**람다와 함수 포인터의 차이**

C의 함수 포인터는 선언된 위치나 선언 방식과 관련없이 어떤 맥락에서든 함수의 매개변수로 활용 가능한 반면 람다는 구체적인 인터페이스의 형태를 띄어야만 사용 가능하다. 또한 람다는 객체화한 메서드를 사용하기에 객체의 범위가 가지는 변수를 활용할 수 있다.

또한 C의 함수 포인터가 malloc을 통해 메모리를 직접적으로 할당해서 변수에 대입하지 않는 이상 C 함수 포인터의 지역 변수는 스택 메모리를 차지했다가 함수가 종료되는 시점에 자동으로 소멸된다. 하지만 람다의 변수는 객체가 후에도 객체가 활용된다면 힙 메모리에 남아있다.

람다 표현식

Myinterface myLambda = ClassName::methodName;

추상 매서드에 매개변수가 하나 뿐이거나 없는 경우 매개변수 생략해서 사용 가능

Or

interface Myinterface {

void mymethod(String s);

}

Myinterface myLambda = (s) -> System.out.println(s);

인터페이스에 매개변수가 여러 개로 정의되어 있거나 하나를 특정해서 람다식으로 표현하고 싶을때(s)