지로 넘어감



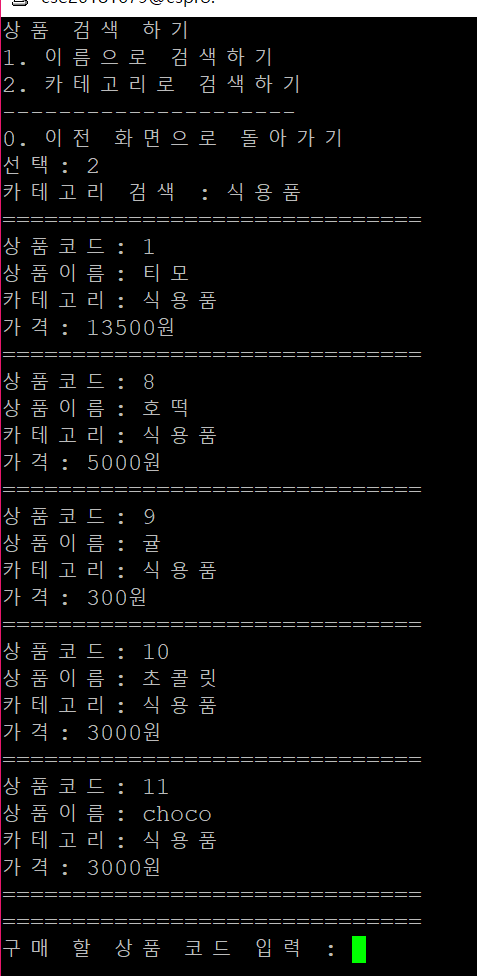
출금 1200원 -> 가상계좌 잔액 보다 많기 때문에 0원일때와 마찬가지로 출금 불가능

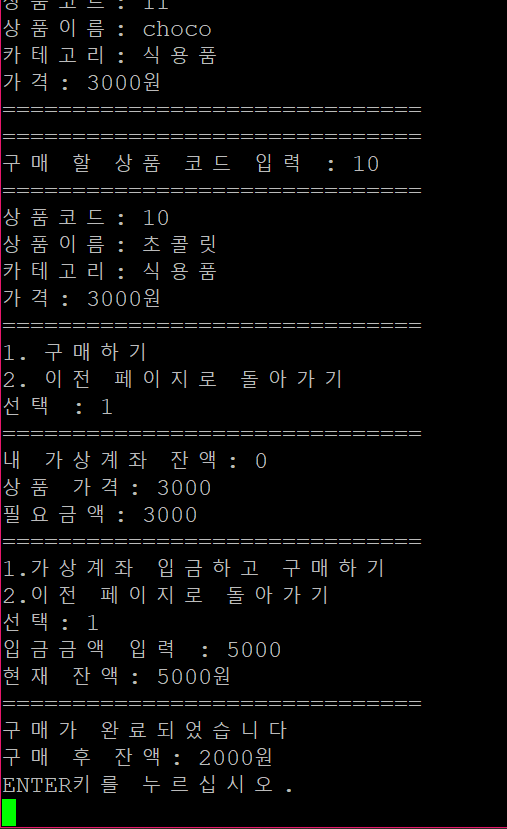


출금 700원 -> 현재 잔액 보여주고 end ->엔터키 누르면 회원페이지로 돌아감

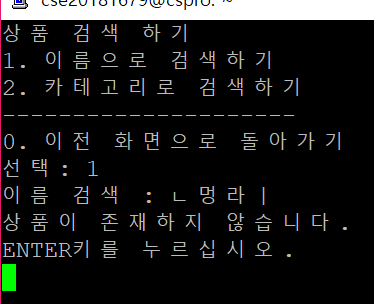


상품 이름 검색 -> 상품 코드 입력 -> 구매하기

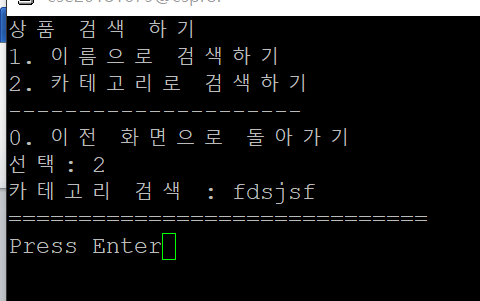




상품 카테고리 검색 -> 상품 코드 입력 -> 구매하기(잔고가 부족할 경우의 상황)

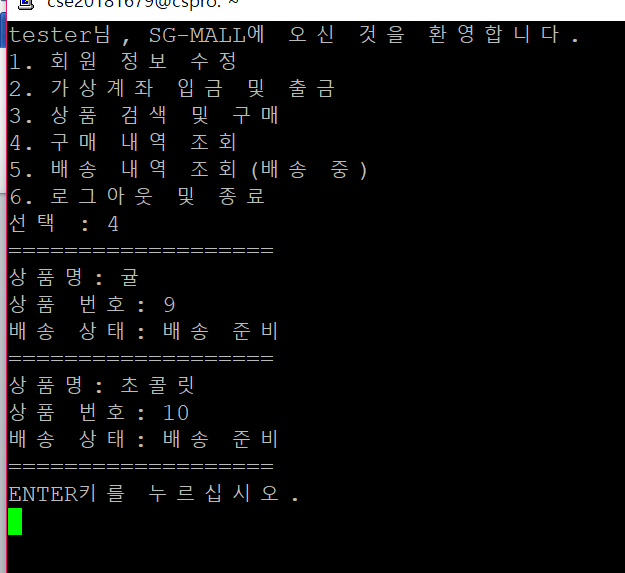


없는 이름 검색 시



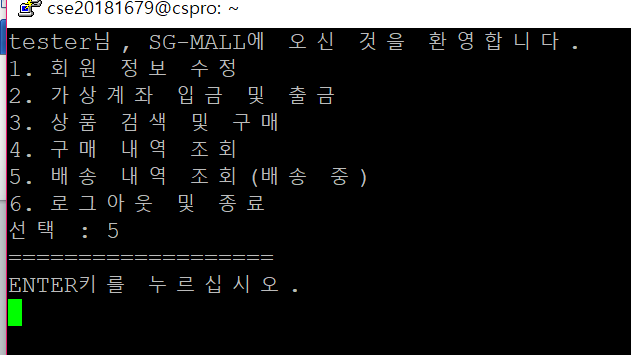
없는 카테고리 이름 입력 시

5. 구매내역 조회



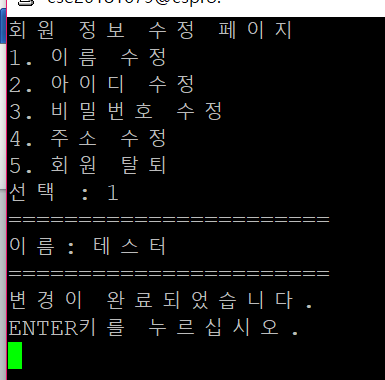
시험에서 5번째 단계 -> 아직 배송 준비 상태

6. 배송내역 조회

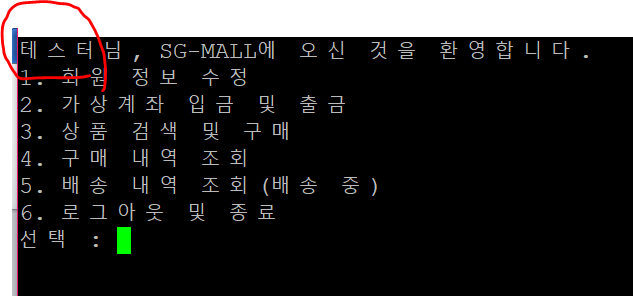


시험에서 6번째 단계 -> 아직 배송 준비 상태

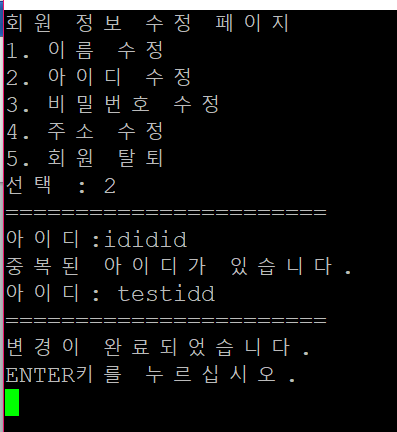
7. 회원 정보 수정



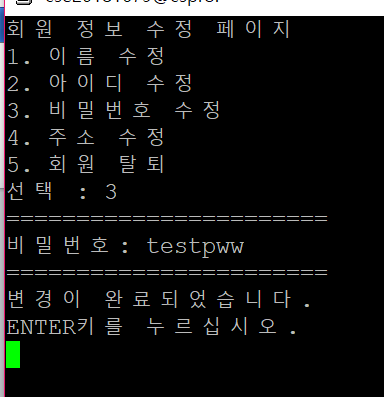
아름 수정



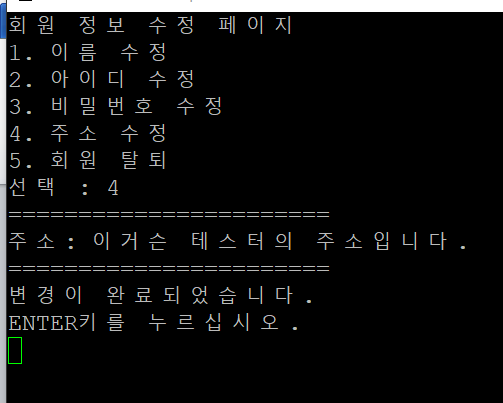
변경 된 부분 확인 가능



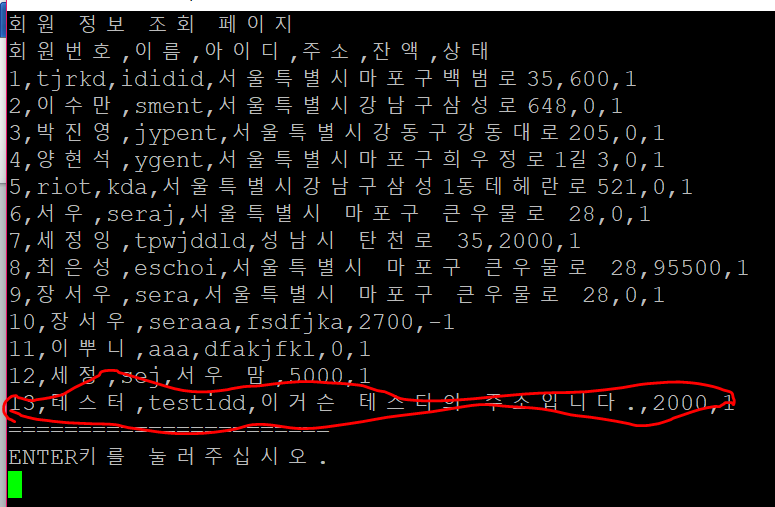
아이디 수정 (아이디 중복 체크 포함)



비밀번호 수정

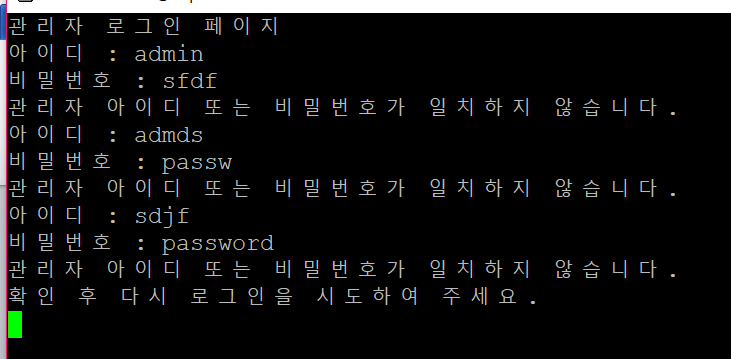


주소 수정

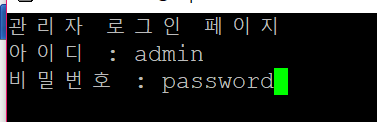


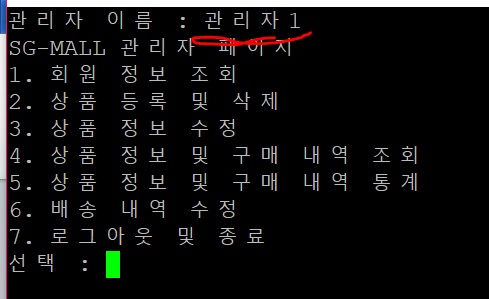
(관리자 회원 정보 조회 부분에서 바뀐 거 확인가능)

8. 관리자 로그인



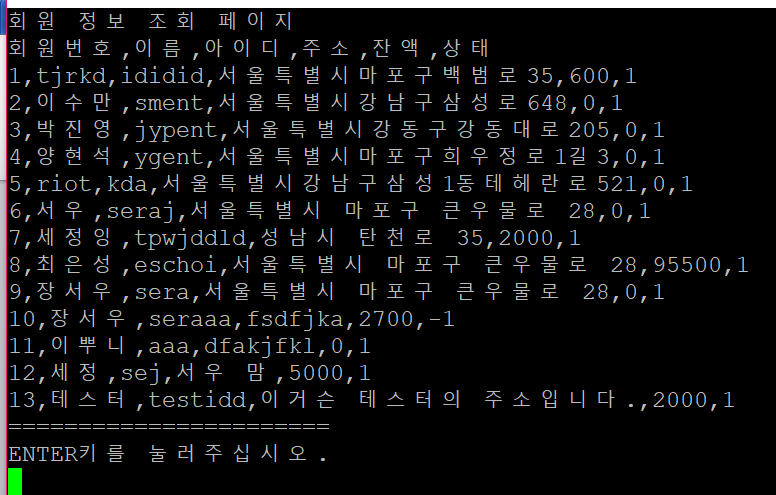
3회이상 잘못 입력 시





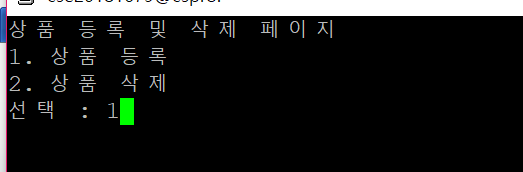
로그인 성공 시

9. 회원정보 조회

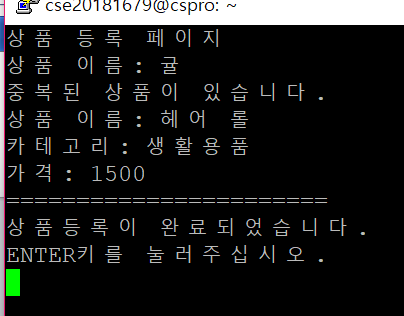


customer.csv에 있는 모든 회원 정보 조회

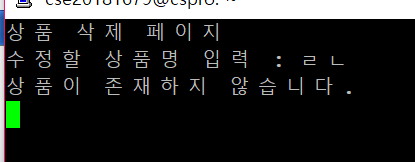
10. 상품 등록 및 삭제



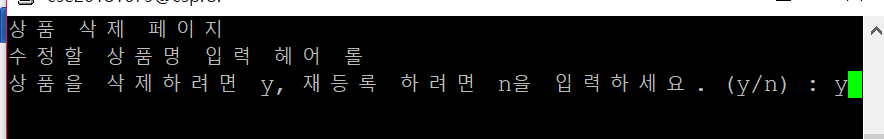
상품 등록



이름 중복체크 포함

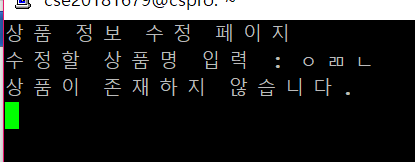


상품 삭제 (일치하는 상품 없을 경우)

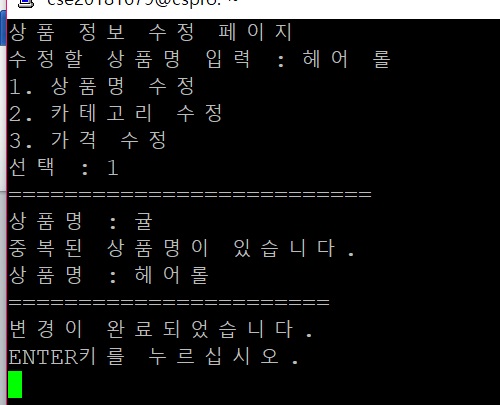


상품 삭제

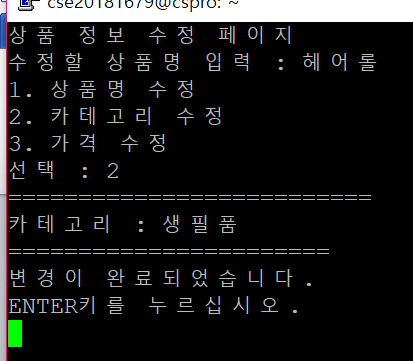
11. 상품 정보 수정



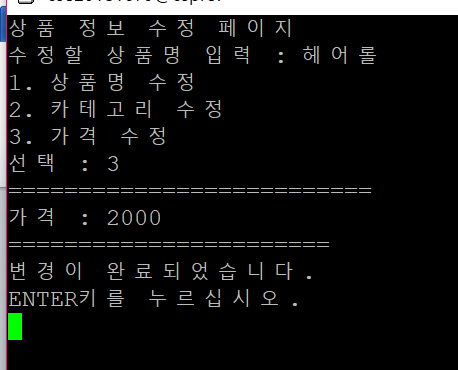
없는 상품명 입력 시



이름 수정 (이름 중복 체크)

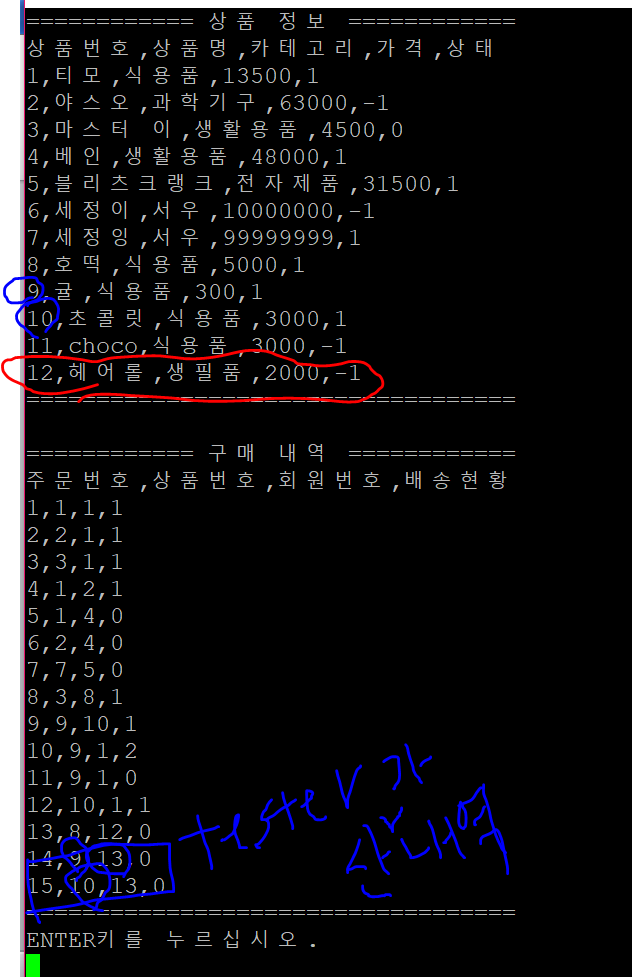


카테고리 수정



가격 수정

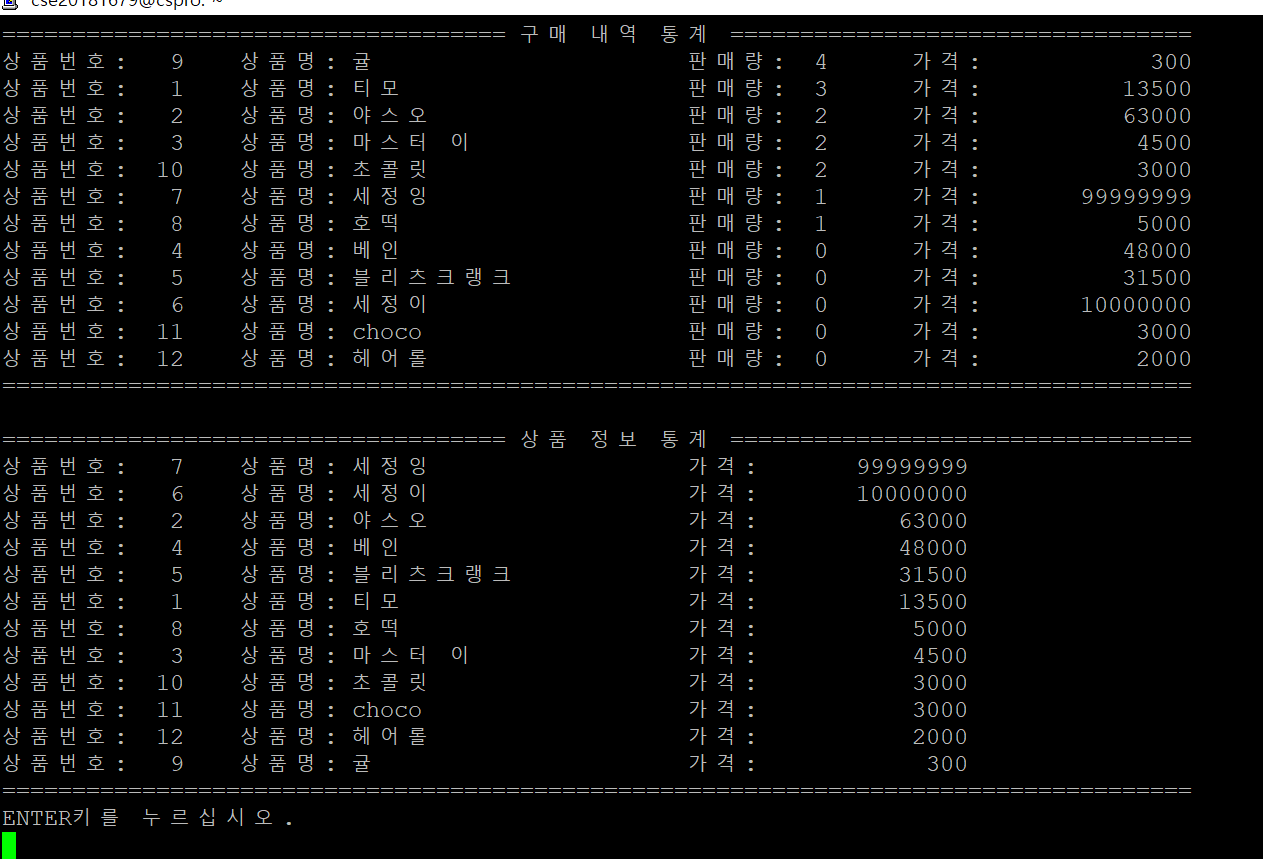
12. 상품 정보 및 구매 내역 조회



헤어 롤 정보 수정한 거 확인 + 상품 삭제한 거 확인

tester가 산 구매내역 확인

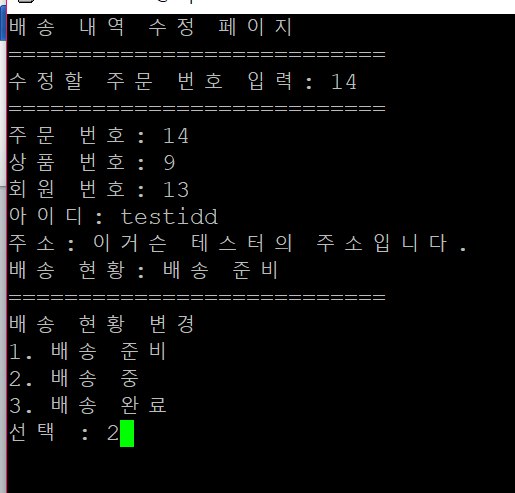
13. 상품 정보 및 구매 내역 통계(구매 내역 조회 했으므로 구매 내역 통계 맞는지 확인/상품 정보도)

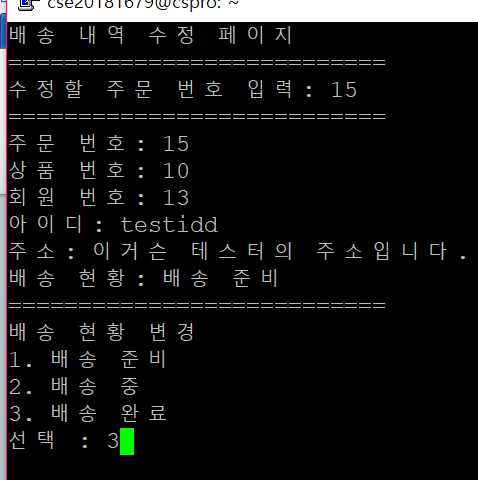


아름다운 결과를 볼 수 있음

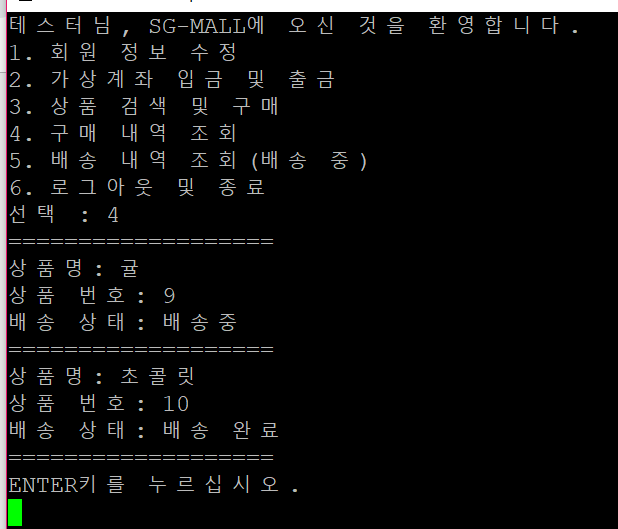
상품 정보 통계는 가격 순으로 구매 내역 통계는 판매량 순으로 정렬

14. 배송 내역 수정(tester의 구매내역을 배송중으로 변경해줌-> 배송 내역 조회 다시 검사)

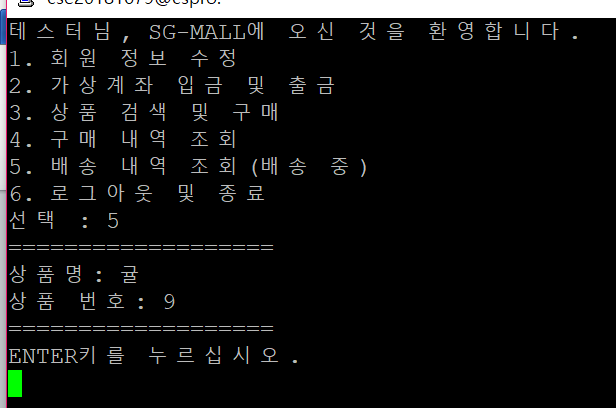




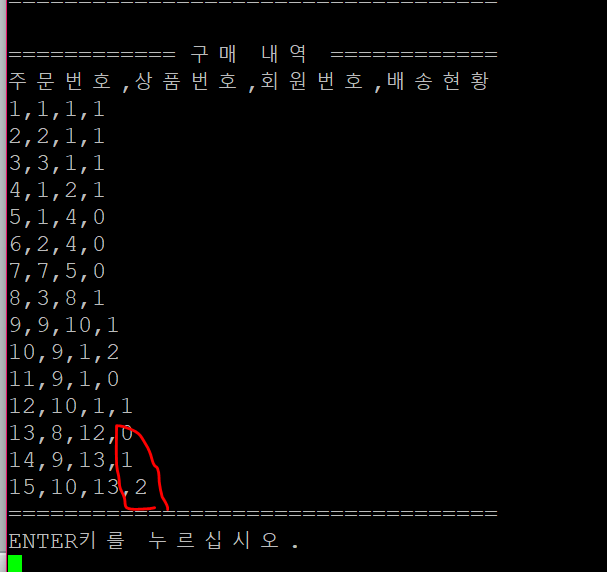
배송 현황 변경



이후 확인을 위해 testidd/testpww로 회원로그인 후 구매내역 조회

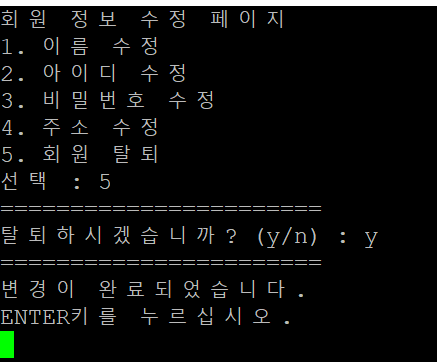


이후 확인을 위해 testidd/testpww로 회원로그인 후 배송내역 조회

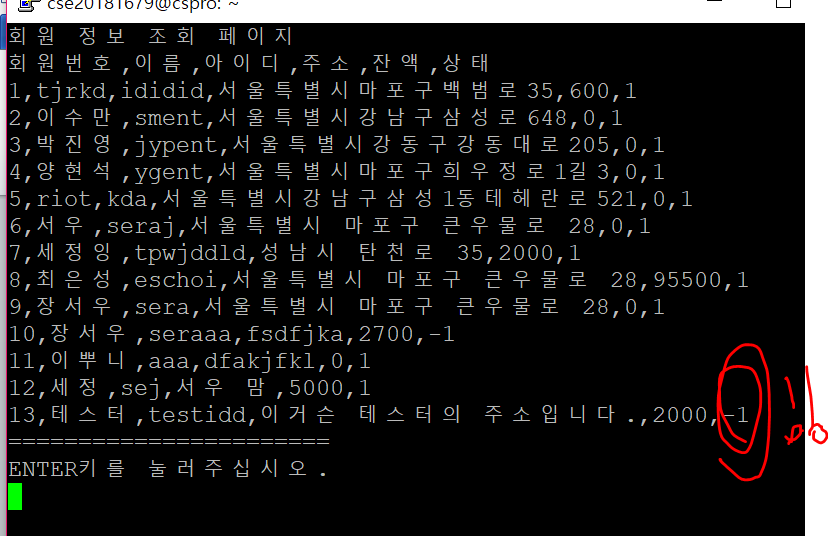


또는 관리자 구매내역조회에서 이런 식으로도 확인 가능

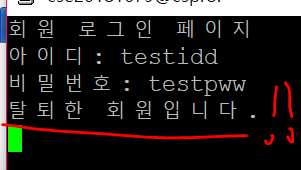
15. 회원 탈퇴



회원탈퇴 기능



이후 확인 하기 위해 관리자 로그인 해서 관리자 페이지에서 회원 정보 조회



또는 testidd/testpww로 회원로그인 시도

**2.6 보건 및 안정**

각 struct 내에서 정적 배열을 사용하여 글자 수에 제한을 두었다. scanf에서 띄어쓰기를 포함한 주소나 상품이름 등을 받을 경우에는 에딧셋을 사용하여 받았다. 관리자 구매 내역 통계부분에서 판매량 순대로 정렬을 하기 위해 리스트를 정렬한 후 리스트가 원래의 리스트로 돌아가야 다른 기능에서도 문제가 없기 때문에 다시 원상복구(상품번호로 정렬) 해주었다. 또한 관리자 상품 통계 부분에서도 상품 가격으로 정렬을 해준 후 다시 원래의 리스트로 원상복구 해주는 기능이 필요하였다.

**2.7 생산성과 내구성**

Makefile을 사용하였다. 각각 구조체마다 리스트를 만들었고, get \_ ~~뭐시기~~ \_data 함수를 사용하여서 쉽게 리스트를 불러올 수 있었다. 이중 구조체를 사용하였다.

**2.8 산업 표준**

gcc 5.4.0(cspro 이용) 와 makefile을 사용하였다. 또한 전역변수를 사용하지 않았다. 시스템에서 사용되는 모든 정보는 파일입출력을 통해 저장하였고 누락되는 정보가 없었다. 파일 입출력은 .csv를 사용하였고 파일의 행에서 열 간 구분은 ,를 사용하였다. 정적배열을 남용하지 않았다. (struct 내 정적배열은 허용되었다.)

**3. 기타**

**3.1 환경 구성**

운영체제는 Ubuntu 16.04를 사용했으며, gcc 5.4.0 버전을 사용하였다. 소스 코드는 모두 ansi C표준을 준수하여 개발하였다. 또한 텍스트 편집기로는 vi를 사용하였다. 사용한 라이브러리는 <stdio.h>, <stdlib.h>, <string.h>가 있다.

**3.2 참고 사항**

파일 입출력 <https://www.ibm.com/support/knowledgecenter/en/ssw_ibm_i_73/rtref/fopen.htm> 을 참고하였다.

**3.3 팀 구성**

개인 프로젝트로 진행한다.

**3.4 수행기간**

2018.11.14(수)~2018.12.14(금)