



Day 5

❖ 배열

같은 자료형들끼리 모아두는 하나의 묶음

담을수 있는 자료의 타입과 길이가 먼저 정해진다.

·선언 형식1

데이터타입[] 변수이름;

·선언 형식2

데이터타입 변수이름[];

❖ 배열

· 배열 생성 형식1 : 배열 객체만 생성하는 방법

```
데이터타입 변수이름[] = new 데이터타입[길이];
```

📌 예제] 정수 자료 6개를 넣을 배열을 만드세요.

```
int[] noArray = new int[6];
```

· 배열 생성 형식2 : 데이터를 생성과 동시에 입력하는 방법

```
데이터타입 변수이름[] = {데이터1, 데이터2, 데이터3, ...};
```

📌 예제] 정수 자료 6개(1, 2, 3, 4, 5, 6)를 넣을 배열을 만드세요.


```
int[] noArray = {1, 2, 3, 4, 5, 6};
```

❖ 향상된 for 명령

시퀀스 자료형(인덱스가 있는 자료형)에서 유용하게 쓸 수 있는 반복 명령
인덱스 순서로 하나씩 꺼내서 변수에 대입한다.

· 형식 : 매개변수에 담겨있는 데이터 타입과 동일한 타입의 변수를 만들어준다.

```
for( 데이터타입 변수이름 : 시퀀스자료 변수이름 ){  
    실행문;  
}
```

 예제] 정수 자료 6개(1, 2, 3, 4, 5, 6)를 넣은 배열의 데이터를 순서대로 출력


```
int[] noArray = {1, 2, 3, 4, 5, 6};  
int cnt = 0;  
for(int num : noArray){  
    System.out.println("index : " + cnt++ + " / 데이터 : " + num);  
}
```

❖ while 명령

for 반복문과 비교해서 카운터변수를 따로 만들어줘야한다는 차이점이 있다.
조건이 참이면 반복하는 명령
카운터변수의 생성과 처리는 따로 해줘야 한다.

· 형식

```
카운터변수 선언;  
while( 반복조건 ){  
    실행문;  
    카운터변수 증감식;  
}
```

 예제] 정수 자료 6개(1, 2, 3, 4, 5, 6)를 넣은 배열의 데이터를 순서대로 출력


```
int[] noArray = {1, 2, 3, 4, 5, 6};  
int cnt = 0;  
while( cnt < noArray.length ){  
    System.out.println("index : " + cnt++ + " / 데이터 : " + num);  
}
```

❖ do - while 명령

다른 반복문과 비교해서 최소 한번은 반드시 실행한다는 차이점이 있다.
while 문과 비교해 조건이 맨 마지막에 온다는 점이 다르다.

· 형식

```
카운터변수 선언;  
do {  
    실행문;  
    카운터변수 증감식;  
} while ( 반복조건 );
```

 예제] 정수 자료 6개(1, 2, 3, 4, 5, 6)를 넣은 배열의 데이터를 순서대로 출력

```
int[] noArray = {1, 2, 3, 4, 5, 6};  
int cnt = 0;  
do{  
    System.out.println("index : " + cnt + " / 데이터 : " + num);  
} while( cnt++ < noArray.length )
```

❖ break 명령

반복문과 switch문 또는 레이블이 붙은 반복문을 종료
가장 가까운 반복문을 종료

❖ continue 명령

반복문을 다음 회차로 다시 반복
가장 가까운 반복문을 다시 실행
continue 다음에 레이블이 붙을 경우 해당 레이블이 붙은 반복문의 다음회차로 진행

· 형식

```
반복문( 반복조건 ){  
    if( 조건식 ) {  
        break( 또는 continue ) [레이블];  
    }  
    실행문;  
}
```

 **예제]** 정수를 입력받아 그 숫자가 짝수인지 홀수인지 출력하세요.
7번 반복하고 중간에 3이 나오면 종료하세요.

```
import java.util.*;
... 생략 ...

Scanner sc = new Scanner(System.in);
int no = 0;
int cnt = 0;

while(cnt++ < 7){
    no = sc.nextInt();
    if ( no == 3 ){
        break;
    } else {
        System.out.println(
            cnt + " 번째 입력값 : " + no + " / " + (no % 2 == 0 ? "짝수" : "홀수");
        continue;
    }
}
```