

## 데이터 타입

### 불리언 표현

- BOOL, 값 : YES or NO

### 숫자표현

- 정수 — 부호가 있는 정수 : NSInteger
  - 부호가 없는 정수 : NSUInteger
- 실수 — 32bit 실수 표현 : float
  - 64bit 실수 표현 : double
  - core graphic용 실수 표현 : CGFloat
- 객체형 — NSNumber : @ 뒤에 숫자를 써 주었던 것이 숫자를 객체로 사용한 것
- 객체형이 아닌 표현은 모두 [Value Type](값)
- Value Type을 직접 표현할 때에는 포인터를 사용하지 않는다.
- Reference Type을 표현할 때에는 포인터를 사용한다.
- 포인터(@)를 뜻한다.

### 문자표현

- 문자 — 단 하나의 문자를 표현 : char [Value Type]
- 문자열 — 문장 또는 문단을 표현 : NSString [Reference Type]

## 데이터 타입

- 프로퍼티 또는 매개변수의 데이터 타입을 명시해 준 경우, 다른 타입의 값이 할당되거나 전달이 된다면 오류가 발생가능
- id라는 타입은 모든 객체 타입을 의미

### 타입 종류 설명

```
//-----  
//타입 종류 설명  
//불리언타입//  
//    죽음에대한 설명        결과  
BOOL haveBlood = NO;  
BOOL dontHaveBlood = YES;  
  
//부호가있는 정수타입//  
//    부호를 사용함            부호사용숫자  
NSInteger signedInteger = -100;  
NSInteger twoHundred = 200;  
  
//부호가없는 정수타입//  
//    부호를 사용하지않음      부호사용하지않은숫자  
NSUInteger unsignedinteger = 100;  
NSUInteger hundred = 100;  
//dontHaveBlood에 (괄호안의값)을 넣어준다.  
dontHaveBlood = (twoHundred > hundred);  
//hundred의 "값"을 anotherNumber에 복사한다.  
NSInteger anotherNumber = hundred;  
  
//객체형 숫자 타입//  
//선언이후 사용할때는 *를 붙이지 않는다.  
//    객체형 숫자를 생성, 초기화  
NSNumber *someNumberObject = [[NSNumber alloc] initWithInt:100];  
//    someNumberObject의 "주소"를 anotherNumberVariable에 복사한다.  
NSNumber *anotherNumberVariable = someNumberObject;  
  
//실수형 숫자 타입//  
//키, 몸무게, 시력  
CGFloat height = 200.3;  
CGFloat weight = 100.5;  
CGFloat eyesight = 1.0;  
  
//한글자만 표현 하는 타입  
char somecharacter = 'a';  
  
//-----
```

### 형식지정자 설명

```
//  
// 형식지정자 //  
// 정수타입 - 부호가 있는 32bit 정수타입 > 10진수 (%d)  
// 정수타입 - 부호가 있는 64bit 정수타입 > 10진수 (%ld)  
NSLog(@"height : %ld", jack.height);  
  
// 정수타입 - 부호가 없는 32bit 정수타입 > 10진수 (%u)  
// 정수타입 - 부호가 없는 64bit 정수타입 > 10진수 (%lu)  
NSLog(@"height : %lu", jack.height);  
  
// 정수타입 (16진수)  
NSLog(@"height : %lx", jack.height);  
  
// 정수타입 (8진수)  
NSLog(@"height : %lo", jack.height);  
  
// 실수타입 - 32bit 실수타입 > 10진수 (%f)  
// 실수타입 - 64bit 실수타입 > 10진수 (%lf)  
NSLog(@"float value : %lf", soser.eyesight);  
  
// 불리언 타입 %d YES,NO가 0,1로 표현되기때문에 32비트 10진수와 같다  
NSLog(@"Boolean value ; %d", YES);  
NSLog(@"Boolean value ; %d", NO);  
  
// 캐릭터 타입 %c  
NSLog(@"character : %c %c %c", 'a','b','c');  
  
// %를 표현할때 %%  
NSLog(@"몇 500%% 인가요?");  
  
// 주소값을 확인할때 %p  
NSLog(@"jack object : %@, mamory address: %p",jack,jack);  
  
// 줄바꿈 \n  
NSLog(@"줄\n바꾸기");  
  
// 탭 표현 \t  
NSLog(@"사이에 \t을 넣어준다");  
  
// 응용  
NSLog(@"키는 %ld이고 \n몸무게는 %ld \n발사이즈%ld \n %d",jack.height,jack.kg,jack.footsize,jack.isdead);  
  
// long : 32bit 부호가 있는 정수 타입  
// int : 64bit 부호가 있는 정수 타입  
  
// 형식지정자 찾아보기  
// %-51d  
// %-041d  
// %+31d  
// %5.2f  
// %-10.3f  
// %10.0f  
// %.3f
```