

본 강의에서 수업자료로 이용되는 저작물은  
저작권법 제25조 수업목적 저작물 이용 보상금제도에 의거,  
한국복제전송저작권협회와 약정을 체결하고 적법하게 이용하고 있습니다.  
약정범위를 초과하는 사용은 저작권법에 저촉될 수 있으므로  
수업자료의 재 복제, 대중 공개·공유 및 수업 목적 외의 사용을 금지합니다.

2024. 8. 30.

부천대학교·한국복제전송저작권협회

**C#**

## **2장 언어 구조1**

# C# 프로그래밍 입문



## 2. 언어 구조 1



# 목차

■ 어휘 구조

■ 자료형

■ 연산자

■ 형 변환

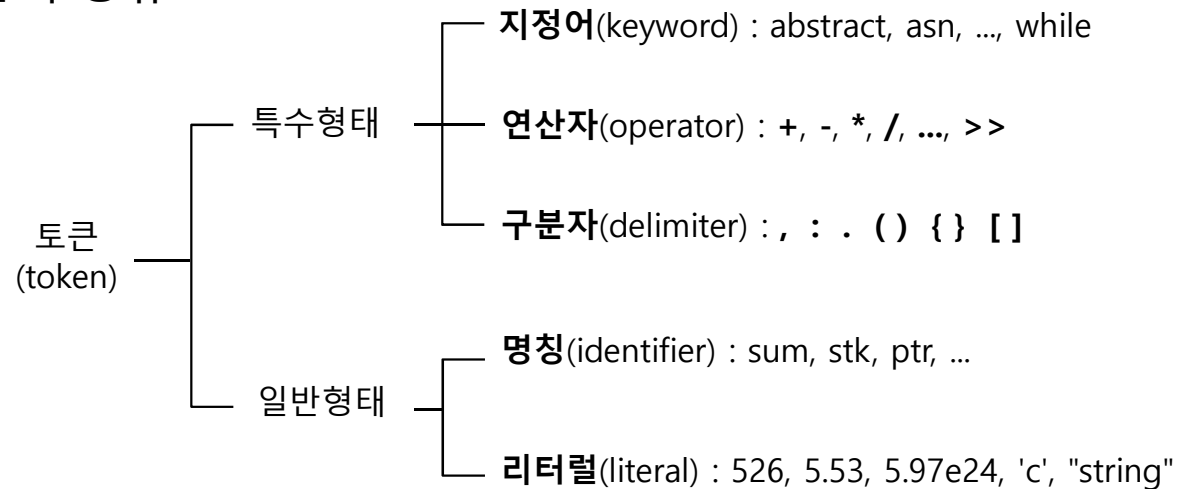


# 어휘 구조

## ■ 어휘

- 프로그램을 구성하고 있는 기본 소자
- 토큰(token)이라 부름
- 문법적으로 의미 있는 최소 단위

## ■ 토큰의 종류





## 지정어

- 프로그래밍 언어 설계 시에 그 기능과 용도가 이미 정의되어 있는 단어
- C# 지정어 (77개) – C# language specification (ECMA TC39/TG2)

abstract	as	base	bool	break
byte	case	catch	char	checked
class	const	continue	decimal	default
delegate	do	double	else	enum
event	explicit	extern	false	finally
fixed	float	for	foreach	goto
if	implicit	in	int	interface
internal	is	lock	long	namespace
new	null	object	operator	out
override	params	private	protected	public
readonly	ref	return	sbyte	sealed
short	sizeof	stackalloc	static	string
struct	switch	this	throw	true
try	typeof	uint	ulong	unchecked
unsafe	ushort	using	virtual	void
volatile	while			



# 리터럴

## ■ 리터럴의 의미

- 자신의 표기법이 곧 자신의 값이 되는 상수

## ■ 리터럴의 종류

리터럴 상수

정수형 상수: 10진수, 16진수(C#에서는 8진수(octal)을 지원하지 않음)

실수형 상수: 고정소수점, 부동 소수점

부울형 상수: true, false

문자 상수

스트링 상수

객체 참조 리터럴: null



# 주석

## ■ 프로그램을 설명하기 위한 문장

- 프로그램의 실행에는 무관
- 프로그램 유지보수에 중요

## ■ 주석의 종류

### ■ // comment

- //부터 새로운 줄 전까지 주석으로 간주
- 예) `int size = 100; //size는 100으로 초기화`

### ■ /\* comment \*/

- /\*와 다음 \*/ 사이의 모든 문자들은 주석으로 간주
- 주석문 안에서 또 다른 주석이 포함될 수 없음
- 예) /\* C# 언어에서는 여러 줄의 주석을 위해 지금 사용하고 있는  
다. \*/

주석의 형태를 지원하고 있





## 주석

- `///` comment
  - `///` 다음의 문자들은 주석으로 간주
  - C# 프로그램 내에서 사용되지 않고 C# 프로그램에 대한 웹 보고서를 작성하는데 사용하는 방법
  - XML 태그를 이용하여 기술
  - 컴파일 시에 `/doc` 옵션을 사용하여 XML 문서 생성



```
using System;
/// <summary>
/// CommentApp Class : Convert a hexadigit into binary value
/// </summary>
class Program
{
    /// <summary>
    /// Main Method
    /// </summary>
    public static void Main()
    {
        int value = 0;
        char ch;
        Console.Write("Enter a super digit : "); // prompt message
        ch = (char) Console.Read();
        switch (ch)
        {
            case 'A': case 'B': case 'C':
            case 'D': case 'E': case 'F':
                value = ch - 'A' + 10; // code value => binary value
                break;
            case 'a': case 'b': case 'c':
            case 'd': case 'e': case 'f':
                value = ch - 'a' + 10; // code value => binary value
                break;
            default: Console.WriteLine(ch + " is not a hexadigit");
                break;
        }
        Console.WriteLine(value);
    } // end of Main()
}
```

**\*표준 입출력 메서드**

- Console.Read()
- Console.ReadLine()
- Console.Write
- Console.WriteLine()

실행 결과: Enter a super digit = F       $\text{value} = 70(\text{ch}) - 65('A') + 10 = 15$

Dec	Hex	문자	Dec	Hex	문자	Dec	Hex	문자	Dec	Hex	문자
0	0	NULL	20	14	DC4	40	28	(	60	3C	<
1	1	SOH	21	15	NAK	41	29	)	61	3D	=
2	2	STX	22	16	SYN	42	2A	*	62	3E	>
3	3	ETX	23	17	ETB	43	2B	+	63	3F	?
4	4	EOL	24	18	CAN	44	2C	,	64	40	@
5	5	ENQ	25	19	EM	45	2D	-	65	41	A
6	6	ACK	26	1A	SUB	46	2E	.	66	42	B
7	7	BEL	27	1B	ESC	47	2F	/	67	43	C
8	8	BS	28	1C	FS	48	30	0	68	44	D
9	9	HT	29	1D	GS	49	31	1	69	45	E
10	A	LF	30	1E	RS	50	32	2	70	46	F
11	B	VT	31	1F	US	51	33	3	71	47	G
12	C	FF	32	20	space	52	34	4	72	48	H
13	D	CR	33	21	!	53	35	5	73	49	I
14	E	SO	34	22	"	54	36	6	74	4A	J
15	F	SI	35	23	#	55	37	7	75	4B	K
16	10	DLE	36	24	\$	56	38	8	76	4C	L
17	11	DC1	37	25	%	57	39	9	77	4D	M
18	12	DC2	38	26	&	58	3A	:	78	4E	N
19	13	DC3	39	27	'	59	3B	;	79	4F	O



## 표준 입출력

[ch2\_quiz1\_xxxx] 다음 프로그램의 실행 결과 값을 표현 하시오.

```
using System;
class Program
{
    public static void Main()
    {
        int i; char c;
        Console.Write("Enter a digit and a character = ");
        i = Console.Read() - 48; //'0'
        c = (char) Console.Read();
        Console.Write("i = " + i);
        Console.WriteLine(", c = " + c);
        //Console.WriteLine("i = {0} , c = {1}", i, c );
    }
}
```

\*표준 입출력 메서드  
- Console.Read()  
- Console.ReadLine()  
- Console.Write  
- Console.WriteLine()

실행 결과:



## 자료형 [1/2]

### ■ 자료형의 의미

- 자료 객체가 갖는 형으로 구조 및 개념, 값의 범위, 연산 등을 정의

### ■ 자료형의 종류





## 자료형 [2/2]

- C# 의 자료형은 공통자료형 시스템(CTS)에서 정의한 형식으로 표현할 수 있다.

// 다음 두 선언의 의미는 동일하다.

System.Int32 x;     // CTS 형으로 정수형 변수 x의 선언

int x;             // C# 형으로 정수형 변수 x의 선언

- CTS 형과 C# 자료형과의 관계

CTS 자료형	의미	C#자료형	CTS 자료형	의미	C#자료형
System.Object	객체형	object	System.Int64	64비트 정수형	long
System.String	스트링형	string	System.UInt64	64비트 부호없는 정수형	ulong
System.Sbyte	부호있는 바이트형	sbyte	System.Char	문자형	char
System.Byte	바이트형	byte	System.Single	단일 정밀도 실수형	float
System.Int16	16비트 정수형	short	System.Double	이중 정밀도 실수형	double
System.UInt16	16비트 부호없는 정수형	ushort	System.Boolean	불린형	bool
System.Int32	32비트 정수형	int	System.Decimal	10진수형	decimal
System.UInt32	32비트 부호없는 정수형	uint			