**[ 5 ] 배열** ; 데이터를 관리해 주는 중요한 요소

배열이란, 배열문법, 배열의 메모리 구조, 레퍼런스의 이해, 배열복사, 다차원배열

1. 배열이란? 동일 자료형의 집합. 하나의 이름으로 여러개의 데이터를 사용할 수 있음.

- 하나의 데이터만 가지고 있던 변수와 달리 여러 개의 데이터를 저장한 곳의 주소를 가리킴

- 즉, 여러 개의 변수를 모아 놓은 또 하나의 변수라고 생각하면 쉽다.

int kang, jung, kim, na, park, shin, yu, jeon, jo, lee;

int s1, s2, s3, s4, s5, s6, s7, s8, s9, s10;

→ int[] student = new int[10];

2. 배열의 문법(배열의 선언와 생성)

- 여러 개의 데이터가 모여 있어 ‘{ }’를 이용하여 초기화한다.

- 배열의 크기는 **최초에 한번 설정되면 변경이 불가** 하다.

- 배열을 객체로 취급.

- 배열선언 → 배열의 메모리 할당(배열 생성) → 배열이용

(cf)변수 선언 : int i = 10;

(cf)배열 선언① : // 배열의 크기(length)는 5 / 인덱스(순서)는 0~4

int[] iArr = {10, 20, 30, 40, 50}; // 선언과 초기화를 한번에

배열 선언② : int[] iArr = new int[5]; // 배열선언과 배열 생성(메모리 할당)

int iArr[] = new int[5];

iArr[0] = 10; // 각 배열방에 값 할당

iArr[1] = 20;

iArr[2] = 30;

iArr[3] = 40;

iArr[4] = 50;

배열 선언③ : int[] iArr; //배열 변수선언(생성된 배열을 다루는데 사용될 참조변수 선언)만

iArr = new int[5]; //배열을 생성한다(5개의 int값을 저장할 수 있는 공간생성)

iArr[0] = 10;

iArr[1] = 20;

iArr[2] = 30;

iArr[3] = 40;

iArr[4] = 50;

**public** **static** **void** main(String[] args){

**int** i = 10; // 변수선언, 초기화

**int**[] iArr = {10,20,30,40,50};//①배열선언,초기화

**for**(i=0 ; i<iArr.length ; i++)

System.***out***.println(iArr[i]);

**int**[] jArr = **new** **int**[5];//②배열선언과메모리할당

jArr[0] = 100;

jArr[1] = 200;

jArr[2] = 300;

jArr[3] = 400; //jArr[4]=X

**for**(i=0 ; i<jArr.length ; i++)

System.***out***.println(jArr[i]);

**int**[] score; //③변수선언

score = **new** **int**[3]; //메모리확보

score[0] = 1000; score[1]=2000; score[2]=11; //값 할당

//score[3] = 99; // runtime error

**for**(i=0 ; i<score.length ; i++)

System.***out***.println(score[i]);

}

* 확장 for문을 이용할 수도 있다

**public** **static** **void** main(String[] args) {

String[] names = {"홍길동","김길동","김제동"};

**for**(**int** idx = 0 ; idx<names.length ; idx++){

System.***out***.println("이름"+idx +":"+names[idx]);

}

// for(단위 변수 : 배열변수){ ~ }

**for**(String name : names){

System.***out***.println("이름:"+name);

}

}

3. 변수와 같이 이해하는 배열의 메모리 구조의 이해

① int i=10; 메모리에 i주머니가 만들어 지고, i 주머니 안에 10이라는 데이터가 들어있다.

② int[] iArr = {10,20,30}; 메모리에 iArr 주머니가 만들어지고 iArr주머니안에는 배열을 구성하고 있는 데이터의 주소값이 들어있다.

4. 레퍼런스의 이해

- 배열은 배열을 구성하고 있는 데이터들의 주소값을 가지고 있다고 하였다.

- 이때 주소값을 '레퍼런스'라고 한다.

- 만약 동일한 주소값을 가지고 있다면, 같은 데이터를 가리키고 있다고 볼 수 있다.

**public** **static** **void** main(String[] args) {

String name ="HongGilDong";

**int**[] s = {10,20,30,40,50};

**int**[] ss = s;

**for**(**int** i=0 ; i<s.length ; i++)

System.***out***.printf("s[%d]=%d\t ss[%d]=%d\n",i, s[i], i, ss[i]);

ss[0]=3; ss[1]=3;

**for**(**int** i=0 ; i<s.length ; i++)

System.***out***.println("s["+i+"]="+s[i]+"\t ss["+i+"]="+ss[i]);

}

5. 배열의 복사

①for문을 이용한 배열 복사

**public** **static** **void** main(String[] args) {

**int**[] num = {0,1,2,3,4};

**int**[] newNum = **new** **int**[5];

**for**(**int** i=0 ; i<newNum.length ; i++){

// num배열을 newNum으로 copy

newNum[i]=num[i];

}

**for**(**int** i=0 ; i<num.length ; i++){

System.***out***.printf("num[%d]=%d\t",i,num[i]);

System.***out***.printf("newNum[%d]=%d\n",i,newNum[i]);

}//for

}//main

②System.arraycopy(원본배열객체, int 원본시작위치, 복사본배열객체, int 복사본시작위치, int 복사길이);

System.arraycopy(num, 0, newNum, 0, num.length);

**public** **static** **void** main(String[] args) {

**int**[] num={0,1,2,3,4};

**int**[] newNum = **new** **int**[5];

**for**(**int** i=0 ; i<num.length ; i++){

System.***out***.printf("num[%d]=%d\t",i,num[i]);

System.***out***.printf("newNum[%d]=%d\n",i,newNum[i]);

}

System.*arraycopy*(num, 0, newNum, 0, num.length);

**for**(**int** i=0 ; i<num.length ; i++){

System.***out***.printf("num[%d]=%d\t",i,num[i]);

System.***out***.printf("newNum[%d]=%d\n",i,newNum[i]);

}

}

(실습예제)

1. -1 배열을 이용하여 학생들(길동, 현식, 땅땅, 슘당, 대갈)의 키를 입력받고, 평균치를 구해보자..

**public** **static** **void** main(String[] args) {

String[] arrName={"영희","철수","길동","영수","말자"};

**int**[] arrHeight = **new** **int**[5];

**int** totalHeight =0;

Scanner scanner = **new** Scanner(System.***in***);

**for**(**int** i=0 ; i<arrHeight.length ; i++){

System.***out***.print(arrName[i]+"의 키는? ");

arrHeight[i]=scanner.nextInt();

totalHeight += arrHeight[i];

}

System.***out***.print("평균키 : ");

System.***out***.println(totalHeight/arrName.length);

scanner.close();

}

①-2 위의 예제에서 가장 큰 학생과 가장 작은 학생을 구분해 보자

**public** **static** **void** main(String[] args) {

String[] arrName={"영희","철수","길동","몽룡","춘향"};

**int** [] arrHeight = **new** **int**[arrName.length];

**int** totalHeight =0;

Scanner scanner = **new** Scanner(System.***in***);

**for** (**int** i=0 ; i<arrName.length ; i++) {

System.***out***.print(arrName[i] +"의 키는 ?");

arrHeight[i]= scanner.nextInt();

totalHeight += arrHeight[i];

}

System.***out***.print("평균키 : ");

System.***out***.println(totalHeight/arrName.length);

scanner.close();

**int** maxIdx=0, maxHeight=0; // 최장신을 구하기 위한 변수

**int** minIdx=0, minHeight=999; // 최단신을 구하기 위한 변수

**for** (**int** i=0 ; i<arrHeight.length ; i++) {

**if**(arrHeight[i]>maxHeight) {

maxHeight = arrHeight[i];

maxIdx = i;

}//if : 가장 큰 학생의 키와 index를 구함

// if(가장 작은 학생의 키와 index를 구하는 if)문을 완성시키시오

}//for

System.***out***.printf("가장 작은 학생은 : %s, 키 : %d\n", arrName[minIdx], minHeight);

System.***out***.printf("가장 큰 학생은 : %s, 키 : %d\n", arrName[maxIdx], maxHeight);

}

※ 쉬어가는 코너 : main(String[] args)의 의미를 알아보자

**package** com.lec.ex;

**public** **class** Hello {

**public** **static** **void** main(String[] args) {

System.***out***.print("Hello");

**for**(**int** i=0 ; i<args.length ; i++) {

System.***out***.print(", "+args[i]);

}

}//main

}

탐색기의 d:\ai\source\1\_javaSrc\ch05\bin 의 주소창에 cmd를 실행하고

콘솔창이 뜨면 java com.lec.ex.Hello Tom실행하면

1

Hello, Tom이 출력

Quiz1) 76,45,34,89,50 5개의 값을 배열로 초기화 하고 이들 값들을 크기 순으로 나타내는 프로그램을 작성 하시요.

**for**(**int** i=0 ; i<arr.length-1 ; i++) {

**for**(**int** j=i+1 ; j<arr.length ; j++) {

**if**(arr[i] > arr[j]) {

**int** temp = arr[i];

arr[i] = arr[j];

arr[j] = temp;

}

}//for-j

}

Quiz2) 배열에 담긴 값의 합계, 평균, 최대값, 최소값을 구하는 프로그램을 작성하시오.

int[] arr = { 10, 20, 30, 40, 50, 12, 13}

프로그램 결과 :

합 계 = 175

평 균 = 25.0

최대값 = 50

최소값 = 10