

1. 과제 내용 : 영화 티켓박스 프로그램 구현

2. 수행 방법 : 개인 프로그램 작성

1) 프로젝트 명: 영문이니셜\_TICKET

3. 제출 항목 : 프로젝트 소스를 압축하여 제출

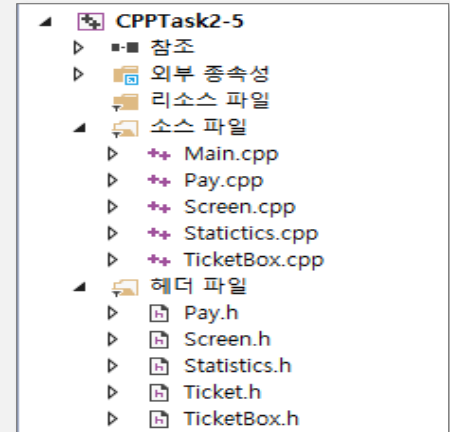
1) 압축 파일 명 : 이름\_TICKET.zip

2) 소스 설명서 : 이름\_TICKET.doc or hwp (5장 이하)

-> 코드: 객체지향 특성 설명 (가상함수, 추상 클래스, 다형성, 동적할당, 복사생성자등)

4. 제출 기한 : 6월 12일까지

5. 기능 설명 (총 8점)



기능(배점)		상영관 별 스크린 기능	프로젝트 파일
기본 (5점)	1	1. 영화 정보 보기 2. 좌석 예약 현황 보기	Main.cpp, TicketBox.cpp, Screen.cpp, Ticket.h, TicketBox.h, Screen.h
	1	좌석 예약 기능 (0.5)	
추가 (3점)	2	좌석 예약 취소 기능(0.5)	
	3	티켓 결제 기능(1.0)	Pay.cpp, Pay.h
	4	관리 통계 기능(1.0)	Statistics.cpp, Statistics.h

\* 추가 기능은 순서대로 구현할 것

## 기본기능 : Main.cpp 와 결과 화면

```
int main() {
    KpuTicketBox tBox;
    Screen * screen = NULL;
    bool bScreenMenu = true;
    int select = 0;

    tBox.Initialize(); // 3개의 스크린 객체 생성
    while (1) {
        if (bScreenMenu) {
            screen = tBox.selectMenu();
            bScreenMenu = false;
            if (!screen) return 0;
        }
        screen->showMovieMenu();
        cout << "메뉴를 선택하세요: ";
        cin >> select;      cout << endl;
        switch (select) {
            case 1:          // 영화 정보
                screen->showMovieInfo();
                break;
            case 2:          // 좌석 현황
                screen->showSeatMap();
                break;
            case 7:          // 스크린 메인 메뉴
                bScreenMenu = true;
                break;
            .....
        }
    }
    return 0;
}
```

Main.cpp

```
C:\WINDOWS\system32\cmd.exe

-----
상영관 메인 메뉴
1. 유럽 영화 1관
2. 아시아 영화 2관
3. 프리미엄 영화 3관

선택 (1~3) 그외 종료 : 1

-----
영화 메뉴 - 비포 선택 라미즈
1. 상영 영화 정보
2. 좌석 예약 현황
7. 메인 메뉴 이동

메뉴를 선택하세요 : 1

-----
[ 비포 선택 라미즈 ]
영화관 : 유럽 영화 1관
상영관 : 에디트 후크 (블리 랩)
출시일 : 제지와 절친 (11월 10일)
가작 : 10000

-----
영화 메뉴 - 비포 선택 라미즈
1. 상영 영화 정보
2. 좌석 예약 현황
7. 메인 메뉴 이동

메뉴를 선택하세요 : 2

-----
[ 좌석 예약 현황 ]
[ 1 ] [ 2 ] [ 3 ] [ 4 ] [ 5 ] [ 6 ] [ 7 ] [ 8 ] [ 9 ] [ 10 ]
[ 1 ] [ 1 ] [ 1 ] [ 1 ] [ 1 ] [ 1 ] [ 1 ] [ 1 ] [ 1 ] [ 1 ]
[ 2 ] [ 1 ] [ 1 ] [ 1 ] [ 1 ] [ 1 ] [ 1 ] [ 1 ] [ 1 ] [ 1 ]
[ 3 ] [ 1 ] [ 1 ] [ 1 ] [ 1 ] [ 1 ] [ 1 ] [ 1 ] [ 1 ] [ 1 ]
[ 4 ] [ 1 ] [ 1 ] [ 1 ] [ 1 ] [ 1 ] [ 1 ] [ 1 ] [ 1 ] [ 1 ]
[ 5 ] [ 1 ] [ 1 ] [ 1 ] [ 1 ] [ 1 ] [ 1 ] [ 1 ] [ 1 ] [ 1 ]
[ 6 ] [ 1 ] [ 1 ] [ 1 ] [ 1 ] [ 1 ] [ 1 ] [ 1 ] [ 1 ] [ 1 ]
[ 7 ] [ 1 ] [ 1 ] [ 1 ] [ 1 ] [ 1 ] [ 1 ] [ 1 ] [ 1 ] [ 1 ]
[ 8 ] [ 1 ] [ 1 ] [ 1 ] [ 1 ] [ 1 ] [ 1 ] [ 1 ] [ 1 ] [ 1 ]
[ 9 ] [ 1 ] [ 1 ] [ 1 ] [ 1 ] [ 1 ] [ 1 ] [ 1 ] [ 1 ] [ 1 ]
[ 10 ] [ 1 ] [ 1 ] [ 1 ] [ 1 ] [ 1 ] [ 1 ] [ 1 ] [ 1 ] [ 1 ]

-----
영화 메뉴 - 비포 선택 라미즈
1. 상영 영화 정보
2. 좌석 예약 현황
7. 메인 메뉴 이동

메뉴를 선택하세요 : 7
```

## 기본기능 : TicketBox.h

TicketBox.h

```
#ifndef TICKETBOX_H
#define TICKETBOX_H

class TicketBox {

public:
    virtual Screen * selectMenu() { return NULL; }    // 상영관 선택 메뉴 ( 1관, 2관, 3관)
    void Initialize() {}

};

class KpuTicketBox : public TicketBox {

    EuropeScreen * pEuropeScreen;
    AsiaScreen * pAsiaScreen;
    PremiumScreen * pPremiumScreen;

public:
    KpuTicketBox();
    ~KpuTicketBox();
    Screen * selectMenu();
    void Initialize();

};

#endif

// 스크린 객체 생성
void KpuTicketBox::Initialize() {
    // Screen(영화제목, 티켓가격, 좌석(nRowMax), 좌석(nColMax))
    pEuropeScreen = new EuropeScreen("비포 선 라이즈", 10000, 10, 10);
    pAsiaScreen = new AsiaScreen("패왕별희", 12000, 10, 10);
    pPremiumScreen = new PremiumScreen("프리즌 이스케이프", 30000, 6, 6);
}
```

## 기본기능 : Screen.h

Screen.h

```

#ifndef SCREEN_H
#define SCREEN_H
#include <string>
#include "Ticket.h"
using namespace std;

class Screen {
protected:
    int nTicketPrice;           // 티켓 가격
    int nRowMax, nColMax;       // 좌석 행과 열 최대 값
    int nCurrentReservedId;    // 발급된 마지막 예약 번호
    Ticket ** pSeatArray;       // 스크린 좌석에 대한 티켓 배열
    string strMovieName;       // 스크린 상영중인 영화 제목

public:
    Screen(string name, int price, int r, int c);
    ~Screen();
    void showSeatMap();         // 좌석 예약 여부 맵으로 보기
    virtual void showMovieMenu(); // 영화 예매 메뉴
    virtual void showMovieInfo() = 0; // 영화 소개 정보
};

class EuropeScreen : public Screen {
public:
    EuropeScreen(string name, int price, int r, int c) : Screen( name, price, r, c) {}
    void showMovieInfo();
};

class AsiaScreen : public Screen {
public:
    AsiaScreen(string name, int price, int r, int c) : Screen(name, price, r, c) {}
    void showMovieInfo();
};

class PremiumScreen : public Screen {
public:
    PremiumScreen(string name, int price, int r, int c) : Screen(name, price, r, c) {}
    void showMovieInfo();
};

#endif

```

## 기본기능 : Ticket.h

```

#ifndef TICKET_H
#define TICKET_H

#define SEAT_EMPTY_MARK      '.'
#define SEAT_RESERVED_MARK  'R'
#define SEAT_COMPLETION_MARK '$'

class Ticket {
    int nRow;
    int nCol;
    char charCheck;
    int nReservedID;
    int nPayAmount;
    int nPayMethod;

public:
    Ticket() {}
    void setCheck(char check) { charCheck = check; }
    char getCheck() { return charCheck; }
};

#endif

```

Ticket.h

```

// 좌석행
// 좌석번호
// 예약번호
// 예약금액
// 결제수단

```

Screen.cpp

```

Screen::Screen(string name, int price, int r, int c) {
    strMovieName = name;
    nRowMax = r;
    nColMax = c;
    nTicketPrice = price;
    pSeatArray = new Ticket*[nRowMax];
    for (int r = 0; r < nRowMax; r++) {
        pSeatArray[r] = new Ticket[nColMax];
    }
    for (int i = 0; i < nRowMax; i++) {
        for (int j = 0; j < nColMax; j++) {
            pSeatArray[i][j].setCheck(SEAT_EMPTY_MARK);
        }
    }
}

```

## \* 기본 기능 구현 시 헤더 파일 참조

```
#include <iostream>
#include <string>
#include "Screen.h"
using namespace std;
```

Screen.cpp

```
#include <iostream>
#include "Screen.h"
#include "TicketBox.h"
using namespace std;
```

TicketBox.cpp

```
#include <iostream>
#include <string>

#include "Screen.h"
#include "TicketBox.h"
using namespace std;
```

Main.cpp

## 추가 기능 1 : 좌석 예약과 결과 화면

## Screen.h

```
public:
    void reserveTicket();    // 좌석 예약하기
```

Ticket.h

```
public:
    // 좌석 번호 저장
    void setSeat(int r, int c) { nRow = r; nCol = c; }
    // 좌석 예약 번호 저장
    void setReservedID(int reserved) { nReservedID = reserved; }
```

The image displays a sequence of seven screenshots from a Korean cinema program, illustrating the user interface for selecting a movie and a seat.

- Screenshot 1:** Shows the main menu with options for movie selection (영화 메뉴 - 패랑별회) and seat selection (좌석 예약). The user has selected movie 1, 'The 13th Warrior' (13전사).
- Screenshot 2:** Shows the movie selection screen with options for movie 1, 'The 13th Warrior' (13전사), and movie 2, 'The 13th Warrior' (13전사). The user has selected movie 1.
- Screenshot 3:** Shows the seat selection screen with options for seat 1, 'The 13th Warrior' (13전사), and seat 2, 'The 13th Warrior' (13전사). The user has selected seat 1.
- Screenshot 4:** Shows the seat selection screen with options for seat 1, 'The 13th Warrior' (13전사), and seat 2, 'The 13th Warrior' (13전사). The user has selected seat 1.
- Screenshot 5:** Shows the seat selection screen with options for seat 1, 'The 13th Warrior' (13전사), and seat 2, 'The 13th Warrior' (13전사). The user has selected seat 1.
- Screenshot 6:** Shows the seat selection screen with options for seat 1, 'The 13th Warrior' (13전사), and seat 2, 'The 13th Warrior' (13전사). The user has selected seat 1.
- Screenshot 7:** Shows the seat selection screen with options for seat 1, 'The 13th Warrior' (13전사), and seat 2, 'The 13th Warrior' (13전사). The user has selected seat 1.

## 추가 기능 2 : 좌석 예약 취소와 결과 화면

```

C:\WINDOWS\system32\cmd.exe

-----
상영관 메인 메뉴
-----
1. 유럽 영화 1관
2. 아시아 영화 2관
3. 프리미엄 영화 3관

선택(1~3) 그외 종료 : 3

-----
영화 메뉴 - 프리즌 이스케이프
-----
1. 영화 정보
2. 예약 현황
3. 예약 하기
4. 예약 취소
7. 메인 메뉴 이동

메뉴를 선택하세요: 3

[ 좌석 예약 ]
좌석 행 번호 입력(1 - 6) : 2
좌석 열 번호 입력(1 - 6) : 4
행[2] 열[4] 100 예약 번호로 접수되었습니다.

-----
영화 메뉴 - 프리즌 이스케이프
-----
1. 영화 정보
2. 예약 현황
3. 예약 하기
4. 예약 취소
7. 메인 메뉴 이동

메뉴를 선택하세요: 2

[ 좌석 예약 현황 ]
[1] [2] [3] [4] [5] [6]
[2] [-] [-] [-] [-] [-] [-]
[3] [-] [-] [-] [-] [-] [-]
[4] [-] [-] [-] [-] [-] [-]
[5] [-] [-] [-] [-] [-] [-]
[6] [-] [-] [-] [-] [-] [-]
    
```

```

-----
영화 메뉴 - 프리즌 이스케이프
-----
1. 영화 정보
2. 예약 현황
3. 예약 하기
4. 예약 취소
7. 메인 메뉴 이동

메뉴를 선택하세요: 4

[ 좌석 예약 취소 ]
예약 번호 입력 : 100
100 예약 번호가 취소 처리되었습니다.

-----
영화 메뉴 - 프리즌 이스케이프
-----
1. 영화 정보
2. 예약 현황
3. 예약 하기
4. 예약 취소
7. 메인 메뉴 이동

메뉴를 선택하세요: 2

[ 좌석 예약 현황 ]
[1] [2] [3] [4] [5] [6]
[2] [-] [-] [-] [-] [-] [-]
[3] [-] [-] [-] [-] [-] [-]
[4] [-] [-] [-] [-] [-] [-]
[5] [-] [-] [-] [-] [-] [-]
[6] [-] [-] [-] [-] [-] [-]

-----
영화 메뉴 - 프리즌 이스케이프
-----
1. 영화 정보
2. 예약 현황
3. 예약 하기
4. 예약 취소
7. 메인 메뉴 이동

메뉴를 선택하세요:
    
```



## 추가 기능 2 : 좌석 예약 취소와 결과 화면

Screen.h

```
Ticket * checkReservedID(int id); // 예약 번호 확인  
public:  
    void cancelReservation();      // 좌석 예약 취소하기
```

Ticket.h

```
public:  
    // 좌석 예약 번호 읽기  
    int getReservedID() { return nReservedID; }
```

## 추가 기능 3 : 티켓 결제와 결과 화면

```

C:\WINDOWS\system32\cmd.exe

-----
상영관 메인 메뉴
-----
1. 유럽 영화 1관
2. 아시아 영화 2관
3. 프리미엄 영화 3관

선택 (1~3) 그외 종료 : 3

-----
영화 메뉴 - 프리즌 이스케이프
-----
1. 영화 정보
2. 예약 현황
3. 예약하기
4. 예약취소
5. 결제하기
7. 메인 메뉴 이동

메뉴를 선택하세요: 1

-----
[ 프리즌 이스케이프 ]
-----
영화관 : 프리미엄영화 3관
좌석공백 : 다니엘 래드클리프
출거리 : 인권을통가 팀과 스티븐은 억울한 판결로 투옥된다. ...

가격 : 30000

-----
영화 메뉴 - 프리즌 이스케이프
-----
1. 영화 정보
2. 예약 현황
3. 예약하기
4. 예약취소
5. 결제하기
7. 메인 메뉴 이동

메뉴를 선택하세요: 3

[ 좌석 예약 ]
좌석 행 번호 입력 (1 - 6) : 2
좌석 열 번호 입력 (1 - 6) : 5
행[2] 열[5] 100 예약 번호로 접수되었습니다.
    
```

```

-----
영화 메뉴 - 프리즌 이스케이프
-----
1. 영화 정보
2. 예약 현황
3. 예약하기
4. 예약취소
5. 결제하기
7. 메인 메뉴 이동

메뉴를 선택하세요: 5

[ 좌석 예약 결제 ]
예약 번호 입력 : 100

-----
결제 방식 선택
-----
1. 모바일 결제
2. 은행 이체 결제
3. 카드 결제

결제 방식 (1~3) : 1
[ 모바일 결제 ]
핸드폰 번호 입력 (11자리수) : 01033334444
이름 : 홍길동
모바일 결제가 완료되었습니다. : 36000

-----
영화 메뉴 - 프리즌 이스케이프
-----
1. 영화 정보
2. 예약 현황
3. 예약하기
4. 예약취소
5. 결제하기
7. 메인 메뉴 이동

메뉴를 선택하세요: 2

[ 좌석 예약 현황 ]
[1] [2] [3] [4] [5] [6]
[2] [-] [-] [-] [-] [-] [-]
[3] [-] [-] [-] [-] [-] [-]
[4] [-] [-] [-] [-] [-] [-]
[5] [-] [-] [-] [-] [-] [-]
[6] [-] [-] [-] [-] [-] [-]
    
```

## 추가 기능 3 : 티켓 결제와 결과 화면

영화 메뉴 - 프리즌 이스케이프

1. 영화 정보  
2. 예약 현황  
3. 예약 취소하기  
4. 예약 결제하기  
5. 결제하기  
6. 메인 메뉴 이동

메뉴를 선택하세요 : 3

[ 좌석 예약 ]  
좌석 행 번호 입력 (1 - 6) : 4  
좌석 열 번호 입력 (1 - 6) : 2  
행 [4] 열 [2] 101 예약 번호로 접수되었습니다.

영화 메뉴 - 프리즌 이스케이프

1. 영화 정보  
2. 예약 현황  
3. 예약 취소하기  
4. 예약 결제하기  
5. 결제하기  
6. 메인 메뉴 이동

메뉴를 선택하세요 : 5

[ 좌석 예약 결제 ]  
예약 번호 입력 : 101

결제 방식 선택

1. 모바일 결제  
2. 은행 이체  
3. 카드 결제

결제 방식 (1~3) : 2  
[ KPU 은행 결제 ]  
은행 번호 입력 (12자리수) : 678934562134  
이름 : 김철호  
KPU 은행 결제가 완료되었습니다. : 33000

영화 메뉴 - 프리즌 이스케이프

1. 영화 정보  
2. 예약 현황  
3. 예약 취소하기  
4. 예약 결제하기  
5. 결제하기  
6. 메인 메뉴 이동

메뉴를 선택하세요 : 2

[ 좌석 예약 현황 ]  

	1	2	3	4	5	6
[1]	-	-	-	-	-	-
[2]	-	-	-	-	\$	-
[3]	-	-	-	-	-	-
[4]	-	\$	-	-	-	-
[5]	-	-	-	-	-	-
[6]	-	-	-	-	-	-

영화 메뉴 - 프리즌 이스케이프

1. 영화 정보  
2. 예약 현황  
3. 예약 취소하기  
4. 예약 결제하기  
5. 결제하기  
6. 메인 메뉴 이동

메뉴를 선택하세요 : 3

[ 좌석 예약 ]  
좌석 행 번호 입력 (1 - 6) : 5  
좌석 열 번호 입력 (1 - 6) : 4  
행 [5] 열 [4] 102 예약 번호로 접수되었습니다.

영화 메뉴 - 프리즌 이스케이프

1. 영화 정보  
2. 예약 현황  
3. 예약 취소하기  
4. 예약 결제하기  
5. 결제하기  
6. 메인 메뉴 이동

메뉴를 선택하세요 : 5

[ 좌석 예약 결제 ]  
예약 번호 입력 : 102

결제 방식 선택

1. 모바일 결제  
2. 은행 이체  
3. 카드 결제

결제 방식 (1~3) : 3  
[ 신용카드 결제 ]  
카드 번호 입력 (12자리수) : 234567890123  
이름 : 김철호  
신용카드 결제가 완료되었습니다. : 34500

영화 메뉴 - 프리즌 이스케이프

1. 영화 정보  
2. 예약 현황  
3. 예약 취소하기  
4. 예약 결제하기  
5. 결제하기  
6. 메인 메뉴 이동

메뉴를 선택하세요 : 2

[ 좌석 예약 현황 ]  

	1	2	3	4	5	6
[1]	-	-	-	-	-	-
[2]	-	-	-	-	\$	-
[3]	-	-	-	-	-	-
[4]	-	\$	-	-	-	-
[5]	-	-	-	\$	-	-
[6]	-	-	-	-	-	-

## 추가 기능 3 : 티켓 결제

Screen.h

```
public:  
    int getTicketPrice() { return nTicketPrice; }  
    void Payment();    // 결제하기
```

Ticket.h

```
public:  
    void setPayAmount(int amount) { nPayAmount = amount; }  
    int getPayAmount() { return nPayAmount; }  
    void setPayMethod(int method) { nPayMethod = method; }
```

## 추가 기능 3 : 티켓 결제

Pay.h

```

#ifndef PAY_H
#define PAY_H

#define MOBILE_PHONE_PAYMENT          1
#define BANK_TRANSFER_PAYMENT        2
#define CREDIT_CARD_PAYMENT          3
#define MOBILE_PHONE_INTEREST_RATE   0.2 // 모바일 결제 수수료 비율(이자율)
#define BANK_TRANSFER_INTEREST_RATE   0.1 // 은행 이체 결제 수수료 비율(이자율)
#define CREDIT_CARD_INTEREST_RATE     0.15 // 신용카드 결제 수수료 비율(이자율)

class Pay {
public:
    virtual int charge(int amount) = 0;    // 결제하기
};

class CardPay : public Pay {
    double interest;
public:
    CardPay(double rate) { this->interest = rate; }
    int charge(int amount);                // 결제금액 = 티켓 가격 + 티켓 가격 * 카드수수료 비율
};

class BankTransfer : public Pay {
    double interest;
public:
    BankTransfer(double rate) { this->interest = rate; }
    int charge(int amount);                // 결제금액 = 티켓 가격 + 티켓 가격 * 은행수수료 비율
};

class MobilePay : public Pay {
    double interest;
public:
    MobilePay(double rate) { this->interest = rate; }
    int charge(int amount);                // 결제금액 = 티켓 가격 + 티켓 가격 * 모바일수수료 비율
};

#endif

```

## 추가 기능 4 : 통계 관리 결과 화면

- 1) 상영관 메인 메뉴에 ( 9번 통계 관리 ) 기능 추가
  - 스크린 관 별 티켓 결제 금액 총액과 전체 티켓 판매량
  - 관리자 인증 처리
  - 9번 통계 관리 수행 후 자동 종료

```

-----
영화 메뉴 - 프리즌 이스케이프
-----
1. 영화 정보
2. 예약 현황
3. 예약 하기
4. 예약 취소
5. 결제 하기
7. 메인 메뉴 이동

메뉴를 선택하세요 : 7

-----
상영관 메인 메뉴
-----
1. 유럽 영화 1관
2. 아시아 영화 2관
3. 프리미엄 영화 3관
9. 통계 관리

선택(1~3,9) 그외 종료 : 9

-----
관리자 메뉴
-----
관리자 비밀번호 입력 : admin

1. 유럽영화 상영관 결제금액 : 12000
2. 아시아영화 상영관 결제금액 : 13200
3. 프리미엄영화 상영관 결제금액 : 34500
4. 전체 티켓 판매량 : 3

계속하려면 아무 키나 누르십시오 . . .
  
```

## 추가 기능 4 : 통계 관리

```
int getRowMax() { return nRowMax; }  
int getColMax() { return nColMax; }  
Ticket** getTicketArray() { return pSeatArray; }
```

Screen.h

```
#ifndef STATISTICIS_H  
#define STATISTICIS_H  
  
#include "Screen.h"  
class Statistics {  
public:    // Static 함수 구현  
    static int totalSum(Screen* pScreen);  
    static int totalCount(Screen* pScreen);  
};  
#endif
```

Statistics.h

```
#define    TICKETBOX_MANAGER_PWD    "admin"  
  
public:  
    void Manage();
```

TicketBox.h