**객체지향 프로그래밍** | 최윤정 교수님 | Summary 이서현 2276219  
[Chapter 03] week 3: 자바 기본 프로그래밍 (1) 반복문 (2) 배열 (~43p)

(1) 반복문

자바 반복문의 종류: for문, while문, do while문

for문: for(int i=0, i<10, i++){} //c++와 비슷. for문 안에 선언한 변수는 for문 종료시 사라짐

* for문의 특이한 형태: 반복 조건이 true거나/비어 있으면(true로 간주) 무한반복

while문: while(){} //반복 조건이 true이면 반복, false이면 반복 종료. 무한루프: while(true)

* while에서 for문처럼 i 증가시켜 사용시 유의 🡪 while(i<10){ i++; // i 증가 필요 }

do-while문: do{}while(); //일단 실행한 후 조건에 맞는지 검사.

\*중첩 반복: 반복문이 다른 반복문을 내포하는 구조. (for 안에 for… 과다한 중첩 반복은 X!! 2중 또는 3중 반복이 적당)

**continue**: 현재 반복문을 빠져나가지 않으면서 다음 반복으로 진행. (조건식으로 분기) 아래를 무시하고 처음으로 돌아감

**break**: 현재 반복문을 완전히 빠져나감, 한 번에 하나의 반복문만 벗어남. (exit 문장은 프로그램 종료)

(2) 배열

배열(array)

* 인덱스와 인덱스에 대응하는 데이터들로 이루어진 자료 구조. (배열을 이용하면 한 번에 많은 메모리 공간 할당 가능.)
* 같은 타입의 데이터들이 순차적으로 저장. (인덱스를 이용하여 원소 데이터 접근. 반복문 이용해 처리하기에 적합)
* 배열 인덱스: 0부터 시작. (인덱스는 배열의 시작 위치에서부터 데이터가 있는 상대 위치)

1차원 배열 만들기: 배열 선언과 배열 생성의 두 단계 필요

* 선언과 생성 동시에: int [] array = new int[10];
* 선언과 함께 초기화: int [] array = {0, 1, 2};
* <주의> 잘못된 배열 선언 예) int array[3]; //컴파일 오류. 선언 시 배열의 크기를 지정하면 안 된다.

레퍼런스 변수와 배열

1. 배열에 대한 레퍼런스 변수 array 선언: int배열타입 []배열 선언 array배열에 대한 레퍼런스 변수;
2. 배열 생성: array배열에 대한 레퍼런스 변수 = new배열생성 int타입 [5]원소 개수;

배열 인덱스와 원소 접근

* 배열 원소 접근: 배열 변수명과 []사이에 원소의 인덱스를 적어 접근
* 배열 인덱스의 범위: 0에서 시작 ~ (배열 크기 -1)까지
* <주의> 반드시 배열 생성후 접근하기

레퍼런스 치환과 배열 공유: 하나의 배열을 다수의 레퍼런스가 참조 가능

배열의 크기, length 필드

배열은 자바에서 객체로 관리함. 배열 객체 내에 length 필드는 배열의 크기를 나타냄.

int size = array.length; //괄호 없음 주의

배열과 for-each 문: for (배열의 원소타입 변수 : 배열)

* 배열이나 나열의 각 원소를 순차적으로 접근하는데 유용한 for 문

int[] num = {1, 2, 3, 4, 5};

for (int k : num) //반복될 때마다 k는 num[0], num[1], …, num[size-1] 값으로 설정  
 sum += k;

* int[] num 뿐 아니라 String[] names, enum Week{}의 예시

enum Week{월, 화, 수, 목, 금, 토, 일}

for (Week day : Week.values()) //반복될 때마다 day는 월, 화, …, 일로 설정

System.out.print(day+”요일 ”);