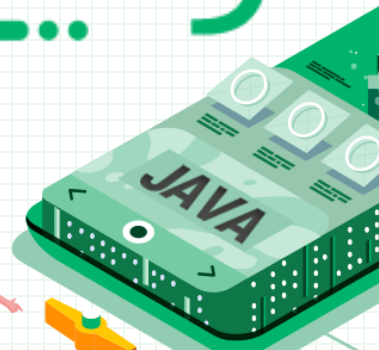




# 안드로이드 프로그래밍을 위한 자바기초 \_...



## ② 자료형과 변수의 기본 문법 이해하기



## 학습목표

- 자료형과 상수를 이해하고 프로그래밍에 적용할 수 있다.
- 변수의 개념과 사용법을 이해하고 프로그래밍 적용할 수 있다.
- 표준 입출력을 이용하여 프로그래밍을 작성할 수 있다.



## 학습내용

- 자료형과 상수 이해하기
- 변수 이해하기
- 표준 입출력 이해하기

## ② 자료형과 변수의 기본 문법 이해하기

### 자료형과 상수 이해하기

#### ▶ 자료형(Data Type) 이해하기

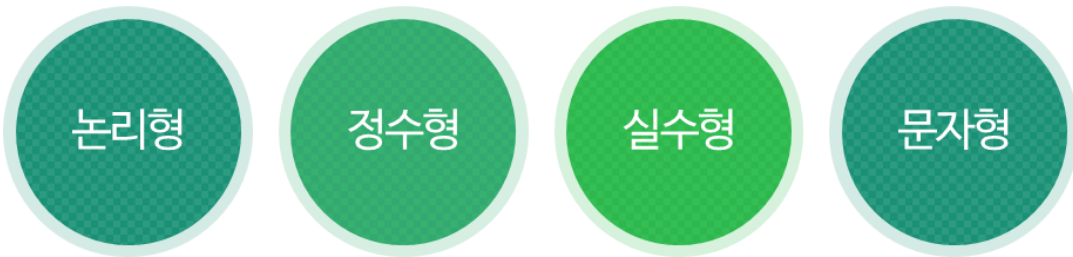
1) 데이터의 종류에 따라 컴퓨터 메모리의 크기를 정해놓은 것

- 데이터의 종류
  - ✓ 정수 - 123,567
  - ✓ 실수 - 3.14
  - ✓ 문자 - 'a'
  - ✓ 문자열 - "programming"
- 데이터 종류에 따라 메모리의 크기가 달라짐

2) 종류

- 기본 자료형
- 참조형 자료형

3) 기본 자료형



(1) 논리형

기본 표현	자료 크기	데이터 표현 범위
boolean	1byte	<ul style="list-style-type: none"> <li>• 참 : true</li> <li>• 거짓 : false</li> </ul>

## ② 자료형과 변수의 기본 문법 이해하기

### 자료형과 상수 이해하기

#### (2) 정수형

기본 표현	자료 크기	데이터 표현 범위
byte	1byte	$-2^7 \sim 2^7 - 1$ (-128 ~ 127)
short	2byte	$-2^{15} \sim 2^{15} - 1$ (-32,768 ~ 32,767)
int	4byte	$-2^{31} \sim 2^{31} - 1$
long	8byte	$-2^{63} \sim 2^{63} - 1$

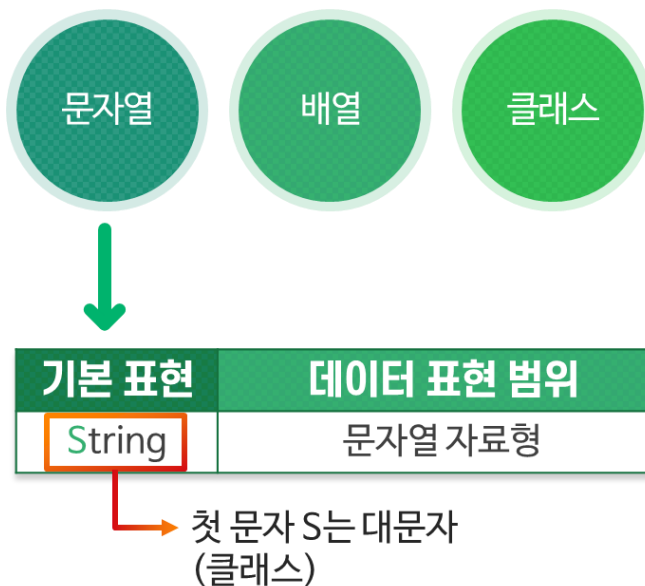
#### (3) 실수형

기본 표현	자료 크기	데이터 표현 범위
float	4byte	$\pm 1.4 \times 10^{-45} \sim \pm 3.4 \times 10^{+38}$
double	8byte	$\pm -4.9 \times 10^{-324} \sim \pm 1.7 \times 10^{+308}$

#### (4) 문자형

기본 표현	자료 크기	데이터 표현 범위
char	2byte	0 ~ 65,535

#### 4) 참조형 자료형



## ② 자료형과 변수의 기본 문법 이해하기

### 자료형과 상수 이해하기

#### ➤ 상수(constant) 이해하기

##### 1) 상수란?

- 프로그램 수행 중 값이 변하지 않는 데이터
- 종류
  - ✓ 논리
  - ✓ 정수
  - ✓ 실수
  - ✓ 문자
  - ✓ 문자열

##### 2) 논리 상수값

- 참 : true, 거짓 : false

자료형	상수 표현
boolean	true, false

##### 2) 정수 상수값

- (1) 자료형이 표현할 수 있는 값의 범위 내에서 사용하여야 함
- (2) 접미사 'L' 또는 'l' (일반적으로 대문자 사용)
  - ✓ 'L'을 사용하면 long형 상수값이 됨
  - ✓ 'L'을 사용하지 않으면 int형 상수값이 됨

##### (3) 진수 표현

10진수	123
16진수	0x123
8진수	0123

##### 정수상수의표현

자료형	상수 표현
byte	123, 0x7B, 0173
short	123, 0x7B, 0173
int	123, 0x7B, 0173
long	123L, 0x7BL, 0173L

## ② 자료형과 변수의 기본 문법 이해하기

## 자료형과 상수 이해하기

## 4) 실수 상수값

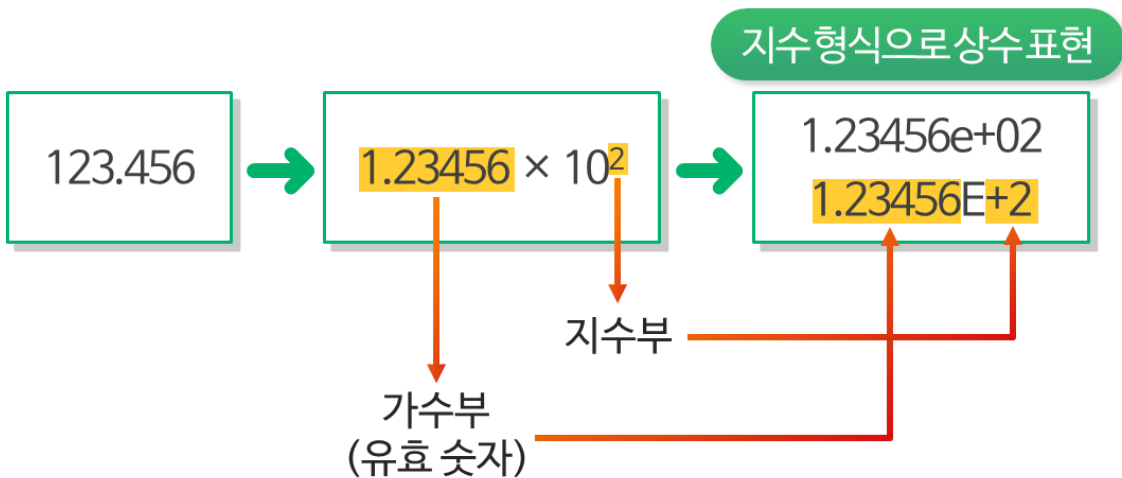
- (1) 자료형이 표현할 수 있는 값의 범위 내에서 사용하여야 함
- (2) 접미사 'f' 또는 'F' (일반적으로 소문자 사용)
  - ✓ 'f'를 사용하면 float형 상수값이 됨
  - ✓ 'f'를 사용하지 않으면 double형 상수값이 됨

## 실수상수의표현

자료형	상수 표현
float	123.45f
double	123.45

## (3) 지수 형식 표기 방법

- ✓ 알파벳 소문자 e 또는 대문자 E를 이용
- ✓ 가수부와 지수부를 분리해서 표현



## ② 자료형과 변수의 기본 문법 이해하기

## 자료형과 상수 이해하기

## 5) 문자/ 문자열 상수값

- (1) 문자 상수 - 작은 따옴표 사용
- (2) 문자열 상수 - 큰 따옴표 사용

## 문자/문자열상수의표현

구분	자료형	상수 표현
문자	char	‘자’, ‘a’, ‘*’
문자열	String	“자바 프로그래밍”

## ② 자료형과 변수의 기본 문법 이해하기

## 변수 이해하기

## ▶ 변수(Variables) 이해하기

## 1) 식별자란?

## ▪ 식별자 (identifier)

- ✓ 프로그래머가 필요시 프로그램 내에 정의 또는 선언하여 사용하는 이름
- ✓ 예) 변수 이름, 메소드 이름 등

## (1) 식별자 만드는 규칙

- ✓ 구성
  - 알파벳 대소문자, 한글, 숫자, \_(언더바), \$(달러 표시)를 사용, 이외는 사용불가(공백 포함)
- ✓ 식별자 첫 문자
  - 알파벳 대소문자, \_(언더바), 한글로 시작하여야 함, 숫자는 사용불가
- ✓ 예약어는 사용 불가
- ✓ 알파벳 대소문자는 구분함

## (2) 식별자 예제

## 올바른 식별자

```
start //소문자알파벳으로시작
Total //대문자알파벳으로시작
_sum //_(언더바)로시작
합계 //한글로시작
```

## 잘못된 식별자

```
9start //숫자로시작
Countgood //중간에공백
int //예약어사용
Loop^total //특수문자사용
```



## ② 자료형과 변수의 기본 문법 이해하기

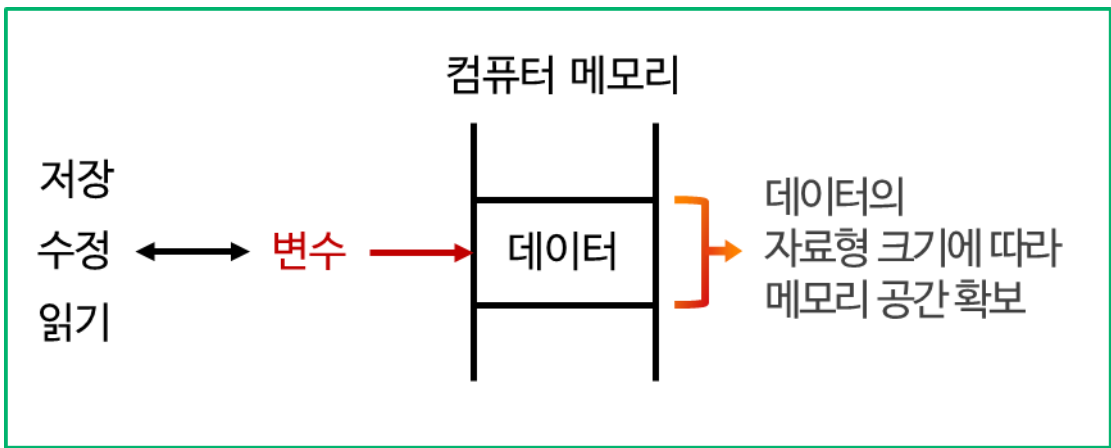
## 변수 이해하기

## 2) 예약어란?

- 예약어(Reserved Word)
  - ✓ 자바 언어에서 사용하는 문법적 단어들
  - ✓ 식별자로 사용할 수 없음
- 예약어 (예)
  - ✓ byte, short, int, long, float, double, char, String, if, switch, else, case for, while, do

## 3) 변수(Variables)란?

- (1) 데이터를 저장하는 컴퓨터 메모리 공간을 가리키는 식별자
- (2) 변수에 데이터를 저장, 수정, 읽기 가능
- (3) 프로그램이 실행되는 동안 변수의 데이터 값이 유효함
- (4) 데이터의 자료형 크기에 따라 컴퓨터 메모리 공간이 확보됨

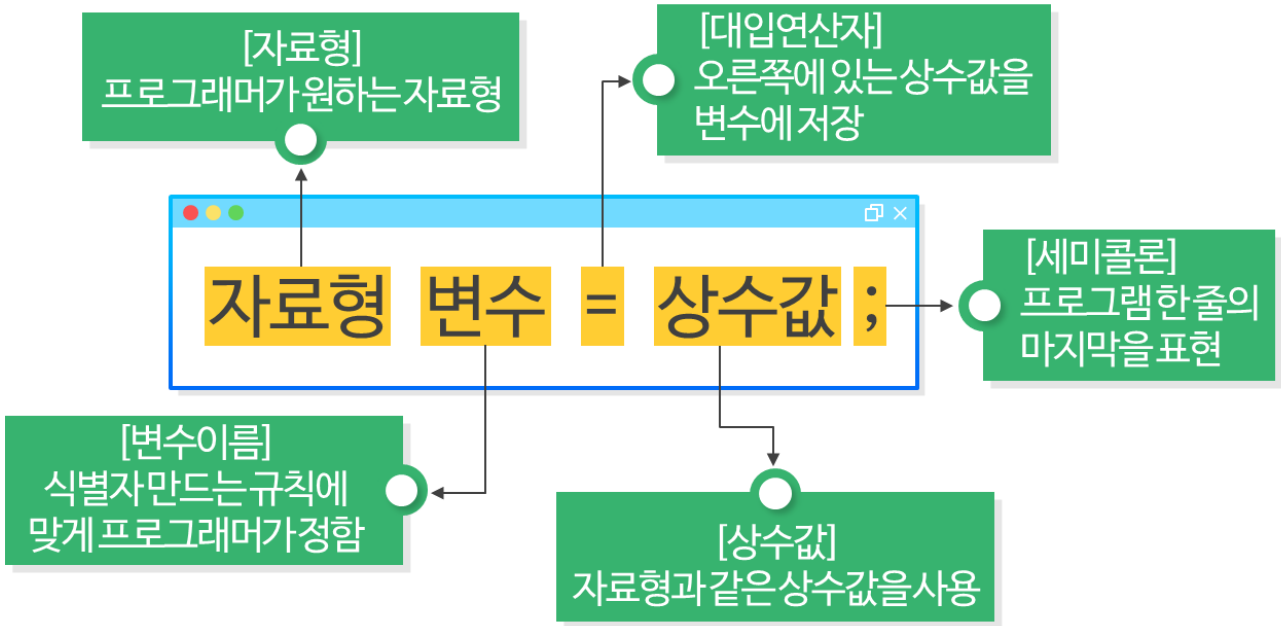


## ② 자료형과 변수의 기본 문법 이해하기

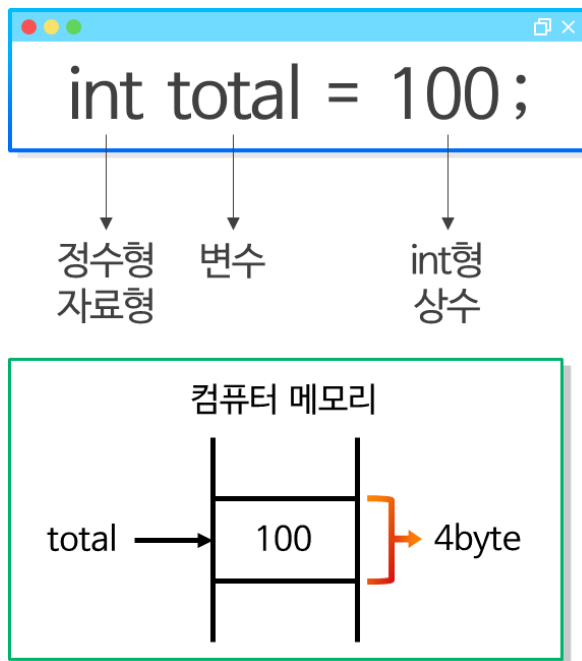
### 변수 이해하기

#### ▶ 변수 사용하기

##### 1) 변수 선언 방법



##### 2) 변수 선언 예제

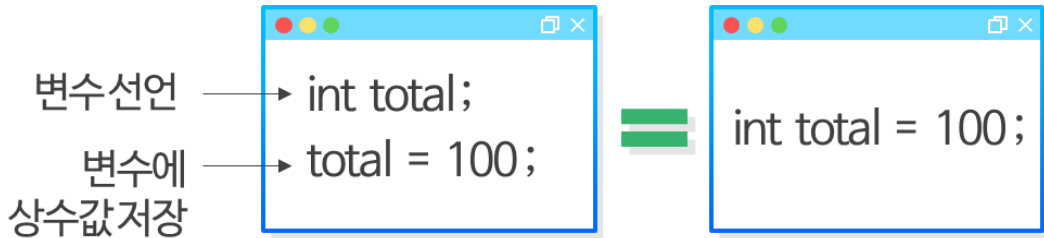


## ② 자료형과 변수의 기본 문법 이해하기

## 변수 이해하기

## 3) 변수 선언하는 부분과 상수값을 저장하는 부분을 관리

- ✓ 변수 선언하는 부분과 상수값을 저장하는 부분을 분리



## 4) 변수 이름 표기법

## (1) 가독성 있게 표현

- ✓ 의미 없는 변수 이름은 사용하지 않음(abc, a1, b2) 등
- ✓ 변수 사용 목적에 맞게 표기

## (2) 표기법

- ✓ 카멜 표기법
  - 2개의 단어를 연결하여 사용
  - 첫 번째 단어는 소문자로, 두번째 단어 대문자로 시작(scoreCount, myTotal 등)
- ✓ 스네이크 표기법(Snake Case)
  - 2개의 단어를 연결하여 사용
  - 첫 번째 단어와 두 번째 단어를 \_(언더바)를 이용하여 연결(score\_count, my\_total 등)

## ② 자료형과 변수의 기본 문법 이해하기

## 변수 이해하기

## 5) 변수 선언 위치와 사용 범위

## (1) 변수 선언 위치

- ✓ 프로그램 내에 원하는 위치에 선언 가능(중복 선언 불가)
- ✓ 선언하는 위치에 따라서 사용 범위가 정해짐

## (2) 메소드 내부에 선언한 변수 사용범위

- ✓ 변수 선언 아래쪽으로 사용 가능

```

public class Hello {
    public static void main(String[] args) {
        int countNum = 1;
        .....
        int countTotal = 1;
        .....
    }
}

```

main()  
메소드countNum 변수 선언 이후  
모든 영역에서 사용 가능countTotal 변수 선언 이후  
모든 영역에서 사용 가능  
(선언 이전에는 사용 불가)

## ② 자료형과 변수의 기본 문법 이해하기

## 변수 이해하기

## ▶ 변수의 데이터 표현 범위

## 1) 오버플로 (Overflow)

- (1) 변수의 자료형에 따라 표현 할 수 있는 데이터의 최소값과 최대값이 있음
- (2) 변수가 표현할 수 있는 최대값보다 큰 값이 변수에 지정될 때 발생
- (3) 오버플로 발생 시 문법적 오류가 발생함

## 정수형 자료형 short의 데이터 표현범위

-32,768~32,767

```
short test1 = 32767;
```

short 자료형 최대값  
이하이므로 정상임

```
short test2 = 32768;
```

short 자료형 최대값보다  
크므로 **오버플로 발생**

## 2) 언더플로 (Underflow)

- (1) 변수가 표현할 수 있는 최소값보다 작은 값이 변수에 지정될 때 발생
- (2) 언더플로 발생 시 문법적 오류가 발생함

## 정수형 자료형 short의 데이터 표현범위

-32,768~32,767

```
short test1 = -32768;
```

short 자료형 최대값  
이상이므로 정상임

```
short test2 = -32769;
```

short 자료형 최소값보다  
작으므로 **언더플로 발생**

## ② 자료형과 변수의 기본 문법 이해하기

## 변수 이해하기

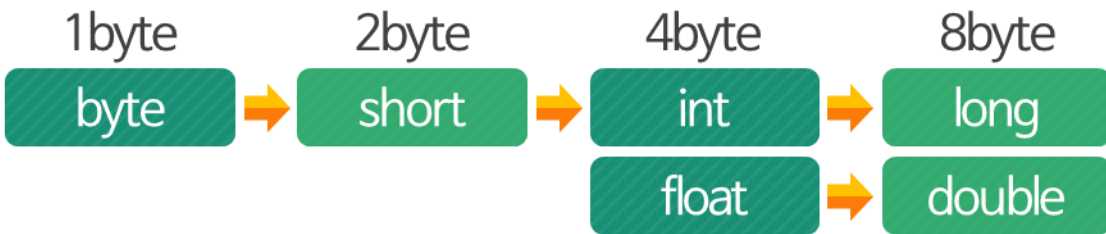
## ▶ 자료형 변환

## 1) 자료형 변환이란?

- A라는 자료형의 값을 B라는 자료형 값으로 변환하는 것
- 종류
  - ✓ 자동 자료형 변환
  - ✓ 강제 자료형 변환

## 2) 자동 자료형 변환

- 메모리 크기가 작은 자료형의 값을 메모리 크기가 큰 자료형 값으로 변환하는 것



```

byte by = 100;
short sh = 123;
int in = 4567;
long lo = 12345;
float fl = 123.34f;
double dou = 456.78;
  
```

변수선언

```

sh = by; // byte → short
in = sh; // short → int
lo = in; // int → long
dou = fl; // float → double
  
```

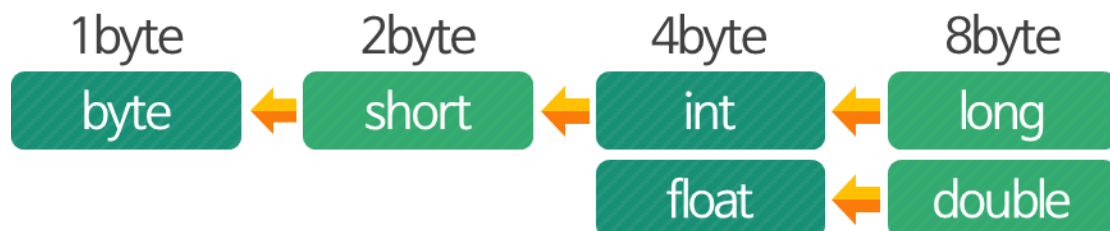
자동자료형변환예제

## ② 자료형과 변수의 기본 문법 이해하기

## 변수 이해하기

## 3) 강제 자료형 변환

- (1) 메모리 크기가 큰 자료형의 값을 메모리 크기가 작은 자료형의 값으로 변환하는 것
- (2) 변수의 값은 작은 자료형 크기에 맞는 값이어야 함 (오버플로 주의)
- (3) 실수형 자료형의 값이 정수형 자료형의 값으로 변환될 때는 소수점 이하는 버려짐



## (4) 캐스트(cast) 연산자를 사용하여 강제 자료형 변환

## 형식

(자료형)변수이름

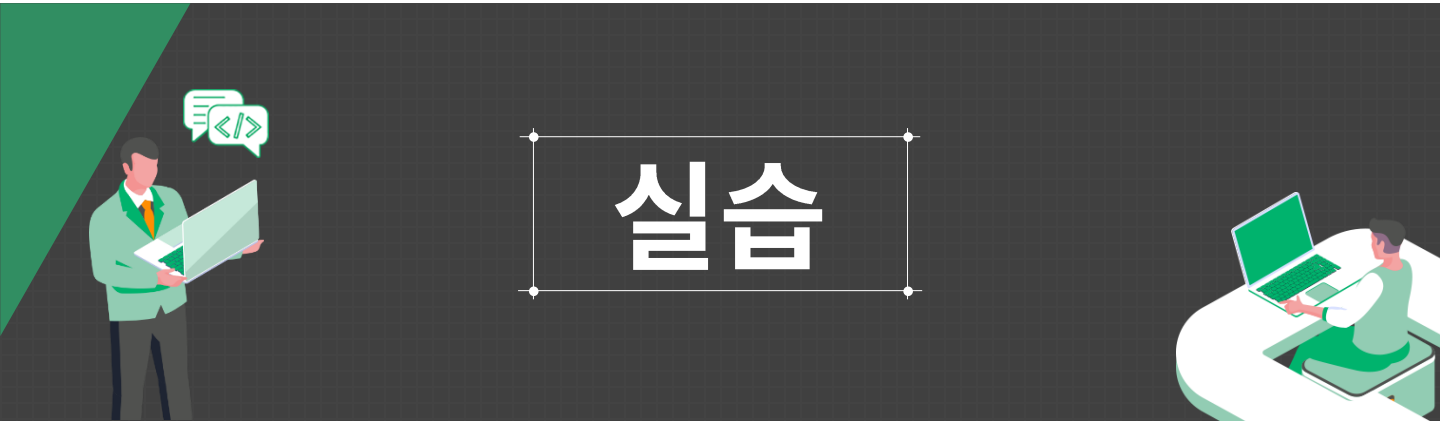
```
byte by = 123;
short sh = 123;
int in = 4567;
long lo = 12345L;
float fl = 123.34f;
double dou = 456.78;
```

## 변수 선언

```
by = (byte)sh; // short → byte
sh = (short)in; // int → short
in = (int)fl; // float → int
fl = (float)dou; // double → float
```

## 강제 자료형 변환예제

## ② 자료형과 변수의 기본 문법 이해하기



### 변수 사용하기 실습



#### 실행 화면

(출력 내용 없음)

- 소스 파일명 : [UseVariable.java]
- 자세한 내용은 실습 영상을 확인해보세요.



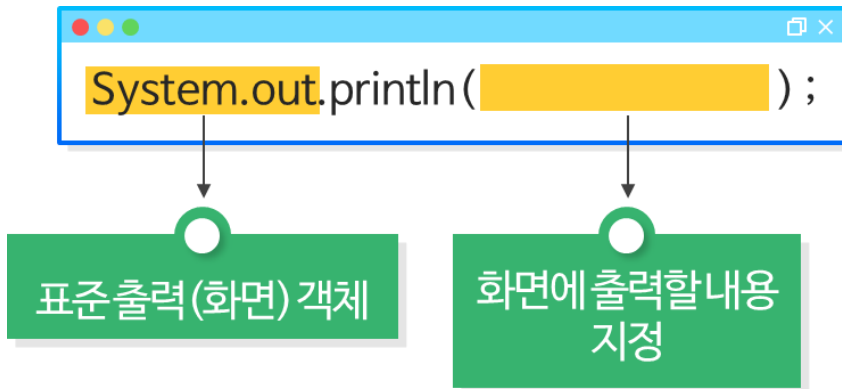
## ② 자료형과 변수의 기본 문법 이해하기

## 표준 입출력 이해하기

## ▶ 표준 출력 메소드 이해하기

## 1) System.out.println() 메소드

- (1) 화면(표준 출력)에 내용을 출력해 주는 메소드
- (2) 소괄호 사이에 화면에 출력할 내용을 지정
- (3) 화면에 내용을 출력한 후 줄 바꿈 실행됨



## (4) 소괄호 사이에 화면에 출력할 내용 지정 방법

- ✓ 큰 따옴표(" ") 사이에 있는 문자열은 화면에 그대로 출력됨
- ✓ 변수는 변수의 값이 화면에 출력됨
- ✓ 상수는 상수의 값이 그대로 화면에 출력됨
- ✓ 수식은 수식이 계산된 결과값이 화면에 출력됨
- ✓ 화면에 출력될 항목들은 "+" 기호로 연결함

```
int a = 100;
System.out.println( "a의 값은" + a + ", " + 200 + "을 더하면" + (a+200) );
```

실행 결과  
(화면에 출력되는 내용)



a의 값은 100, 200을 더하면 300

## ② 자료형과 변수의 기본 문법 이해하기

## 표준 입출력 이해하기

## 2) System.out.print() 메소드

- (1) System.out.println() 메소드와 사용법 동일
- (2) 화면에 내용을 출력한 후 줄 바꿈 실행 안함

```
System.out.println("안녕");
System.out.println("하세요!");
```

실행 결과

안녕  
하세요!

```
System.out.print("안녕");
System.out.print("하세요!");
```

실행 결과

안녕하세요!

## ▶ 표준 입력 메소드 이해하기

## 1) 표준 입력 메소드 사용 방법

- (1) Scanner 클래스를 import 하기
  - (2) 표준 입력 객체를 이용하여 Scanner 클래스의 객체 생성하기
  - (3) Scanner 객체를 이용하여 표준 입력(키보드)로부터 데이터 입력 받기
- ✓ 자료형에 따라 입력 받는 메소드가 정해져 있음

```
import java.util.Scanner;
Scanner s = new Scanner(System.in);
int a = s.nextInt();
```

Scanner 객체

표준 입력(키보드) 객체

## ② 자료형과 변수의 기본 문법 이해하기

표준 입출력 이해하기

정수형자료입력(int)



```
int a = s.nextInt();
```

실수형자료입력(double)



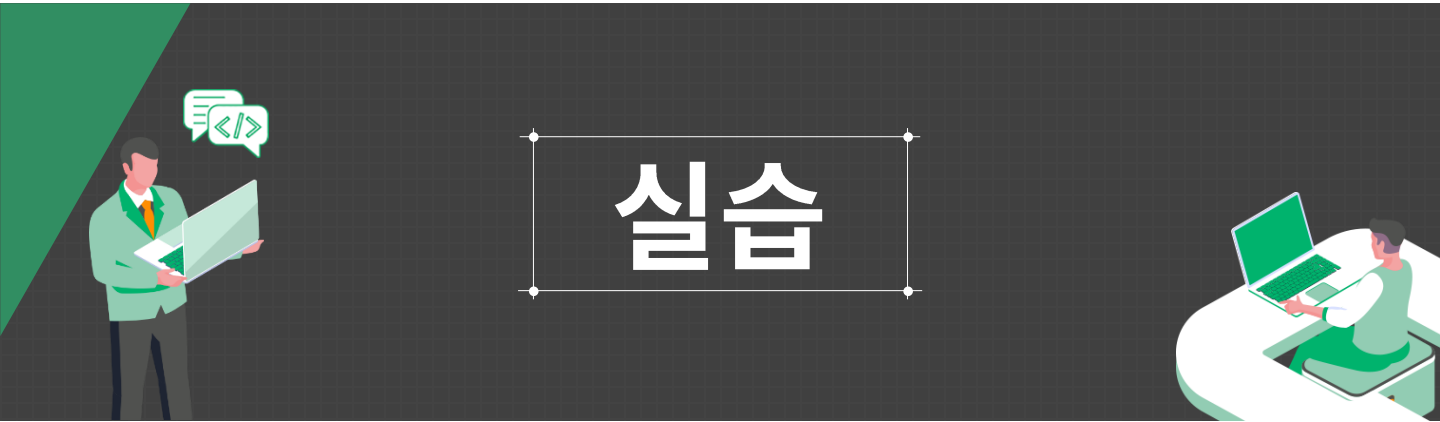
```
double b = s.nextDouble();
```

문자열자료입력(String)



```
String c = s.nextLine();
```

## ② 자료형과 변수의 기본 문법 이해하기



## 표준 입출력 프로그램 실습



## 실행 화면

```
100
300
BOOLEAN : true, false
BYTE : 127, SHORT : 32767, INT : 100, LONG :
123
FLOAT : 123.45, 123.45, DOUBLE : 123.45,
0.012345
CH1 : 자, CH2 : A, CH3 : @
자바 프로그래밍 (JAVA PROGRAMMING)
FLOAT : 123.45 } INT : 123
DOUBLE : 123.45 } INT : 123
CHAR : A } SHORT : 65
정수를 입력하시오 : 123...
```

- 소스 파일명 : [UseInOut.java]
- 자세한 내용은 실습 영상을 확인해보세요.



## 정리하기

### ■ 자료형과 상수 이해하기

- 자료형(Data Type) 이해하기
  - 데이터의 종류에 따라 컴퓨터 메모리의 크기를 정해놓은 것
    - 데이터의 종류 : 정수, 실수, 문자, 문자열
    - 데이터 종류에 따라 메모리의 크기가 달라짐
  - 종류
    - 기본 자료형 : 논리형, 정수형, 실수형, 문자형
    - 참조형 자료형 : 문자열, 배열, 클래스
- 상수(constant)란?
  - 프로그램 수행 중 값이 변하지 않는 데이터
  - 종류 : 논리, 정수, 실수, 문자, 문자열



## 정리하기

### ■ 변수 이해하기

- 식별자(identifier) 란?
  - 프로그래머가 필요시 프로그램 내에 정의 또는 선언하여 사용하는 이름
  - 식별자 만드는 규칙
    - 구성 : 알파벳 대소문자, 한글, 숫자, \_(언더바), \$(달러 표시)를 사용, 이외는 사용불가(공백 포함)
    - 식별자 첫 문자 : 알파벳 대소문자, \_(언더바), 한글로 시작하여야 함, 숫자는 사용불가
    - 예약어는 사용불가
    - 알파벳 대소문자는 구분함
- 예약어(Reserved Word)란?
  - 자바 언어에서 사용하는 문법적 단어들
  - 식별자로 사용할 수 없음
- 변수(Variables)란?
  - 데이터를 저장하는 컴퓨터 메모리 공간을 가리키는 식별자
  - 변수에 데이터를 저장, 수정, 읽기 가능
  - 프로그램이 실행되는 동안 변수의 데이터 값이 유효함
  - 데이터의 자료형 크기에 따라 컴퓨터 메모리 공간이 확보됨
- 자료형 변환이란?
  - A라는 자료형의 값을 B라는 자료형 값으로 변환하는 것
  - 종류 : 자동 자료형 변환, 강제 자료형 변환



## 정리하기

### ■ 표준 입출력 이해하기

- 표준 출력 메소드
  - `System.out.println()` 메소드
    - 화면(표준 출력)에 내용을 출력해 주는 메소드
    - 소괄호 사이에 화면에 출력할 내용을 지정
    - 화면에 내용을 출력한 후 줄 바꿈 실행됨
  - `System.out.print()` 메소드
    - `System.out.println()` 메소드와 사용법 동일
    - 화면에 내용을 출력한 후 줄 바꿈 실행 안 함
- 표준 입력 메소드 사용 방법
  - `Scanner` 클래스를 import 하기
  - 표준 입력 객체를 이용하여 `Scanner` 클래스의 객체 생성하기
  - `Scanner` 객체를 이용하여 표준 입력(키보드)로 부터 데이터 입력 받기