

# C언어

---

LESSON 02.

**상수, 변수, 자료형 및  
표준 입력 함수의 이해**

# 1. 상수의 이해

# 1. 상수의 이해

## 01 개념

메모리의 일정 공간에 저장된 변경이 불가능한 데이터

```
#include <stdio.h>

void main()
{
    printf("%d ₩n", 10);
    printf("%f ₩n", 1.2);
    printf("%c ₩n", 'k');
    printf("%s ₩n", "abcd");
}
```

# 1. 상수의 이해

## 02 수치 상수

### 고정소수점 상수

- 정수형 상수
- 소수점을 포함하지 않는 상수

### 부동소수형 상수

- 실수형 상수
- 소수점을 포함하는 상수

# 1. 상수의 이해

## 03 문자 상수

### 1 문자형 상수

- ✓ 단일 인용부호(' ') 사용
- ✓ ASCII코드로 표현 가능

### 2 문자열 상수

- ✓ 이중 인용부호(" ") 사용
- ✓ 문장의 끝을 나타내는 NULL(\0)문자가 포함됨

# 1. 상수의 이해

## 03 문자 상수

### 3 ASCII CODE

- ✓ 정보를 표현하기 위해 약속한 기본 코드
- ✓ 영문자(52) / 숫자(10) / 특수문자(66)

# 1. 상수의 이해

## 04 예 : 수치 상수와 문자 상수

```
#include <stdio.h>

void main()
{
    printf("정수형 상수 : %d \n", 10);
    printf("실수형 상수 : %f \n", 1.2);
    printf("문자형 상수 : %c \n", 'k');
    printf("문자열 상수 : %s \n", "abcd");
}
```

# 1. 상수의 이해

## 05 예 : ASCII CODE의 이용

```
#include <stdio.h>

void main()
{
    printf("아스키 문자 : %c \n", 65);
    printf("아스키 숫자 : %d \n", 'A');
}
```



**1. 상수의 이해에 대한  
강의가 끝났습니다.**

## 2. 변수와 자료형의 이해

## 2. 변수와 자료형의 이해

### 01 변수

메모리의 일정 공간으로 데이터의 저장 및 변경이 가능

➡ 변수를 생성하는 명령을 변수 선언이라 함

## 2. 변수와 자료형의 이해

### 02 변수 선언

#### 데이터 저장 공간을 확보하고 이름 붙이는 작업

- 형식 : 자료형 공간이름;
- 자료형을 통해 데이터 저장 공간을 확보함
- 설정한 공간 이름을 변수명이라 함

## 2. 변수와 자료형의 이해

### 03 자료형

데이터 종류에 따라 필요 메모리 공간을 확보하는 명령

- 공간에 저장될 데이터의 성격을 결정하는 기준이 됨
- 데이터 저장에 필요한 공간의 크기가 약속되어 있음

## 2. 변수와 자료형의 이해

### 03 자료형

데이터 종류	자료 형	크기(바이트)
정 수	int	4
	long int	4
	unsigned int	4
실 수	float	4
	double	8
문 자	char	1
	unsigned char	1

## 2. 변수와 자료형의 이해

### 04 예 : 변수의 이용 - 정수 및 실수 데이터

```
#include <stdio.h>

void main()
{
    int age = 21;
    float weight = 50.0, height = 163.5;

    printf("나의 나이 : %d ₩n", age);
    printf("나의 몸무게 : %f ₩n", weight);
    printf("나의 키 : %f ₩n", height);
}
```

## 2. 변수와 자료형의 이해

### 05 예 : 변수의 이용 - 문자 데이터

```
#include <stdio.h>

void main()
{
    char ch;
    ch = 'D';
    printf("#.1 ch : %c \n", ch);

    ch = 'G';
    printf("#.2 ch : %c \n", ch);
}
```



## 2. 변수와 자료형의 이해

### 06 예 : 변수의 이용 - 실수 데이터 연산

```
#include <stdio.h>

void main()
{
    float h_inch = 70.0;
    float h_cm = h_inch * 2.54;

    printf("나의 키는 %f (inch)이고, ");
    printf("%f (cm)입니다.\n", h_inch, h_cm);
}
```

## 2. 변수와 자료형의 이해

### 07 자료형 변환

#### 1 대입식에서의 자료형 변환

- ✓ 대입연산자 우측의 데이터 자료형이 좌측의 데이터 자료형으로 변환

#### 2 연산식에서의 자료형 변환

- ✓ 서로 다른 자료형 데이터의 연산이 진행되는 경우 저장 공간의 크기가 큰 데이터의 자료형으로 일괄적으로 변환

## 2. 변수와 자료형의 이해

### 07 자료형 변환

#### 3 Cast연산자에 의한 자료형 변환

- ✓ 변수 또는 상수에 대하여 명시한 자료형으로 강제적 변환

## 2. 변수와 자료형의 이해

### 08 예 : 자료형 변환 - 대입식

```
#include <stdio.h>

void main()
{
    int n1;
    float n2 = 2.5;

    n1 = n2;
    printf("n1 = %d \n", n1);
}
```

## 2. 변수와 자료형의 이해

### 09 예 : 자료형 변환 - 연산식

```
#include <stdio.h>

void main()
{
    printf("#.1 결과 : %d \n", 5 / 2);
    printf("#.2 결과 : %f \n", 5 / 2.0);
    printf("#.3 결과 : %f \n", 5.0 / 2);
}
```

## 2. 변수와 자료형의 이해

### 10 예 : 자료형 변환 - 강제 형변환

```
#include <stdio.h>

void main()
{
    printf("#.1 결과 : %d \n", 5 / 2);
    printf("#.2 결과 : %f \n", (float)5 / (float)2);
}
```

## 2. 변수와 자료형의 이해

### 11 출력 형식 지정 제어 문자의 확장 기호

#### 출력 형식을 결정하는 제어 문자의 확장 기능

확장기호	기 능	사 용 예
m	출력되는 인수의 필드 폭을 표시	%5d
		%3c
-	인수의 값을 필드의 좌측 기준으로 출력 우측에 남는 공간은 공백으로 채움	%-7d
		%-10c
m.n	실수형 인수 출력 시 소수점 이하 자릿수를 표시 n : 소수점 이하 자릿수	%.2f
		%-
L (또는 l)	출력 인수가 정수인 경우 : long int형 출력인수가 실수인 경우 : double형	%Ld
		%Lf

## 2. 변수와 자료형의 이해

### 12 예 : 제어 문자의 확장 - %d

```
#include <stdio.h>
```

```
void main()
```

```
{
```

```
    printf("#.1 결과 : %d \n", 10);
```

```
    printf("#.2 결과 : %Ld \n", 10000000001);
```

```
    printf("#.3 결과 : %5d \n", 10);
```

```
    printf("#.4 결과 : %-5d \n", 10);
```

```
}
```

#.3				1	0
#.4	1	0			



## 2. 변수와 자료형의 이해

### 13 예 : 제어 문자의 확장 - %f

```
#include <stdio.h>
```

```
void main()
```

```
{
```

```
    printf("#.1 결과 : %f \n", 12.34);
```

```
    printf("#.2 결과 : %.2f \n", 12.34);
```

```
    printf("#.3 결과 : %7.2f \n", 12.34);
```

```
    printf("#.4 결과 : %-7.2f \n", 12.34);
```

```
}
```

#.3			1	2	.	3	4
#.4	1	2	.	3	4		

## 2. 변수와 자료형의 이해

### 14 예 : 제어 문자의 확장 - %c, %s

```
#include <stdio.h>

void main()
{
    char ch = 'A';
    char str[20] = "abc 123";

    printf("#.1 문자 : %3c \n", ch);
    printf("#.2 문자열 : %-20s \n", str);
}
```

**2. 변수와 자료형의 이해에 대한  
강의가 끝났습니다.**