## Clss란

c++에서 클래스는 구조체의 상위 호환으로 이해할 수 있다. C++의 구조체는 멤버로 함수를 포함할 수 있어, C언어보다 확장된 의미를 갖는다. 클래스와 구조체는 둘 다 객체를 묶는 문법적인 역할을 하며, 다른 점은 접근 제어 지시자에서 차이가 난다. (뒤에서 살펴보겠다!) 클래스의 멤버 변수를 property, 멤버 함수를 method라고도 한다.

### OOP (Object Oriented Programming)

객체 지향 프로그래밍에서는 모든 데이터를 객체로 취급하며, 프로그래밍의 중심이 된다.

`객체`란, 물리적으로 존재하거나 추상적으로 생각할 수 있는 것들 중에서 자신의 속성을 가지고 있고, 다른 것과 식별 가능한 것을 의미한다. 객체의 상태(state)와 행동(behavior)을 구체화하는 것이 객체 지향 프로그래밍이다. 또한 객체를 만들어내기 위한 틀이 클래스(class)라고 볼 수 있다.

### OOP 특징

1. 추상화 (abtraction)

2. 캡슐화 (encapsulation)

3. 정보 은닉 (data hiding)

4. 상속 (inheritance)

5. 다형성 (polymorphism)

---

## Class의 선언

클래스는 추상화를 사용자 정의 타입으로 구현한 것이라고 할 수 있다. C++에서 클래스를 선언하는 방법은 C에서 구조체를 선언하는 방법과 거의 유사하다. 구조체에서 struct 키워드를 이용했지만, 클래스에서는 class 키워드와 접근 제어 지시자를 함께 사용한다.

C++에서 클래스는 아래와 같이 정의한다.

노트북, 화면, 컴퓨터이(가) 표시된 사진

자동 생성된 설명

> class: 키워드

ClassName: 클래스 이름

private ~ public: 클래스의 멤버

### 멤버 함수의 정의

클래스에서 멤버 함수를 정의하는 방법은 일반적인 함수 정의와 크게 다르지 않다. C++에서는 멤버 함수를 클래스 선언 내,외에서 모두 정의 가능하다. 클래스의 선언 밖에 멤버 함수를 정의할 때는 `범위 지정 연산자 (::)`를 사용하여 해당 함수가 어느 클래스에 속하는 지를 명시해야 한다.

> [return type] class name :: 멤버 함수 이름 (파라미터) {함수 몸체}

텍스트, 화면, 앉아있는, 모니터이(가) 표시된 사진

자동 생성된 설명

### 멤버 함수의 호출

멤버 함수는 멤버 참조 연산자(.)를 사용하여 호출할 수 있다.

```

객체이름.멤버함수이름();

객체이름.멤버함수이름(인자);

```

---

## 접근 제어

### 정보 은닉

정보 은닉이란 사용자가 굳이 알 필요가 없는 정보는 사용자로부터 숨겨야 한다는 개념이다. 이를 통해 사용자는 언제나 최소한의 정보만으로 프로그램을 손쉽게 사용할 수 있다.

### 접근 제어

C++에서 정보 은닉을 위해 접근 제어라는 기능을 제공한다. 접근 제어는 접근 제어 지시자를 사용하여 클래스 외부에서의 직접적인 접근을 허용하지 않는 멤버를 설정할 수 있도로 하여, 정보 은닉을 구체화하는 것을 의미한다.

### 접근 제어 지시자

\* public

\* private

\* protected

클래스의 기본 접근 제어 권한은 Private이며, 구조체의 경우 public이다.

#### Public

public 접근 제어 지시자를 사용하여 선언된 클래스 멤버를 외부로 공개하거나 접근할 수 있다. public 멤버 함수는 해당 객체의 private 멤버와 프로그램 사이의 인터페이스 역할을 하게 된다.

시계이(가) 표시된 사진

자동 생성된 설명

#### Private

private 접근 제어 지시자를 사용하여 선언된 클래스 멤버는 외부에 공개되지 않으며, 외부에서 직접 접근할 수도 없다. 해당 객체의 public 멤버 함수를 통해서만 접근할 수 있다.

시계, 측정기이(가) 표시된 사진

자동 생성된 설명

#### Protected

protected 접근 제어 지시자는 파생 클래스에 대해서는 public 멤버처럼 취급되며, 외부에서는 private 멤버처럼 취급된다.

protected 멤버에 접근할 수 있는 영역은 다음과 같다.

> 1. 이 멤버를 선언한 클래스의 멤버 함수

> 2. 이 멤버를 선언한 클래스의 프렌드

> 3. 이 멤버를 선언한 클래스에서 public 또는 protected 접근 제어로 파생된 클래스

지도이(가) 표시된 사진

자동 생성된 설명

---

## this 포인터

### 멤버 함수의 공유

C++에서 하나의 클래스에서 생성된 인스턴스는 각각 독립된 메모리 공간에 저장된 자신만의 멤버 변수를 가지지만, 멤버 함수는 모든 인스턴스가 공유한다.

따라서 A 클래스를 이용해 2개의 인스턴스를 생성하여, madeA()라는 멤버 함수는 1개만 존재한다. 즉, madeA()라는 함수는 2개의 인스턴스가 공유하게 된다.

### this 포인터

this 포인터는 해당 멤버 함수를 호출한 객체를 가리키며, 호출된 멤버 함수의 숨은 인자로 전달된다. 이렇게 하면 호출된 멤버 함수는 자신을 호출한 객체가 무엇인지 알 수 있다.

### 특징

\* this 포인터는 클래스, 구조체 또는 열거체 타입의 비정적 멤버 함수에서만 사용 가능하다.

\* static 멤버 함수는 this 포인터를 가지지 않는다.

\* this 포인터는 언제나 포인터 상수이며, 값을 재할당 할 수 없다.