



자료형

입력한 문자가 숫자인지, 알파벳인지 구분하기 위해서는 데이터형(자료형)이라는 것을 만들고 선언해 주어야 한다.

int

정수형 데이터를 위한 데이터 타입

ATmega : 2바이트 메모리 할당 ARM : 4바이트 메모리 할당

unsigned int

정수형 데이터를 위한 데이터 타입

int : +, - 가능 unsigned int : + 가능

ATmega : 2바이트 할당

int는 MSB가 부호를 나타내는 비트, unsigned int는 그냥 데이터의 크기를 나타내는 비트

byte

1바이트 할당

0~255

char

하나의 문자를 저장하기 위한 데이터 타입

1바이트 할당

문자가 저장되지만 실질적으로는 아스키코드 값이 저장

| -128~127

unsigned char

| 하나의 문자를 저장하기 위한 데이터 타입

0~255

| 확장아스키코드의 사용 가능

boolean

| 참, 거짓을 의미하는 논리값을 저장하는 데이터타입

true(1), false(0)의 값이 들어감

| 1바이트 사용

float

| 실수형의 데이터를 저장하기 위한 데이터 타입

4바이트 할당

부호비트 1비트, 지수부 8비트, 나머지 23비트는 가수부

| 정밀도, 연산속도 : float < double

double

| 실수형의 데이터를 저장하기 위한 데이터 타입

| ATmega : 4바이트 할당 나머지 : 8바이트 메모리 할당

String

| 문자를 사용하기 위해 사용하는 전용 클래스

| 다양한 멤버함수, 연산자 사용 - char보다 다양하게 문자 사용

const

상수를 선언하기 위한 것 - 소스내에서 값 변경 불가

변수에 하나의 값만 넣어서 변함없이 사용하고 싶다면 데이터타입 앞에 선언

static

선언한 지역을 벗어나도 소멸되지 않고 값을 유지

정적변수

그 밖에 사용되는 데이터형

u가 붙어있는 것은 Unsigned를 나타내고 숫자는 바이트

Specifier	Common Equivalent	Signing	Bits	Bytes	Minimum Value	Maximum Value
int8_t	signed char	Signed	8	1	-128	127
uint8_t	unsigned char	Unsigned	8	1	0	255
int16_t	short	Signed	16	2	-32,768	32,767
uint16_t	unsigned short	Unsigned	16	2	0	65,535
int32_t	long	Signed	32	4	-2,147,483,648	2,147,483,647
uint32_t	unsigned long	Unsigned	32	4	0	4,294,967,295
int64_t	long long	Signed	64	8	-9,223,372,036,854,775,808	9,223,372,036,854,775,807
uint64_t	unsigned long long	Unsigned	64	8	0	18,446,744,073,709,551,615