

스마트 웹 & 콘텐츠 개발(JAVA, 안드로이드앱) 과정

자리 예약 프로그램

포트폴리오 # 01

2019.05.20(월)

김문선

목차

1. 개요	2
2. 요구사항분석	2
① 차별성 3C	2
② 예약기능에 대한 요구사항	2
③ 조회기능에 대한 요구사항	2
④ 취소기능에 대한 요구사항	3
3. 소스코드 정의	3
① ReserveProgram.java(메인자바파일)	3
② Seat.java	3
③ SeatType.java	4
4. 프로그램 코드	5
① ReserveProgram.java	5
② Seat.java	6
③ SeatType.java	6
5. 실행결과화면	7
✓ 초기화면	7
✓ 예약기능	8
✓ 조회기능	8
✓ 취소기능	9
6. 매뉴얼	9
① 초기화면	9
② 예약기능	10
③ 조회기능	10
④ 취소기능	11
⑤ 종료	12
7. 프로그램의 차별성	12

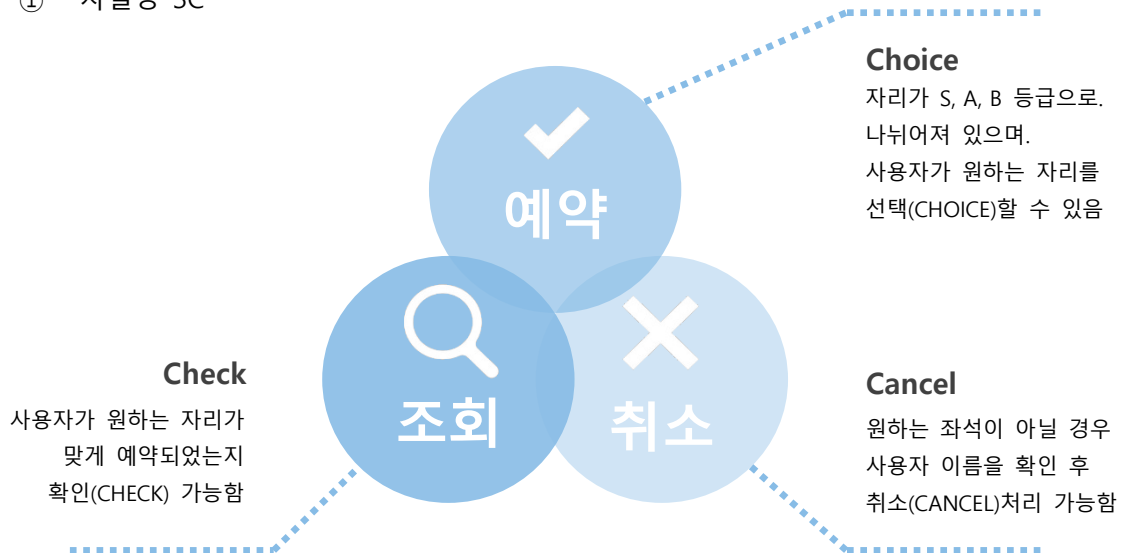
자리 예약 프로그램

1. 개요

자리를 예약하고 예약한 자리를 조회하며 취소할 수 있는 자리에예약프로그램

2. 요구사항분석

① 차별성 3C



② 예약기능에 대한 요구 사항

예약	입력	<ul style="list-style-type: none"> 좌석 등급 입력 사용자 이름과 좌석 번호 선택 후 자리 예약
	출력	<ul style="list-style-type: none"> 사용자가 좌석 등급 선택하면 좌석 현황을 보여줌.
	검색	<ul style="list-style-type: none"> 사용자가 선택한 좌석이 이미 예약된 좌석인지 선택한 자리를 검색함 - 예약된 좌석이면 다시 선택, 비어있는 자리면 예약!

③ 조회기능에 대한 요구 사항



④ 취소기능에 대한 요구사항

취소	입력	<ul style="list-style-type: none"> ☞ 메뉴에서 취소 선택 시 ☞ 예약자 이름과 좌석등급 확인 후, 자리 취소함 ☞ 예약자 이름 오 입력 시, 취소 불가
	출력	☞ 사용자가 좌석 등급 선택하면 좌석 현황을 보여줌
	검색	☞ 입력한 좌석등급과 이름이 맞을 경우 취소 처리됨

3. 소스코드 정의

① ReserveProgram.java(메인 자바 파일)

자료형	변수명	내용	초기값
Int	choice	예약프로그램 기능 선택 [입력] ☞ 1: 예약 / 2: 조회 / 3: 취소 / 4. 종료	0
int	type	예약프로그램 예매 자리등급 [입력] ☞ 1: S등급 / 2: A등급 / 3: B등급	0
SeatType[] aSeatType = new SeatType[3];			
	aSeatType[0]	S 타입 좌석 생성	10
	aSeatType[1]	A 타입 좌석 생성	10
	aSeatType[2]	B 타입 좌석 생성	10

② Seat.java

자료형	변수명	내용	초기값
String	name	좌석 예약자의 이름 [입력]	

자료형	클래스 명칭	클래스 기능
public	Seat	예약자 이름과 좌석 관리기능
자료형	메소드 명칭	내용
boolean	match()	예약자 이름과 예약 취소자 이름이 같은지 체크 - 이름이 같으면 true / 다르면 false를 반환해줌
boolean	isOccupied()	예약 시 좌석이 공석인지 체크 - 공석이면 true / 아니면 false를 반환해줌
void	setName()	좌석 예약자 이름을 저장
void	resetName()	취소 시 예약자 이름을 null 값으로 변경
String	getName()	name(예약자 이름)을 반환해줌

③ SeatType.java

자료형	변수명	내용	초기값
char	type	좌석 타입	
Seat[]	aSeat	좌석 배열	

SeatType			
자료형	명칭	기능, 내용	
public	SeatType	좌석 타입과 좌석 배열 생성 ✓ 매개변수 : char type, int num	
자료형	변수명	내용	
char	type	좌석 타입(S / A / B 등급)	
Seat[]	aSeat	배열 방을 생성	
int	num	좌석 개수	
int	i	For문 순환용 변수	

reserve()			
자료형	명칭	기능, 내용	
boolean	reserve	예약 기능 메소드. 정보를 입력 받아 예약 처리 ✓ 매개변수 : 없음 ✓ return 값 : true or false	
자료형	변수명	내용	
int	seatNum	몇 번째 자리에 앉을 것인지 사용자로부터 입력 받는 변수	
String	name	예약자 이름을 입력 받는 변수	

cancel()			
자료형	명칭	기능, 내용	
boolean	cancel	예약한 자리를 취소하는 메소드 ✓ 매개변수 : 없음 ✓ return 값 : true or false	
자료형	변수명	내용	
String	name	사용자로부터 취소할 예약자 이름을 입력 받는 변수	

Show()			
자료형	함수명	기능, 내용	
void	show	좌석이 예약되어 있는지 공석인지를 출력해주는 메소드 ✓ 매개변수 : 없음	

	✓ return 값 : 없음
--	-----------------

4. 프로그램 코드

① ReservrProgram.java

```
//package Reserve_Hardprogram;
import java.util.Scanner;

public class ReserveProgram {
    public static void main(String[] args) {
        System.out.println("*****");
        System.out.println("*****");
        System.out.println("***** [ 예약프로그램을 시작합니다 ] *****");
        System.out.println("*****");
        System.out.println();

        Scanner sin = new Scanner(System.in);
        SeatType[] aSeatType = new SeatType[3];
        aSeatType[0] = new SeatType('S', 10); // S 타입 좌석 생성
        aSeatType[1] = new SeatType('A', 10); // A 타입 좌석 생성
        aSeatType[2] = new SeatType('B', 10); // B 타입 좌석 생성
        int choice = 0;
        int type = 0;

        while (choice != 4) {
            System.out.print("예약(1), 조회(2), 취소(3), 끝내기(4)>>");
            choice = Integer.parseInt(sin.nextLine()); // 기능선택!!

            switch (choice) {
                case 1:// 예약
                    System.out.println("***** 예약메뉴를 선택하셨습니다.");
                    do {
                        System.out.print("좌석구분 S(1), A(2), B(3)>>");
                        type = Integer.parseInt(sin.nextLine());
                        if (type < 1 || type > 3) {
                            System.out.println("잘못된 좌석 타입입니다.");
                        }
                    } while (type > 3);
                    aSeatType[type - 1].reserve(); // 예약: reserve 메소드 호출
                    break;

                case 2:// 조회
                    System.out.println("***** 조회메뉴를 선택하셨습니다.");
                    for (int i = 0; i < aSeatType.length; i++)
                        aSeatType[i].show(); // 조회: show 메소드 호출
                    System.out.println("*****");
                    break;

                case 3:// 취소
                    System.out.println("***** 취소메뉴를 선택하셨습니다.");
                    do {
                        System.out.print("좌석구분 S(1), A(2), B(3)>>");
                        type = Integer.parseInt(sin.nextLine());
                        if (type < 1 || type > 3) {
                            System.out.println("잘못된 좌석 타입입니다.");
                        }
                    } while (type > 3);
                    aSeatType[type - 1].cancel(); // 취소: cancel 메소드 호출

                    break;

                case 4:// 끝내기
                    System.out.println("**** 예약프로그램을 종료합니다 ****");
                    break;

                default:
                    System.out.println("잘못 입력하셨습니다.");
            }
        }
    }
}
```

② Seat.java

```
//package Reserve_Hardprogram;
public class Seat {
    private String name; // 좌석 예약자 이름

    public Seat() {
        name = null;
    } // 생성자: name에 null을 저장

    public String getName() {
        return name;
    } // name을 반환

    public void resetName() {
        name = null;
    } // name에 null을 저장

    public void setName(String name) {
        this.name = name;
    } // 좌석 예약자 이름을 저장

    public boolean isOccupied() { // 좌석이 비어 있는지 체크
        if (name == null) // 좌석이 비어 있으면 false 반환
            return false;
        else // 좌석이 예약되어 있으면 true 반환
            return true;
    }

    public boolean match(String name) { // 예약자 이름과 예약 취소자 이름이 같은지 체크
        if (name.equals(this.name)) // 예약자 이름과 예약 취소자 이름이 같으면 true 반환
            return true;
        else
            return false;
    }
}
```

③ SeatType.java

```
package Reserve_Hardprogram;
import java.util.Scanner;

public class SeatType {
    private char type; // 좌석 타입
    private Seat[] aSeat; // 좌석 배열

    public SeatType(char type, int num) { // 생성자: 좌석 타입과 좌석 배열 생성
        this.type = type;
        aSeat = new Seat[num]; // 좌석 수 만큼 배열 방을 생성
        for (int i = 0; i < aSeat.length; i++)
            aSeat[i] = new Seat(); // 배열 aSeat[i]에 좌석 객체를 생성하여 저장
    }

    public boolean reserve() { // 예약 메소드: 예약자 정보를 입력받아 예약이 처리되면 true 반환
        Scanner sin = new Scanner(System.in);
        int seatNum;
        String name;

        show(); // 좌석의 현재상황을 보여주는 메소드
        System.out.print("이름>>");
        name = sin.nextLine();
        System.out.print("번호>>");
        seatNum = Integer.parseInt(sin.nextLine());
        //System.out.println();

        if (seatNum < 1 || seatNum > aSeat.length) { // 존재하지 않는 좌석이면 false 반환
            System.out.println("잘못된 좌석번호입니다.");
            return false;
        }

        if (aSeat[seatNum - 1].isOccupied()) { // 이미 예약된 좌석이면 false 반환
            System.out.println("이미 예약된 좌석입니다.");
            return false;
        }
        aSeat[seatNum - 1].setName(name); // setName 메소드 호출
        System.out.println(type+"등급 "+seatNum+"번 자리에 "+name+"님으로 예약되었습니다. ");
        return true; // "예약 잘됐다"
    }
}
```



```

public boolean cancel() { // 취소 메소드: 예약 취소가 정상 처리되면 true 반환
    Scanner sin = new Scanner(System.in);
    show();
    System.out.print("이름>>");
    String name = sin.nextLine(); // 취소할 예약자 이름 입력
    if (name != null) {
        for (int i = 0; i < aSeat.length; i++) {
            if (aSeat[i].match(name)) { // match 메소드 호출: 예약자=취소할 예약자이름?
                aSeat[i].resetName(); // resetName 메소드 호출: 예약자 이름을 다시 null로 저장
                return true;
            }
        }
        System.out.println("예약자 이름을 찾을 수 없습니다.");
    }
    return false; // 예약자 이름을 찾지 못함
}

public void show() { // 조회 메소드: 좌석이 예약되어 있는지 비어 있는지를 출력
    System.out.print(type + ">> ");
    for (int i = 0; i < aSeat.length; i++) {
        System.out.print(i + 1 + ".");
        if (aSeat[i].isOccupied()) // 예약된 좌석인가?
            System.out.print(aSeat[i].getName()); // 예약자 이름 출력
        else // 빈 좌석이면 "---" 출력
            System.out.print("---");
        System.out.print(" ");
    }
    System.out.println();
}
}

```

5. 실행결과 화면

✓ 초기화면

```

명령 프롬프트 - java ReserveProgram
*****
***** [ 예약프로그램을 시작합니다 ] *****
*****
예약(1), 조회(2), 취소(3), 끝내기(4)>>

```


✓ 예약 화면

```

명령 프롬프트 - java ReserveProgram
*****
***** [ 예약프로그램을 시작합니다 ] *****
*****

예약(1), 조회(2), 취소(3), 끝내기(4)>>5
잘못 입력하셨습니다.
예약(1), 조회(2), 취소(3), 끝내기(4)>>1
***** 예약메뉴를 선택하셨습니다.
좌석구분 S(1), A(2), B(3)>>1
S>> 1,--- 2,--- 3,--- 4,--- 5,--- 6,--- 7,--- 8,--- 9,--- 10,---

이름>>김문선
번호>>3
S등급 3번 자리에 김문선님으로 예약되었습니다.
예약(1), 조회(2), 취소(3), 끝내기(4)>>

```

✓ 조회 화면

```

선택 명령 프롬프트 - java ReserveProgram
이름>>박현아
번호>>8
A등급 8번 자리에 박현아님으로 예약되었습니다.
예약(1), 조회(2), 취소(3), 끝내기(4)>>2
***** 조회메뉴를 선택하셨습니다.
S>> 1.김문선 2.천송이 3.이예원 4.성혜경 5.김우승 6.오택환 7.신용권 8.오미경 9.이상아 10.배은정
A>> 1.고영호 2.박재우 3.서혜리 4.주제현 5.남궁업 6.임희주 7.이삭 8.박현아 9.최수정 10.박비파
B>> 1.주형지 2.배미영 3.전훈섭 4.오나연 5.곽보현 6.서예운 7,--- 8.안선미 9.이동형 10.문보경
*****
예약(1), 조회(2), 취소(3), 끝내기(4)>>

```

✓ 취소 화면

```

C:\ 명령 프롬프트 - java ReserveProgram
이름>>박현아
번호>>8
A등급 8번 자리에 박현아님으로 예약되었습니다.
예약(1), 조회(2), 취소(3), 끝내기(4)>>2
***** 조회메뉴를 선택하셨습니다.
S>> 1.김문선 2.천송이 3.이예원 4.성혜경 5.김우승 6.오택환 7.신용권 8.오미경 9.이상아 10.배은정
A>> 1.고영호 2.박재우 3.서혜리 4.주제현 5.남궁업 6.임희주 7.이삭 8.박현아 9.최수정 10.박비파
B>> 1.주형지 2.배미영 3.전훈섭 4.오나연 5.곽보현 6.서예윤 7.--- 8.안선미 9.이동형 10.문보경
*****
예약(1), 조회(2), 취소(3), 끝내기(4)>>3
***** 취소메뉴를 선택하셨습니다.
좌석구분 S(1), A(2), B(3)>>2
A>> 1.고영호 2.박재우 3.서혜리 4.주제현 5.남궁업 6.임희주 7.이삭 8.박현아 9.최수정 10.박비파
이름>>서혜리
예약(1), 조회(2), 취소(3), 끝내기(4)>>2
***** 조회메뉴를 선택하셨습니다.
S>> 1.김문선 2.천송이 3.이예원 4.성혜경 5.김우승 6.오택환 7.신용권 8.오미경 9.이상아 10.배은정
A>> 1.고영호 2.박재우 3.--- 4.주제현 5.남궁업 6.임희주 7.이삭 8.박현아 9.최수정 10.박비파
B>> 1.주형지 2.배미영 3.전훈섭 4.오나연 5.곽보현 6.서예윤 7.--- 8.안선미 9.이동형 10.문보경
*****
예약(1), 조회(2), 취소(3), 끝내기(4)>>

```

6. 메뉴얼

① 초기화면

아래의 그림 1.1은 자리에약 프로그램의 초기화면이다.

```

C:\ 명령 프롬프트 - java ReserveProgram
*****
***** [ 예약프로그램을 시작합니다 ] *****
*****
예약(1), 조회(2), 취소(3), 끝내기(4)>>

```

그림 1.1 초기화면

사용자로부터 기능을 입력 받게 되어 있으며 숫자 키 1~4번 중 원하는 기능을 선택할 수 있다. 그 외의 숫자를 누르면 잘못 선택했다는 오류메시지가 출력되면서 다시 기능선택화면으로 돌아간다. 1번을 누르면 예약, 2번을 누르면 조회, 3번을 누르면 취소, 4번을 누르면 프로그램이 종료된다는 메시지를 출력함과 동시에 프로그램이 종료된다.

② 예약 기능

아래의 그림 1. 2는 [메뉴선택] -> [1. 예약기능] 을 실행한 모습이다.

```

*****
***** [ 예약프로그램을 시작합니다 ] *****
*****

예약(1), 조회(2), 취소(3), 끝내기(4)>>5
잘못 입력하셨습니다.
예약(1), 조회(2), 취소(3), 끝내기(4)>>1
***** 예약메뉴를 선택하셨습니다.
좌석구분 S(1), A(2), B(3)>>1
S>> 1.--- 2.--- 3.--- 4.--- 5.--- 6.--- 7.--- 8.--- 9.--- 10.---

이름>>김문선
번호>>3
S등급 3번 자리에 김문선님으로 예약되었습니다.
예약(1), 조회(2), 취소(3), 끝내기(4)>>
  
```

그림 1. 2 예약기능 실행

초기화면에서 1번을 선택하여 예약기능이 실행된 화면이다. 해당 프로그램은 좌석 등급이 3단계로 구분되어지며 S, A, B등급으로 나뉘어지지만, 개발자 여담으로는 다 동일한 가격의 좌석이라고 한다..(ㅎㅎ) 그리고 각 등급마다 10자리씩 존재한다. 사용자가 선택하고 싶은 좌석등급을 선택하면 그 등급의 좌석현황을 볼 수 있고 상황을 보면서 자리를 예약할 수 있다.

단, S, A, B등급 이외의 좌석을 선택하면 다시 등급을 선택할 수 있는 화면으로 돌아간다. 사용자가 좌석구분을 옳게 입력했으면 예약자 이름과 좌석번호를 입력할 수 있다. 해당 자리가 비었는지 isOccupied() 메소드에서 판별한 후, 공석이면 예약이 되었다는 안내메시지를 출력하고 다시 초기화면으로, 공석이 아니면 이미 예약된 자리라는 메시지가 출력되면서 다시 선택화면으로 돌아간다.

③ 조회 기능

아래의 그림 1. 3은 조회기능을 실행한 모습이다.

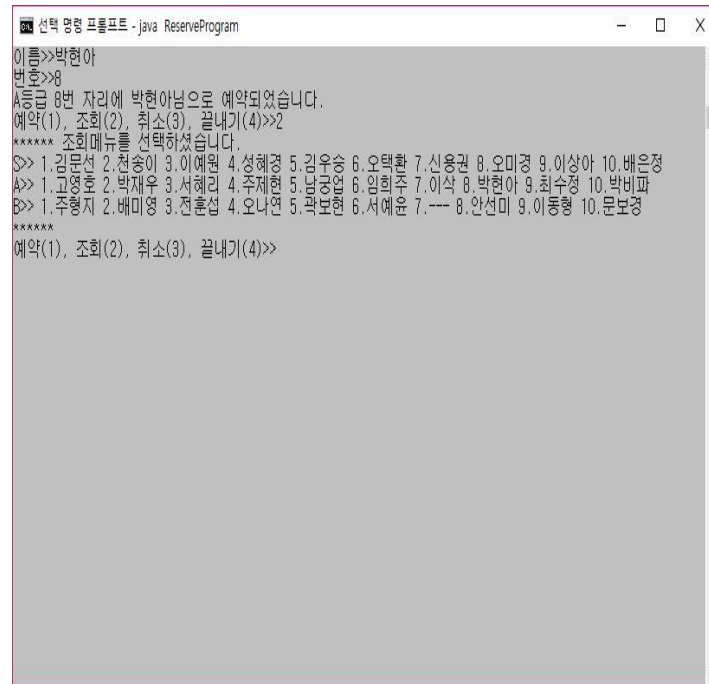


그림 1.3 조회기능 실행

해당 화면은 초기화면에서 2번 조회기능을 선택하면 실행되는 화면이다. 현재까지 예약된 좌석현황을 보여준다. 어느 자리가 공석인지 자리가 윽게 예약되었는지 확인할 수 있다.

④ 취소 기능

아래의 그림 1.4는 취소기능을 실행한 모습이다.

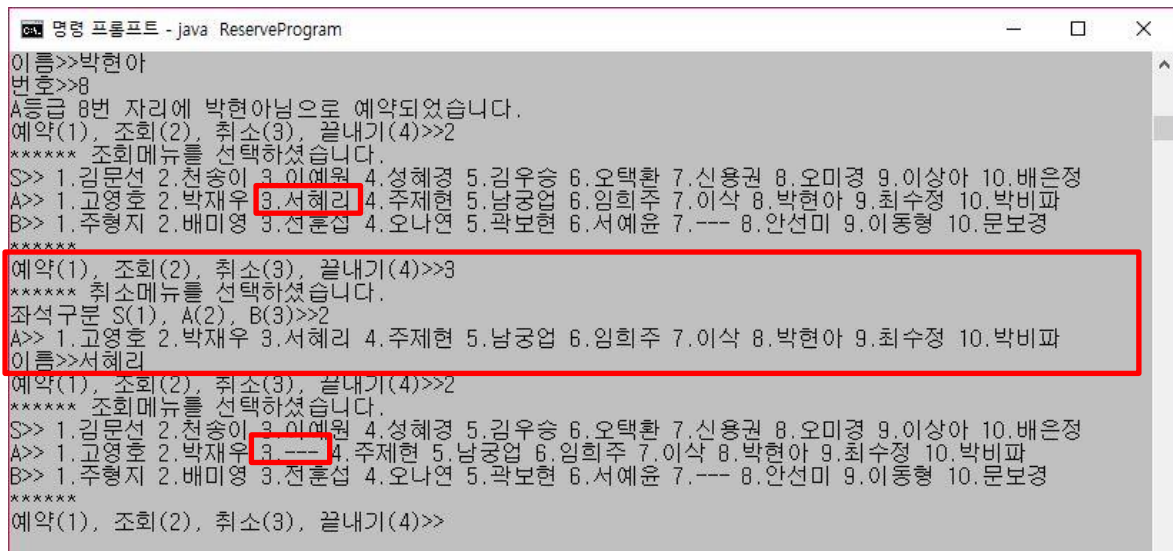


그림 1.4 취소 기능 실행

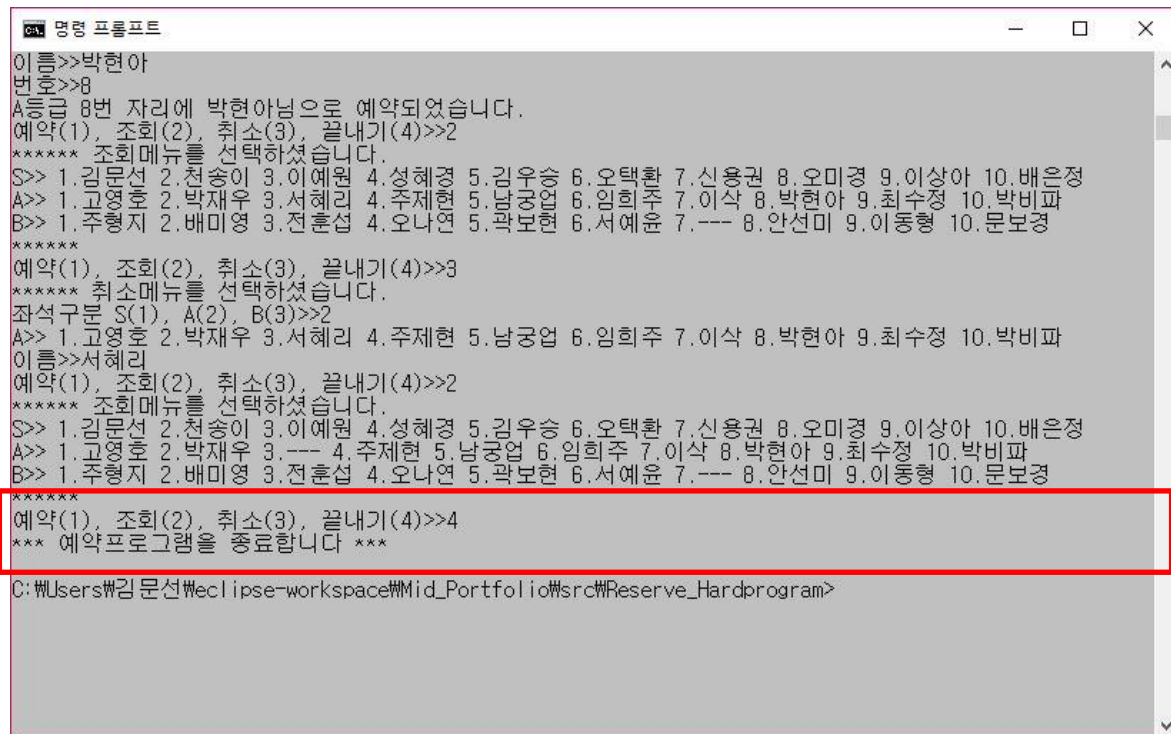
해당 화면은 초기화면에서 3번 취소기능을 실행한 화면이다. 취소 기능을 선택하면 어떤 등급의 자리를 취소할 지 선택하는 안내문이 출력되며 등급을 선택하면 취소할 예약자의 이름을 입력 받는다. match() 메소드를 통해 자리 예매한 사람과 취소할 사람이 같은지 확인 후, 해당

자리를 null 값으로 변경한다.

⑤ 종료

아래의 그림 1. 5는 종료기능을 실행한 모습이다.

다른 기능들과 마찬가지로 종료기능에 해당하는 ' 4 ' 를 입력하여 실행한다. 종료기능을 선택하면 예약프로그램을 종료한다는 안내문을 출력하고 종료한다.



```
명령 프롬프트
이름>>박현아
번호>>8
A등급 8번 자리에 박현아님으로 예약되었습니다.
예약(1), 조회(2), 취소(3), 끝내기(4)>>2
***** 조회메뉴를 선택하셨습니다.
S>> 1.김문선 2.천송이 3.이예원 4.성혜경 5.김우승 6.오택환 7.신용권 8.오미경 9.이상아 10.배은정
A>> 1.고영호 2.박재우 3.서혜리 4.주제현 5.남궁업 6.임희주 7.이삭 8.박현아 9.최수정 10.박비파
B>> 1.주형지 2.배미영 3.전훈섭 4.오나연 5.곽보현 6.서예윤 7.--- 8.안선미 9.이동형 10.문보경
*****
예약(1), 조회(2), 취소(3), 끝내기(4)>>3
***** 취소메뉴를 선택하셨습니다.
좌석구분 S(1), A(2), B(3)>>2
A>> 1.고영호 2.박재우 3.서혜리 4.주제현 5.남궁업 6.임희주 7.이삭 8.박현아 9.최수정 10.박비파
이름>>서혜리
예약(1), 조회(2), 취소(3), 끝내기(4)>>2
***** 조회메뉴를 선택하셨습니다.
S>> 1.김문선 2.천송이 3.이예원 4.성혜경 5.김우승 6.오택환 7.신용권 8.오미경 9.이상아 10.배은정
A>> 1.고영호 2.박재우 3.--- 4.주제현 5.남궁업 6.임희주 7.이삭 8.박현아 9.최수정 10.박비파
B>> 1.주형지 2.배미영 3.전훈섭 4.오나연 5.곽보현 6.서예윤 7.--- 8.안선미 9.이동형 10.문보경
*****
예약(1), 조회(2), 취소(3), 끝내기(4)>>4
*** 예약프로그램을 종료합니다 ***
C:\Users\김문선\workspace\Mid_Portfolio\src\Reserve_Hardprogram>
```

그림 1. 5 종료기능 실행

7. 프로그램의 차별성

- ✓ 사용자가 예약한 정보를 확인할 수 있으며 잘못 예매되었을 때, 취소할 수 있도록 사용자의 불편을 최소화하였다.
- ✓ 사용자가 예약하기 편하도록 해당 등급의 자리 현황을 보여줄 수 있도록 했다.
- ✓ 조회 시 모든 좌석 현황을 출력해 어느 자리가 공석인지를 확인할 수 있도록 시각적으로 표현했다.