

2023.4.19

VI.VII.

<<요약>>

=====

I . array

배열 : 같은 자료형으로 연속된 공간을 만드는 것

- 자료형 배열명[] & 자료형 [] 배열명 =new 자료형 [];
- index는 0부터 시작한다. (index = 배열의 위치)
- 참조형 변수
- length : 길이

```

12      - length : 길이
13      */
14
15
16      int [] arr = new int [3];
17      arr[0] = 1111;
18      arr[1] = 2222;
19      arr[2] = 3333;
20      System.out.println(arr.length); //length - 해당길이
21      System.out.println(arr); //장표간 출력
22
23
24      for(int i=0; i <3; i++) {
25          System.out.println(arr[i]); //arr의 0,1,2 의 값을 출력한다. [i=3]
26      }
27
28      for(int i=0; i <arr.length; i++) {
29          System.out.println(arr[i]);
30      }
31
32      //배열 초기화
33      int [] arr2 = {111,222,333,444,555};
34      System.out.println("arr2 length : " + arr2.length);
35
36      for (int j=0; j<arr2.length; j++) {
37          System.out.println(arr2[j]);
38      }
39
40      int[] arr3 = new int [] {1,2,3,4,5};
41      for (int i=0; i<arr3.length; i++) {
42          System.out.println(arr3[i]);
43      }
44
45      } //method end
46
47  }
48

```

II .for-each

: 배열의 값을 변수에 담아서 중괄호 구간 안으로 전달해준다. 반복문의 종료조건이나 종료조건을 위해서 기준 값을 증가시키는 등의 반복적인 작업을 내부적으로 감춘 것.

```
*for (자료형 변수명 : 배열){
    종속문장
}
```

	for	for each
사용법	<pre>for(초기 값; 종료되는 조건 ; 값의 증감) { 반복 수행할 작업 }</pre>	<pre>for(각 요소 값 : 배열이나 컨테이너 값) { 반복 수행할 작업 }</pre>
예제	<pre>String[] name = {"KIM","JACK","LEE","DACUYA"}; for(int i = 0; i < name.length ; i++) { System.out.println("name is "+name[i]); }</pre>	<pre>String[] name = {"KIM","JACK","LEE","DACUYA"}; for(String Nm : name) { System.out.println("name is "+Nm); }</pre>

III. 문자열 비교

❖ 배열 5개를 만들고 로그인 프로그램을 만드시오.

```
String[] id = new String[5];
```

```
String[] pwd = new String[5];
```

1.로그인시

- 아이디가 없으면 존재하지 않는 아이디 입니다
- 비밀번호가 틀리면 비밀번호가 틀렸습니다
- 아이디와 비밀번호가 일치하면 인증통과

2.회원가입시

- 동일한 아이디가 있으면 동일한 아이디가 존재합니다
- 5개의 배열 모두 사용됐으면 더 이상 저장할 공간이 없습니다
- 회원가입 성공시 가입을 축하합니다

1.로그인

2.회원가입

3.나가기

>>>

<https://limdotori.tistory.com/40>