

# 배열(Array)

배열 : 같은 타입의 여러 변수를 하나의 묶음으로 다루는 것을 배열이라고 한다.

많은 양의 데이터를 손쉽게 다룰 수 있다.

## 1. 배열의 선언

배열을 선언하는 방법은 원하는 타입의 변수를 선언하고 변수 or 타입에 배열임을 의미하는 []를 붙이면 된다.

하지만 가능하면 **타입 뒤에** 붙이도록 하자.

선언방법	선언 예
타입[] 변수이름;	<code>int[] socre;</code>
타입 변수이름[];	<code>int score[];</code>

## 2. 배열의 생성

배열을 선언 하는 것은 단지 생성된 배열을 다루기 위한 참조변수 공간이 만들어질 뿐이다.

배열을 생성해야만 비로소 데이터를 저장할 수 있는 공간이 만들어진다.

```
1 int[] score; //score 라는 변수를 **선언** 한다.
2 score = new int[5]; //배열을 **생성** 한다.
```

배열을 생성 하기 위해서는 **new**라는 연산자와 함께 배열의 타입과 크기를 지정해줘야 한다.

위 예제에서는 score 라는 배열의 타입을 int형 , 크기는 5로 지정한 것이다. 위 코드는 또한 아래와 같이 표현 가능하다.

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

## 배열의 생성과정

### 1. int[] score;

int형 배열 참조변수 score 를 선언한다. 데이터를 저장할 수 있는 공간은 아직 없다.

### 2. score = new int[5];

연산자 **new**에 의해서 메모리의 빈 공간에 5개의 int형 데이터를 저장할 수 있는 공간이 생겼다.

그리고 각 배열요소는 자동적으로 int의 기본값인 0으로 초기화 된다.

마지막으로 할당연산자 =에 의해 배열의 주소가 int형 배열 참조변수 score에 저장된다.

### 3. 배열의 초기화

배열은 기본적으로 자신의 타입에 맞게 초기화가 되지만, 원하는 값으로 초기화는 아래와 같이 한다.

```
1 int[] score = new int[5];
2 score[0]=100;
3 score[1]=90;
4 score[2]=80;
5 score[3]=70;
6 score[4]=60;
```

이렇게 하나하나 지정해주는 것은 좀 불편하니까 아래와 같이 초기화 할 수 있다.

```
1 int[] score = {100,90,80,70,60};
```

배열은 크기를 지정해주지 않으면, 주어진 초기화 값의 갯수에 따라 자동적으로 크기가 결정된다.

```
1 int[] score = new int[] {100,90,80};
```

### 4. 배열의 활용

배열의 각 저장공간에 값을 저장하고, 값을 읽어오기 위해서는 배열 참조변수와 인덱스를 이용한다.

인덱스 : 배열의 각 공간에 자동적으로 주어지는 일련번호.

- 0부터 시작해 1씩 증가하는 연속적인 정수값

```
1 score[3] = 100; // 0,1,2,3 이기 때문에 4번째 공간에 100이라는 값을 저장한다.
```

#### \* for문을 활용한 배열의 처리

배열을 다루는데 for문은 빼먹을 수 없다.

```
1 int[] score = {1,2,3,4,5}; //int 형 배열 score를 선언한다.(공간크기는 5)
2
3 for (int i = 0 ; i < 5 ; i ++){ //0부터 1씩증가(i++)해서 6보다 작을때까지 for문이
   도므로, 6번 돈다.
4     System.out.println(score[i]); //score[0]부터 score[4]까지 출력한다.
5 }
```

위와 같이 코드를 짜면 정상 작동하지만, 만약에 배열의 크기를 잘못 세었다고 치자.

```

1  int[] score = {1,2,3,4}; //int 형 배열 score를 선언한다.(공간크기는 4)
2
3  for (int i = 0 ; i < 5 ; i ++){ //0부터 1씩증가(i++)해서 5보다 작을때까지 for문이
   do므로, 5번 돈다.
4      System.out.println(score[i]); //score[0]부터 score[4]까지 출력한다.
5  }

```

이렇게 되면 배열의 공간은 4개, **score[3]**까지 밖에 없는데 **score[4]**의 값을 가져오려 하므로 에러가 난다.

\*참고: 이런 에러는 `ArrayIndexOutOfBoundsException`이라고 한다.

이렇듯, 배열의 크기를 직접 기입하게 되면 문제가 발생할 가능성이 있기때문에, 가급적이면 **length**라는 배열의 멤버 변수를 사용하는 것이 좋다.

```

1  int[] score = {1,2,3,4}; //int 형 배열 score를 선언한다.(공간크기는 4)
2
3  for (int i = 0 ; i<score.length ; i ++){ //score.length 멤버변수를 사용. 배열
   의 크기를 가져온다.
4      System.out.println(score[i]);
5  }

```