JAVA 프로그래밍

Chapter 07 배열

목차

- 배열의 소개
- 배열의 분석
- 배열의 초기화
- 배열의 복사
- 객체 배열
- 다차원 배열
- 다차원 배열의 분석

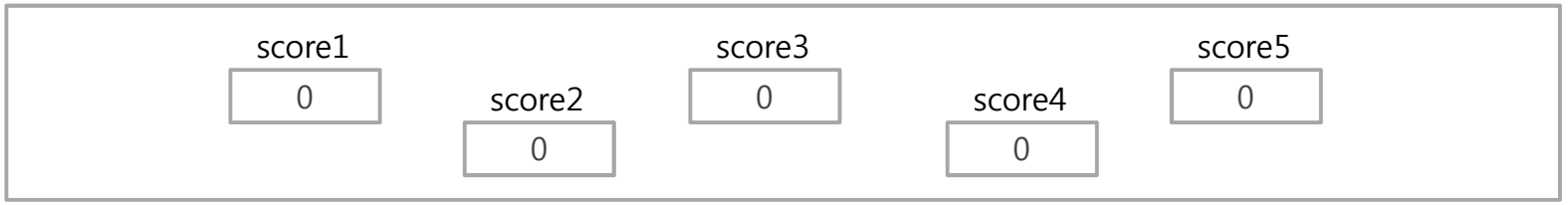
배열의 소개

❖ 배열의 정의

- 동일한 자료 형으로 선언된 데이터 공간을 메모리 상에 연속적으로 나열하여 데이터 관리의 효율성을 높인 것

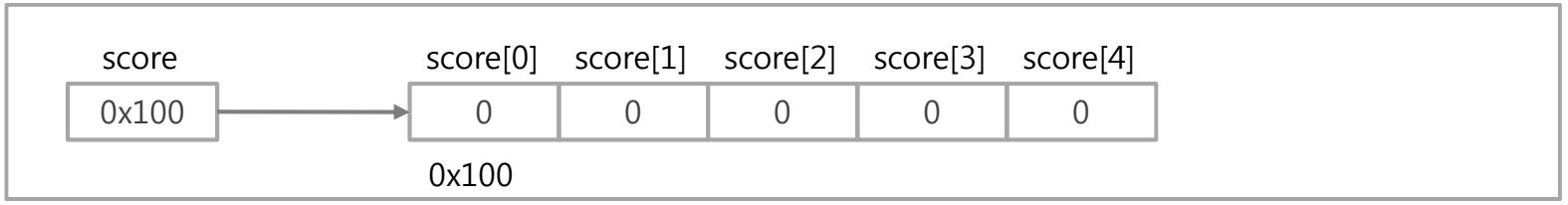
❖ 5개의 int형 변수 선언

- int score1 = 0, score2 = 0, score3 = 0, score4 = 0, score5 = 0;



❖ 배열을 이용한 5개의 int형 변수 선언

- int[] score = new int[5]; 또는 int score[] = new int[5];



배열의 소개

❖ 배열의 종류
- 기본 데이터 타입 배열 (byte[], int[], long[], float[], double[] 등 자바 8가지 기본형)
- 객체 배열 (String[], Integer[] 등 모든 레퍼런스 타입)

❖ 배열의 특징
- 같은 데이터 타입의 변수를 한꺼번에 여러 개 생성할 수 있다.
- 배열의 크기는 배열의 첨자로 결정된다.
- 첨자에 해당하는 만큼의 같은 데이터 타입을 가진 변수가 생성된다.
- 배열의 요소는 변수이다.
- 배열의 메모리는 연속적으로 잡히게 된다.
- 배열의 이름은 연속된 변수들을 참조하기 위한 참조값이다.
- 자바에서 배열은 객체이다.

❖ 배열의 검색
- 배열은 첨자(index)를 통해서 배열 내에 존재하는 모든 변수를 검색할 수 있다.

❖ 배열의 단점
- 배열을 생성할 때 첨자를 이용해서 배열의 크기를 정해버리면 배열의 크기를 변경할 수 없다.

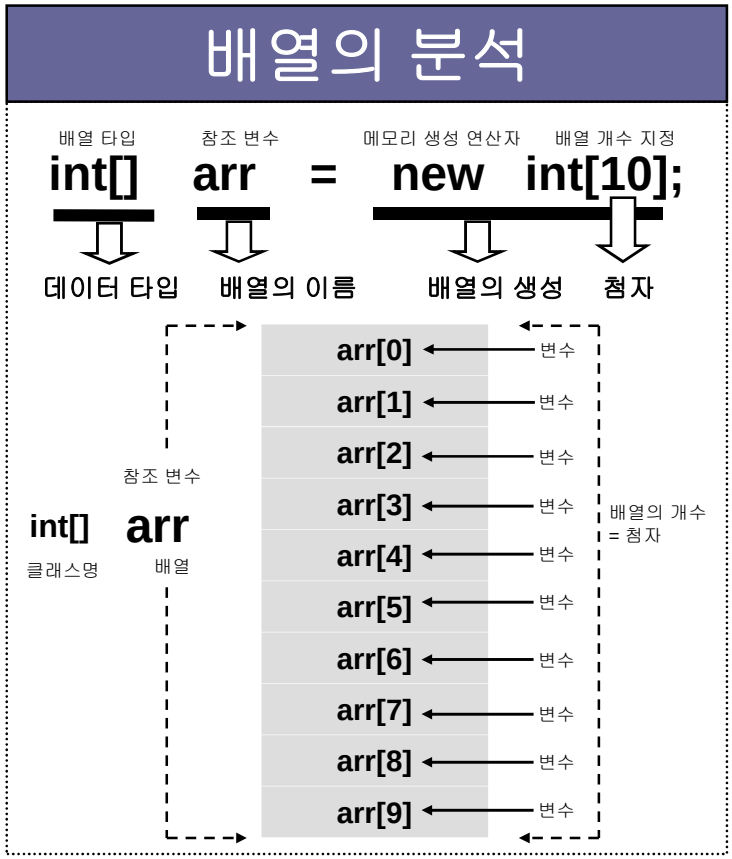
배열의 분석

❖ 배열의 생성
- int[] ar = new int[10];

❖ 기본 데이터 타입 배열
- 기본 데이터 타입은 변수의 선언과 동시에 메모리가 생성되기 때문에 기본 데이터 타입으로 배열을 선언하면 배열의 전체 메모리가 생성된다.

❖ 배열의 특징
- 배열의 첨자는 배열의 크기를 의미한다.
- 배열의 이름은 참조 값이다.
- 배열끼리의 할당은 참조 값에 대한 값 복사다.

❖ 배열의 개수를 확인하는 멤버
- 배열의 개수는 length 멤버를 이용해서 알 수 있다.
- int[] arr = new int[10];
- int size = arr.length; // size는 10이 된다.
- length 필드는 읽기 전용 필드이기 때문에 값을 바꿀 수가 없다.



배열의 초기화

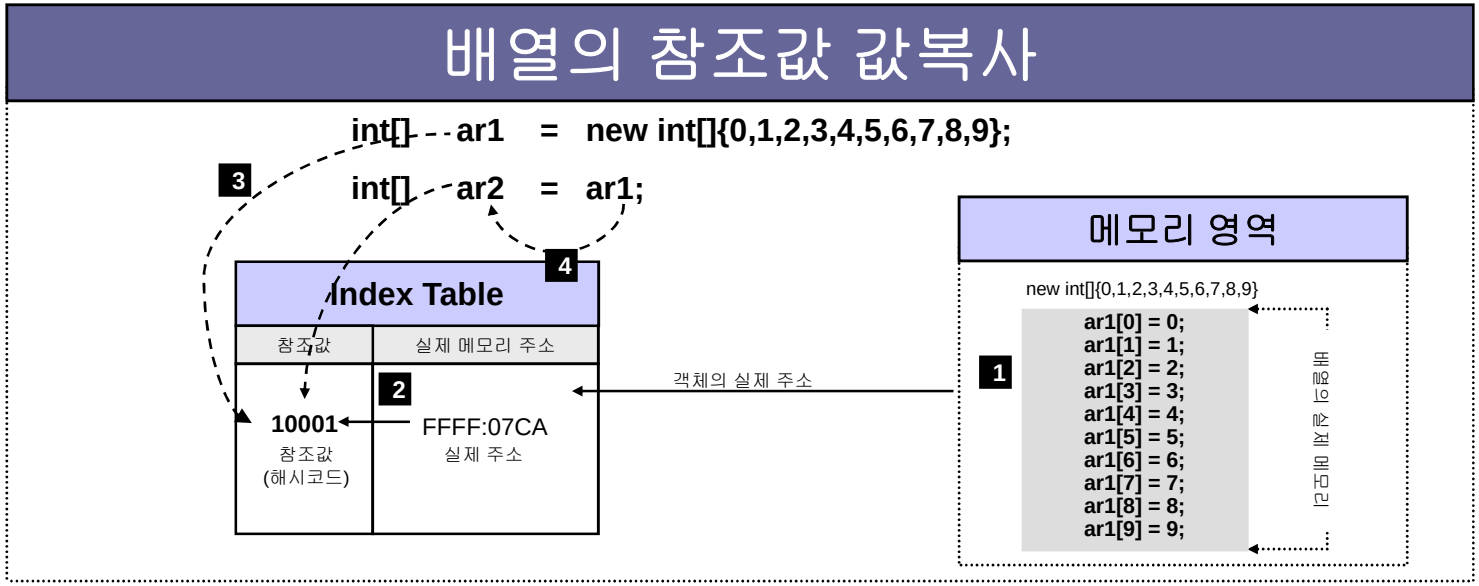
❖ 배열의 생성과 배열의 접근을 테스트하는 예제

```
public static void main(String[] args){
  // 배열의 선언과 초기화를 한번에
  int[] ap = new int[]{0, 1, 2, 3, 4, 5, 6, 7, 8, 9};
  int[] aw = {10, 11, 12, 13, 14, 15, 16, 17, 18, 19};

  //1. int[] ap 출력
  for(int i=0; i < ap.length; i++){
    System.out.print(ap[i] + "\t");
  }

  //2. int[] aw 출력
  for(int i=0; i<aw.length; i++){
    System.out.print(aw[i] + "\t");
  }
}
```

배열의 복사




```
부분 배열 복사를 위한 System.arraycopy()
int[] source  = new int[]{5, 4, 6, 9, 7, 9};
int[] target = {100, 200, 300, 400, 500,
600,700};

//부분 배열 복사의 예
System.arraycopy( source, 2, target, 3, 4 );

for(int i=0; i<target.length; i++) {
    System.out.println("target["+i+"]:" +
target[i]);
}
```

```
clone()을 이용한 메모리 차원의 배열복사
int[] source  = new int[]{5, 4, 6, 9, 7,
9};

//clone()을 이용한 메모리 복사
int[] target = (int[])source.clone();

for(int i=0; i < target.length; i++){
    System.out.println("target["+i+"] : "
+ target[i]);
}
```

객체 배열

❖ 객체 배열

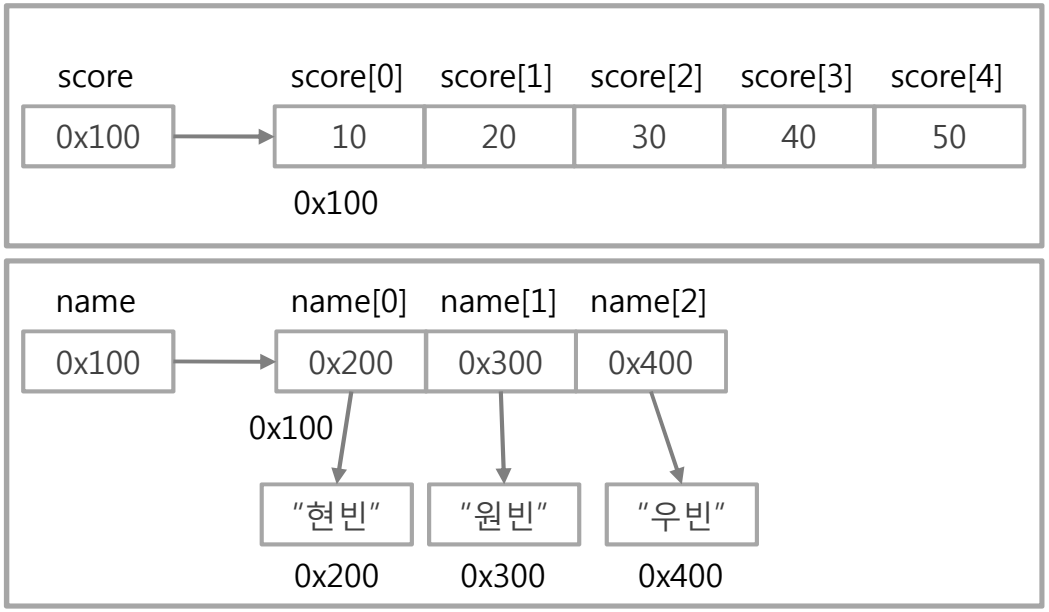
- 객체 배열을 선언했을 때 첨자의 수만큼 참조 변수가 만들어진다.
- 각각의 참조 변수에 대한 객체의 메모리는 생성되지 않는다.

❖ 기본 데이터 타입 배열 vs 객체 배열

- 기본 데이터 타입 배열은 배열 생성과 동시에 메모리가 생성된다.
- 객체 배열은 객체 변수(참조 변수)의 이름들만 생성되고, 객체들의 실제 메모리는 생성되지 않는다.

```
// 크기가 5인 int형 배열을 생성
int[] score = new int[5];
score[0] = 10;   // 각 요소에 직접 값을 저장
score[1] = 20;
score[2] = 30;
score[3] = 40;
score[4] = 50;
```

```
String[] name = new String[3];
name[0] = new String("현빈");
name[1] = new String("원빈");
name[2] = new String("우빈");
```



다차원 배열

❖ 다차원 배열(multi-dimensional array)

- 다차원 배열이란 2차원 이상의 배열을 의미하며, 배열 요소로 또 다른 배열을 가지는 배열을 의미한다.
- 즉, 2차원 배열은 배열 요소로 1차원 배열을 가지는 배열임며, 3차원 배열은 배열 요소로 2차원 배열을 가지는 배열이고, 4차원 배열은 배열 요소로 3차원 배열을 가지는 배열이다.

❖ 2차원 배열

- 2차원 배열이란 배열의 요소로 1차원 배열을 가지는 배열이다.

```
타입[][] 배열이름;
타입 배열이름[][];
타입[] 배열이름[];

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

다차원 배열

```
int[ ][ ] src = new int[ ][ ]{ {100,200,300}, {400,500,600} };
int[ ][ ] tar = { {701,702,703}, {704,705,706} };

//int[][] src의 정보출력
System.out.print("src.length:" + src.length + "₩t");
System.out.print("src[0].length:" + src[0].length + "₩t");
System.out.println("src[1].length:" + src[1].length);

//int[][] tar의 정보출력
System.out.print("tar.length:" + tar.length + "₩t");
System.out.print("tar[0].length:" + tar[0].length + "₩t");
System.out.println("tar[1].length:" + tar[1].length);

//2차원 배열의 출력
for(int i=0; i<src.length; i++){
  for(int j=0; j<src[i].length; j++){
     System.out.print("src[" + i + "][" + j + "]=" + src[i][j] + "₩t");
     System.out.print("tar[" + i + "][" + j + "]=" + tar[i][j] + "₩n");
  }
}
```

2차원 배열의 선언과 초기화를 동시에

1 선언과 동시에 초기화 (배열의 개수는 자동으로 [2][3]이 된다)

int[][] ar = new int[][]{{100, 200, 300}, {400,500,600}};

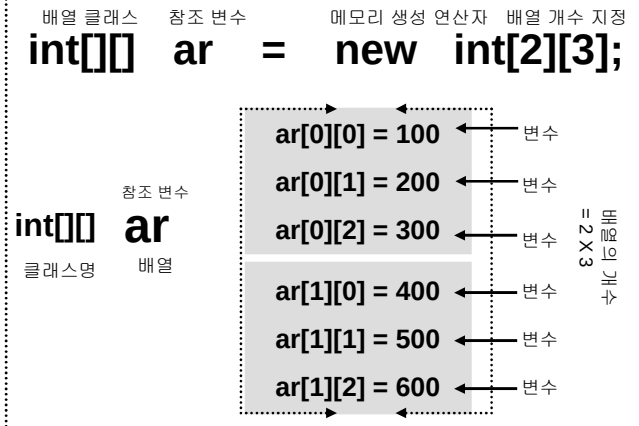
2

int[][] ar = {{100, 200, 300}, {400,500,600}};

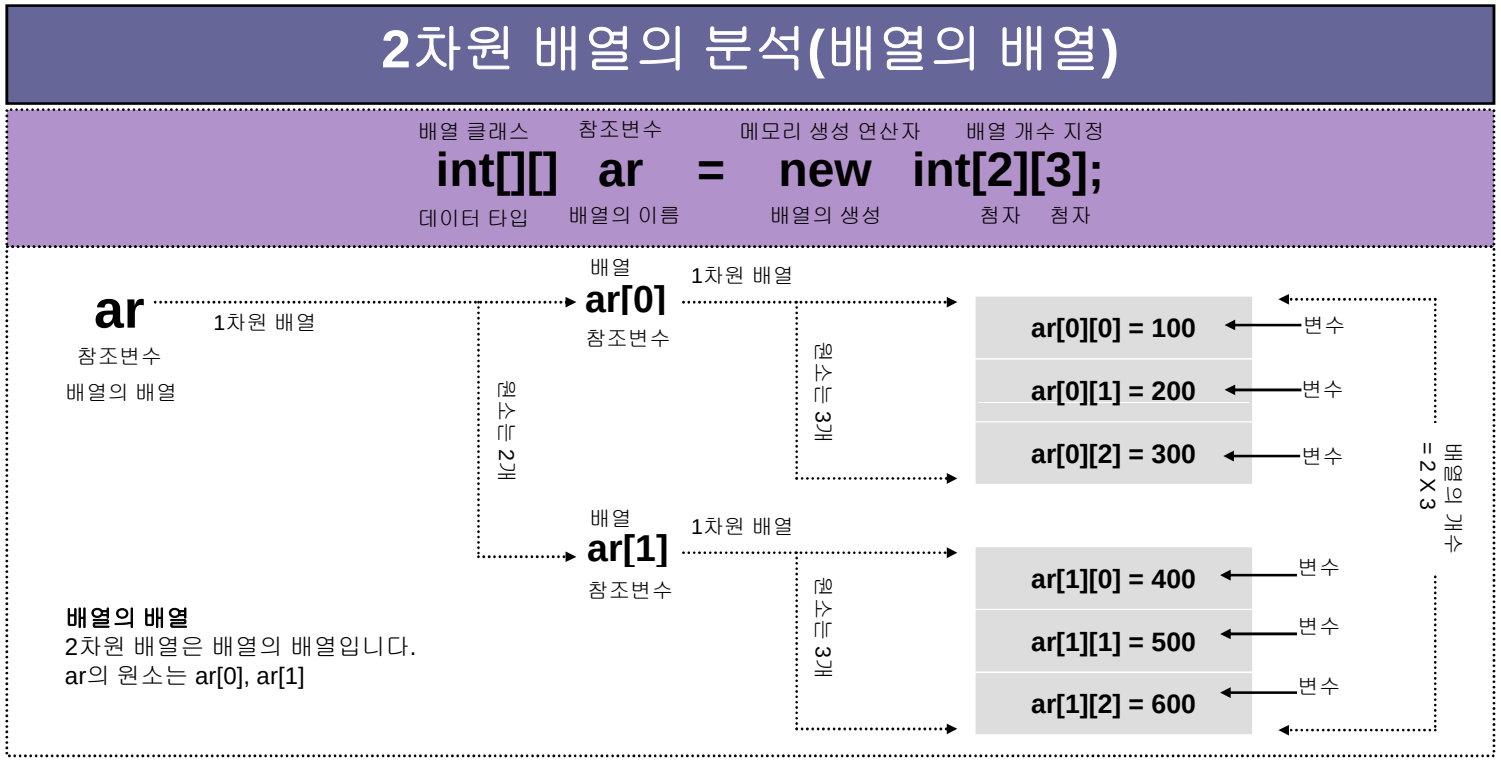
선언과 동시에 초기화 (배열의 개수는 자동으로 [2][3]이 된다)

이 방법들은 선언과 초기화를 동시에 하는 방법
입니다. 위의 방법 둘 다 사용할 수 있습니다.

2차원 배열의 선언과 할당의 분리



다차원 배열의 분석


2차원 배열의 분석(배열의 배열)
배열 클래스
참조변수
메모리 생성 연산자
배열 개수 지정
int[][] ar = new int[2][3];
데이터 타입
배열의 이름
배열의 생성
첨자
첨자
ar
참조변수
배열의 배열
1차원 배열
원소는 2개
배열
ar[0]
참조변수
1차원 배열
원소는 3개
ar[0][0] = 100
ar[0][1] = 200
ar[0][2] = 300
변수
변수
변수
배열
ar[1]
참조변수
1차원 배열
원소는 3개
ar[1][0] = 400
ar[1][1] = 500
ar[1][2] = 600
변수
변수
변수
배열의 개수
= 2 X 3
배열의 배열
2차원 배열은 배열의 배열입니다.
ar의 원소는 ar[0], ar[1]