

기계식 주차시스템

2301110326 나예은

시스템 조건

1. 차량번호를 입력하여 입고나 출고할 수 있다.
2. 중복된 차량번호 입력 시 입고할 수 없다.
3. 출고 시 없는 차량번호 입력 시 출고할 수 없다.
4. 최대 8대까지 주차가 가능하다.
5. 객체지향 언어의 특성을 활용한다. (ParkingSystem, User, Parking)

● ParkingSystem Class

1. state 값에 따라 기능 부여
 - 0 : 메뉴
 - 1 : 차 입고
 - 2 : 차 출고
 - 3 : 주차장 현황
2. 무한으로 돌림으로써 시스템이 끊기지 않음

```
public class ParkingSystem {
    public static void main(String[] args) {
        int state = 0;

        while(true) {
            switch (state) {
                case 0:
                    state = Parking.Menu();
                    break;
                case 1:
                    state = Parking.CarIn();
                    break;
                case 2:
                    state = Parking.CarOut();
                    break;
                case 3:
                    state = Parking.List();
                    break;
            }
        }
    }
}
```

● User Class

1. 유저의 차 번호와 주차상태를 저장하는 객체 생성자
 - carNum (앞 숫자가 0일경우를 대비해 int가 아닌 String 사용)
 - check (입고된 경우 true, 출고된 경우 false)

```
public class User {
    String carNum;
    boolean check;
```

```

    public User (String _carNum) {
        carNum = _carNum;
        check = true;
    }
}

```

● Parking Class

1. 아래 사진과 같은 메뉴를 출력

```

[ 메뉴 ]
1. 입고
2. 출고
3. 주차 목록
>>

```

2. 3보다 크거나 1보다 작은 수를 입력 시 메뉴를 다시 출력하도록 구성

```

public static int Menu() {
    System.out.println(" [ 메뉴 ]");
    System.out.println(" 1. 입고");
    System.out.println(" 2. 출고");
    System.out.println(" 3. 주차 목록\n");

    System.out.print(">> ");
    int state = scan.nextInt();
    if (state > 3 || state < 1) {
        System.out.println("<error> 다시 입력해주세요. \n");
        return 0;
    }
    return state;
}

```

3. state 가 1일 때, 입고 기능 실행
4. 숫자 입력 시 주차장 내를 탐색함
 - 객체가 비어 있을 경우 → 입고 성공
 - 같은 번호의 차가 주차 되어있는 경우 → 입고 실패 및 이유 띄움
 - 중간에 출고되어 비어 있는 공간이 있는 경우 → key 값에 자리를 저장함
 - 꽉 차 있거나 끝까지 돌았을 경우
 - 1) key 값이 기본값 10이 아닐 경우 → key 값 자리에 입고 성공
 - 2) 1) 이 아닐 경우 입고 실패 및 이유 띄움
5. 4번의 각 기능이 실행되고 성공 혹은 실패 시 메뉴로 돌아가는 안내문 띄우고 0 반환

*key값을 넣은 이유: 출고하여 빈 자리에 내가 넣을 값과 그 후에 있는 값이 같은 경우를 대비하여 key값을 통해 다 확인 후 입고하는 방식을 택함

```

public static int CarIn() {
    System.out.print("<system> 입고할 차량번호 4자리를 입력해주세요. : ");
    String number = scan.next();
    int key = 10;
    for (int i = 0; i < customer.length; i++) {
        if (customer[i] == null) {

```

```

        customer[i] = new User(number);

        System.out.printf("<success> 입고에 성공하였습니다. (차량번호: %s)\n",
number);
        System.out.println("<system> 메뉴로 돌아갑니다.\n");
        return 0;
    }
    if (number.equals(customer[i].carNum) && customer[i].check == true) {
        System.out.println("<error> 입고에 실패하였습니다. (사유: 같은 번호의 차량이 주
차 되어있음)");
        System.out.println("<system> 메뉴로 돌아갑니다.\n");
        return 0;
    }
    else if (customer[i].check == false && key == 10) {
        key = i;
    }
    if (customer[i] != null && i >= 7) {
        if (key != 10) {
            customer[i] = new User(number);

            System.out.printf("<success> 입고에 성공하였습니다. (차량번호: %s)\n",
number);

            System.out.println("<system> 메뉴로 돌아갑니다.\n");
            return 0;
        }
        System.out.println("<error> 입고에 실패하였습니다. (사유: 주차 자리가 부족함)");
        System.out.println("<system> 메뉴로 돌아갑니다.\n");
        return 0;
    }
}
return 0;
}

```

6. state 가 2일 때, 출고 기능 실행

7. 숫자 입력 시 주차장 내를 탐색함

- 객체가 비어 있을 경우 → 출고 실패 및 이유 띄움
- 같은 번호의 차가 주차 되어있는 경우 → 출고 성공
- 끝까지 돌아갔는데 없는 경우 → 출고 실패 및 이유 띄움

8. 7번의 각 기능이 실행되고 성공 혹은 실패 시 메뉴로 돌아가는 안내문 띄우고 0 반환

```

public static int CarOut() {
    System.out.print("<system> 출고할 차량번호 4자리를 입력해주세요. : ");
    String number = scan.next();

    for (int i = 0; i < customer.length; i++) {
        if (customer[i] == null) {
            System.out.println("<error> 출고에 실패하였습니다. (사유: 해당 번호의 차량은 입
고되지 않았음)");
            System.out.println("<system> 메뉴로 돌아갑니다.\n");
            return 0;
        }
        if (number.equals(customer[i].carNum) && customer[i].check == true) {
            customer[i].check = false;
            System.out.printf("<succenss> 출고에 성공하였습니다. (차량번호: %s)\n",

```

```

number);
        System.out.println("<system> 메뉴로 돌아갑니다. \n");
        return 0;
    }
    else if (number.equals(customer[i].carNum) == false && i >= 7) {
        System.out.println("<error> 출고에 실패하였습니다. (사유: 해당 번호의 차량은 입
고되지 않았음)");
        System.out.println("<system> 메뉴로 돌아갑니다. \n");
        return 0;
    }
}

return 0;
}

```

9. state 가 3일 때, 현황출력 기능 실행

10. 해당 객체가 비어 있거나 출고된 경우 비어 있음을 출력

11. 비어 있지 않을 경우 자리번호와 차 번호를 출력

12. 8번 자리까지 모두 출력 후 메뉴로 돌아가는 안내문 띄우고 0을 반환

```

public static int List() {
    System.out.println(" [ 목록 ]");
    for (int i = 0; i < customer.length; i++) {
        if (customer[i] == null || (customer[i].check == false)) {
            System.out.printf(" %d번자리 : 비어 있음\n", i+1);
        }
        else if (customer[i] != null) {
            System.out.printf(" %d번자리 : %s\n", i+1, customer[i].carNum);
        }
    }
    System.out.println();
    return 0;
}

```

13. Parking Class 전체 코드

```

import java.util.Scanner;

public class Parking {
    static Scanner scan = new Scanner(System.in);
    static User[] customer = new User[8];

    public static int Menu() { // 메뉴 출력
        System.out.println(" [ 메뉴 ]");
        System.out.println(" 1. 입고");
        System.out.println(" 2. 출고");
        System.out.println(" 3. 주차 목록\n");

        System.out.print(">> ");
        int state = scan.nextInt();
        if (state > 3 || state < 1) {
            System.out.println("<error> 다시 입력해주세요. \n");
            return 0;
        }
        return state;
    }
}

```

```

    }

    public static int CarIn() {
        System.out.print("<system> 입고할 차량번호 4자리를 입력해주세요. : ");
        String number = scan.next();
        int key = 10;

        for (int i = 0; i < customer.length; i++) {
            if (customer[i] == null) {
                customer[i] = new User(number);

                System.out.printf("<success> 입고에 성공하였습니다. (차량번호: %s)\n",
number);

                System.out.println("<system> 메뉴로 돌아갑니다.\n");
                return 0;
            }
            if (number.equals(customer[i].carNum) && customer[i].check == true) { // 같은
번호의 차가 주차되어 있는 경우
                System.out.println("<error> 입고에 실패하였습니다. (사유: 같은 번호의 차량
이 주차 되어있음)");
                System.out.println("<system> 메뉴로 돌아갑니다.\n");
                return 0;
            }
            else if (customer[i].check == false && key == 10) {
                key = i;
            }
            if (customer[i] != null && i >= 7) {
                if (key != 10) {
                    customer[i] = new User(number);

                    System.out.printf("<success>   입고에   성공하였습니다.   (차량번
호: %s)\n", number);
                    System.out.println("<system> 메뉴로 돌아갑니다.\n");
                    return 0;
                }
                System.out.println("<error> 입고에 실패하였습니다. (사유: 주차 자리가 부족
함)");
                System.out.println("<system> 메뉴로 돌아갑니다.\n");
                return 0;
            }
        }

        return 0;
    }

    public static int CarOut() {
        System.out.print("<system> 출고할 차량번호 4자리를 입력해주세요. : ");
        String number = scan.next();

        for (int i = 0; i < customer.length; i++) {
            if (customer[i] == null) {
                System.out.println("<error> 출고에 실패하였습니다. (사유: 해당 번호의 차량
은 입고되지 않았음)");

```

```

        System.out.println("<system> 메뉴로 돌아갑니다.Wn");
        return 0;
    }
    if (number.equals(customer[i].carNum) && customer[i].check == true) {
        customer[i].check = false;
        System.out.printf("<succenss> 출고에 성공하였습니다. (차량번호: %s)Wn",
number);
        System.out.println("<system> 메뉴로 돌아갑니다.Wn");
        return 0;
    }
    else if (number.equals(customer[i].carNum) == false && i >= 7) {
        System.out.println("<error> 출고에 실패하였습니다. (사유: 해당 번호의 차량
은 입고되지 않았음)");
        System.out.println("<system> 메뉴로 돌아갑니다. Wn");
        return 0;
    }
}
return 0;
}

public static int List() {
    System.out.println(" [ 목록 ]");
    for (int i = 0; i < customer.length; i++) {
        if (customer[i] == null || (customer[i].check == false)) {
            System.out.printf("  %d번자리 : 비어있음Wn", i+1);
        }
        else if (customer[i] != null) {
            System.out.printf("  %d번자리 : %sWn", i+1, customer[i].carNum);
        }
    }
    System.out.println();
    return 0;
}
}

```