

기본 입출력

입출력이란 **입력** + **출력**의 줄임말. **Input** + **Output** 이라는 의미로 **IO** 라고 부르기도 한다.

#01. 출력

1) 괄호안의 내용을 출력하고 줄바꿈 수행하기

`println()`의 **ln**은 **line**의 줄임말

```
System.out.println("j");
System.out.println("a");
System.out.println("v");
System.out.println("a");
```

- 출력결과

```
j
a
v
a
```

2) 아무런 내용도 출력하지 않고 줄바꿈만 수행

`println()`에게 출력할 내용을 지정하지 않는다.

```
System.out.println();
```

3) 괄호안의 내용을 출력하고 줄바꿈 안하기

`println()`이 아닌 `print()` 명령어 사용

```
System.out.print("j");
System.out.print("a");
System.out.print("v");
System.out.print("a");
```

- 출력결과

```
java
```

#02. 입력

키보드를 통해 사용자에게 어떤 내용을 입력받아서 변수에 저장 가능함.

이 기능은 명령프롬프트를 통해 실행한 경우에만 사용 가능하다.

1) 구현절차

클래스 정의 상단에 다음의 구문을 추가함

```
import java.util.Scanner;
```

메인 메서드 첫 라인에서 다음의 구문을 추가함

```
Scanner reader = new Scanner(System.in);
```

숫자값을 입력받고자 하는 경우

```
int input = reader.nextInt();
```

문자열을 입력받고자 하는 경우

```
String input = reader.nextLine();
```

사용 완료 후 입력기능 닫기

```
reader.close();
```

💡 ****주의!**** 문자열과 숫자를 하나의 프로그램에서 입력 받아야 하는 경우 ****문자 > 숫자****의 순서로 입력 받아야 합니다.

#03. 이스케이프 문자

출력시에 문자열에 포함시킬 수 없는 줄바꿈, 탭키 등을 처리하기 위한 특수문자.

글자 앞에 ****역슬래시(\)****를 적용하여 두 글자가 모여 한 글자로 인식된다.

이스케이프 문자	의미
----------	----

<code>\n</code>	줄바꿈
-----------------	-----

이스케이프 문자	의미
\t	탭키
\"	쌍따옴표
\'	홀따옴표
\\	역슬래시 자체

#04. 형식문자

문자열을 출력할 때 변수값이 적용될 수 있는 치환자.

1) 종류

형식문자	데이터 타입	의미
%s	문자열	문자열을 지정
%d	정수	10진 정수로 표시
%f	실수	실수형태의 10진수로 표시 무조건 소수점 6자리까지 표시. 그 이상 되는 자리수는 반올림
%g	실수	실수형태의 소수점 10진수로 표시 소수점 4자리까지 표시. 그 이상 되는 자리는 반올림 처리
%%	퍼센트 퍼센트를 표시	

2) 형식문자를 포함하는 문자열 직접 출력하기

구문형식

```
System.out.printf(문자열, 변수1, 변수2 ... 변수3)
```

- 첫 번째 조건값으로 주어지는 문자열에는 형식문자가 포함되어야 한다.
- 두 번째 이후의 조건값들은 첫 번째 조건값인 문자열에 포함된 형식 문자들을 치환할 변수들이 순서대로 나열된다.
- 출력 후 자동 줄바꿈이 되지 않으므로 줄바꿈을 사용하기 위해서는 \n 을 활용하여 직접 처리해야 한다.

3) 형식문자를 포함하는 문자열을 통한 새로운 문자열 만들기

구문형식

```
String 변수이름 = String.format(문자열, 변수1, 변수2 ... 변수3)
```

- 첫 번째 조건값으로 주어지는 문자열에는 형식문자가 포함되어야 한다.
- 두 번째 이후의 조건값들은 첫 번째 조건값인 문자열에 포함된 형식 문자들을 치환할 변수들이 순서대로 나열된다.

4) 자릿수 지정

%다음에 자릿수를 의미하는 숫자를 명시할 수 있다.

출력 되는 내용은 오른쪽에 정렬하고 남는 자리는 공백으로 채워진다.

만약 자릿수가 부족할 경우 원본 값이 그대로 출력 된다.

문자열의 자릿수 지정

- %숫자s 형식으로 명시한다.
- 치환할 문자열의 길이가 주어진 자릿수보다 짧을 경우 왼쪽부터 공백이 추가된다.
- 치환할 문자열의 길이가 주어진 자릿수보다 긴 경우 문자열이 그대로 출력된다.

정수의 자리수 지정

- %숫자d 형식으로 명시한다.
- 치환할 값의 길이가 주어진 자리수보다 짧을 경우 왼쪽부터 공백이 추가된다.
- 치환할 값의 길이가 주어진 자리수보다 긴 경우 문자열이 그대로 출력된다.

실수의 자리수 지정

- 소수점을 포함하여 10자리를 만들기 위해 뒤에 0을 추가한다.
- 소수점도 자릿수에 포함됨에 주의한다.

빈 자리를 0으로 채우기

정수의 경우

- 빈 자리를 0으로 채우도록 지정할 수 있다.
- % 다음에 010 과 같은 형식으로 0을 먼저 명시하고 자리수를 지정한다.

실수의 경우

- 빈 자리를 0으로 채우도록 지정할 경우 소수점 뒤에 적용된다.
- 점(.)을 사용하면 소수점의 기본 자리수를 변경할 수 있다.