

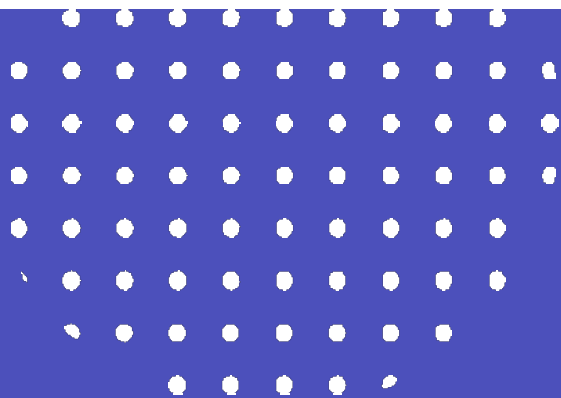
능동적 사고 방식의

java

강사 박주병

Park Ju Byeong

Park Ju Byeong



01

배열의 선언과 생성



— 02

배열의 인덱스

1-1 실습문제 (normal)

int 타입 길이5의 배열을 만들어 임의의 값을 넣고 출력해보자
- 출력은 반복문을 이용해 보자

```
0  
10  
20  
30  
40
```

1-1 문제풀이 (normal)

배열의 시작주소를 저장할 변수

```
int[] arr = new int[5];
```

배열을 메모리에 할당하여 생성한다.

```
arr[0] = 0;  
arr[1] = 10;  
arr[2] = 20;  
arr[3] = 30;  
arr[4] = 40;
```

길이를 리터럴로 적지 않고
length 변수를 이용했다.

```
for(int i=0;i<arr.length;i++)  
    System.out.println(arr[i]);
```

```
for(int i : arr)  
    System.out.println(i);
```

향상된for(foreach) 를 사용해도
동일한 결과가 나온다.

1-2 실습문제 (normal)

1번 문제에서 만들어놓은 배열 요소들의 평균을 구해보자

```
int[] arr = {10,20,30,40,50};
```

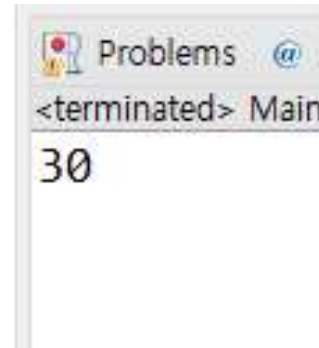
```
arr[0] = 10;
```

```
arr[1] = 20;
```

```
arr[2] = 30;
```

```
arr[3] = 40;
```

```
arr[4] = 50;
```



1-2 문제풀이 (normal)

```
int[] arr = {10,20,30,40,50};
```

```
arr[0] = 10;  
arr[1] = 20;  
arr[2] = 30;  
arr[3] = 40;  
arr[4] = 50;
```

```
int sum=0;
```

```
for(int i = 0 ;i<arr.length; i++)
```

```
    sum +=arr[i];
```

→ 배열 요소들의 합계를 구한다.

```
System.out.println(sum/arr.length);
```

→ 합계를 요소갯수만큼 나눠서
평균을 구한다.

1-3 실습문제 (normal)

int 타입 길이 10의 배열을 만들어 임의의값을 넣고 최대값과 최소값을 출력하자

```
int[] arr = {5,478,41,2,45,758,14,3,97,10};
```

Problems @ Javadoc
<terminated> Main (6) [Java
최대값은 758
최소값은 2

1-3 문제풀이 (normal)

```
int[] arr = {5,478,41,2,45,758,14,3,97,10};  
  
int max,min;  
  
max = arr[0];  
min = arr[0];  
  
for(int i = 1 ;i<arr.length; i++)  
{  
    if(max<arr[i]) max =arr[i];  
    if(min>arr[i]) min =arr[i];  
}  
  
System.out.println("최대값은 " +max);  
System.out.println("최소값은 " +min);
```

→ 첫번째 요소의 값을 각각 최대값과 최소값이라고 가정하고 넣어둔다.

→ 저장해놓은 값보다 크거나 작은게 나오면 교체하면서 모든 요소들을 확인해본다.

1-4 실습문제 (hard)

int타입 길이10의 배열에 0~9 까지의 랜덤한값을 넣은후
오름차순으로 정렬하여보자

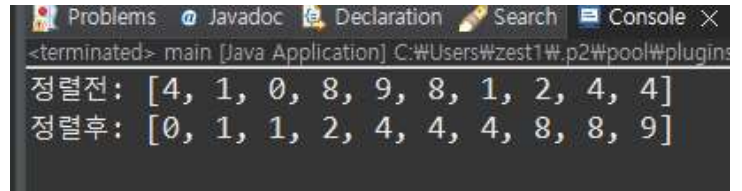
```
int[] arr = new int[10];

for(int i = 0 ;i<arr.length; i++)
    arr[i] = (int)(Math.random()*10);

System.out.println("정렬전: "+ Arrays.toString(arr));

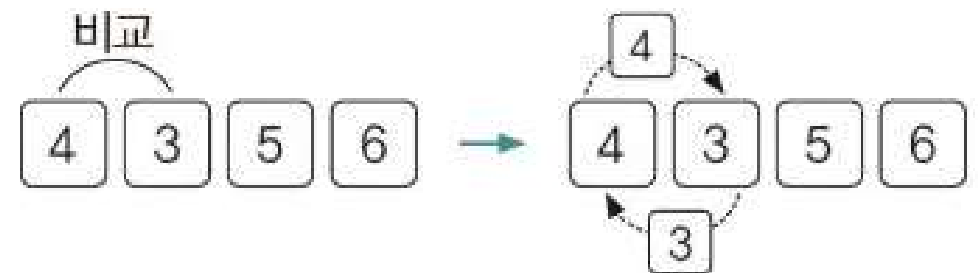
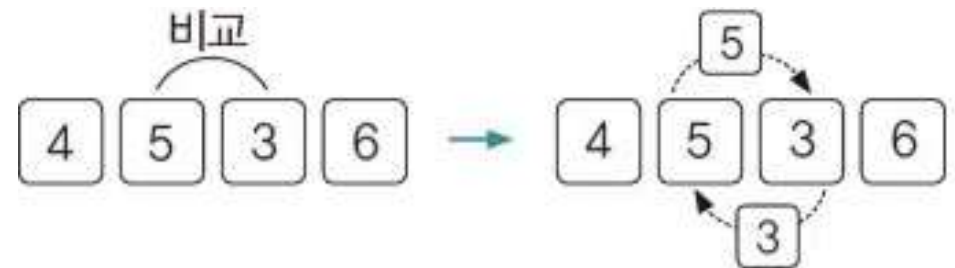
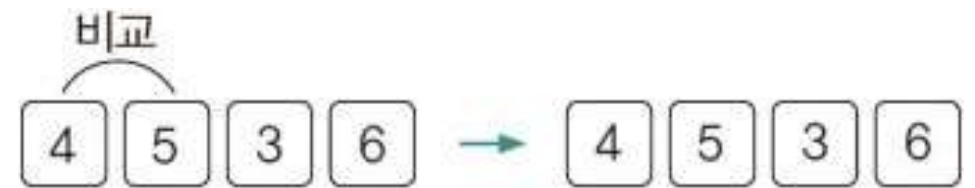
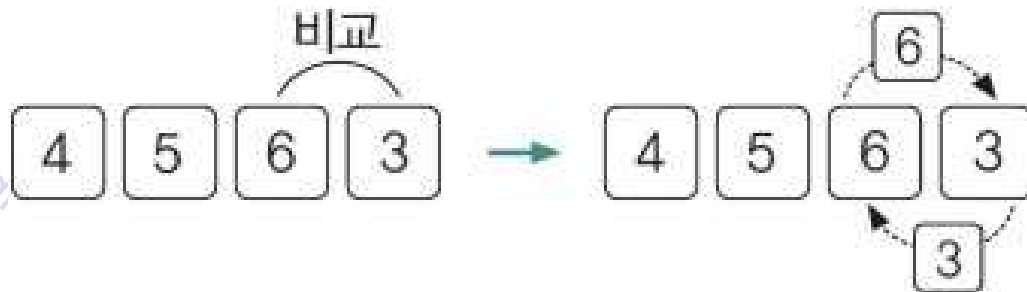
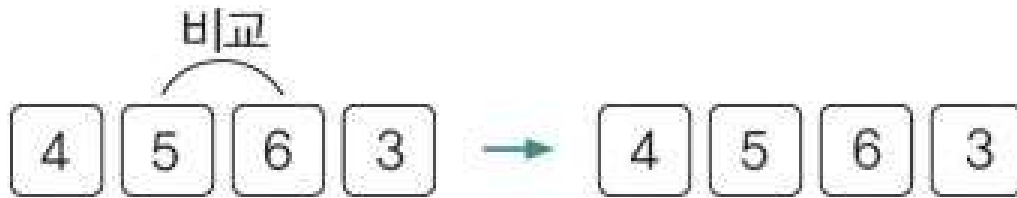
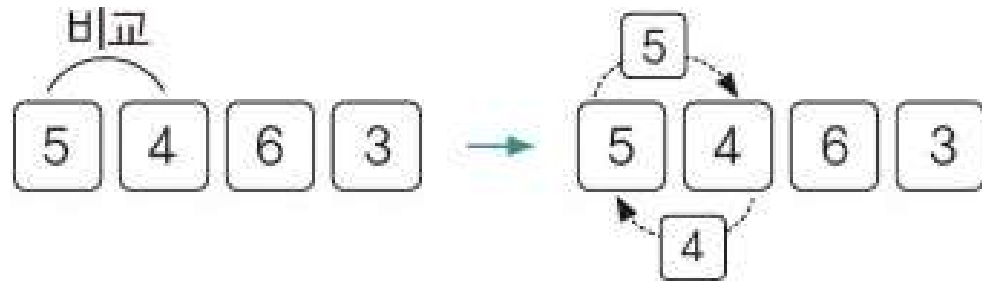
for(int i = 0 ;i<arr.length-1; i++)
    for(int j =0;j<arr.length-1-i;j++)
        if(arr[j] >arr[j+1])
        {
            int temp = arr[j+1];
            arr[j+1] = arr[j];
            arr[j] = temp;
        }

System.out.println("정렬후: "+ Arrays.toString(arr));
```

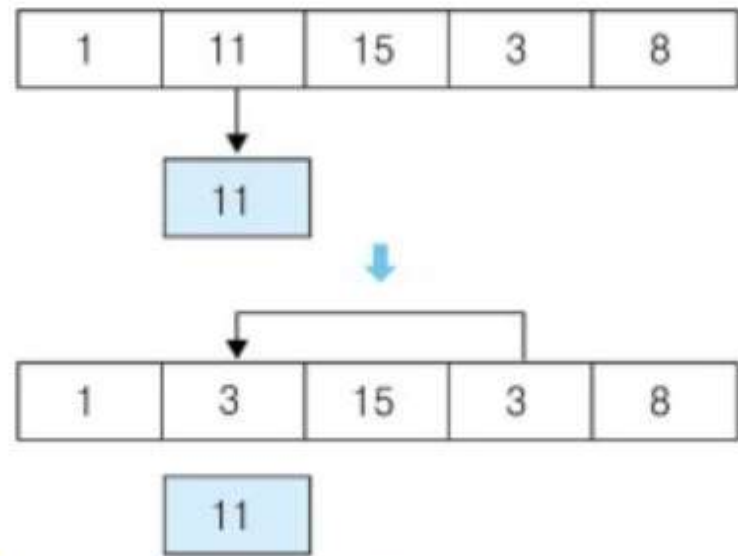
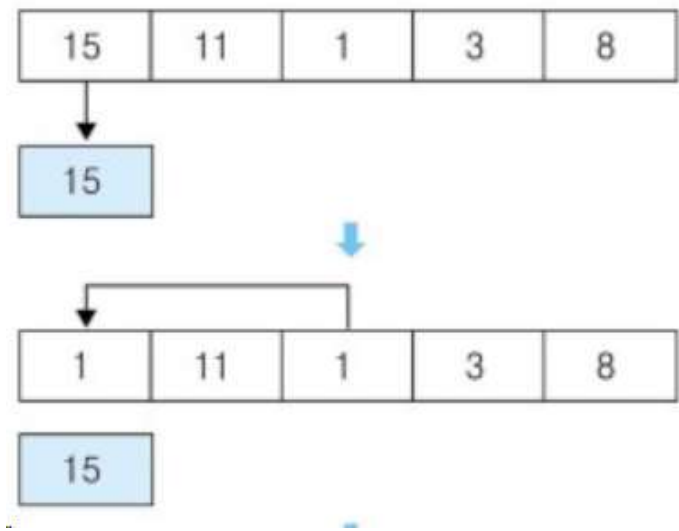


```
Problems Javadoc Declaration Search Console X
<terminated> main [Java Application] C:\Users\Wzest1\p2\pool\plugins
정렬전: [4, 1, 0, 8, 9, 8, 1, 2, 4, 4]
정렬후: [0, 1, 1, 2, 4, 4, 4, 8, 8, 9]
```

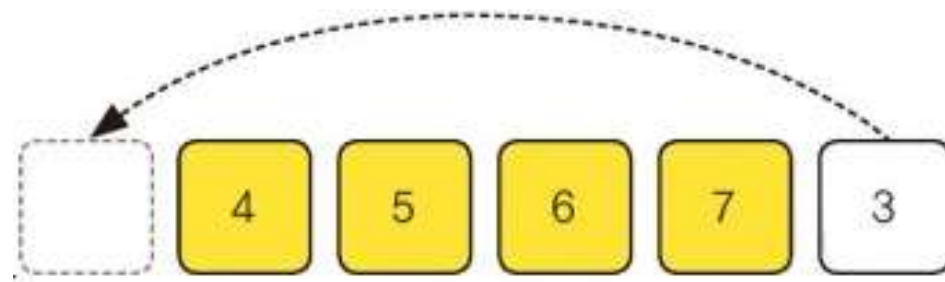
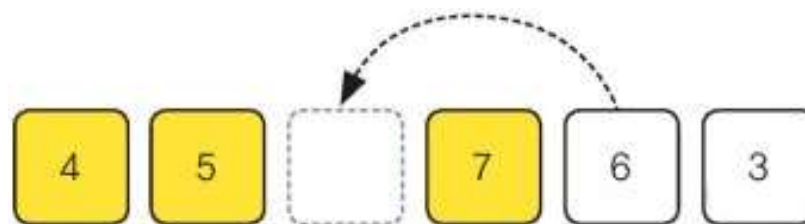
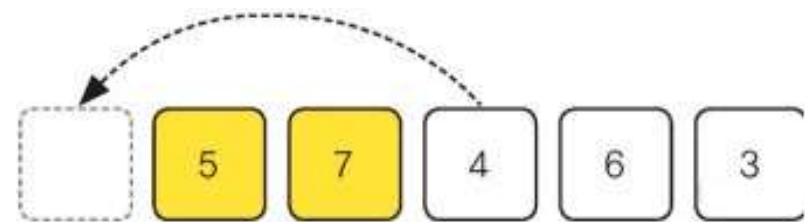
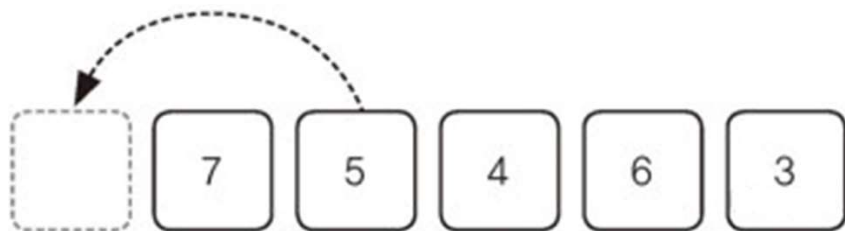
버블 정렬

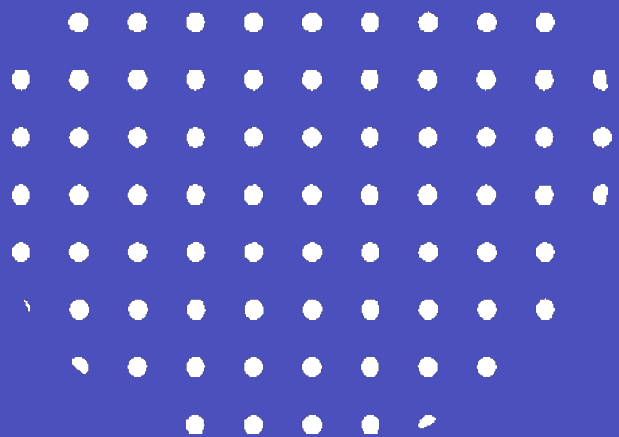


선택 정렬



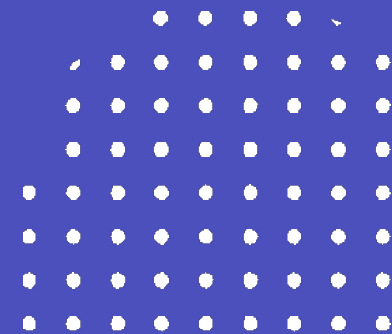
삽입 정렬





03

다차원배열



2-1 실습문제 (normal)

학생 5명의 국어, 영어, 수학 성적을 저장하는 **2차원 배열**을 만들어 저장하고 각 학생의 평균을 출력하여 보자(길이3의 1차원배열 5개를 저장 해야 한다)

- 데이터는 임의로 넣어보자
- 평균도 배열에 포함시켜도 된다.
- 간격을 일정하게 만들고 싶으면 `\t` 를 문자열에 포함하면 된다.

```
int[][] score = new int [][] {  
    {30,50,30}  
    , {70,20,90}  
    , {100,80,70}  
    , {90,40,30}  
    , {10,40,100}  
};  
System.out.println("국어 \t영어 \t수학 \t평균");
```

<terminated> Main (11) [Java Application] C:\Users\USER545\p2\			
국어	영어	수학	평균
30	50	30	36
70	20	90	60
100	80	70	83
90	40	30	53
10	40	100	50

2-1 문제풀이 (normal)

```
int[][] score = new int [][] {  
    {30,50,30}  
    , {70,20,90}  
    , {100,80,70}  
    , {90,40,30}  
    , {10,40,100}  
}  
;  
System.out.println("국어 \t영어 \t수학 \t평균");  
for(int[] student : score)  
{  
    int sum = 0;  
  
    for(int i : student)  
    {  
        sum += i;  
        System.out.print(i+" \t");  
    }  
  
    System.out.print(sum/student.length);  
  
    System.out.println();  
}
```


2-2 실습문제 (hard)

차원배열을 이용하여 숫자를 입력받아 암호화 하는 프로그램을 만들자.

```
char[][] encryptGrid = {{'0',' '},
                        {'1','!'},
                        {'2','@'},
                        {'3','#'},
                        {'4','$'},
                        {'5','%'},
                        {'6','^'},
                        {'7','&'},
                        {'8','*'},
                        {'9','('}};
```

```
Scanner scan = new Scanner(System.in);
System.out.println("암호화할 숫자를 입력하세요 : ");
String input = scan.nextLine();
//사용자로 부터 받은 숫자
char[] plainText = input.toCharArray();
//암호화된 숫자를 저장하기 위한 배열
```

- nextLine() 은 키보드로부터 문자열을 입력받는다.
- input.toCharArray()는 문자열을 char 배열로 변환해준다.

```
<terminated> main [Java Application] C:\Use
암호화할 숫자를 입력하세요 :
45674
평문 : 45674
암호화 : $%^&$
```

2-2 문제풀이 (hard)

```
char[][] encryptGrid = {{'0',' '},
                        {'1','!'},
                        {'2','@'},
                        {'3','#'},
                        {'4','$'},
                        {'5','%'},
                        {'6','^'},
                        {'7','&'},
                        {'8','*'},
                        {'9','('},
                        {}
};

Scanner scan = new Scanner(System.in);
System.out.println("암호화할 숫자를 입력하세요 : ");
String input = scan.nextLine();
//사용자로 부터 받은 숫자
char[] plainText = input.toCharArray();
//암호화된 숫자를 저장하기 위한 배열
char[] encryptText = new char[plainText.length];

System.out.print("평문 : ");
System.out.println(plainText);
for(int i =0;i<plainText.length;i++)
{
    for(char[] temp :encryptGrid)
    {
        //같은 문자라면 암호화된걸로 교체 한다.
        if(temp[0] == plainText[i])
            encryptText[i] = temp[1];
    }
}

System.out.print("암호화 : ");
System.out.println(encryptText);
```

2-3 실습문제 (expert)

1~25까지 숫자를 순서대로 저장하는 2차원 배열[5][5]을 만든후
값을 랜덤하게 섞어보자

- 섞는방법: Math.random() 이용하여 1~25 사이 숫자를 뽑은후
2차원배열에서 동일한 숫자를 찾아 [0][0] 의 값과 교체하는것을 100번
반복하자)

```
int[][] arr = new int[5][5];

System.out.println("---서플전---");
for(int i=0;i<arr.length;i++)
{
    for(int j =0;j<arr[0].length;j++)
    {
        arr[i][j] = i*arr[0].length+(j+1);
        System.out.print(arr[i][j] + "\t");
    }
    System.out.println();
}

System.out.println("-----");
```

```
//숫자 서플
for(int shuffleCnt=0;shuffleCnt<100;shuffleCnt++)
{
    int randomNumber = (int)(Math.random()*25)+1;

    for(int i=0;i<arr.length;i++)
    {
        for(int j =0;j<arr[0].length;j++)
        {
            //랜덤수와 동일한 숫자를 찾으면
            if(arr[i][j] == randomNumber)
            {
                //0,0 의 숫자와 바꿔서 섞는다.
                int temp = arr[i][j];
                arr[i][j] = arr[0][0];
                arr[0][0] = temp;
                break;
            }
        }
    }
}
```

```
---서플전---
1      2      3      4      5
6      7      8      9      10
11     12     13     14     15
16     17     18     19     20
21     22     23     24     25
-----
---서플후---
10     21     3      13     2
11     7      1      19     14
18     4      20     12     5
8      23     16     9      6
25     22     17     24     15
-----
```

2-3 문제풀이 (expert)

```
int[][] arr = new int[5][5];

System.out.println("---서플전---");
for(int i=0;i<arr.length;i++)
{
    for(int j =0;j<arr[0].length;j++)
    {
        arr[i][j] = i*arr[0].length+(j+1);
        System.out.print(arr[i][j] + "\t");
    }
    System.out.println();
}

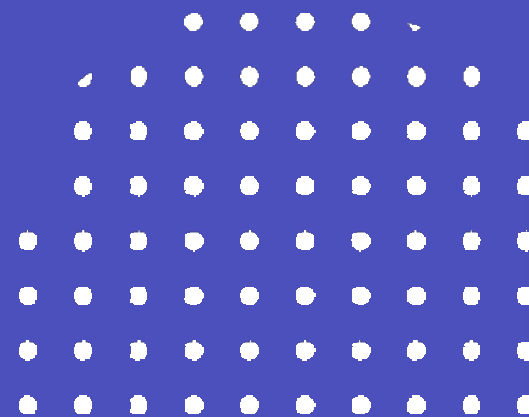
System.out.println("-----");
```

```
//숫자 셔플
for(int shuffleCnt=0;shuffleCnt<100;shuffleCnt++)
{
    int randomNumber = (int)(Math.random()*25)+1;

    for(int i=0;i<arr.length;i++)
    {
        for(int j =0;j<arr[0].length;j++)
        {
            //랜덤수와 동일한 숫자를 찾으면
            if(arr[i][j] == randomNumber)
            {
                //0,0 의 숫자와 바꿔서 섞는다.
                int temp = arr[i][j];
                arr[i][j] = arr[0][0];
                arr[0][0] = temp;
                break;
            }
        }
    }
}
```

— 04

실습문제



3-1 실습문제 (hard)

좌표를 입력 받아 해당좌표에 x를 표시하자

```
char[][] arr = new char[][]{
    { ' ', ' ', ' ', ' ', ' ', ' ', ' ', ' ', ' ' },
    { ' ', ' ', ' ', ' ', ' ', ' ', ' ', ' ', ' ' },
    { ' ', ' ', ' ', ' ', ' ', ' ', ' ', ' ', ' ' },
    { ' ', ' ', ' ', ' ', ' ', ' ', ' ', ' ', ' ' },
    { ' ', ' ', ' ', ' ', ' ', ' ', ' ', ' ', ' ' },
    { ' ', ' ', ' ', ' ', ' ', ' ', ' ', ' ', ' ' },
    { ' ', ' ', ' ', ' ', ' ', ' ', ' ', ' ', ' ' },
    { ' ', ' ', ' ', ' ', ' ', ' ', ' ', ' ', ' ' },
    { ' ', ' ', ' ', ' ', ' ', ' ', ' ', ' ', ' ' },
    { ' ', ' ', ' ', ' ', ' ', ' ', ' ', ' ', ' ' },
};

Scanner scan = new Scanner(System.in);

System.out.println("좌표를 입력하세요:");
int pos = scan.nextInt();
```

좌표를 입력하세요:

24
12345
1
2 X
3
4
5

3-1 문제풀이 (hard)

```
char[][] arr = new char[][]{
    { '1', '2', '3', '4', '5', '6', '7', '8', '9', '0' },
    { '1', '2', '3', '4', '5', '6', '7', '8', '9', '0' },
    { '1', '2', '3', '4', '5', '6', '7', '8', '9', '0' },
    { '1', '2', '3', '4', '5', '6', '7', '8', '9', '0' },
    { '1', '2', '3', '4', '5', '6', '7', '8', '9', '0' },
    { '1', '2', '3', '4', '5', '6', '7', '8', '9', '0' },
    { '1', '2', '3', '4', '5', '6', '7', '8', '9', '0' },
    { '1', '2', '3', '4', '5', '6', '7', '8', '9', '0' },
    { '1', '2', '3', '4', '5', '6', '7', '8', '9', '0' },
    { '1', '2', '3', '4', '5', '6', '7', '8', '9', '0' },
};

Scanner scan = new Scanner(System.in);

System.out.println("좌표를 입력하세요:");
int pos = scan.nextInt();
//입력한 좌표에 x 저장
arr[(pos/10)-1][(pos%10)-1] = 'X';

System.out.println(" 12345");

for(int y=0; y<arr.length; y++)
{
    System.out.print(y+1);
    for(int x=0; x<arr.length; x++)
    {
        System.out.print(arr[y][x]);
    }
    System.out.println();
}
```

3-2 실습문제 (expert)

아래의 그림은 2차원 배열을 이용하여 **┐** 모양의 테트리스 블록을 표현한 것이다.
블록을 시계방향으로 90도 돌려서 **└** 모양을 만들어 보자.

```
char[][] star = {
    {'*', '*', ' ', ' ', ' ', ' '},
    {'*', '*', ' ', ' ', ' ', ' '},
    {'*', '*', '*', '*', '*', '*'},
    {'*', '*', ' ', ' ', ' ', ' '},
    {'*', '*', ' ', ' ', ' ', ' '},
};

char[][] result = new char[star[0].length][star.length];
for(int i=0; i < star.length; i++)
{
    for(int j=0; j < star[i].length; j++)
        System.out.print(star[i][j]);

    System.out.println();
}

System.out.println();
for(int i=0; i < star.length; i++)
{
    for(int j=0; j < star[i].length; j++)
    {
        int x = j;
        int y = star.length-1-i;
        result[x][y]=star[i][j];
    }
}
for(int i=0; i < result.length; i++)
{
    for(int j=0; j < result[i].length; j++)
    {
        System.out.print(result[i][j]);
    }
}
```

- 2차원 배열[5][5]를 만든후 적절한 인덱스에 *을 넣어서 **┐** 모양을 만든다.
- *의 위치를 다른 인덱스로 옮겨서 90도 회전을 시켜보자.

```
**
**
*****
**
**

*****
*****

*
*
*
```


3-2 문제풀이 (expert)

00	01	02	03	04
10	11	12	13	14
20	21	22	23	24
30	31	32	33	34
40	41	42	43	44

40	30	20	10	00
41	31	21	11	01
42	32	22	12	02
43	33	23	13	03
44	34	24	14	04

00	04
01	14
02	24
03	34
04	44
10	03
11	13
12	23
13	33
14	43
20	02
21	12
22	22
23	32
24	42

Park Ju Byeong

Park Ju Byeong

3-3 실습문제 (expert)

1~25 사이의 셔플된 2차원 배열[5][5]을 만든후 사용자로부터 좌표를 입력받아 해당좌표의 숫자를 공개하고 공개된숫자가 15일 경우 게임을 종료한다.

- 해당좌표의 공개여부를 저장하기위해 셔플된 2차원배열[5][5]을 2개 가지는 3차원 배열을 만들어야 한다

```
Scanner scan = new Scanner(System.in);  
|  
//[1][0][0] -> 공개여부 저장 [0][0][0] -> 데이터  
// arr[1][0][0] = 1 -> 0,0 좌표는 공개  
int[0][0][0] arr = new int[2][5][5];  
  
//1~25 순차 입력  
for(int i=0;i<arr[0].length;i++)  
    for(int j =0;j<arr[0][0].length;j++)  
        arr[0][i][j] = i*arr[0][0].length+(j+1);
```

좌표를 입력하여 숫자 15를 찾으시오.:

00

```
-----  
10      *      *      *      *  
*      *      *      *      *  
*      *      *      *      *  
*      *      *      *      *  
*      *      *      *      *
```

좌표를 입력하여 숫자 15를 찾으시오.:

12

```
-----  
10      *      *      *      *  
*      *      *      *      *  
*      1      *      *      *  
*      *      *      *      *  
*      *      *      *      *
```

좌표를 입력하여 숫자 15를 찾으시오.:

3-3 문제풀이 (expert)

[0][][] 데이터

4	15	6	18	16
7	11	20	5	12
8	19	2	21	17
25	3	23	9	24
14	13	10	22	1

[1][][] 공개여부

0	0	0	0	0
1	0	0	0	0
0	0	0	1	0
1	0	1	0	0
0	0	0	0	0



THANK YOU



강사 박주병