state 는 변수, behavior는 함수라고 생각하면 돼

c에서 structure 만들었던 것처럼. 클래스와 객체

oop에서는 data도 만들고 함수도 원하는 함수를 만들어서 객체를 구성할 수 있다.

다음주 휴강이라 오늘 3, 4주차 내용 압축적으로 다룰 예정

c++

sample obj

호출하기 위한 메시지 전달. 전달될 객체, 함수 이름, 함수 수행 시 필요한 인자

scanf와 printf를 똑같이 쓸 수 있음.

대신 % 안쓰고 >>, <<로 cin, cout 쓸 수 있음. 꺾쇠 두개

malloc 쓰지 않고 new와 delete 이용

물론 malloc 쓸 수도 있음.

malloc의 단점은 동적 할당을 할 때 어떤 타입에 대해 할당할 지 명시를 해주어야 한다.

new를 쓰게 되면 그런 거 없이 자동으로 integer 포인터를 할당할 수 있다.

주의) c++에서 new로 할당할 때 c에서 쓰는것처럼 free로 막 섞어서 쓰면 안됨.

c++에서 되게 유용한 기능 : 참조 연산자.

-> 뒤로 가서 보도록 하겠다. p35.

원래 c에서는 아무 값도 가지지 않는다는 의미로 NULL을 사용

c++에서는 nullptr라고 써야 한다.

c에서는 같은 이름의 함수가 있으면 안됨.

c++에서는 같은 이름의 함수가 있어도 컴파일하게 해줌.

lvalue 가 있고 rvalue 가 있는데 컴실에서는 lvalue를 사용한다

lvalue가 하는 역할은 포인터 역할을 대신 해주는 것이다. lvalue 데이터들을 가리킬 수 있는 기능.

레퍼런스(&)를 붙여서 사용할 수 있다.

vector를 처음 보실 수도 있는데..

pass by value

pass by reference 두 가지가 c에 있었는데 엄밀히 말하면 c에는 pass by value 밖에 없는거야.

(p39 보고 있다)

다시 앞으로 돌아와서

뭐라뭐라 하다가ㅏㅏㅏ

상속.

로욜라에서 이용하는 코드를 만들어보자

학생, 교직원, 대학원, 교수에 따라 빌릴 수 있는 권수, 대출 기한 등 각각의 특성이 다를 수 있겠지. 각각의 클래스를 따로따로 만들면 아주 귀찮아지지. 학생도 책을 빌릴 수 있고, 교직원도 빌릴 수 있고, 대학원생도 빌릴 수 있지. 똑같은 기능을 하는 코드인데 그걸 반복적으로 돌려야 하기 때문이야. 모두 ‘책을 빌릴 수 있는 사람’이라는 공통된 기능이 있지. 그렇기 때문에 상속을 이용한다.

코드의 유지보수 차원에서 생각하면, 학생 한 명에게 문제가 생겼으면 모든 사람들에게 그 문제를 해결해야 하지만, 상속을 이용했다면 학생에 대해서만 그 문제를 해결하면 된다.

암튼 그래서 객체지향에서 상속이란 기능이 아주 중요하다.

다중정의

함수 이름이 같아도 인자의 개수나 자료형이 다른 함수를 만드는 걸 c++에서는 지원을 한다.