

연습문제 1

자바기반응용프로그래밍

문제 1

- 제시된 Day class 타입의 객체 배열 days를 갖는 MonthSchedule class에서 각 메소드의 정의부를 작성한다.
- 생성자에서는 각 필드를 초기화한다. scanner와 days 초기화에 주의한다.
- input() 메소드는 1번 입력에 대한 작동을 담당한다. 사용자로부터 날짜와 할 일을 입력 받고 올바른 날짜인지를 체크한다. 정상적인 날짜가 입력되었다면 days 배열의 올바른 위치에 할 일을 저장한다.
- view() 메소드는 2번 보기에 대한 작동을 담당한다. 사용자로부터 날짜를 입력 받고, 올바른 날짜라면 할 일을 출력한다.
- finish() 메소드는 종료 메시지를 출력하고, scanner를 닫는다.
- run() 메소드는 프로그램의 전체 구조를 담당한다. 사용자가 3을 입력하기 전까지 프로그램은 반복적으로 올바르게 기능을 수행할 수 있어야 한다.
- 출력 예시는 어디까지나 예시이다. 추가적인 기능 테스트는 스스로 해야 한다.

문제 1

```
1 import java.util.Scanner;
2
3 class Day {
4     private String work; // 하루의 할 일을 나타내는 문자열
5     public void set(String work) { this.work = work; }
6     public String get() { return work; }
7     public void show() {
8         if(work == null) System.out.println("없습니다.");
9         else System.out.println(work + "입니다.");
10    }
11 }
12
13 public class MonthSchedule {
14     private int nDays;
15     private Day[] days; // Day 객체 배열
16     private Scanner scanner;
17
18     public MonthSchedule(int nDays) {}
19
20     private void input() {}
21     private void view() {}
22     private void finish() {}
23     public void run() {}
24
25     public static void main(String[] args) {
26         MonthSchedule april = new MonthSchedule(30); // 4월달의 할 일
27         april.run();
28     }
29 }
```

이번달 스케줄 관리 프로그램.

할일(입력:1, 보기:2, 끝내기:3) >>0
잘못입력하였습니다.

할일(입력:1, 보기:2, 끝내기:3) >>1
날짜(1~30)?-2
할일(빈칸없이입력)?공부
날짜 잘못 입력하였습니다.!

할일(입력:1, 보기:2, 끝내기:3) >>1
날짜(1~30)?23
할일(빈칸없이입력)?자바시험

할일(입력:1, 보기:2, 끝내기:3) >>1
날짜(1~30)?5
할일(빈칸없이입력)?과제출제

할일(입력:1, 보기:2, 끝내기:3) >>2
날짜(1~30)?5
5일의 할 일은 과제출제입니다.

할일(입력:1, 보기:2, 끝내기:3) >>2
날짜(1~30)?23
23일의 할 일은 자바시험입니다.

할일(입력:1, 보기:2, 끝내기:3) >>3
프로그램을 종료합니다.

문제 1

```
18⊖ public MonthSchedule(int nDays) {  
19     this.nDays = nDays;  
20     this.days = new Day[nDays];  
21     for(int i=0; i<days.length; i++) {  
22         days[i] = new Day();  
23     }  
24     scanner = new Scanner(System.in);  
25 }
```

```
27⊖ private void input() {  
28     System.out.print("날짜(1~30)?");  
29     int day = scanner.nextInt();  
30     System.out.print("할일(빈칸없이입력)?");  
31     String work = scanner.next();  
32     day--;  
33  
34     if(day < 0 || day > nDays) { // 0~29 만 유효  
35         System.out.println("날짜 잘못 입력하였습니다.!");  
36         return;  
37     }  
38     days[day].set(work);  
39 }
```

```
51⊖ private void finish() {  
52     scanner.close();  
53     System.out.println("프로그램을 종료합니다.");  
54 }
```

```
40⊖ private void view() {  
41     System.out.print("날짜(1~30)?");  
42     int day = scanner.nextInt();  
43     day--;  
44     if(day < 0 || day > nDays) { // 0~29 만 유효  
45         System.out.println("날짜 잘못 입력하였습니다.!");  
46         return;  
47     }  
48     System.out.print((day+1) + "일의 할 일은 ");  
49     days[day].show();  
50 }
```

```
55⊖ public void run() {  
56     System.out.println("이번달 스케줄 관리 프로그램.");  
57  
58     while(true) {  
59         System.out.print("할일(입력:1, 보기:2, 끝내기:3) >>");  
60         int menu = scanner.nextInt();  
61         switch(menu) {  
62             case 1: input(); break;  
63             case 2: view(); break;  
64             case 3: finish(); return;  
65             default :  
66                 System.out.println("잘못입력하였습니다.");  
67             }  
68         System.out.println();  
69     }  
70 }
```

문제 2

- ArrayUtil class의 두 메소드를 작성한다.
- 27-28 라인의 static 메소드를 호출하는 방식을 눈여겨본다.

```
1 class ArrayUtil {  
2⊕    public static int [] concat(int[] a, int[] b) { // 배열 a와 b를 연결한 새로운 배열 리턴.  
14⊕    public static void print(int[] a) { // 배열 a 출력.  
21 }  
22  
23 public class StaticEx {  
24⊖    public static void main(String [] args){  
25        int [] array1 = { 1, 5, 7, 9 };  
26        int [] array2 = { 3, 6, -1, 100, 77 };  
27        int [] array3 = ArrayUtil.concat(array1, array2);  
28        ArrayUtil.print(array3);  
29    }  
30 }
```

```
[ 1 5 7 9 3 6 -1 100 77 ]
```

문제 2

```
public static int [] concat(int[] a, int[] b) { // 배열 a와 b를 연결한 새로운 배열 리턴
    int tmp [] = new int [a.length + b.length]; // 배열 a와 b를 합한 크기의 배열 생성
    for(int i=0; i<a.length; i++) // 배열 a를 tmp로 복사
        tmp[i] = a[i];

    for(int i=0; i<b.length; i++) { // 배열 b를 tmp로 복사
        int index = a.length + i;
        tmp[index] = b[i];
    }

    return tmp;
}
```

```
public static void print(int[] a) { // 배열 a 출력
    System.out.print("[ ");
    for(int i=0; i<a.length; i++) {
        System.out.print(a[i] + " ");
    }
    System.out.println("]");
}
```