



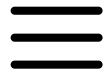
# C++ Study

## C++의 기본 자료형

울산대학교 | 박예찬 | Github:devpyc | yechan6855@naver.com



```
#include <iostream>
using namespace std;
int main() {
    cout << "int: " << sizeof(int) << " bytes\n";
    cout << "double: " << sizeof(long long) << " bytes\n";
    cout << "bool: " << sizeof(bool) << " bytes\n";
}
```



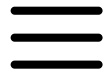
## C++ Data Type



정수형	char	1byte
	short	2byte
	int	4byte
	long	4byte
	long long	8byte
실수형	float	4byte
	double	8byte
	long double	8byte 이상



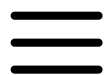
# 정수



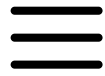
C++ Data Type



부호	자료형	크기(Byte)	값 범위
signed	short	2	-32,768 ~ 32,767
	int	4	-2,147,483,648 ~ 2,147,483,647
	long long	8	-9,223,372,036,854,775,808 ~ 9,223,372,036,854,775,807
unsigned	unsigned short	2	0 ~ 65535
	unsigned int	4	0 ~ 4294967295
	unsigned long long	8	0 ~ 18.446,744,073,709,551,615



실수



## C++ Data Type



자료형	크기(Byte)
float	4
double	8
long double	16

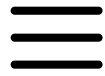


## 실수의 정밀도

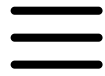
```
#include <iostream>
using namespace std;

int main() {
    double pi = 3.14159265358979;
    cout << pi; // 3.14159 까지만 출력됨
}
```





**소수점 이하 자리수를 고정하고 싶다면?**



# cout.precision()

```
#include <iostream>
#include <iomanip> // fixed 사용시 필요
using namespace std;

int main() {
    double pi = 3.14159265358979;
    cout << fixed;
    cout.precision(10); // 10번째 자리에서 반올림
    cout << pi << endl;    // 3.141592654
}
```



# 변수 (Variable)



## 변수 선언

변수는 데이터를 저장하는 이름이 있는 공간

```
int a = 10;
```

```
double b = 3.14
```

```
char c = 'c'
```



# 타입 자동 추론이 있다?



## 그럼 auto만 쓰면 되는거 아닌가?

auto가 사용된 변수는 반드시 초기화식이 필요함



# 상수 (Constant)



## 값이 변하지 않는 데이터

```
const int MAX = 1000000007;
```

#define 으로도 가능하지만 const 사용을 권장





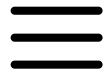
# 타입 변환



## 자동 변환과 직접 변환

```
double d = 3;
```

```
int n = (int)3.14;
```



# 오버플로우를 방지하는 방법



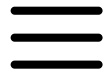
## 1LL(long long 타입의 1) 연산

a와 b가 int일때, int 범위 초과시 오버플로우가 발생하지만,  
1LL과 곱해준다면 첫 연산부터 long long으로 바뀜

```
int a = 100000, b = 100000;
```

```
long long sum1 = a * b;
```

```
long long sum2 = 1LL * a * b;
```



# return 0;

울산대학교 | 박예찬 | Github:devpyc | yechan6855@naver.com