

JAVA 프로그램 실습

다차원 배열

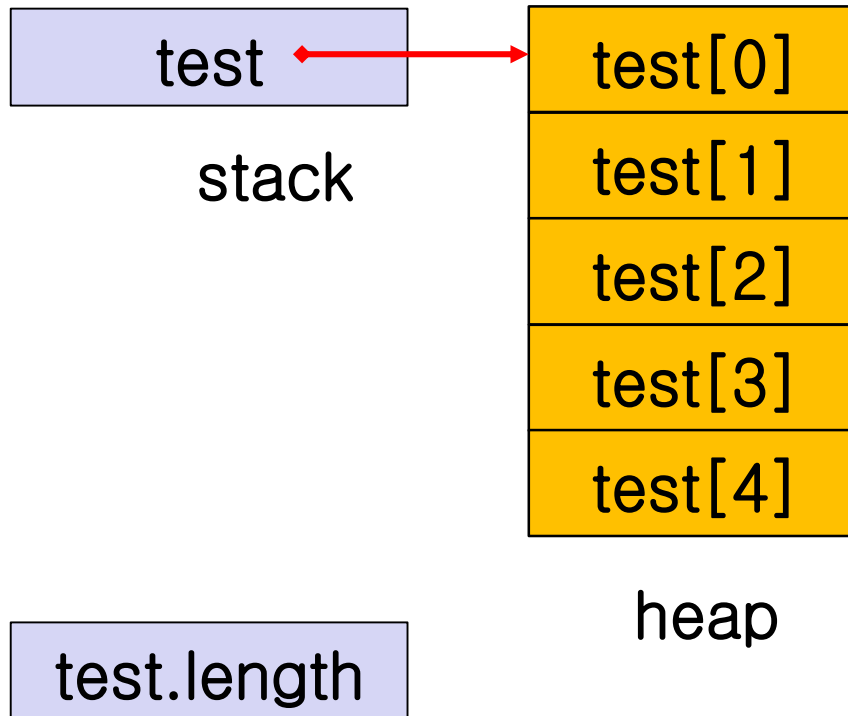
경북대학교
소프트웨어융합과
배 희 호 교수

1차원 배열

■ 1차원 배열

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

행

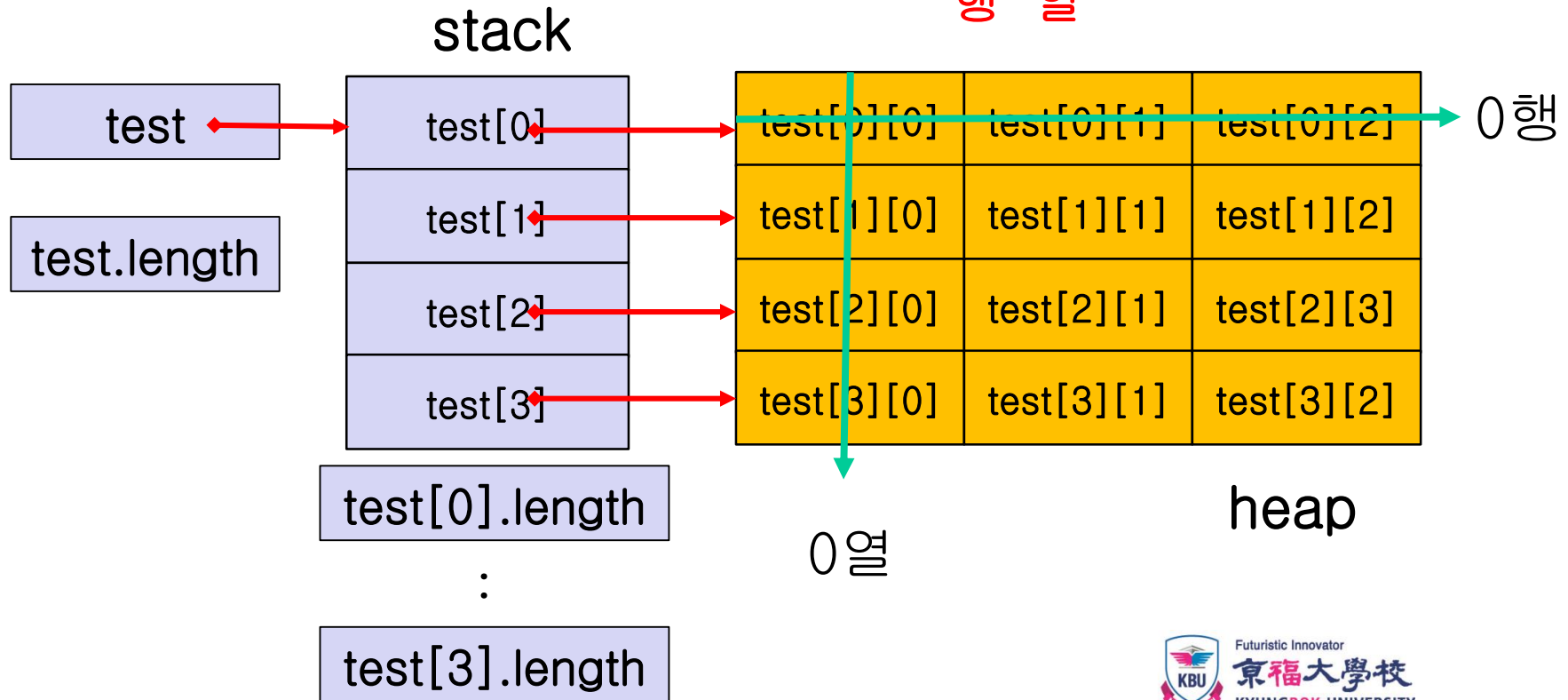


2차원 배열

- 2차원 배열을 간단한 Table로 생각하는 것이 직관적이긴 하지만, 실제 구조는 좀 더 복잡함
- 2차원 배열은 실제로 “배열들의 배열”

```
int[][] test = new int[4][3];
```

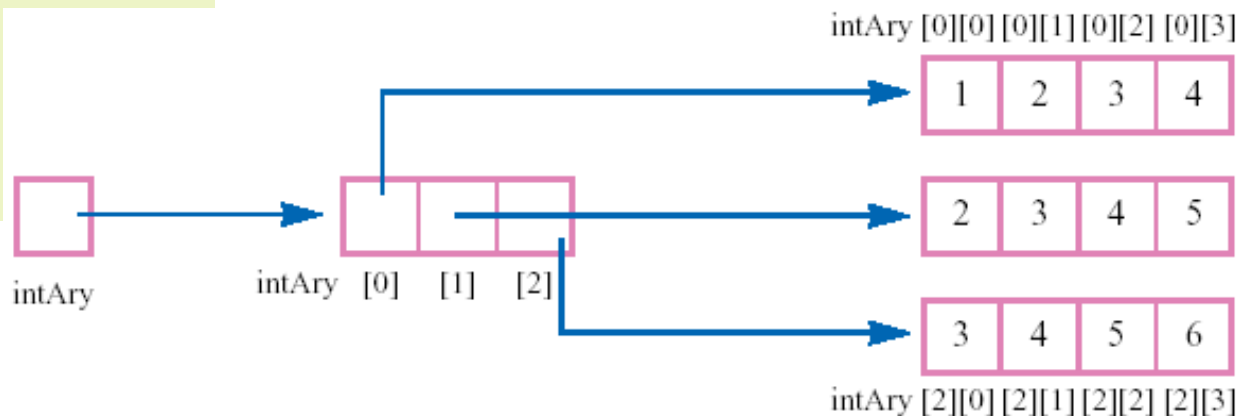
행 열



다차원 배열

- 배열의 배열(array of array)
 - 배열의 원소가 다시 배열이 되는 배열
 - 2차원 배열

```
int [][] intAry = new int[3][4];  
...  
//초기 값 지정  
intAry[0][0] = 1;  
intAry[0][1] = 2;  
intAry[0][2] = 3;  
intAry[0][3] = 4;  
...  
intAry[2][3] = 6;
```



2차원 배열 선언과 생성

■ 2차원 배열

- 2차원 배열이란 배열의 요소로 1차원 배열을 가지는 배열
- JAVA에서는 2차원 배열을 나타내는 Type을 따로 제공하지 않음
- 대신에 1차원 배열의 배열 요소로 또 다른 1차원 배열을 사용하여 2차원 배열을 나타낼 수 있음

2차원 배열 선언과 생성

■ 선언

- 1) 자료형[][] 배열명; // 추천 방법
- 2) 자료형[] 배열명[];
- 3) 자료형 배열명[][];

■ 초기화

- 1) 배열명 = new 자료형[개수1][개수2];
- 2) 배열명 = {{...}, {...}, {...}, ...};
- 3) 배열명 = new 자료형[][] {{...},{...}...};
- 4) 배열명 = new 자료형[개수][];

2차원 배열 선언과 생성

■ 2차원 배열 선언

```
int intArray[][];  
char charArray[][];  
float floatArray[][];
```

```
int[][] intArray;  
char[][] charArray;  
float[][] floatArray;
```

■ 2차원 배열 생성

```
intArray = new int[2][5];  
charArray = new char[5][5];  
floatArray = new float[5][5];
```

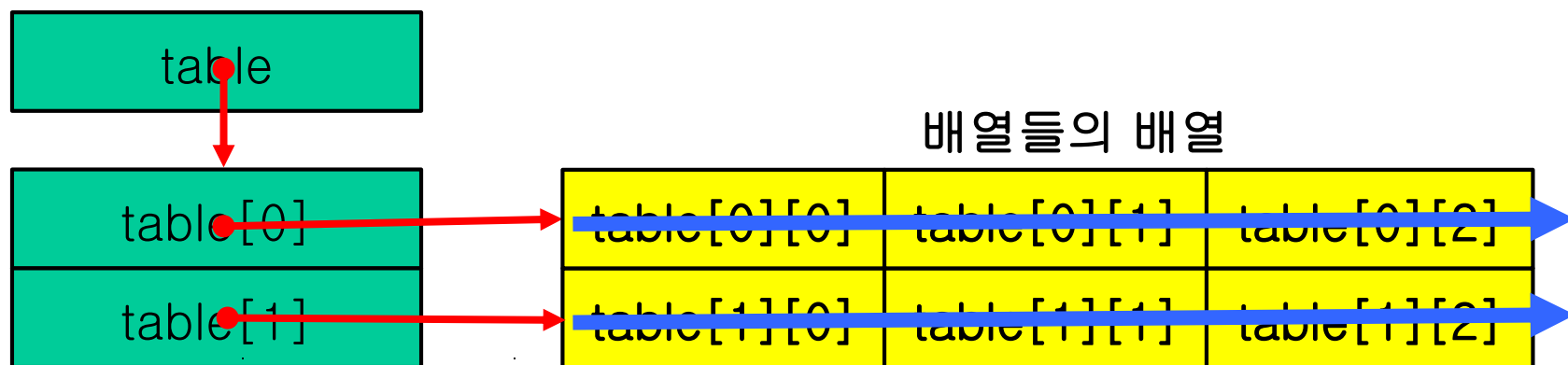
■ 2차원 배열 선언과 생성

```
int[][] intArray = new int[2][5];  
char[][] charArray = new char[5][5];  
Float[][] floatArray = new float[5][5];
```

2차원 배열 선언과 생성

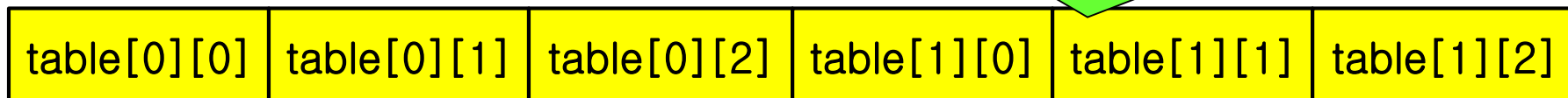
```
int[][] table; // 2차원 정수배열을 가리키는 참조변수 선언  
table = new int[2][3]; // 2 x 3 2차원 배열 생성
```

```
int[][] table = new int[2][3];
```



배열 요소의 수 : $2 * 3 = 6$

실제 기억장소에 저장된 모습



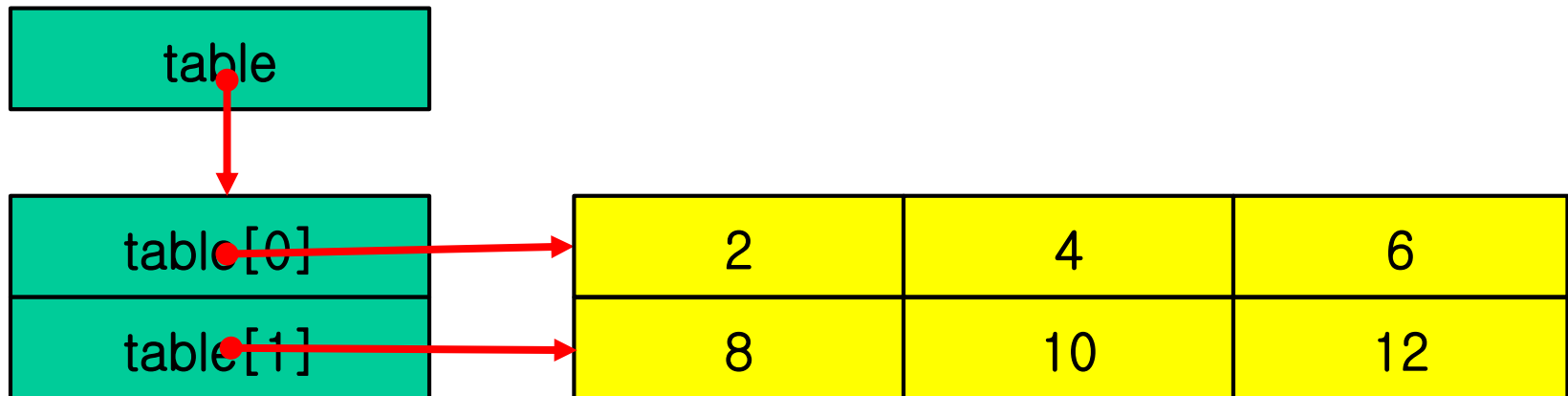
Row-Major Order (행 우선 법칙)

2차원 배열 초기화

- 선언, 생성, 초기화를 한꺼번에 하는 방법

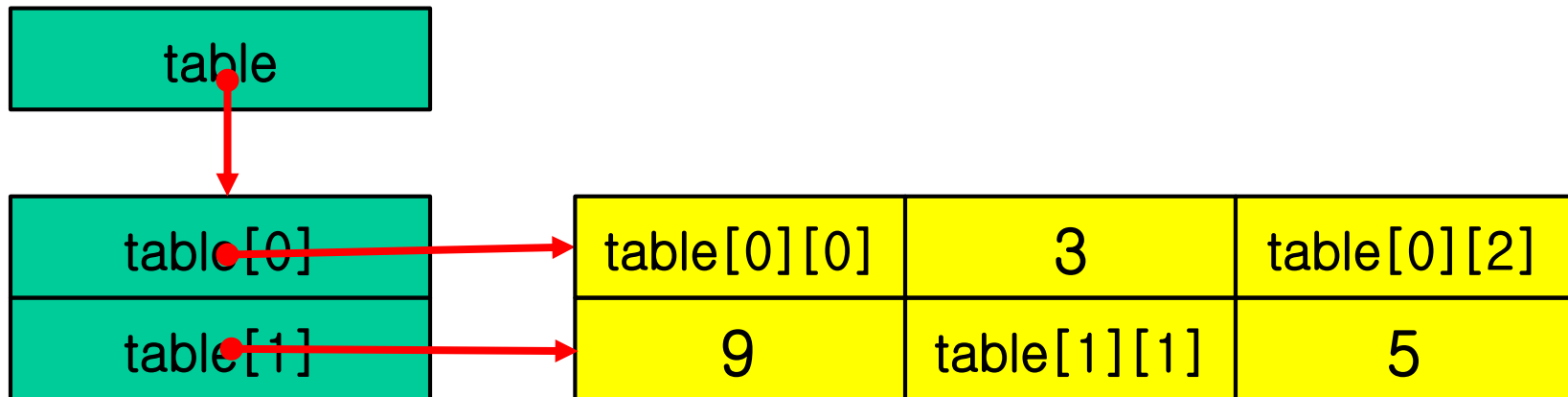
```
int[][] table = {{2, 4, 6}, {8, 10, 12}};
```

```
int[][] table = new int[][] {{2, 4, 6}, {8, 10, 12}};
```



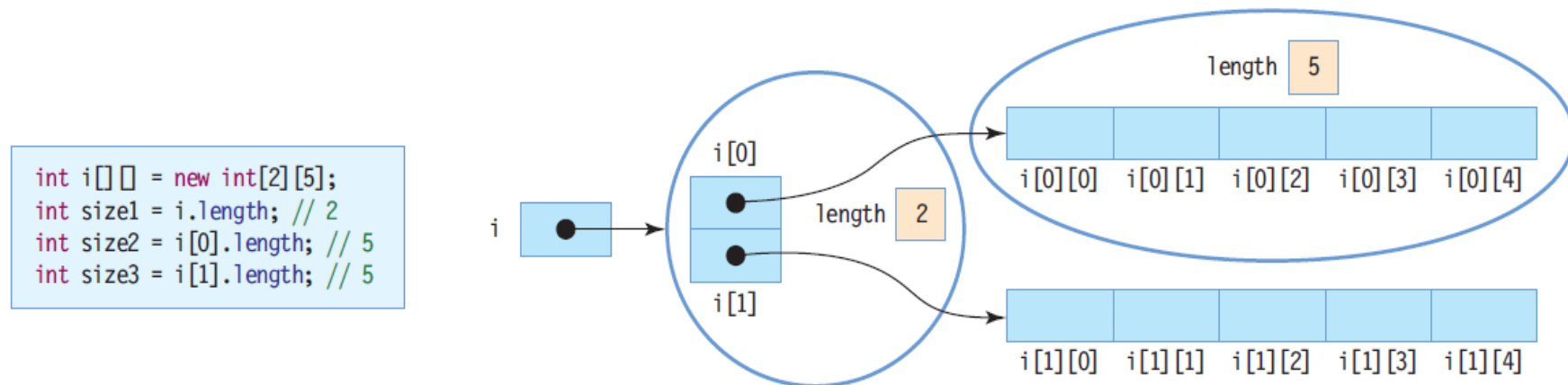
2차원 배열 조작

```
table[1][2] = 5;  
table[0][1] = 3;  
table[1][0] = table[1][2] + 4;
```



2차원 배열 조작

■ 2차원 배열의 length 필드



- `i.length` -> 2차원 배열의 행의 개수로서 2
- `i[n].length` -> n번째 행의 열의 개수
- `i[0].length` -> 0번째 행의 열의 개수로서 5
- `i[1].length` -> 1번째 행의 열의 개수로서 역시 5

2차원 배열 예제 0

- 2차원 배열의 요소값을 초기화하고, 요소의 값 출력하는 프로그램을 작성하여라.

2차원 배열 예제 0

```
public static void main(String[] args) {  
    int[][] score = new int[5][3];  
    score[0][0]=10; score[0][1]=90; score[0][2]=70;  
    score[1][0]=60; score[1][1]=80; score[1][2]=65;  
    score[2][0]=55; score[2][1]=60; score[2][2]=85;  
    score[3][0]=90; score[3][1]=75; score[3][2]=95;  
    score[4][0]=60; score[4][1]=30; score[4][2]=80;  
  
    for (int row = 0; row < score.length; row++) {  
        for (int col = 0; col < score[row].length; col++)  
            System.out.print(" " + score[row][col]);  
        System.out.println();  
    }  
}
```

평균 평점 구하기

```
public static void main(String[] args) {  
    double[][] score = {{3.3, 3.4},{3.5, 3.6}, {3.7, 4.0}, {4.1, 4.2}};  
    double sum = 0;  
  
    for(int year = 0; year < score.length; year++)  
        for(int term = 0; term < score[year].length; term++)  
            sum += score[year][term];  
    int n = score.length;  
    int m = score[0].length;  
    System.out.println("4학년 전체 평점 평균은" + sum/(n*m));  
}
```

2차원 배열 예제 1

- 학생 5명의 국어, 영어, 수학 점수를 처리하기 위해서는 데이터를 다음과 같이 저장할 수 있는데 JAVA 프로그램에서 어떤 데이터형을 사용하면 될까요 ?

	국어	영어	수학
1번 학생	85	60	70
2번 학생	90	95	80
3번 학생	75	80	100
4번 학생	80	70	95
5번 학생	100	65	80

2차원 배열은 행렬형태로 데이터를 처리할 때 사용

- 행 단위에는 각 학생의 정보가 저장
- 열 단위로는 각 과목별 점수가 저장

2차원 배열 예제 1

- JAVA의 선언문을 작성해보자

```
int[][] score = new int[5][3];
```

	0열 (국어)	1열(영어)	2열(수학)
0 행 (1번 학생)	score[0][0]	score[0][1]	score[0][2]
1 행(2번 학생)	score[1][0]	score[1][1]	score[1][2]
2 행(3번 학생)	score[2][0]	score[2][1]	score[2][2]
3 행(4번 학생)	score[3][0]	score[3][1]	score[3][2]
4 행(5번 학생)	score[4][0]	score[4][1]	score[4][2]

2차원 배열 score는 15(5 x 3)개의 원소를 저장하는 메모리가 할당

- 3번 학생의 영어 점수를 80점으로 수정하여보자?

```
score[2][1] = 80;
```


2차원 배열 예제 1

```
public static void main(String[] args) {  
    // int table[][] = new int[3][4]; // 3×4 배열의 선언  
    // int table[][] = {{0, 0, 0, 0}, {0, 1, 2, 3}, {0, 2, 4, 6}};  
    int table[][] = new int[][] {{0, 0, 0, 0}, {0, 1, 2, 3}, {0, 2, 4, 6}};  
  
    for (int row = 0; row < table.length; row++) {  
        for (int col = 0; col < table[row].length; col++) {  
            table[row][col] = row * col;  
        }  
    }  
  
    for (int row = 0; row < table.length; row++) {  
        for (int col = 0; col < table[row].length; col++) {  
            System.out.print(table[row][col] + " ");  
        }  
        System.out.println();  
    }  
}
```

0	0	0	0
0	1	2	3
0	2	4	6

2차원 배열 예제 2

- 5행 5열의 배열을 다음과 같이 초기화하고, 대각선이 지나
는 원소의 값을 합산하여 보자

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

2차원 배열 예제 2

```
public static void main(String[] args) {  
    int[][] table = new int[5][5];  
  
    int value = 1;  
    for (int i = 0; i < table.length; i++) {  
        for (int j = 0; j < table[i].length; j++) {  
            table[i][j] = value++;  
        }  
    }  
  
    int total = 0;  
    for (int i = 0; i < table.length; i++) {  
        for (int j = 0; j < table[i].length; j++) {  
            if (i == j)  
                total += table[i][j];  
        }  
    }  
    System.out.println("대각선 합 = " + total);  
}
```

2차원 배열 예제 3(매출 총액)

- 어느 회사의 최근 3년간 분기별 매출이 다음과 같을 때 3년간의 매출 총액과 연평균 매출을 구하여라

1분기	2분기	3분기	4분기
1230	2345	1495	1543
2387	1288	2191	3465
1675	1678	3215	2701

2차원 배열 예제 3(매출 총액)

```
public static void main(String[] args) {  
    int sales[][] = {{1230, 2345, 1495, 1543},  
                     {2387, 1288, 2191, 3465},  
                     {1675, 1678, 3215, 2701}};  
  
    int sum = 0;  
  
    for (int i = 0; i < sales.length; i++)  
        for (int j = 0; j < sales[i].length; j++)  
            sum += sales[i][j];  
  
    System.out.printf("지난 3년간 매출 총액은 %,d 이며 연평균  
                        매출은 %,1f 입니다.\n",  
                      sum, (float) sum / sales.length);  
}
```

2차원 배열 예제 4(총원 계산하기)

- 경북고등학교의 재학생 집계표가 아래와 같다. 학년별 총인원과 재학생 총원을 구하는 프로그램을 작성하여라.

	1반	2반	3반	4반	5반	6반	합계
1학년	41	43	45	42	44	39	
2학년	38	37	40	39	40	37	
3학년	35	37	41	38	38	37	

재학생 총원 : 명

2차원 배열 예제 4(총원 계산하기)

■ 출력 결과

경북고등학교 학생 집계표

1반 2반 3반 4반 5반 6반 총합

41 43 45 42 44 39 254

38 37 40 39 40 37 231

35 37 41 38 38 37 226

총원 : 711 명

2차원 배열 예제 4(총원 계산하기)

- 몇 차원의 크기는 어떠한 배열을 생성하면 될까요 ?
 - 배열의 모든 원소의 데이터형이 정수형이므로 2차원 배열의 3행 7열의 배열을 생성
- 배열을 초기화해보자

```
int[][] school = new int[3][7];  
school = new int[][]{{41, 43, 45, 42, 44, 39, 0},  
                     {38, 37, 40, 39, 40, 37, 0},  
                     {35, 37, 41, 38, 38, 37, 0}};
```


2차원 배열 예제 4(총원 계산하기)

```
public static void main(String[] args) {  
    int[][] school = new int[3][7];  
    school = new int[][]{{41, 43, 45, 42, 44, 39, 0},  
                          {38, 37, 40, 39, 40, 37, 0},  
                          {35, 37, 41, 38, 38, 37, 0}};  
  
    for (int i = 0; i < school.length; i++){  
        for (int j = 0; j < school[i].length - 1; j++) {  
            school[i][school[i].length-1] += school[i][j];  
        }  
    }  
  
    int total = 0;  
    for (int i = 0; i < school.length; i++)  
        total += school[i][school[i].length-1];  
}
```



2차원 배열 예제 3(총원 계산하기)

```
System.out.println("Wt 경북고등학교 학생 집계표");
System.out.println("*****");
System.out.println("  1반  2반  3반  4반  5반  6반  총합 ");
System.out.println("*****");
for (int i = 0; i < school.length; i++){
    for (int j = 0; j < school[i].length; j++) {
        System.out.printf(" %4d", school[i][j]);
    }
    System.out.println();
}
System.out.println("*****");
System.out.printf("WtWtWt 총원 : %d 명Wn", total);
}
```

2차원 배열 예제 5(성적 출력)

- 5명의 3과목(국어, 영어, 수학)의 성적이 다음과 같을 때 개인의 성적의 합과 평균을 구하시오.

이름	국어	영어	수학	총점	평균
홍길동	90	90	95	275	91.67
이혜인	87	88	90	265	88.33
김명성	75	100	95	270	90.00
경복대	100	100	99	299	99.67
이대학	100	75	89	264	88.00

2차원 배열 예제 5(성적 출력)

■ 실행결과

이름	국어	영어	수학	총점	평균
홍길동	90	90	95	275	91.67
이혜인	87	88	90	265	88.33
김명성	75	100	95	270	90.00
경복대	100	100	99	299	99.67
이대학	100	75	89	264	88.00

2차원 배열 예제 5(성적 출력)

- 이름(name), 국어(korean), 수학(math), 영어(english), 총점(total), 평균(avg)을 인스턴스 변수로 선언한다.
- 이름과 국어, 수학, 영어, 총점 그리고 평균의 데이터형이 각각 문자열(String), 정수형(int), 실수형(float)이므로 이름은 문자열 1차원 배열, 점수는 정수형 2차원 배열, 평균은 실수형 1차원 배열을 사용

2차원 배열 예제 5(성적 출력)

■ 배열 선언과 생성 및 초기화 방법

```
String[] name = {"홍길동", "이혜인", "김명성", "경북대", "이대학"};  
int[][] score = {{90, 90, 95, 0}, {87, 88, 90, 0},  
                 {75, 100, 95, 0}, {100, 100, 99, 0},  
                 {100, 75, 89, 0}};  
float[] avg = new float[name.length];
```

2차원 배열의 연습 문제 1

- 2차원 배열을 이용하여 다음 표의 빈칸을 채우는 프로그램을 작성하시오

	홍길동	이순신	정약용	과목평균
국어	90	80	65	
영어	70	86	99	
수학	78	90	99	
총점				
평균				
등수				

2차원 배열의 연습 문제 2

- 5명의 오른쪽 시력과 왼쪽 시력의 평균을 구하는 프로그램을 작성해보자

0.1	4	C	그					
0.15	2	0	가					
0.2	3	7	0	느	✈			
0.3	5	6	0	스	🦋			
0.4	2	3	0	구	🐦			
0.5	7	4	C	그	🐟			
0.6	3	6	0	0	가	🚗		
0.7	4	2	7	0	0	나	🦋	🐾
0.8	7	6	2	0	0	느	🐾	🐾
0.9	3	7	4	0	0	구	🐟	☂
1.0	4	2	6	0	0	시	🚗	🦋
1.2	7	3	2	0	0	그	🐦	✈
1.5	6	7	3	0	0	나	☂	🐾
2.0	4	2	7	0	0	스	🦋	🐾

2차원 배열의 연습 문제 3

- 다음은 불규칙 동사의 과거와 과거분사이다. 이들을 배열에 기억시켜서 현재형 단어를 입력하면 과거와 과거분사를 출력하는 프로그램을 작성하라.

현재	awake	become	build	catch	choose	do	go
과거	awoke	became	built	caught	chose	did	went
과거분사	awoke	become	built	caught	chosen	done	gone

2차원 배열의 연습 문제 4

- 대각선이 지나는 오른쪽 요소들을 모두 합산하기

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

$$\begin{aligned} Y = & A(1, 1) + A(1, 2) + A(1, 3) + A(1, 4) + A(1, 5) \\ & + A(2, 2) + A(2, 3) + A(2, 4) + A(2, 5) \\ & + A(3, 3) + A(3, 4) + A(3, 5) \\ & + A(4, 4) + A(4, 5) \\ & + A(5, 5) \end{aligned}$$

2차원 배열의 연습 문제 5

- 대각선의 왼쪽에 있는 요소들을 합산하기

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

$$\begin{aligned} Y = & A(1, 1) + \\ & A(2, 1) + A(2, 2) + \\ & A(3, 1) + A(3, 2) + A(3, 3) + \\ & A(4, 1) + A(4, 2) + A(4, 3) + A(4, 4) + \\ & A(5, 1) + A(5, 2) + A(5, 3) + A(5, 4) + A(5, 5) \end{aligned}$$

2차원 배열의 연습 문제 6

■ 대각선의 아래쪽의 요소만 합산하기

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

$$\begin{aligned} Y = & A(1, 5) + \\ & A(2, 4) + A(2, 5) + \\ & A(3, 3) + A(3, 4) + A(3, 5) + \\ & A(4, 2) + A(4, 3) + A(4, 4) + A(4, 5) + \\ & A(5, 1) + A(5, 2) + A(5, 3) + A(5, 4) + A(5, 5) \end{aligned}$$

2차원 배열의 연습 문제 7

- 배열의 중심에서 시작하여 좌우로 하나씩 늘려서 삼각형 형태의 요소들만 합산하기

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

$$\begin{aligned} Y = & A(3, 3) + \\ & A(4, 2) + A(4, 3) + A(4, 4) + \\ & A(5, 1) + A(5, 2) + A(5, 3) + A(5, 4) + A(5, 5) \end{aligned}$$

2차원 배열의 연습 문제 8

- 배열의 왼쪽에서 오른쪽으로 중심까지 블록 삼각형에 해당되는 요소들만 합산하기

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

$$\begin{aligned} Y = & A(1, 1) + A(2, 1) + A(3, 1) + A(4, 1) + A(5, 1) + \\ & A(2, 2) + A(3, 2) + A(4, 2) + \\ & A(3, 3) \end{aligned}$$

2차원 배열의 연습 문제

- 다음은 불규칙 동사의 과거와 과거분사이다. 이들을 배열에 기억시켜서 현재형 단어를 입력하면 과거와 과거분사를 출력하는 프로그램을 작성하라.

현재	awake	become	build	catch	choose	do	go
과거	awoke	became	built	caught	chose	did	went
과거분사	awoke	become	built	caught	chosen	done	gone

지역의 주택수 통계

- 아래의 표를 보고 각 통의 주택수, 각 동의 주택수, 구의 총 주택수를 구하여 출력하는 프로그램을 작성하여라.

행복구의 주택 집계표

행복구											
행복 1 동						행복 2 동					
1 통			2 통			1 통			2 통		
1반	2반	3반	1반	2반	3반	1반	2반	3반	1반	2반	3반
32	26	41	28	29	34	29	40	45	47	42	50

행렬 덧셈

- 3 x 3 정방행렬 2개를 읽어 이의 합을 구하는 프로그램을 작성하여라.

>>행렬의 덧셈과 뺄셈<<

두 행렬 A, B가 서로 같은 꼴일 때
서로 같은 위치에 있는 성분끼리 +, -

$$A = \begin{pmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{pmatrix}, B = \begin{pmatrix} b_{11} & b_{12} \\ b_{21} & b_{22} \end{pmatrix}$$

$$A \pm B = \begin{pmatrix} a_{11} \pm b_{11} & a_{12} \pm b_{12} \\ a_{21} \pm b_{21} & a_{22} \pm b_{22} \end{pmatrix}$$

행렬 덧셈

```
public static void main(String[] args) {  
    int[][] test = {{1, 2, 3}, {3, 4, 5}};  
    int[][] value = {{4, 5, 6}, {6, 7, 8}};  
  
    for (int i = 0; i < test.length; ++i) {  
        for (int j = 0; j < test[i].length; ++j) {  
            System.out.printf("%4d", test[i][j] + value[i][j]);  
        }  
        System.out.println();  
    }  
}
```

3년 매출과 년 평균 매출

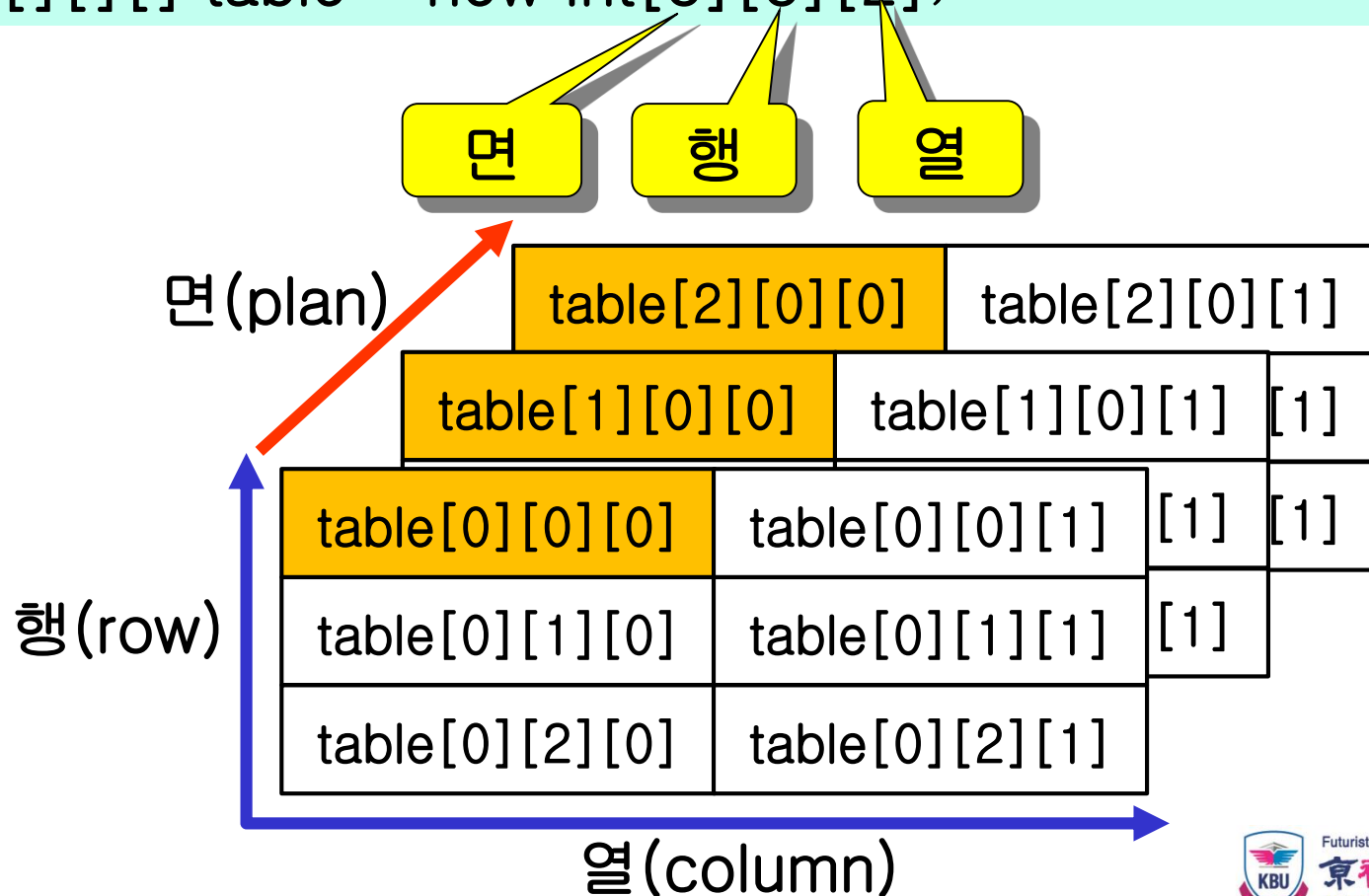
```
public class SalesRevenue {  
    public static void main (String[] args) {  
        int intArray[][] = {{90, 90, 110, 110}, {120, 110, 100, 110},  
                             {120, 140, 130, 150}};  
  
        double sum = 0;  
        for (int i = 0; i < intArray.length; i++)  
            for (int j = 0; j < intArray[i].length; j++)  
                sum += intArray[i][j];  
        System.out.println("지난 3년간 매출 총액은 " + sum +  
                            "이며 연평균 매출은 " + sum/intArray.length + "입니다.");  
    }  
}
```

지난 3년간 매출 총액은 1380.0이며 연평균 매출은 460.0입니다

다차원 배열 선언

- 3차원 이상의 배열도 사용이 가능함
- 3차원 배열 선언, 구성

```
int[][][] table = new int[3][3][2];
```



대리점 관리

- 어느 전자회사의 각 대리점의 상품별 연도별 매출표를 보고 다음의 조건을 계산하는 프로그램을 작성하여라.

상품	년도	대리점				
		서울	부산	대구	인천	광주
TV	2018	9	7	4	3	2
	2019	5	6	7	8	7
	2020	14	21	7	6	9
냉장고	2018	8	9	6	7	8
	2019	12	14	8	9	12
	2020	20	16	12	11	18

상품별, 연도별, 대리점별 매출고 (단위 1,000만원)

대리점 관리

- 상품별 매출액을 집계하여라.
- 전자회사의 총 매출액을 집계하여라
- <추가>
 - 대리점의 년도별 매출액을 계산하여라.
 - 전자회사의 년도별 매출액을 집계하여라.

대리점 관리

■ 자료구조

년도	대리점					TV
	서울	부산	대구	인천	광주	
2018	년도	대리점				
2019		서울	부산	대구	인천	광주
2020	2018	8	9	6	7	8
	2019	12	14	8	9	12
	2020	20	16	12	11	18

냉장고

```
int[][][] dealer = new int[2][4][6];
```

대리점 관리

```
public static void main(String[] args) {  
    int[][][] dealer = new int[2][4][6];  
    String[] item = {"TV", "냉장고"};  
  
    dealer = new int[][][]{{{9, 7, 4, 3, 2, 0},  
                             {5, 6, 7, 8, 7, 0},  
                             {14, 21, 7, 6, 9, 0},  
                             {0, 0, 0, 0, 0, 0}},  
                             {{8, 9, 6, 7, 8, 0},  
                             {12, 14, 8, 9, 12, 0},  
                             {20, 16, 12, 11, 18, 0},  
                             {0, 0, 0, 0, 0, 0}}};  
}
```


대리점 관리

```
for (int i = 0; i < dealer.length; i++) {  
    for (int j = 0; j < dealer[i].length; j++) {  
        for (int k = 0; k < dealer[i][j].length - 1; k++) {  
            dealer[i][j][dealer[i][j].length - 1] += dealer[i][j][k];  
        }  
    }  
}  
  
for (int i = 0; i < dealer.length; i++) {  
    for (int j = 0; j < dealer[i].length - 1; j++) {  
        for (int k = 0; k < dealer[i][j].length; k++) {  
            dealer[i][dealer[i].length - 1][k] += dealer[i][j][k];  
        }  
    }  
}  
  
int sum = 0;  
for (int i = 0; i < dealer.length; i++) {  
    sum += dealer[i][dealer[i].length - 1]  
        [dealer[i][dealer[i].length - 1].length - 1];  
}
```

대리점 관리

```
System.out.println("Wt 전자회사 매출 집계표");
for (int i = 0; i < dealer.length; i++) {
    System.out.printf("WtWtWtWtWt 품목 : %sWn", item[i]);
    System.out.println("*****");
    System.out.println("  서울  부산  대구  인천  광주  총합 ");
    System.out.println("*****");
    for (int j = 0; j < dealer[i].length; j++) {
        for (int k = 0; k < dealer[i][j].length; k++) {
            System.out.printf(" %4d", dealer[i][j][k]);
        }
        System.out.println();
    }
    System.out.println("*****");
    System.out.printf("WtWtWt %s 매출총액 : %,d 천만원WnWn",
                                                              item[i], dealer[i][3][5]);
}
System.out.printf("WtWt 매출 총액 : %,d 천만원Wn", sum);
}
```

소망 빌라

- 소망 빌라는 3동이 5층으로 이뤄져 있으며 각 층은 3세대로 이루어져 있다.
- 프로그램 실행 시 사용자로부터 각 동의 1층에서 5층까지 각층의 세대에 사는 사람의 수를 입력 받은 후, 소망빌라에 사는 동별, 층별 및 전체 사람의 수와 소망빌라에 사는 세대별 평균 사람의 수를 계산하여 출력하는 프로그램 작성하여라.



소망 빌라

■ 자료 구조

		1호	2호	3호	합계	가동
5		1호	2호	3호	합계	나동
4	5층	1호	2호	3호	합계	다동
3	4층	5층				
2	3층	4층				
1	2층	3층				
	1층	2층				
		1층				

```
int[][][] villa = new int[3][5][3];
int[][] total = new int[3][1];
```

```
int[][][] villa = new int[3][5][4];
```

소망 빌라

```
public static void main(String[] args) {  
    Scanner keyboard = new Scanner(System.in);  
    int[][][] villa = new int[3][5][4];  
    String[] dong = {"가동", "나동", "다동"};  
    int[] total = new int[villa.length];  
  
    for (int i = 0; i < villa.length; i++) {  
        for (int j = 0; j < villa[i].length; j++) {  
            for (int k = 0; k < villa[i][j].length - 1; k++) {  
                System.out.printf("%s %d층 %d호 가족수 : ", dong[i], j + 1, k + 1);  
                // villa[i][j][k] = keyboard.nextInt();  
                villa[i][j][k] = (int) ((Math.random() * (6 - 1 + 1)) + 1);  
            }  
        }  
    }  
}
```

소망 빌라

```
for (int i = 0; i < villa.length; i++) {  
    for (int j = 0; j < villa[i].length; j++) {  
        for (int k = 0; k < villa[i][j].length - 1; k++) {  
            villa[i][j][villa[i][j].length - 1] += villa[i][j][k];  
        }  
    }  
}
```

```
for (int i = 0; i < villa.length; i++) {  
    for (int j = 0; j < villa[i].length; j++) {  
        total[i] += villa[i][j][villa[i][j].length - 1];  
    }  
}
```

```
int sum = 0;  
for (int i = 0; i < total.length; i++)  
    sum += total[i];
```



Futuristic Innovator

京福大學校
KYUNGBOK UNIVERSITY

소망 빌라

```
System.out.println("Wt 소망빌라 주민 집계표");
for (int i = 0; i < villa.length; i++) {
    System.out.printf("WtWtWtWt %sWn", dong[i]);
    System.out.println("*****");
    System.out.println("    1호 2호 3호 총합 ");
    System.out.println("*****");
    for (int j = 0; j < villa[i].length; j++) {
        for (int k = 0; k < villa[i][j].length; k++) {
            System.out.printf(" %4d", villa[i][j][k]);
        }
        System.out.println();
    }
    System.out.println("*****");
    System.out.printf("WtWtWt %s 총원 : %d 명Wn", dong[i], total[i]);
}
System.out.printf("WtWt 소망빌라 총원 : %,d 명Wn", sum);
}
```