본인이 Github 계정에 만든 mini project repository 주소

현재 진행정도 Check List CRUD/MENU/다중데이터/파일 저장/파일 불러오기/검색1/검색2/검색3

CRUD -완료

MENU – 완료

다중데이터 – 완료

파일 저장 – 완료(data.txt)

불러오기 – 완료

검색1 – 완료

검색2 – 완료

검색3 – 완료

모든 소스파일 텍스트 복&붙 제출

- **Makefile**

```
CC = gcc

shop : main.c manager.o product.o

        $(CC) -o $@ $^

clean :

        rm *.o shop
```

- **Makefile_macro**

```
CC = gcc

TARGET = shop
```

```
OBJECTS = main.o manager.o product.o

all : $(TARGET)

$(TARGET) : $(OBJECTS)

        $(CC) -o $@ $^

clean:

        rm *.o shop
```

- **Main.c**

```
#include "product.h"

#include "manager.h"


int main() {

        Product plist[100];

        int index = 0;

        int count = 0 , menu;


        count = loadProduct(plist);

        index = count;

        if(count == 0)

                printf("=> NO FILE!!");

        else

                printf("=> LOAD SUCCESS!!");
```

```c
while(1) {

    menu = selectProduct(&plist);

    if(menu == 0) break;

    if(menu == 1 || menu == 3 || menu == 4)

        if(count == 0) continue;

    if(menu == 1) {

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

        printf("No\t Fruit\t Price\t Weight\t Grade\t Stars\n");

        if(count > 0) listProduct(plist, index);

    }
    else if(menu == 2) {

        count += addProduct(&plist[index++]);

    }
    else if(menu == 3) {

        int no = selectDataNum(plist, index);

        if( no == 0 ) {

            printf("=> Canceled!!");

            continue;

        }

        updateProduct(&plist[no-1]);

    }
    else if(menu == 4) {

        int no = selectDataNum(plist, index);

        if(no == 0) {

            printf("=> Canceld!!!");
```

```c
					continue;

				}

				int delok;

				printf("=> READY TO DELETE? (1) ");

				scanf("%d", &delok);

					if(delok == 1) {

							if(deleteProduct(&plist[no-1]))

							count--;

							printf("=> DELETED!!");

					}

		}

		else if(menu == 5) {

				saveProduct(plist, index);

		}

		else if(menu == 6) {

				searchProduct(plist, index);

		}

	}

	return 0;

}
```

- **Manager.c**

```c
#include "manager.h"

#include "product.h"
```

```c
int loadProduct(Product *p) {

        int count = 0, i = 0;

        FILE *fp;

        fp = fopen("data.txt", "rt");


        if(fp == NULL) return 0;

        else {

                for(i = 0 ; i < 100 ; i++) {

                        fscanf(fp, "%s", p[i].name);

                        if(feof(fp)) break;

                        fscanf(fp, "%d", &p[i].price);

                        fscanf(fp, "%d", &p[i].weight);

                        fscanf(fp, "%d", &p[i].star_grade);

                        fscanf(fp, "%d", &p[i].count_star);

                }

                fclose(fp);

                return i;

        }

}


int selectDataNum(Product *p, int index) {

        int no;

        listProduct(p, index);

        printf("Select Number :");

        scanf("%d", &no);

        return no;
```

```c
}


int listProduct(Product *p, int index) {

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

                if(p[i].price == -1) continue;

                printf("%d\t", i+1);

                readProduct(&p[i]);

        }

}


int saveProduct(Product *p, int index) {

        FILE *fp;

        fp = fopen("data.txt", "wt");


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

                if(p[i].price == -1) continue;

                fprintf(fp, "%s %d %d %d %d\n", p[i].name, p[i].price, p[i].weight, p[i].star_grade,
p[i].count_star);

        }

        fclose(fp);

        printf("=> Saved!!!\n");

}


int searchProduct(Product *p, int index) {

        int scnt = 0;

        char search[20];
```

```c
        printf("Search the Fruit? :");

        scanf("%s", search);


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

                if(p[i].price == -1) continue;

                if(strstr(p[i].name, search)) {


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

                        printf("No\t Fruit\t Price\t Weight\t Grade\t Stars\n");

                        printf("%d\t", i+1);

                        readProduct(&p[i]);

                        scnt++;

                }
        }
        if(scnt == 0) printf("=> No Data!!\n");
}
```

- **Manager.h**

```c
#include "product.h"


int loadProduct(Product *p);

int selectDataNum(Product *p, int index);

int listProduct(Product *p, int index);

int saveProduct(Product *p, int index);

int searchProduct(Product *p, int index);
```

- **Product.c**

```c
#include "product.h"

int selectProduct(Product *p) {

        int menu;

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

        printf("1. List Menu\n");

        printf("2. Add Menu\n");

        printf("3. Update Menu\n");

        printf("4. Delete Menu\n");

        printf("5. Save Menu\n");

        printf("6. Search Fruit\n");

        printf("0. Exit the program\n");

        printf("Enter the Menu:   ");

        scanf("%d", &menu);

        printf("\n");


        return menu;

}


int readProduct(Product *p) {

        printf("%s\t %d\t %d\t %d\t %d\n\n", p->name, p->price, p->weight, p->star_grade,
p->count_star);

        return 1;

}
```

```c
int addProduct(Product *p) {

        printf("Fruit?: ");

        scanf("%s", p->name);

        printf("Price?: ");

        scanf("%d", &p->price);

        printf("Weight?: ");

        scanf("%d", &p->weight);

        printf("Grade?: ");

        scanf("%d", &p->star_grade);

        printf("Star?: ");

        scanf("%d", &p->count_star);

        return 1;

}


int updateProduct(Product *p) {

        printf("Fruit?: ");

        scanf("%s", p->name);

        printf("Price?: ");

        scanf("%d", &p->price);

        printf("Weight?: ");

        scanf("%d", &p->weight);

        printf("Grade?: ");

        scanf("%d", &p->star_grade);

        printf("Star?: ");

        scanf("%d", &p->count_star);
```

```c
        printf("Update Success!!!\n\n");


        return 1;

}


int deleteProduct(Product *p) {

        strcpy(p->name, "NULL");

        p->price = -1;

        p->weight = -1;

        p->star_grade = -1;

        p->count_star = -1;


        return 1;

}
```

- **Product.h**

```c
#ifndef _HYUN__

#define _HYUN__

#include <stdio.h>

#include <string.h>


typedef struct {

        char name[20];

        int weight;

        int price;
```

```
        int star_grade;

        int count_star;

} Product;


int selectProduct();

int readProduct(Product *p);

int addProduct(Product *p);

int updateProduct(Product *p);

int deleteProduct(Product *p);

#endif
```

- **Data.txt**

Apple 5000 800 5 14323

Banana 7500 300 4 1232

Orange 8000 1300 4 43211

모든 메뉴 실행결과 텍스트 복&붙 제출

=> LOAD SUCCESS!!

****** Menu *********

1. List Menu

2. Add Menu

3. Update Menu

4. Delete Menu

5. Save Menu

6. Search Fruit

0. Exit the program

Enter the Menu: 1

==

No	Fruit	Price	Weight	Grade	Stars
1	Apple	5000	800	5	14323
2	Banana	7500	300	4	1232
3	Orange	8000	1300	4	43211

****** Menu *********

1. List Menu

2. Add Menu

3. Update Menu

4. Delete Menu

5. Save Menu

6. Search Fruit

0. Exit the program

Enter the Menu: 2

Fruit?: Kiwi

Price?: 19000

Weight?: 300

Grade?: 4

Star?: 12344

****** Menu *********

1. List Menu

2. Add Menu

3. Update Menu

4. Delete Menu

5. Save Menu

6. Search Fruit

0. Exit the program

Enter the Menu: 3

1	Apple	5000	800	5	14323
2	Banana	7500	300	4	1232
3	Orange	8000	1300	4	43211
4	Kiwi	19000	300	4	12344

Select Number :2

Fruit?: berry

Price?: 9000

Weight?: 200

Grade?: 5

Star?: 123

Update Success!!!

****** Menu *********

1. List Menu

2. Add Menu

3. Update Menu

4. Delete Menu

5. Save Menu

6. Search Fruit

0. Exit the program

Enter the Menu: 1

==

No	Fruit	Price	Weight	Grade	Stars
1	Apple	5000	800	5	14323

2	berry	9000	200	5	123
3	Orange	8000	1300	4	43211
4	Kiwi	19000	300	4	12344

****** Menu *********

1. List Menu

2. Add Menu

3. Update Menu

4. Delete Menu

5. Save Menu

6. Search Fruit

0. Exit the program

Enter the Menu: 4

1	Apple	5000	800	5	14323
2	berry	9000	200	5	123
3	Orange	8000	1300	4	43211
4	Kiwi	19000	300	4	12344

Select Number :3

=> READY TO DELETE? (0) 1

=> DELETED!!

****** Menu *********

1. List Menu

2. Add Menu

3. Update Menu

4. Delete Menu

5. Save Menu

6. Search Fruit

0. Exit the program

Enter the Menu: 5

=> Saved!!!

****** Menu *********

1. List Menu

2. Add Menu

3. Update Menu

4. Delete Menu

5. Save Menu

6. Search Fruit

0. Exit the program

Enter the Menu: 1

==

No   Fruit   Price   Weight   Grade   Stars

1   Apple   5000   800   5   14323

2   berry   9000   200   5   123

4   Kiwi   19000   300   4   12344

****** Menu *********

1. List Menu

2. Add Menu

3. Update Menu

4. Delete Menu

5. Save Menu

6. Search Fruit

0. Exit the program

Enter the Menu: 6

Search the Fruit? :Kiwi

```
==================================================

No          Fruit    Price    Weight  Grade    Stars

4           Kiwi     19000    300       4       12344
```

****** Menu *********

1. List Menu

2. Add Menu

3. Update Menu

4. Delete Menu

5. Save Menu

6. Search Fruit

0. Exit the program

Enter the Menu: 6

Search the Fruit? :Apple

```
==================================================

No          Fruit    Price    Weight  Grade    Stars

1           Apple    5000     800       5       14323
```

****** Menu *********

1. List Menu

2. Add Menu

3. Update Menu

4. Delete Menu

5. Save Menu

6. Search Fruit

0. Exit the program

Enter the Menu: 0