

CPP Base

CHAPTER 6 (함수 중복과 Static 멤버)

**함수 중복

*함수 중복 (1.cpp 참고)

- 설명 : c++에서는 c와달리 같은 이름의 함수를 여러개 만들 수 있으며 이것을 함수 중복이라고 말함. 함수 중복은 다형성으로서 전역 함수와 멤버함수에 모두 적용됨 또한 ktdthr 관계에 있는 기본 클래스와 파생클래스 사이에도 허용 됨.

- 중복 함수 조건

1. 중복된 함수들의 이름은 동일
2. 중복된 함수들은 매개 변수 타입이나 매개변수의 개수가 달라야 함.
3. 함수 중복에 리턴 타입은 고려되지 않음. (리턴 타입만 다른 경우 에러난다는 소리임)

- 생성자 함수도 중복이 가능하다.

*디폴트 매개변수

-설명 : 함수가 호출될 때 매개 변수에 값이 넘어오지 않는다면, 미리 정해진 디폴트 값으로 받도록 선언된 매개 변수를 디폴트 매개변수 혹은 매개변수라고 부른다.

ex) void msg(int id, string text="Hello");

-디폴트 매개변수에 관한 제약 조건

1. 디폴트 매개변수는 항상 끝쪽에 몰려있어야 한다

ex) void (int a=5, int b) //x

ex) void (int b, int a=5) //O

즉 디폴트 매개변수는 일반 매개 변수 앞에 선언될 수 없다.

-매개 변수에 값을 정하는 규칙

1. 디폴트 매개 변수를 가지고 있는 함수의 호출문을 컴파일할 때, 호출문에 나열된 실인자 값들은 앞에서부터 순서대로 전달함.

*함수 중복의 모호성

1. 중복된 함수 중에서 참조 매개변수를 가진 함수가 있는 경우, 이들 사이에 모호성이 존재할 수 있다.
2. 디폴트 매개변수를 가진 함수가 보통 매개 변수를 가진 함수와 중복 작성될 때, 모호성이 존재할 수 있다.

* static의 특성

- 설명 : static은 변수와 함수의 생명 주기와 사용 범위를 지정하는 방식중 하나로서, static으로 선언된 변수와 함수의 생명주기와 사용 법은 다음과 같음.

1. 생명주기 : 프로그램이 시작할 때 생성되고 프로그램이 종료할 때 소멸
2. 사용범위 ; 변수나 함수가 선언된 범위 내에서 사용. 전역 또는 지역으로 구분

-c++에서 클래스 멤버 변수와 멤버 함수를 포함하여 모든 변수와 함수는 static 지정자로 선언 가능.

-보통 변수와 함수의 static선언은 c언어와 동일함.

-static 멤버 변수는 외부에 전역 변수로 선언 되어야 한다.(별도로 전역공간에서 변수의 공간을 할당받는 선언문이 추가적으로 필요하다.)

-static 멤버들은 객체가 생기기 전부터 사용이 가능하다. 클래스명과 범위지정 연산자를 이용하면, 객체를 언급하지 않고도 static멤버를 활용할 수 있다.

*static의 활용

1. 객체 지향 언어에서 추구하는 핵심 가치는 ‘캡슐화’이다. 하지만 c++은 c언어와의 호환성 때문에 100% 캡슐화하지 못한다. 따라서 우리는 c++에서 전역변수와 전역함수를 사용하지 않고 클래스의 static멤버로 선언하여 캡슐화를 진행하여야 한다.

2. 객체 사이에 공유 변수를 만들고자 할 때

static 멤버는 클래스의 모든 인스턴스(객체) 들이 공유하는 변수나 함수를 만들고자 할 때 사용된다.

3.static 멤버 함수는 오직 static 멤버들만 접근이 가능하다.

4.static 멤버 함수는 this를 사용할수 없다. 즉 static 멤버함수는 객체가 생기기 전부터 호출이 가능하기 때문에 this를 통해 가리킬 수 없다.