

멀티방 객실 키오스크 프로그램

멀티방 이용자를 위한 객실 내 키오스크 프로그램 ver.1

안녕하세요
~!



목차



1 프로젝트 개요

- 프로젝트 구상 이유
- 플랫폼 사용 목적

2 팀 소개

- 팀원 소개
- 담당 파트 설명

3 각파트별 설명

- 도서 열람 파트 특징
- 영화 관람 파트 특징
- 게임 실행 파트 특징

Part 1,

프로젝트 개요

프로젝트 구상 이유와 플랫폼 사용 목적



최신 트렌드는

키오스크

이다.

1

프로젝트 구상

2

플랫폼 사용 목적

STEP 1

멀티방 점포의
감소 원인 추적.

>>

STEP 2

멀티방의 주문이 대부분 전화.

최근 젊은 층의 전화 기피.

인터폰 주문을 꺼림.

키오스크를 선호하는 경향.

>>

STEP 3

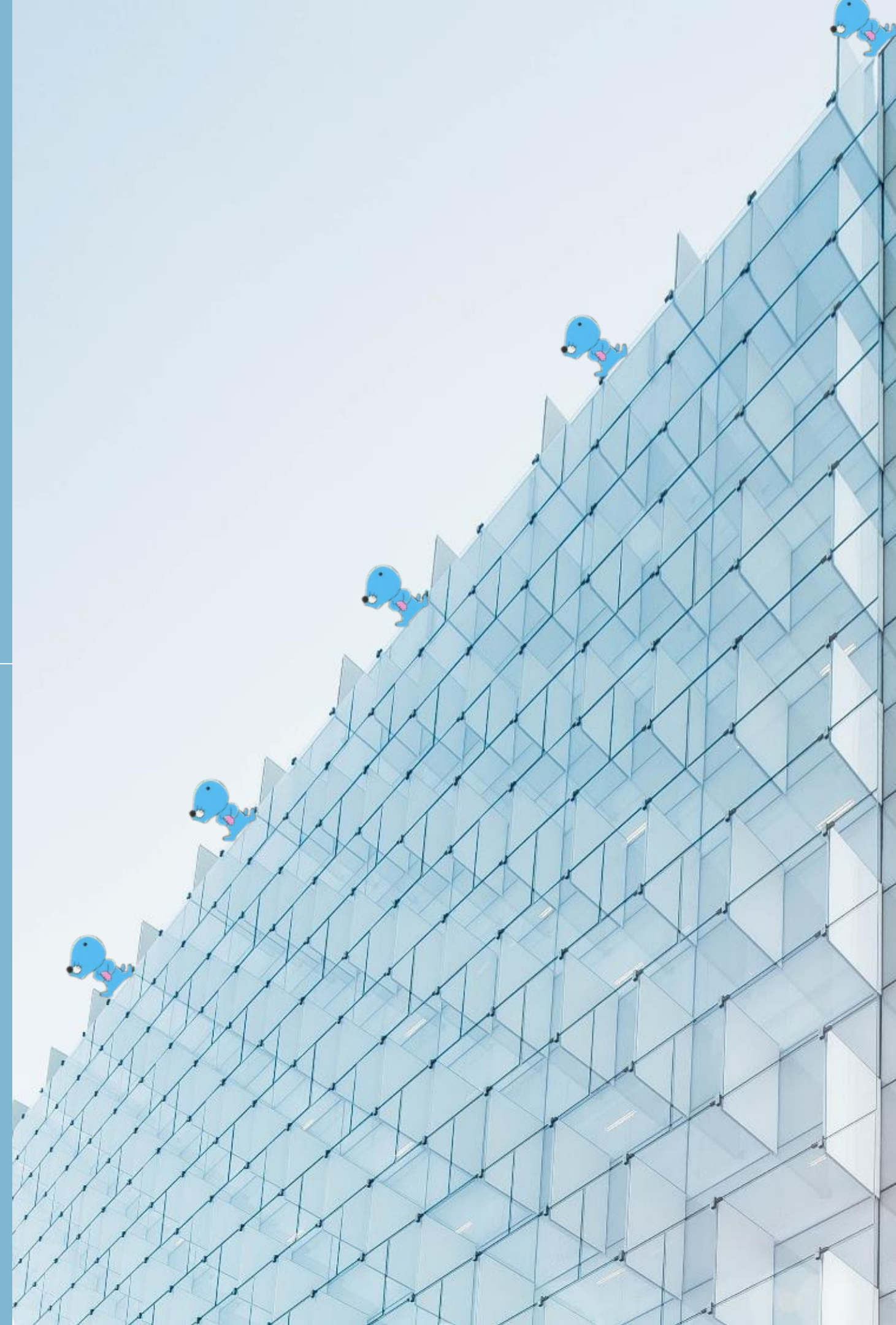
키오스크를 선호하는

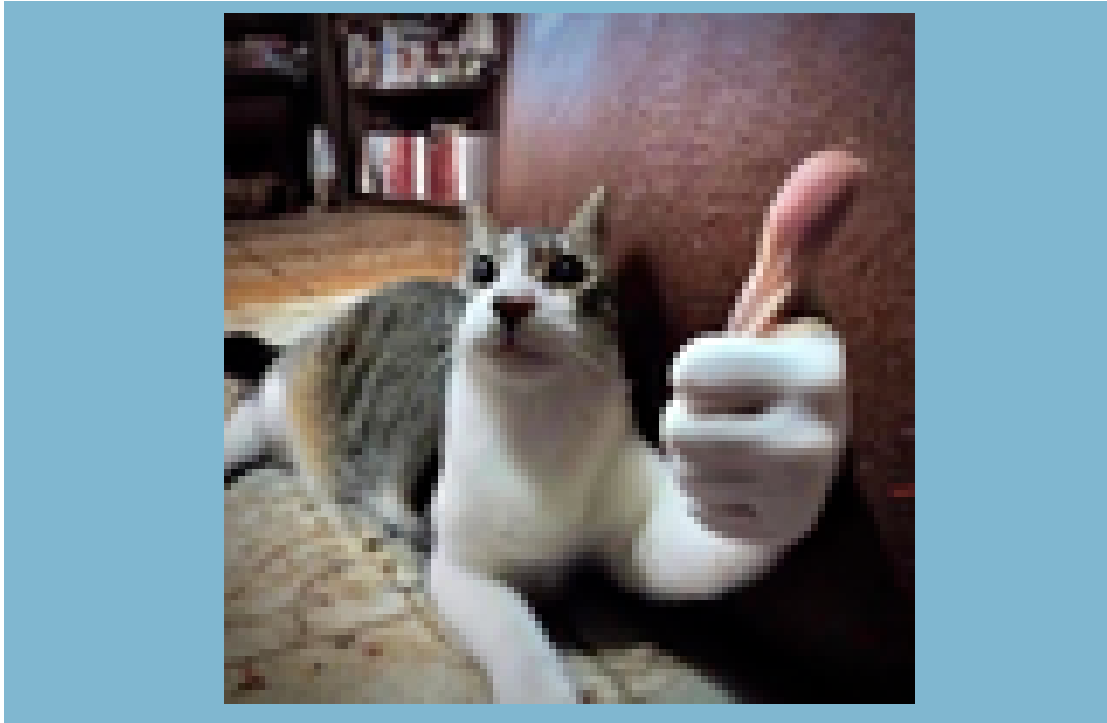
트랜드에 맞춘 프로그램 제작

Part 2,

팀 소개

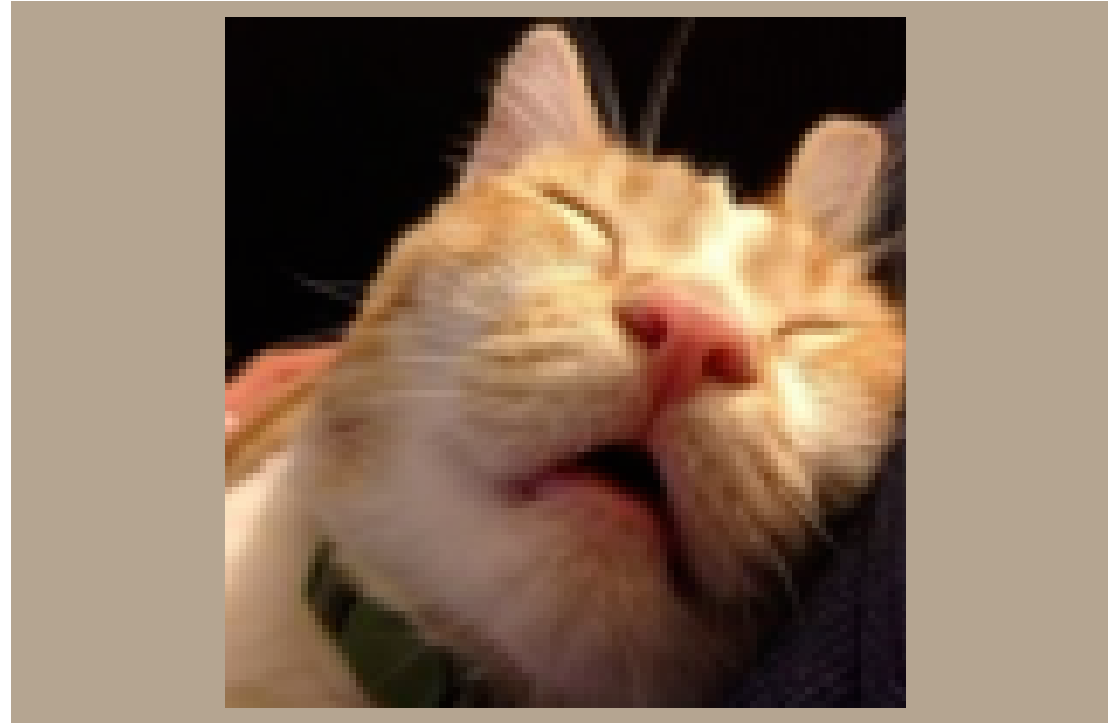
팀 소개 및 담당 파트 설명





이경민

- 도서 파트 및 PPT 담당
- 베스트셀러 조사 및 도서 추천
- 각 해당 코드 작성
- 코드 프레임 구상



유지웅

- 영화 파트 담당
- 박스오피스 조사 및 영화 추천
- 각 해당 코드 작성
- 헤더 파팅 어드바이스



권병진

- 게임 파트 및 메인 타이틀 담당
- 스네이크 게임 코드 작성
- 메인 타이틀 코드 작성
- 각 파트 어드바이스 및 총괄 정리

Part 3,

각 파트 별 설명

영화, 도서 그리고 게임 파트들에 대한 설명과 시연

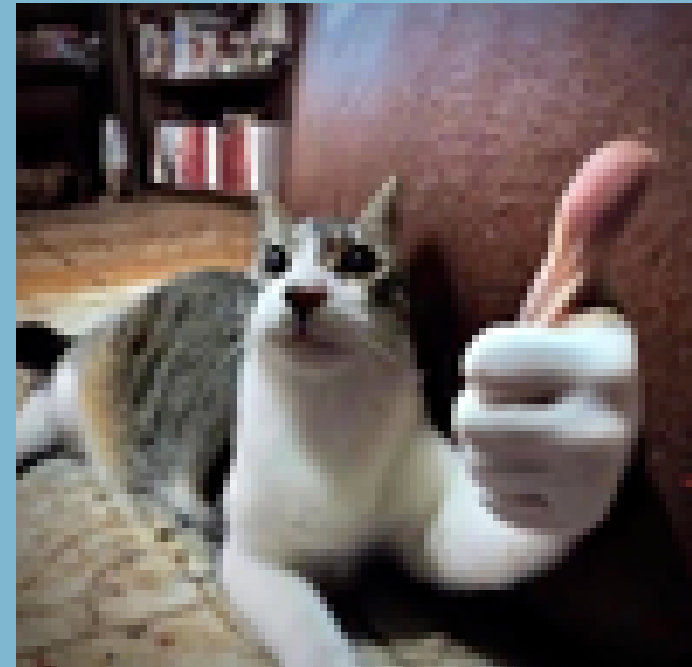


<발표 순서>



- 도서 파트

이경민



- 23년 베스트셀러 TOP 10 순위 도서 코드 작성
- 개발자 추천 도서 코드 작성
- 해당 데이터 파일 작성
- 헤더 파일 분리
- 사용언어 : C언어 (visual studio)

```
// 도서 Top10 데이터 저장 함수
void bookTopData()
{
    FILE* book_top10_data = fopen("book_top10_data.bin", "wb");
    BOOK data[10] = { 0 };
    if (book_top10_data == NULL)
    {
        puts("오류");
        exit(0);
    }
    for (int i = 0; i < 10; i++)
    {
        strcpy(data[i].book_name, book_name[i]);
        strcpy(data[i].book_writer, book_writer[i]);
        strcpy(data[i].book_category, book_category[i]);
        strcpy(data[i].book_count, book_count[i]);
        fwrite(&data[i], sizeof(BOOK), 1, book_top10_data);
    }
    fclose(book_top10_data);
}
```

[illegible]

```
srand(time(0));
int random = rand() % 5;
for (int i = 0; i < 5; i++)
{
    strcpy(data[i].recommend_book, recommend_book[random]);
    strcpy(data[i].recommend_writer, recommend_writer[random]);
    strcpy(data[i].recommend_comments, recommend_comments[random]);
    fwrite(&data[i], sizeof(BOOK), 1, book_recommend_data);
}
fclose(book_recommend_data);
```

- ▶ 폴더 헤더 파일
 - ▶ book_file_data.h
 - ▶ book_info.h

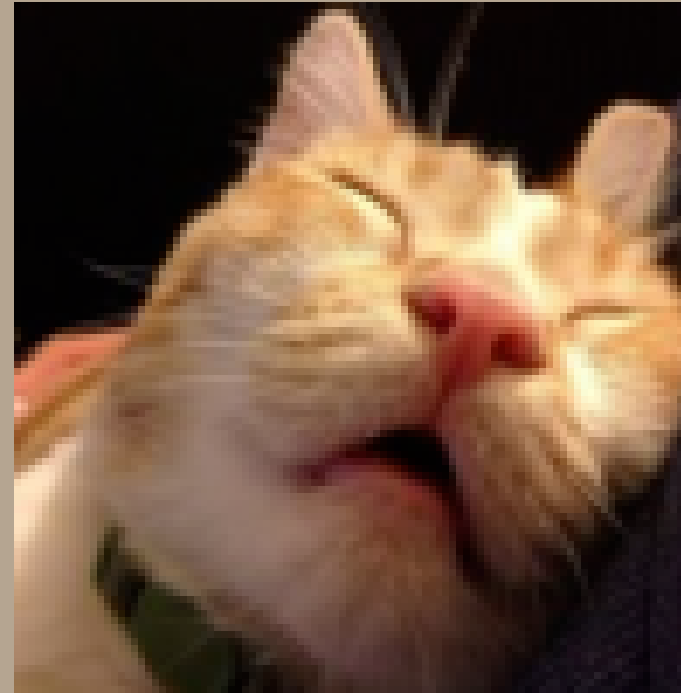
```
typedef struct BOOK
{
    char book_name[50];
    char book_writer[50];
    char book_category[50];
    char book_count[50];
    char recommend_book[50];
    char recommend_writer[50];
    char recommend_comments[1000];
} BOOK;

void bookTopData();
void bookTopRead();
void bookRecData();
void bookRecRead();
```

```
const char* book_name[] = { "스즈메의 문단속", "0  
"역행자", "만일 내가  
const char* book_writer[] = { "신카이 마코토", "중  
const char* book_category[] = { "일본소설", "한국  
const char* book_count[] = { "약 20만", "약 25만"  
const char* recommend_book[] = { "날개", "모든 것  
const char* recommend_writer[] = { "이상", "모리  
const char* recommend_comments[] = { "1936년에 발  
"기발한 무대
```


- 영화 파트

유지웅



- 국내 박스오피스 TOP 10 순위 도서 코드 작성
- 개발자 추천 영화 코드 작성
- 해당 데이터 파일 작성
- 헤더 파일 분리
- 사용언어 : C언어 (visual studio)

```
void enterMovieData()
```

```
{
FILE* movieData = fopen("movieData.bin", "wb");
MOVIE data[10] = { 0 };
if (movieData == NULL) {
    printf("파일 검색 오류!\n");
    exit(0);
}
for (int i = 0; i < 10; i++) {
    strcpy(data[i].movieName, movieName[i]);
    strcpy(data[i].directorName, directorName[i]);
    strcpy(data[i].phrase, phrase[i]);
    strcpy(data[i].count, count[i]);
}
```

[illegible]

```
}
printf("=====");
system("cls");
fclose(movieData);
```

```
void movieTop10()
```

```
{
    FILE* movieData = fopen("movieData.bin", "rb");
    MOVIE data = { 0 };
    if (movieData == NULL) {
        printf("파일 검색 오류!\n");
        exit(0);
    }
    int count = 1;
    if
```

[illegible]

Part 3, 각 파트 별 설명

- 게임 파트 & 메인 파트

권병진



행운이 오진 않지만
일단 귀엽습니다

- 스네이크 게임 코드 작성
- 메인 타이틀 코드 작성 및 내용 총괄 추합
- 해당 데이터 파일 작성
- 헤더 파일 분리
- 사용언어 : C언어 (visual studio)

Part 3, 각 파트 별 설명 • 게임 파트

```
#include <stdio.h>
#include <stdlib.h>
#include <conio.h>
#include <windows.h>
#include <string.h>
#include <math.h>

void Map();
void TempToE();
void Del();
void Food();
void Snake();
void GameOver();
void GameExplain();

int snakegame();

int main()
{
    Map();
    TempToE();
    Del();
    Food();
    Snake();
    GameOver();
    GameExplain();
    snakegame();
    return 0;
}

void Map()
{
    for (int i = 0; i < 20; i++)
    {
        for (int j = 0; j < 20; j++)
        {
            if (i == 0 || i == 19 || j == 0 || j == 19)
            {
                printf("%c", 'X');
            }
            else
            {
                printf(" ");
            }
        }
        printf("\n");
    }
}

void TempToE()
{
    int x = 10;
    int y = 0;
    system("cls");
    font_color(LIGHT_RED);
    gotoxy(x + 17, y + 2); printf("*****");
    gotoxy(x + 17, y + 3); printf(" ");
    gotoxy(x + 17, y + 4); printf(" ");
    gotoxy(x + 40, y + 4); printf("게임 오버!");
    gotoxy(x + 17, y + 5); printf(" ");
    gotoxy(x + 40, y + 5); printf("입구로 돌아갑니다.\n");
    gotoxy(x + 17, y + 6); printf(" ");
    gotoxy(x + 17, y + 7); printf("*****");
    system("pause");
    main();
}

void Del()
{
    bool isDel = false;
    for (int i = 0; i < 20; i++)
    {
        for (int j = 0; j < 20; j++)
        {
            if (body[i][j] == 'S')
            {
                isDel = true;
            }
        }
    }
    if (isDel)
    {
        for (int i = 0; i < 20; i++)
        {
            for (int j = 0; j < 20; j++)
            {
                body[i][j] = ' ';
            }
        }
    }
}

void Food()
{
    int x = 10;
    int y = 0;
    system("cls");
    font_color(LIGHT_RED);
    gotoxy(x + 17, y + 2); printf("*****");
    gotoxy(x + 17, y + 3); printf(" ");
    gotoxy(x + 17, y + 4); printf(" ");
    gotoxy(x + 40, y + 4); printf("게임 오버!");
    gotoxy(x + 17, y + 5); printf(" ");
    gotoxy(x + 40, y + 5); printf("입구로 돌아갑니다.\n");
    gotoxy(x + 17, y + 6); printf(" ");
    gotoxy(x + 17, y + 7); printf("*****");
    system("pause");
    main();
}

void Snake()
{
    int x = 10;
    int y = 0;
    system("cls");
    font_color(LIGHT_RED);
    gotoxy(x + 17, y + 2); printf("*****");
    gotoxy(x + 17, y + 3); printf(" ");
    gotoxy(x + 17, y + 4); printf(" ");
    gotoxy(x + 40, y + 4); printf("게임 오버!");
    gotoxy(x + 17, y + 5); printf(" ");
    gotoxy(x + 40, y + 5); printf("입구로 돌아갑니다.\n");
    gotoxy(x + 17, y + 6); printf(" ");
    gotoxy(x + 17, y + 7); printf("*****");
    system("pause");
    main();
}

void GameOver()
{
    if (body[17][17].x >= 38 || body[17][17].x <= 0)
    {
        int x = 10;
        int y = 0;
        system("cls");
        font_color(LIGHT_RED);
        gotoxy(x + 17, y + 2); printf("*****");
        gotoxy(x + 17, y + 3); printf(" ");
        gotoxy(x + 17, y + 4); printf(" ");
        gotoxy(x + 40, y + 4); printf("게임 오버!");
        gotoxy(x + 17, y + 5); printf(" ");
        gotoxy(x + 40, y + 5); printf("입구로 돌아갑니다.\n");
        gotoxy(x + 17, y + 6); printf(" ");
        gotoxy(x + 17, y + 7); printf("*****");
        system("pause");
        main();
    }
    if (body[17][17].y >= 19 || body[17][17].y <= 0)
    {
        int x = 10;
        int y = 0;
        system("cls");
        font_color(LIGHT_RED);
        gotoxy(x + 17, y + 2); printf("*****");
        gotoxy(x + 17, y + 3); printf(" ");
        gotoxy(x + 17, y + 4); printf(" ");
        gotoxy(x + 40, y + 4); printf("게임 오버!");
        gotoxy(x + 17, y + 5); printf(" ");
        gotoxy(x + 40, y + 5); printf("입구로 돌아갑니다.\n");
        gotoxy(x + 17, y + 6); printf(" ");
        gotoxy(x + 17, y + 7); printf("*****");
        system("pause");
        main();
    }
}

void GameExplain()
{
    gotoxy(70, 10);
    printf("*****");
    printf("게임 설명");
    printf("*****");
}

int snakegame()
{
    int key;
    Map();
    CursorView(0);
    Init();
    Output();
    FoodOutput();

    while (1)
    {
        GameExplain();

        if (_kbhit())
        {
            key = _getch();

            if (key == 'q')
            {
                break;
            }

            switch (key)
            {
                case UP:
                    if (l.Down == true)
                    {
                        l.Up = false;
                    }
                    else
                    {
                        l.Up = true;
                        l.Down = false;
                        l.Right = false;
                        l.Left = false;
                    }
                    Move();
                    break;
            }
        }
    }
}
```


Part 3, 각 파트 별 설명

- 메인 파트

```
int main()
{
    CursorView(NOCURSORS);
    font_color(LIGHT_RED);
    title();
    system("pause");
    system("cls");
    font_color(WHITE);
    switch (enterMenu()) {
    case 1: {
        while (1) {
            system("cls");
            switch (main_menu()) {
            case 1: {
                bookTopData();
                bookRecData();
                system("cls");
                switch (bookMenu()) {
                case 1: {
                    font_color(LIGHT_YELLOW);
                    bookTopRead();
                    break;
                }
                case 2: {
                    font_color(LIGHT_YELLOW);
                    bookRecRead();
                    break;
                }
                case 3: {
                    break;
                }
                default: {
                    printf("!!!!!!보기 내에서 선택해 주십시오.\n");
                    system("pause");
                    break;
                }
            }
        }
        break;
    }
}
```

[illegible]

Part 3, 각 파트 별 설명

- 타이틀 파트

[illegible]

```

char fColor[] = {
BLUE,
GREEN,
CYAN,
RED,
PURPLE,
YELLOW,
WHITE,
GRAY,
LIGHT_BLUE,
LIGHT_GREEN,
LIGHT_CYAN,
LIGHT_RED,
LIGHT_PURPLE,
LIGHT_YELLOW,
LIGHT_GRAY
};

srand(time(NULL));
while(1)
{
    int random = 0;
    random = rand() % 15;
    font_color(fColor[random]);
    gotoxy(x + 17, y + 2); printf("★=====★");
    gotoxy(x + 16, y + 3); printf(" $ Welcome To Our Multi Tasking Room $");
    gotoxy(x + 17, y + 4); printf("☆=====☆");
    gotoxy(x + 23, y + 6); printf("□□□□□□□□□□□□□□");
    gotoxy(x + 23, y + 7); printf("□          |          □");
    gotoxy(x + 23, y + 8); printf("□          |          □");
    gotoxy(x + 23, y + 9); printf("□          |          □");
    gotoxy(x + 23, y + 10); printf("□          ◎|◎          □");
    gotoxy(x + 23, y + 11); printf("□          |          □");
    gotoxy(x + 23, y + 12); printf("□          |          □");
    gotoxy(x + 23, y + 13); printf("□          |          □");
    Sleep(100);
    font_color(LIGHT_GREEN);
    gotoxy(x + 50, y + 16); printf(" 「 Multi Room v1.0 」 ");
    gotoxy(x + 50, y + 17); printf("개발자 : 「 이경민 」 , 「 권병진 」 , 「 유지웅 」 ");
    gotoxy(x + 50, y + 18); printf("문의 사항 : ☎ 010 - 6344 - 5463ㄴㄴㄴㄴㄴㄴㄴㄴ");
    if (_kbhit()) {
        int escape;
        escape = _getch();
        if (escape == 'q')
            break;
    }
}

```

코드 구현 테스트 영상

