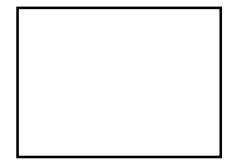
변수 : 값을 저장하기 위한 메모리 공간(기억장소)

프로그래머가 메모리공간에 이름을 붙이고 사용함

예시) int kor;  정수형데이터, 4바이트 크기를 갖는다




변수선언시
저장될 데이터의 유형과 크기를 고려해야 한다!

변수명 :
첫글자 소문자, 카멜법, 스네이크법, 첫글자가 숫자안된다
의미있는 이름으로 변수명을 짓자!

```
int a ;
int kookmin_account;
int shinhanAccount;
```

(리터럴 literal) =문자 그대로의 , 값자체

: 프로그램이 실행되는 동안 변하지 않는 값 (상수라고도 한다)

| | |
|---|---|
| 논리형 | true, false |
| 문자 | 'a' , '한' |
| 문자열 | "hello" |
| 정수형 | 40 ,100 |
| 실수형 | 24.2 |

## 리터럴도 크기를 갖는다 - 상수영역에저장됨

int kor = 32500;

32500이라는 값이 저장( 상수영역)에 저장되고 사용되어짐

정수형은 기본 int형으로 저장됨
실수형은 기본 double형으로 저장됨

long point =2200000000; //오류발생

long point =2200000000L;

point라는 변수에 22억을 저장하려고한다
그러나 오류가 발생됨

22억이 int형자료형의 범위를 넘어서기 때문이다
long형 상수로 저장하고 싶을때는 L을 붙여야 한다

22억 리터럴이 상수영역에 저장될 때 문제를
일으킨다

## 기본형변수와 참조형변수

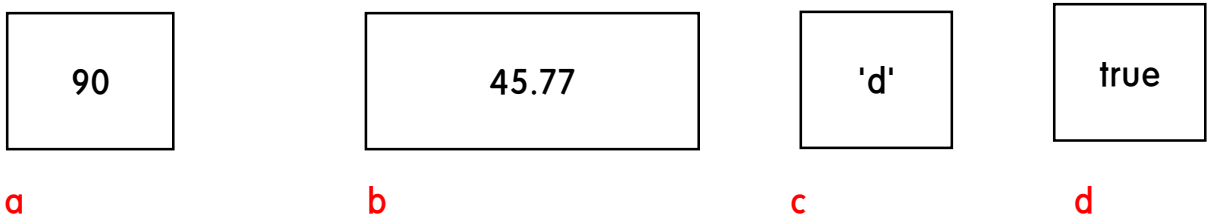
| | |
|---|---|
| 기본형 | 실제 값(DATA)을 저장 |
| 참조형 | 실제값이 있는 공간의<br>주소를 저장한다.<br>(객체와 배열에서 참조형변수를 사용한다) |

```
int kor =90;
int[ ] arr = new int[3];
```

## 자료형 : 크기 + 해석의 도구

| | | |
|---|---|---|
| 논리형 | boolean (1byte) | true, false |
| 문자형 | char (2byte)-unicode | 'a' , '한' |
| 정수형 | byte(1byte)<br>short(2byte)<br>int (4byte)<br>long(8byte) | 127<br>32767<br>21억<br>922경 922L<br>$-2^{n-1}$ ~$2^{n-1}$ -1<br>$-128$~$127$ |
| 실수형 | float (4byte)<br>double(8byte) | 23.2F<br>23.2 |

```
int a=90;
double b=45.77;
char c=  'd'  ;
boolean d=true;
```



프로그래머에게 문자?

문자와 문자열을 구분해서 이야기 합니다.
문자와 문자열은 다르다

문자 : 문자 하나 'a' ,' l'
문자열 : 문자의 집합 "hello"

(문자배열로 처리함)

## 형변환( cast )

: 자동형변환
  명시적형변환

| | |
|---|---|
| 큰변수 = 작은변수 | 자동형변환 |
| 작은변수 = (작은변수형) 큰변수 | 강제형변환(명시적형변환) |

```
int   kor =98;          //가능
double result  = kor;
```

```
double   kor =98.89;    //불가능
int result  = kor;

double   kor =98.89;
int result  = (int) kor;  //강제형변환
```

정수연산

```
int avg  = 10/3;
double avg2 = 10/3;
```

실수연산

```
double avg3  = 10/ (double)3;
```

피연산자에 따라
정수연산과
실수연산으로 나뉜다

정수연산의 기본자료형은 int이다
정수연산의 결과는 정수이다 !! 기억

그래서 소수이하의 데이터가 필요하다면
실수연산이 이루어 질 수 있도록 해야 한다
(주의)

1바이트 :컴퓨터구조에서 의미있는 값을 저장할 수 있는 최소한의 단위

8bit로 구성되어 있다

| | $2^7$ | $2^6$ | $2^5$ | $2^4$ | $2^3$ | $2^2$ | $2^1$ | $2^0$ |
|---|---|---|---|---|---|---|---|---|
| | 부호 | | | | | | | |
| 양수 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| 음수 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

값의 범위 -128 ~ 127

부호없는 값의 범위 0~255

bit

0,1 의 값만 표시

2진법

1byte =8bit

1byte 값의 범위

$2^7$ $2^6$ $2^5$ $2^4$ $2^3$ $2^2$ $2^1$ $2^0$

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| | | | | | | | |

2진수로 변환문제
5
13
56
124

사람은
10진법

126

$10^2$ $10^1$ $10^0$

| | | |
|---|---|---|
| 1 | 2 | 6 |

100 *1 => 100
10 *2 => 20
1 *6 => 6

컴퓨터가 음수를  저장를 저장하는 방식

2의 보수로 저장함

3에 대한 10의 보수 ? 7

n의 보수?
더했을 때  n이 되는 수

## 2의 보수란

2가 되기 위해 보충되는 수? ( 여기서 단순히 숫자2를 말하는 것이 아니다)

2를 이진수로 표현하면  2  -> 10 (2)

2진수 10은  자리올림이 발생되고 0이 되는 수를 뜻한다.

5에 대한 2의 보수 구하라

00000101

\+

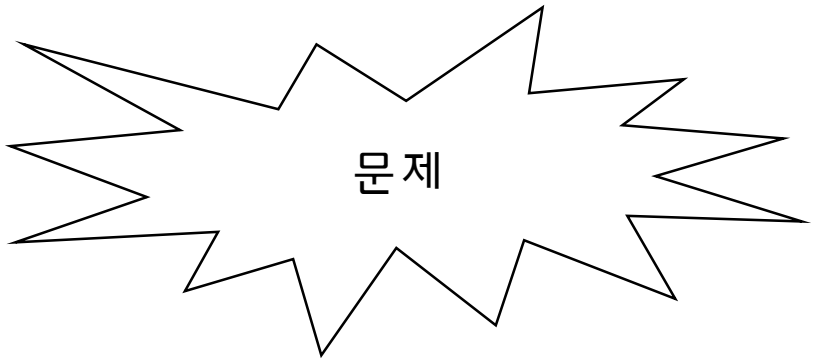
1

00000000

이런 결과를 만들어 낼 수 있게 하는 수

쉽게 2의 보수를 구하는 방법
1의 보수 + 1

| -6 | | |
|---|---|---|
| 1. 절대값을 2진수로 표현한다 | 00000110 | |
| 2. 1의 보수를 구한다 | 11111001 | |
| 3. 2의 보수를 구한다.<br>(1의보수+1) | **11111010** | **-6** |




-8

1. 절대값을 2진수로 표현한다

2. 1의 보수를 구한다

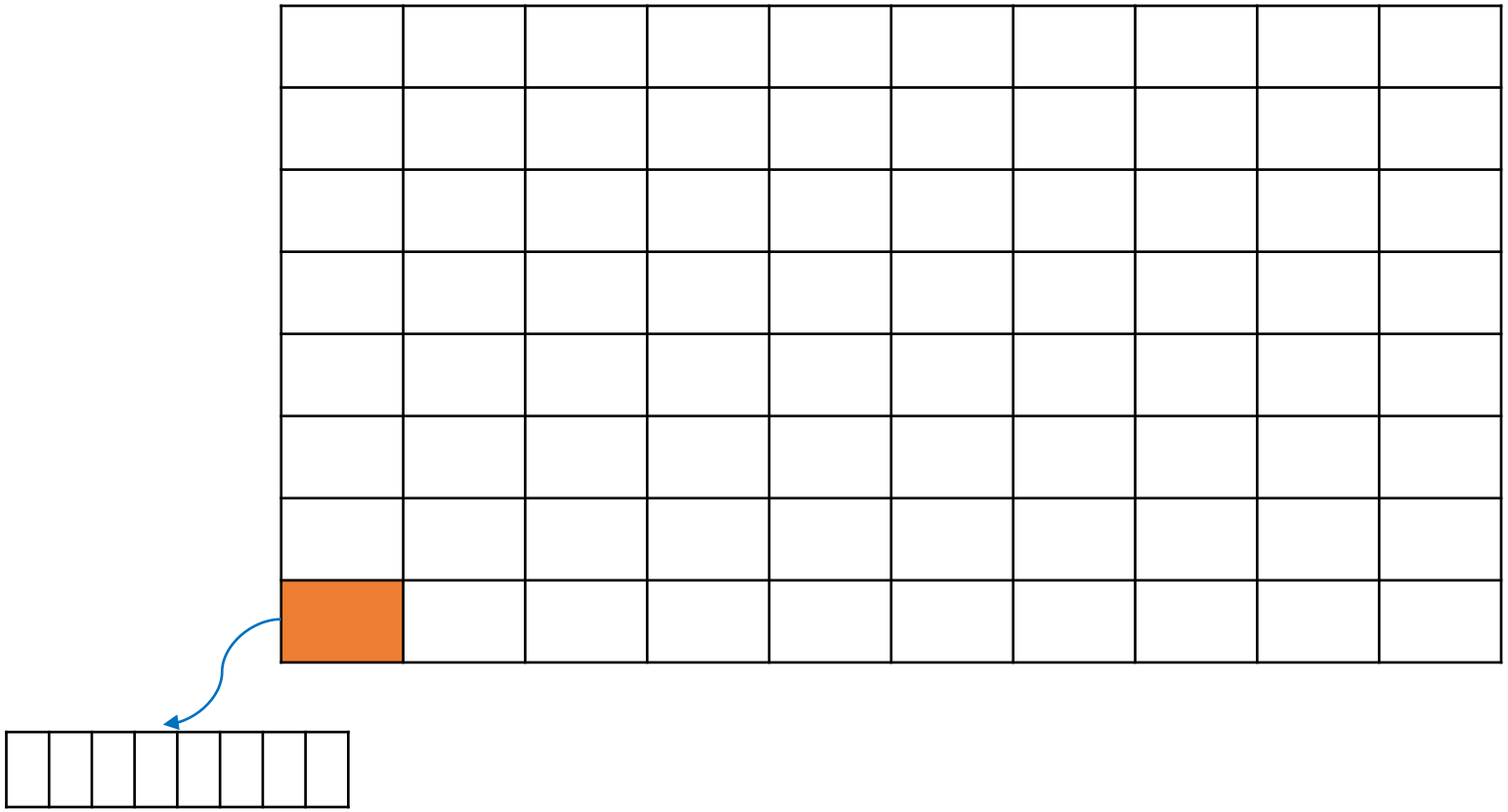
3. 2의 보수를 구한다.
(1의보수+1)

-8

1byte별로
메모리 주소 부여됨

0번지 시작

## 4G 메모리 : 약 42억개 바이트



1byte : 8bit로 구성됨

**모든데이터는 숫자이다**

**문자-인코딩, 디코딩**

**유니코드 문자표**

| 문자 | 코드 |
|---|---|
| 0 | 48 |
| 1 | 49 |
| 2 | 50 |
| 3 | 51 |
| 4 | 52 |

| 문자 | 코드 |
|---|---|
| A | 65 |
| B | 66 |
| C | 67 |
| D | 68 |
| E | 69 |

| 문자 | 코드 |
|---|---|
| a | 97 |
| b | 98 |
| c | 99 |
| d | 100 |
| e | 101 |

```
char ch ='A';
```

```
char ch ='한';
```