Ch4
배열과 문자열

객체지향프로그래밍(기본) 2019

경상대학교 항공우주및소프트웨어공학전공

4.1 배열

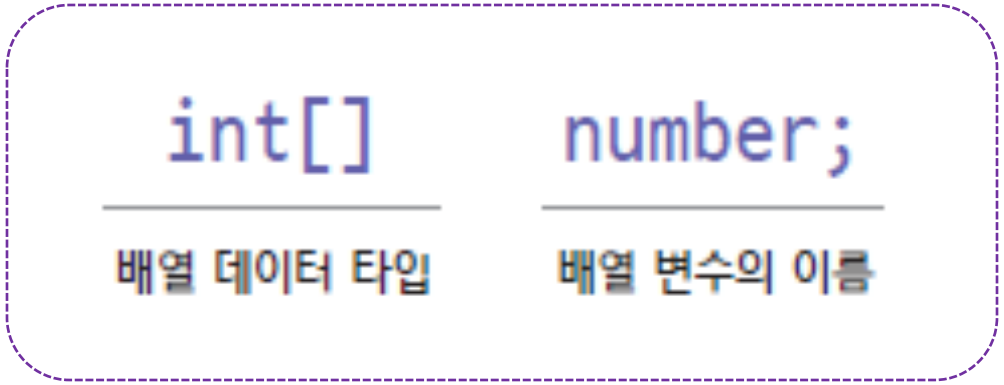
배열(array)은 같은 데이터타입을 연속적인 저장 공간에 배치하고 인덱스(index)를 이용하여 배열 내의 상대 위치에 있는 데이터를 읽거나 쓸 수 있는 자료구조이다.

인덱스	0	1	2	3	4
값	10	20	30	40	50

자바에서 배열의 인덱스는 0부터 시작되며
마지막 인덱스는 (배열의 크기-1)이다.

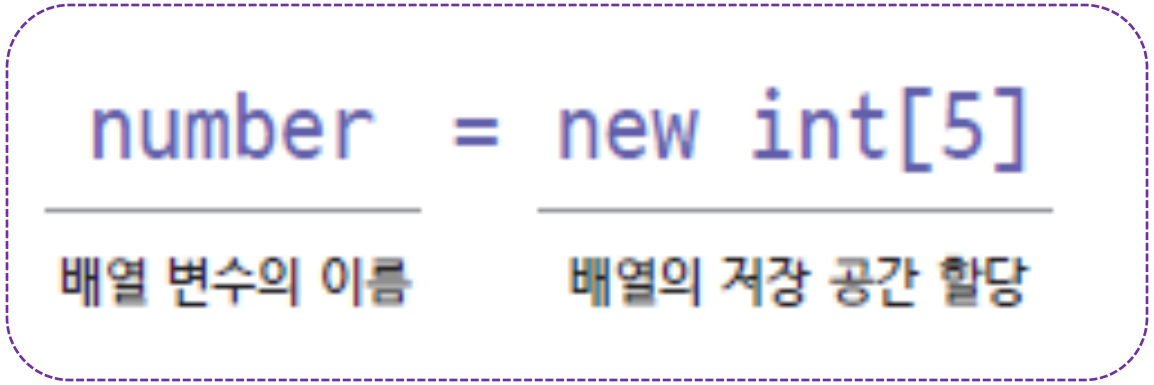
배열의 선언

자바에서 배열은 동일한 종류의 데이터 타입의 집합이다.
배열을 선언하기 위해 어떤 종류의 데이터 타입을 사용 할 것인지 선언해야 하며 데이터 타입 뒤에 []를 붙인 후 배열의 이름을 정의한다.



배열의 메모리 공간 할당

배열은 데이터의 집합이기 때문에 배열을 사용하기 위해서는 사용되는 배열의 원소 개수의 크기만큼 저장 공간을 할당해 주어야 한다.
배열에 저장 공간을 할당하기 위해서 먼저 선언된 배열의 이름에 new 배열의 데이터 타입 [데이터 개수]를 대입하는 형태로 선언한다.



```
1  public class Example {
2    public static void main(String[] args) {
3      int[] number;
4      number = new int[5];
5
6      number[0] = 10;
7      number[1] = 20;
8      number[2] = 30;
9      number[3] = 40;
10     number[4] = 50;
11
12     for(int i = 0 ; i <5 ; i++)
13     {
14       System.out.print(number[i]+" ");
15     }
16   }
17 }
```

결과

```
10 20 30 40 50
```

배열의 초기화

배열에 사용될 데이터들이 지정되어 있다면 배열을 선언하면서 초기화하여 사용할 수 있다

```
1) int[] number = {10,20,30,40,50};
2) int[] number = new int[] {10,20,30,40,50};
```

배열의 선언 / 배열의 초기화

배열의 초기화

```
1  public class Example {
2    public static void main(String[] args) {
3      int[] number  = {10, 20, 30, 40, 50};
4      //int[] number  = new int[] {10, 20, 30, 40, 50};
5
6      for(int i = 0 ; i < 5 ; i++)
7      {
8        System.out.print(number[i]+" ");
9      }
10   }
11 }
```

결과

```
10 20 30 40 50
```

배열의 크기

배열의 크기는 (배열의 이름).length에 저장되어 있다.

```
1  public class Example {
2    public static void main(String[] args) {
3      int[] number  = {10, 20, 30, 40, 50};
4
5      for(int i = 0 ; i < number.length ; i++)
6      {
7        System.out.print(number[i]+" " );
8      }
9    }
10 }
```

결과

```
10 20 30 40 50
```

배열을 위한 반복문 형식

배열을 위한 반복문 형식은 배열의 크기를 알지 못하더라도 사용 할 수 있어 오류가 적고 인덱스를 사용하지 않아도 배열의 원소에 접근이 가능하다.

```
for ( 배열의 원소 저장 변수 : 배열) {
        반복할 문장
        }
```

```
1  public class Example {
2    public static void main(String[] args) {
3      int[] number  = {10, 20, 30, 40, 50};
4
5      for(int data : number)
6      {
7        System.out.print(data+" ");
8      }
9    }
10 }
```

결과

```
10 20 30 40 50
```

배열의 응용(버블 정렬)

정렬은 데이터의 크기에 따른 순서를 맞추는 것으로 그 중 가장 간단한 방식인 버블 정렬이 있다. 버블 정렬은 맨 앞의 값부터 그 뒤의 값을 순차적으로 비교하여 크기 순서대로 나열하는 방법이다.

(50 30) 40 20 10	(30 40) 20 10 50	(30 20) 10 40 50	(20 10) 30 40 50
비교	비교	비교	비교
30 (50 40) 20 10	30 (40 20) 10 50	20 (30 10) 40 50	10 20 30 40 50
비교	비교	비교	
30 40 (50 20) 10	30 20 (40 10) 50	20 10 30 40 50	
비교	비교		
30 40 20 (50 10)	30 20 10 40 50		
비교			
30 40 20 10 50			

배열의 응용(버블 정렬)

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

```
5개의 숫자를 입력 하시오 :
50
30
40
20
10
정렬된 숫자 :
10 20 30 40 50
```

4.2 다차원 배열

다차원 배열은 2차원 이상의 구조를 가지는 배열로 2차원 배열의 경우 행렬로, 3배열의 경우 육면체로 표현하여 이해한다. 그 이상 차원의 배열의 경우 사람의 인지로는 이해하기 힘들어 잘 사용하지 않는다.

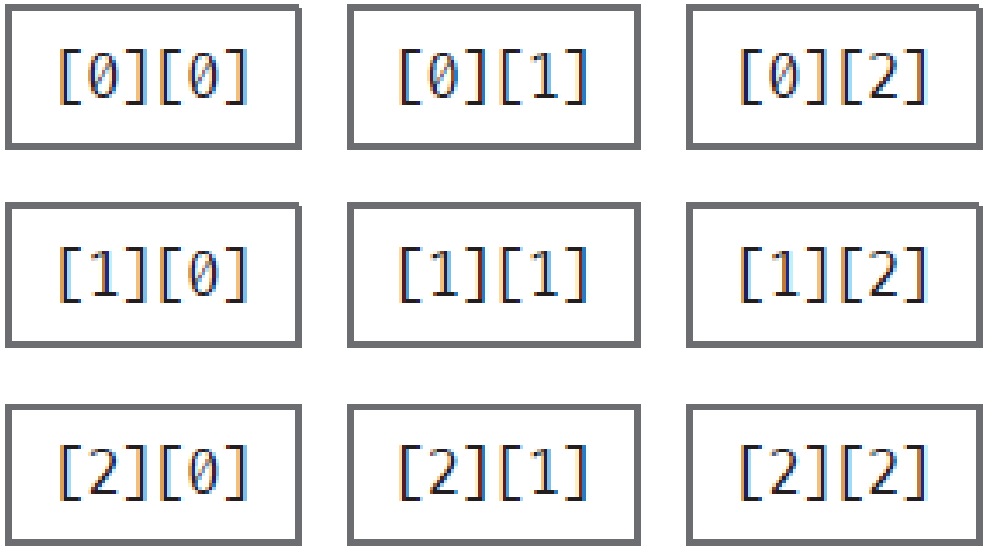


그림 4-7 2 차원 배열의 구조

행렬의 합

```
1  public class Example {
2    public static void main(String[] args) {
3      int[][] A = {{1,2,3},{4,5,6},{7,8,9}};
4      int[][] B = {{2,4,6},{8,10,12},{14,16,18}};
5      int[][] C = new int[3][3];
6
7      for(int i=0; i<3 ;i++)
8      {
9        for(int j = 0 ; j<3; j++)
10       {
11          C[i][j] = A[i][j] + B[i][j];
12       }
13     }
14
15     for(int i=0; i<3 ;i++)
16     {
17       for(int j = 0 ; j<3; j++)
18       {
19          System.out.print(C[i][j] + " ");
20       }
21       System.out.println();
22     }
23   }
24 }
```

결과

```
3 6 9
12 15 18
21 24 27
```

4.3 ArrayList

ArrayList는 배열과 비슷한 역할을 하지만 배열에 비하여 다양한 일을 쉽게 할 수 있다. 배열은 만들어 질 때 배열의 길이를 결정해야 하지만 ArrayList의 경우 객체의 크기를 지정하지 않고 사용 할 수 있다. 배열은 배열의 길이를 초과하는 원소의 개수를 저장 할 수 없지만 ArrayList에는 이러한 제약이 없다.

-ArrayList 주요 메소드

- **add: 새로운 데이터를 추가**
- **get: 원하는 원소를 가져옴**
- **contains: 원소가 있는지를 확인하여 부울 값으로 출력**
- **remove: 원소를 지울 때 사용**

4.4 문자열

문자열이란 문장을 뜻한다. 자바에서 문자열에 해당하는 자료형은 String이다. 문자열을 표현하는 방법은 다음과 같다.

```
String str = "hello JAVA";
```

```
String str = new String("hello JAVA");
```

문자열의 비교 1

```
1  public class Example {
2    public static void main(String[] args) {
3      String str1 = "hello JAVA";
4      String str2 = new String("hello JAVA");
5
6      System.out.println(str1);
7      System.out.println(str2);
8
9      if(str1 == str2)
10     {
11       System.out.println("같다.");
12     }
13     else
14     {
15       System.out.println("다르다.");
16     }
17   }
18 }
```

결과

```
hello JAVA
hello JAVA
다르다.
```

문자열의 비교 2

```
1  public class Example {
2    public static void main(String[] args) {
3      String str1 = "hello JAVA";
4      String str2 = new String("hello JAVA");
5
6      System.out.println(str1);
7      System.out.println(str2);
8
9      if(str1.equals(str2))
10     {
11       System.out.println("같다.");
12     }
13     else
14     {
15       System.out.println("다르다.");
16     }
17   }
18 }
```

결과

```
hello JAVA
hello JAVA
같다.
```

```java
import java.util.*;

public class Example {
  public static void main(String[] args) {
    Scanner input =new Scanner(System.in);

    System.out.print("입력 숫자: ");
    int number = input.nextInt();

    if(number>0)
      System.out.print("양수입니다.");

    if(number<0)
      System.out.print("음수입니다.");
  }
}
```

2개의 if문

결과

```
입력 숫자: 10
양수입이다.
```

결과

```
입력 숫자: -10
음수입이다.
```

문자열의 길이와 인덱스 확인

```
1  public class Example {
2    public static void main(String[] args) {
3      String str1 = "hello JAVA";
4
5      System.out.println(str1.length());
6      System.out.println(str1.indexOf("h"));
7      System.out.println(str1.indexOf("e"));
8      System.out.println(str1.indexOf("l"));
9      System.out.println(str1.indexOf("0"));
10     System.out.println(str1.indexOf("JA"));
11   }
12 }
```

결과

```
10
0
1
2
-1
6
```

문자열의 문자 변환 및 반환

```
1  public class Example {
2    public static void main(String[] args) {
3      String str1 = "hello JAVA";
4
5      System.out.println(str1.replace("hello", "Hello"));
6      System.out.println(str1.substring(4));
7      System.out.println(str1.substring(6,10));
8    }
9  }
```

결과

```
Hello JAVA
o JAVA
JAVA
```

대소문자 변환

```
1  public class Example {
2    public static void main(String[] args) {
3      String str1 = "hello JAVA";
4
5      System.out.println(str1);
6      System.out.println(str1.toUpperCase());
7      System.out.println(str1.toLowerCase());
8    }
9  }
```

문자열을 모두 대문자로 변환

문자열을 모두 소문자로 변환

결과

```
hello JAVA
HELLO JAVA
hello java
```

문자열 숫자를 숫자로 변환

```
1  public class book {
2    public static void main(String[] args) {
3      String str = "1024";
4      int number;
5
6      System.out.println(str+256);
7
8      number = Integer.parseInt(str);
9
10     System.out.println(number+256);
11   }
12 }
```

결과

```
1024256
1280
```

숫자를 문자열 숫자로 변환

```
1  public class Example {
2    public static void main(String[] args) {
3      int number = 1024;
4      String str ;
5
6      System.out.println(number+256);
7
8      str = Integer.toString(number);
9
10     System.out.println(str+256);
11   }
12 }
```

결과

```
1280
1024256
```