

# 컴퓨터공학 All in One

---

C/C++ 문법, 자료구조 및 심화 프로젝트 (나동빈)  
제 85강 - C# 구조체와 클래스

# C# 구조체와 클래스

## C# 구조체

- C# .NET의 기본 데이터 타입은 struct로 정의되어 있습니다.
- int, float, double 등의 데이터 타입은 struct로 정의된 형식입니다.
- C#의 구조체는 클래스와 흡사한 형태를 가지고 있으나 상속은 할 수 없습니다.
- 객체 지향 프로그래밍을 위하여 C#에서도 클래스를 사용해야 합니다.

# C# 구조체와 클래스

## C# 프로젝트 생성

```
namespace ConsoleApp1
{
    struct Adder
    {
        public int x, y;
        public Adder(int x, int y)
        {
            this.x = x;
            this.y = y;
        }
        public string GetResult()
        {
            return x + " + " + y + " = " + (x + y);
        }
    }
    class Program
    {
        static void Main(string[] args)
        {
            Adder adder = new Adder(10, 20);
            Console.WriteLine(adder.GetResult());
            Console.ReadLine();
        }
    }
}
```

# C# 구조체와 클래스

## C# 클래스

- C#의 클래스의 기본적인 특성은 C++의 클래스와 동일합니다.
- 메소드(Method): 특정한 기능을 수행하는 함수를 의미합니다. C#에서는 앞 문자가 대문자로 시작합니다.
- 필드(Field): 클래스의 내부 변수를 의미합니다. C++과 마찬가지로 접근제한자를 사용할 수 있습니다. 일반적으로 내부 변수는 객체 지향 프로그래밍의 원리에 따라 private을 이용합니다.

# C# 구조체와 클래스

## C# 프로젝트 생성

```
public class Student
{
    private int id;
    private string name;
    private int score;
    public Student(int id, string name, int score)
    {
        this.id = id;
        this.name = name;
        this.score = score;
    }
    public void Show()
    {
        Console.WriteLine("학번: " + id);
        Console.WriteLine("이름: " + name);
        Console.WriteLine("점수: " + score);
    }
}
```

# C# 구조체와 클래스

## C# 프로젝트 생성

```
namespace ConsoleApp1
{
    class Program
    {
        static void Main(string[] args)
        {
            Student student = new Student(20153157, "나동빈", 98);
            student.Show();
            Console.ReadLine();
        }
    }
}
```

# C# 구조체와 클래스

## C# 프로젝트 생성

C:\Users\나동빈\source\repos\ConsoleApp1\ConsoleApp1\bin\Debug\ConsoleApp1.exe

학번 : 20153157  
이름 : 나동빈  
점수 : 98