

C언어 스터디

4.0주차

<구조체>

CAPS

구조체

구조체란?

자기가 원하는 자료형을 만드는 것!

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

struct MyStruct
{
    int a, b;
};

int main(void) {
    struct MyStruct tmp;
    tmp.a = 1;
    tmp.b = 2;
    cout << tmp.a << " " << tmp.b;
}
```

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

struct MyStruct
{
    string str;
    int count = 0;
};

struct MyStruct arr[2]; // 배열의 인자를 구조체로 정의하는 방법입니다.
int main(void) {
    for (int i = 0; i < 2; i++) {
        cin >> arr[i].str >> arr[i].count;
    }
    for (int i = 0; i < 2; i++) {
        cout << arr[i].str << " " << arr[i].count << "\n";
    }
}
```

```

#include<iostream>
using namespace std;

typedef int hellint; // 자료형의 이름을 마음대로 지을 수 있음!

struct lily // 이렇게 구조체를 만들 거나
{
    int a, b;
};

typedef
struct idol
{
    int a, b;
}
nmixx; // 자료형의 이름을 바꿀 수 있음

int main(void) {
    hellint a = 1; cout << a << '\n';

    struct lily b = { 1,2 }; // 선언과 동시에 중괄호로 초기화가 가능합니다!
    cout << b.a << " " << b.b << "\n";

    struct idol c = { 100,200 }; // typedef 안의 형식도 쓸 수 있음. (자료구조)
    연결리스트라는 데 씁니다.
    cout << c.a << " " << c.b << "\n";

    nmixx d;
    d.a = 200; d.b = 300;
    cout << d.a << " " << d.b;
}

```

```

#include<iostream>
#include<string>
#include<vector>
using namespace std;

typedef struct element
{
    string str;
    int count = 0;
}element;
vector<element> arr(2); // 벡터 안의 인자를 구조체로 담는 방법입니다.

int main(void) {
    for (int i = 0; i < 2; i++) {
        cin >> arr[i].str >> arr[i].count;
    }
    for (int i = 0; i < 2; i++) {
        cout << arr[i].str << " " << arr[i].count << "\n";
    }
}

```