1.

/home/s21900706/2022OSS/mini\_project

2.

main.c 소스

#include "product.h"

#include "manager.h"

int selectProduct(){

int Product;

printf("\n\*\*\*\*\*\* 제품 판매 \*\*\*\*\*\*\n1. 제품 조회\n2. 제품 추가\n3. 제품 수정\n4. 제품 삭제\n0. 종료\n");

printf("\n=> 원하는 메뉴는? ");

scanf("%d", &Product);

return Product;

}

int main(void){

product sp[30];

int count = 0, a;

int index=0;

#ifdef DEBUG

printf("DEBUG MODE ON:\n");

#endif

while (1){

a = selectProduct();

if (a == 0) break;

if (a == 1 || a == 3 || a == 4) {

if (count==0) continue;

}

if (a == 1) {

listProduct(sp, index);

}

else if (a == 2) {

count += createProduct(&sp[index++]);

#ifdef DEBUG

printf("\nDEBUG MODE: Count added is: %d\n", count);

#endif

}

else if (a == 3) {

int no=selectDataNo(sp, index);

if (no==0) {

printf("=> 취소됨!\n");

continue;

}

updateProduct(&sp[no-1]);

}

else if (a == 4) {

int no=selectDataNo(sp, index);

if (no==0) {

printf("=> 취소됨!\n");

continue;

}

int deleteok;

printf("정말로 삭제하시겠습니까? (삭제 : 1)");

scanf("%d", &deleteok);

if (deleteok == 1) {

if (deleteProduct(&sp[no-1])) {

#ifdef DEBUG

printf("DEBUG MODE: Previous Count Number: %d\n", count);

#endif

count--;

#ifdef DEBUG

printf("DEBUG MODE: After Count Number: %d\n", count);

#endif

}

}

}

}

printf("종료됨!\n");

manager.c 소스

#include "manager.h"

void listProduct(product \*s, int count) {

printf("\n\*\*\*\*\*\*\*\*\* 제품 \*\*\*\*\*\*\*\*\*\*\n");

printf("\n번호\t제품이름\t\t제품정보\t\t무게 가격 배송방식\n");

printf("===================================================================================\n");

int a=count;

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

if (s[i].price==-1) {

a--;

continue;

}

printf("%2d", i+1);

readProduct(s[i]);

}

printf("\n총 %d개의 주문이 있습니다.\n", a);

}

int selectDataNo (product \*s, int count) {

int no;

listProduct(s, count);

printf("번호는 (취소 : 0)? ");

scanf("%d", &no);

return no;

}

manager.h 소스

#include "product.h"

int selectDataNo (product \*s, int count); // 특정 제품의 데이터 위치를 알아내기 위한 함수

void listProduct(product \*s, int count); // 현재있는 제품 알려주는 함수(삭제된 값은 스킵함)

~

Makefile 소스

CC = gcc

TARGET = main.c

SRCS = manager.c product.c

OBJS = $(SRCS:.c=.o)

EXE1 = main

EXE2 = main\_debug

$(EXE1) : $(OBJS)

$(CC) $(TARGET) $^ -o $@

%.o: %.c

$(CC) -c $(SRCS)

$(EXE2) : $(OBJS)

$(CC) -DDEBUG $(TARGET) $^ -o $@

clean :

rm -f $(OBJS) $(EXE1) $(EXE2)

3. mini project 폴더 ls -al 결과

텍스트, 전자기기이(가) 표시된 사진

자동 생성된 설명

makefile 컴파일 및 빌드 화면 사진

텍스트이(가) 표시된 사진

자동 생성된 설명

4.

사용자용 프로그램 실행결과 사진

**s21900706@walab-HGU**:**~/2022OSS/miniproject**$ ./main

\*\*\*\*\*\* 제품 판매 \*\*\*\*\*\*

1. 제품 조회

2. 제품 추가

3. 제품 수정

4. 제품 삭제

0. 종료

=> 원하는 메뉴는? 2

제품이름 ? 제주 귤

제품정보 ? 제주도산 한라봉

제품 무게 ? 10킬로

제품 가격 ? 100000

배송 방법 ? (0:새벽 배송, 1:택배 배송) 1

=> 제품 추가 완료 !

\*\*\*\*\*\* 제품 판매 \*\*\*\*\*\*

1. 제품 조회

2. 제품 추가

3. 제품 수정

4. 제품 삭제

0. 종료

=> 원하는 메뉴는? 2

제품이름 ? 안이천 쌀

제품정보 ? 국내산 이천 쌀

제품 무게 ?

30킬로

제품 가격 ? 50000

배송 방법 ? (0:새벽 배송, 1:택배 배송) 0

=> 제품 추가 완료 !

\*\*\*\*\*\* 제품 판매 \*\*\*\*\*\*

1. 제품 조회

2. 제품 추가

3. 제품 수정

4. 제품 삭제

0. 종료

=> 원하는 메뉴는? 1

\*\*\*\*\*\*\*\*\* 제품 \*\*\*\*\*\*\*\*\*\*

번호 제품이름 제품정보 무게 가격 배송방식

=================================================================

1 제주 귤 제주도산 한라봉 10킬로 100000 택배배송

2 이천 쌀 국내산 이천 쌀 30킬로 50000 새벽배송

총 2개의 주문이 있습니다.

\*\*\*\*\*\* 제품 판매 \*\*\*\*\*\*

1. 제품 조회

2. 제품 추가

3. 제품 수정

4. 제품 삭제

0. 종료

=> 원하는 메뉴는? 0

종료됨!

5.디버깅용 코드 사진 및 설명

텍스트이(가) 표시된 사진

자동 생성된 설명

해당 코드는 현재 실행 중인 프로그램이 디버깅용 프로그램임을 먼저 알려주기 위해 DEBUG MODE를 출력하게 하였다

텍스트이(가) 표시된 사진

자동 생성된 설명

createProduct는 사용자가 물건을 추가하고 싶을 경우 함수를 실행하여 이를 수행하고 완료시 return 1을 하는 함수이다. 따라서 물건을 추가할 때마다 count에 이 리턴값을 추가하여 현재 추가된 물건의 수를 확인한다. 따라서 createProduct 함수가 성공적으로 1을 리턴하여 count가 1씩 올랐는지 확인하기 위해 그 아래 디버깅코드로 count값을 출력하여 확인할 수 있도록 만들었다.

텍스트이(가) 표시된 사진

자동 생성된 설명

deleteProduct 함수는 사용자가 추가한 물건 중 삭제하고 싶은 경우 해당 물건 번호를 지정해 이를 이용해 특정 물건 정보를 함수로 보내 물건 가격을 -1로 바꾸어 나중에 확인할 때 삭제한 물건으로 인식하게 하는 함수이다. 이 함수가 성공적으로 수행되면 1의 값을 반환하는데 이는 true를 나타내므로 함수가 성공했으면 if(deleteProduct(&sp[no-1])의 경우는 실행하게 된다. 따라서 위의 디버깅 코드는 해당 함수의 성공 여부와 물건의 수가 줄었는지 확인하고자 추가하였다.