#include<stdio.h> //헤더파일 stdio.h포함

#include<string.h>//헤더파일 string.h포함

#define NAME\_LEN 20

#define TEL\_LEN 20

#define ADDR\_LEN 30

#define DB\_SIZE 100

#define MAX 100

struct person

{

char name[DB\_SIZE][NAME\_LEN];

char addr[DB\_SIZE][ADDR\_LEN];

char number[DB\_SIZE][TEL\_LEN];

};

struct person list[MAX]; //구조체 선언

int x = 0;

int select;

void InsertTel(); //등록 함수 원형

void DeleteTel(); //삭제 함수 원형

void SearchTel(); //검색 함수 원형

void PrintTel(); //출력 함수 원형

void ExitProg(); //종료 함수 원형

void PrintMenu(); //메뉴 함수 원형

void LoadData(); //데이터 불러오는 함수 원형

void StoreData(); //데이터 저장 함수 원형

int main(void)

{

int i, menu = 0;

LoadData(); //데이터 불러오기

while (menu != 5)

{

PrintMenu();

scanf("%d", &menu);

switch (menu)

{

case 1:

InsertTel(); //등록

break;

case 2:

DeleteTel(); //삭제

break;

case 3:

SearchTel(); //찾기

break;

case 4:

PrintTel(); //출력

break;

case 5:

ExitProg();

StoreData(); //데이터 저장

return 0;

//종료 함수 사용

default: //case문의 경우의 수 이외의 값이 발생할때 실행

printf("다시입력하세요[1-5]\n");

}

menu = 0; //변수 초기화

while (getchar() != '\n');

}

return 0; //0을 반환하여서 종료

}

void PrintMenu() //메뉴 함수 정의

{

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

printf("1.등록<Insert>\n");

printf("2.삭제<Delete>\n");

printf("3.검색<Search>\n");

printf("4.출력<Print All>\n");

printf("5.종료<Exit>\n");

printf("-------------------\n");

printf("메뉴를 선택하세요[1-5]:");

}

void InsertTel() //등록 함수 정의

{

while (getchar() != '\n');

if (x < DB\_SIZE) {

printf("[INSERT]\n");

printf("이름을 입력하시오 :");

gets\_s(list[x].name, 20);

printf("전화번호를 입력하시오:");

gets\_s(list[x].number, 20);

printf("주소를 입력하시오:");

gets\_s(list[x].addr, 30);

printf("정상적으로 저장되었습니다.\n");

x++;

}

}

void DeleteTel() //삭제 함수 정의

{

char del[20];

int count = 0;

int n, i;

printf("[DELETE]\n");

if (x > 0)

{

printf("NAME:");

scanf("%s", del);

for (n = 0; n < x; n++)

{

if (strcmp(list[n].name, del) == 0) //입력받은 del 변수의 값과 구조체 안의 name배열이 일치하는지 검사하고 일치한다면 삭제후 한칸 앞당긴후 x--

{ //strcmp 비교하는 함수

count++;

printf("DELETE SUCCESS\n");

for (i = n; i < x; i++)

list[i] = list[i + 1];

x--;

}

}

if (count == 0)

{

printf("일치하는 데이터가 없습니다.\n"); //일치하지 않으면 일치하지 않음 문구 출력

}

}

if (x == 0)

printf("NO DATA\n"); //저장된 사람이 1명도 없으면 데이터 없음 출력

}

void SearchTel() //검색 함수 정의

{

char sname[20];

int count = 0;

int h;

if (x > 0)

{

while (getchar() != '\n');

printf("[SEARCH]\n");

printf("NAME:");

gets\_s(sname, 20);

for (h = 0; h < x; h++) //입력받은 변수 sname이 name , number , addr 배열에 일치하는 것이 있는지 비교

{

if (strstr(list[h].name, sname) != NULL) {

count++;

printf("이름 : %-20s / 전화번호 : %-20s / 주소 : %-20s\n", list[h].name, list[h].number, list[h].addr);

}

else if (strstr(list[h].number, sname) != NULL) {

count++;

printf("이름 : %-20s / 전화번호 : %-20s / 주소 : %-20s\n", list[h].name, list[h].number, list[h].addr);

}

else if (strstr(list[h].addr, sname) != NULL) {

count++;

printf("이름 : %-20s / 전화번호 : %-20s / 주소 : %-20s\n", list[h].name, list[h].number, list[h].addr);

}

}

}

if (count == 0)

printf("NO DATA\n");

}

void PrintTel() //출력 함수 정의

{

int c = 0;

if (x > 0) {

for (c = 0; c < x; c++) {

if (list[c].name[0] != 0)

{

printf("이름:%-20s /전화번호:%-20s /주소:%-30s \n", list[c].name, list[c].number, list[c].addr);

}

}

}

if (x == 0)

printf("NO DATA\n ");

}

void ExitProg() //종료 함수 정의

{

printf("[EXIT]\n");

}

void LoadData()

{

FILE\* fp = fopen("input.txt", "r"); //파일 읽기모드로 열기

if (fp == NULL) //파일 없으면 열기 실패

{

printf("OPEN FAILED\n");

return;

}

else //파일이 있으면

{

while (!feof(fp)) //파일을이 끝날때까지 반복

{

fscanf(fp, "%s %s %s", list[x].name, list[x].number, list[x].addr); /\* 파일에 내용을

name nuber addr 배열에 저장\*/

x++; //파일에 한줄을 읽으면 x 는 1증가

}

x--;

fclose(fp); //파일 닫기

}

}

void StoreData()

{

int i;

FILE\* fp;

fp = fopen("input.txt", "w"); //파일 쓰기 모드로 열기

if (fp == NULL)

{

printf("OPEN FAILED\n"); //파일 없으면 열지 못했다고 출력

return;

}

else {

for (i = 0; i < x; i++) //입력받은 값들이 전부 저장

{

fprintf(fp, "%-20s %-20s %-30s \n", list[i].name, list[i].number, list[i].addr);

}

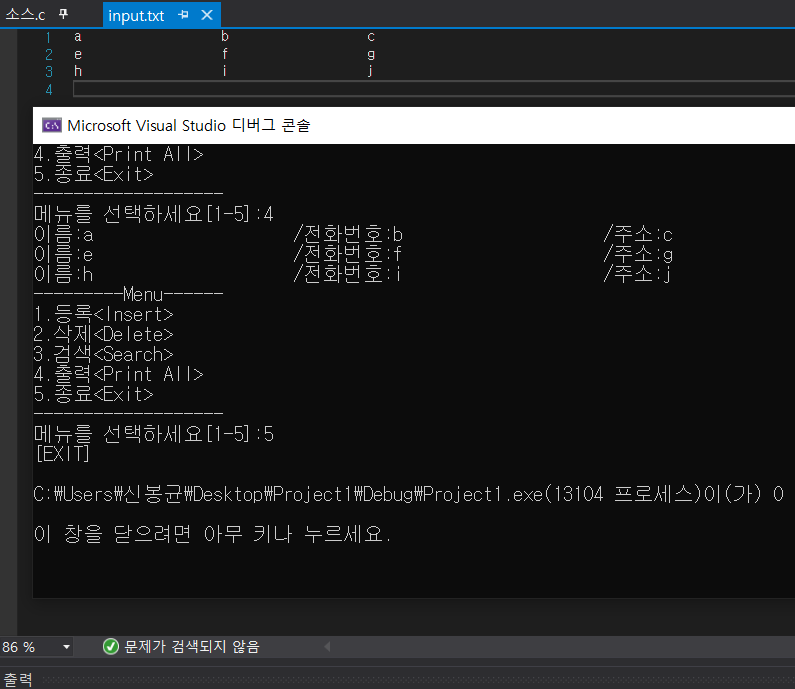
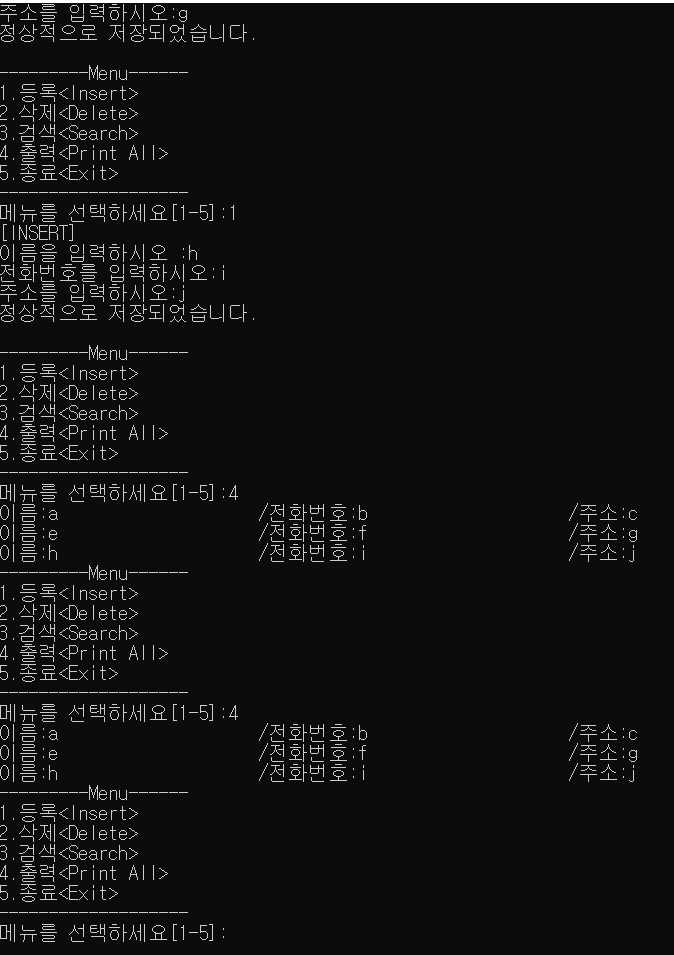
fclose(fp); //파일 닫기

}

}

실행결과

1. 데이터 입력 2. 종료시켜 데이터들을 input.txt 저장



3. 저장된 데이터들 불러오기

