

## #04. Structure

전미정

### 1. Structure

값 형태(자료형)의 구조로, 비슷한 속성의 여러 자료를 하나의 자료형으로 정의할 때 사용한다. 구조체를 정의하는 방식은 아래와 같다.

```

struct [structure tag]
{
    member1_declaration;
    member2_declaration;
    member3_declaration;
    ...
    memberN_declaration;
}; [one or more struct variables]
  
```

①

```

struct sam
{
    char Name[30];
    char MPhoneNum[20];
};

struct sam friend1, friend2;
  
```

②

```

struct sam
{
    char Name[30];
    char MPhoneNum[20];
};

typedef struct sam SAM;
SAM friend1, friend2;
  
```

③

```

typedef struct sam
{
    char Name[30];
    char MPhoneNum[20];
} SAM;

SAM friend1, friend2;
  
```

```

typedef struct
{
    char Name[30];
    char MPhoneNum[20];
} SAM;
  
```

일반 변수는 접근할 때 변수 이름을 그대로 사용하지만, 구조체형 변수 멤버를 사용할 때는 구조체 멤버 참조 연산자(.)를 사용한다.

## 2. Structure vs Class

### 유사점

- 구조체와 클래스 모두 container 구조로 다양한 형식을 멤버로 포함할 수 있다(생성자, 메소드, 속성, 필드, 상수, 열거형 등).
- 구조체와 클래스 모두 개별화된 액세스 수준을 가질 수 있다. (Public, Private)
- 인터페이스를 구현할 수 있으면서, 대리자를 선언할 수 있다.

### 차이점

- 구조체는 값 형식이고, 클래스는 참조 형식이다. 구조체는 자체에 데이터를 포함한다.
- 구조체는 스택 영역에 할당되고, 클래스는 힙 영역에 할당된다.
- 구조체 멤버는 기본적으로 public 이고, 클래스의 변수와 상수는 기본적으로 private 이다.
- 구조체는 상속되지 않지만 클래스는 상속된다.

## 3. Rules

- C 구조체 내부에 변수로 id type, class 를 사용할 수 없다.

```
struct X { id x; float y; };
```

=> won't compile

구조체가 해제되기 전에 참조형인 x 가 먼저 해제되는 경우가 있어 ARC mode 에서는 id 타입을 구조체에서 사용 할 수 없다. 구조체 보다 객체를 쓰는게 좋으며, 만약 객체를 써야한다면 `__unsafe_unretained` 를 써야한다.

<https://goo.gl/MsxGfO>, <https://goo.gl/N2Aq0V>