

Java 8 - Lambda Expression I

우리는 Java를 배우면서 열심히 객체지향의 세계에서 해엄쳐 왔다.

Java는 객체지향 프로그래밍의 대표 주자인데, 함수형 프로그래밍을 JDK 8에서 도입하였다.

그렇다면 함수형 프로그래밍 (Functional Programming)은 무엇이며,

람다식 (Lambda Expression)이라는 것은 무엇일까?

수학 - 함수 (Function)

필자는 질풍노도의 학창시절을 보냈고, 수학과 거의 담을 쌓은 수포자였다.

수학과 완전 담 쌓아뒀던 분들은 함수에 대해서 이해할 필요가 생겼으므로

중,고등과정에서의 함수를 살펴보자.

• 대응 (Mapping) ->

학생 집합 X		
학생 a	학생 b	학생 c
김밥	떡볶이	라면
분식집 메뉴집합 Y		

- 학생 a -> 김밥
- 학생 b -> 라면
- 학생 c -> 떡볶이

집합 X 의 원소 x 가 집합 Y 의 원소 y 를 선택하는 것을 **대응 (Mapping)**이라고 한다.

• 변수

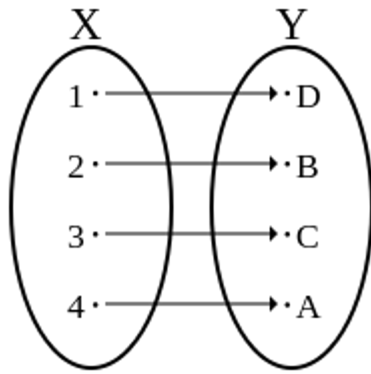
집합 X 의 **원소 x**

집합 Y 의 **원소 y**

- 원소 x 에 따라 대응되는 y 원소가 바뀐다.
- **값이 변하는 수**

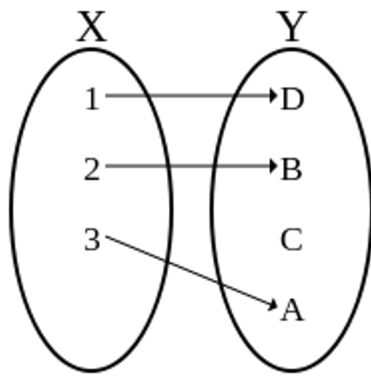
• 함수 (Function) **$f(x)$**

i. 집합 X 의 모든 원소는 반드시 집합 Y 의 원소를 대응해야 한다.

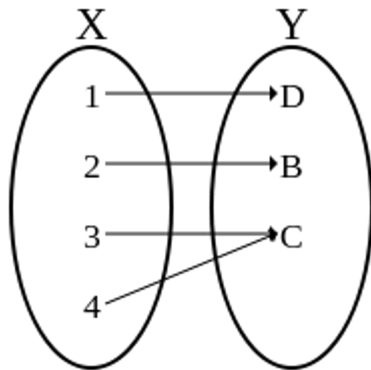


(가장 이상적인 함수 모델)

- ii. 집합 X의 원소는 반드시 집합 Y의 하나의 원소만 선택한다.
- iii. 대응되지 못한 집합 Y의 원소가 있더라도, 집합 X의 원소가 모두 대응되는 관계에 있다면 함수

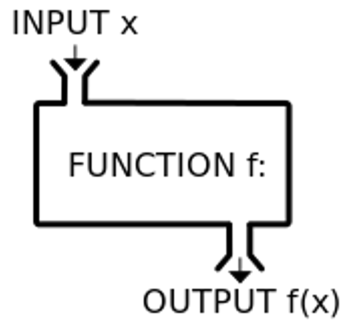


- iv. 집합 Y의 원소는 둘 이상의 집합 X의 원소에 대응될 수 있다.



○ 위의 모든 조건을 만족하면 함수가 된다.

- 집합 X, Y에서 집합 X의 원소가 집합 Y에 하나씩 대응되는 관계
- 입력 x가 주어지면 함수 f로 인해 결정된 출력 y가 나오게 하는 식

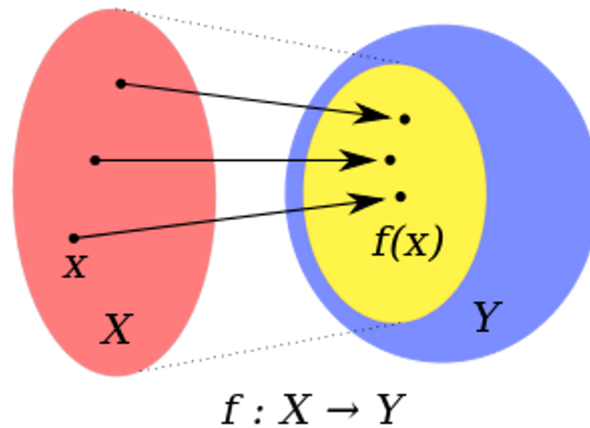


$$f : X \rightarrow Y$$

- 함수 f : (는) 집합 X 에서 집합 Y 로 가리키는(\rightarrow) 함수이다.
- 함수 f 로 인해 X 가 Y 를 가리키게 된다.

$$y = f(x)$$

- 함수 f 가 적용된 x 에 대응되는 y 값, x 의 함수값은 y 이다.



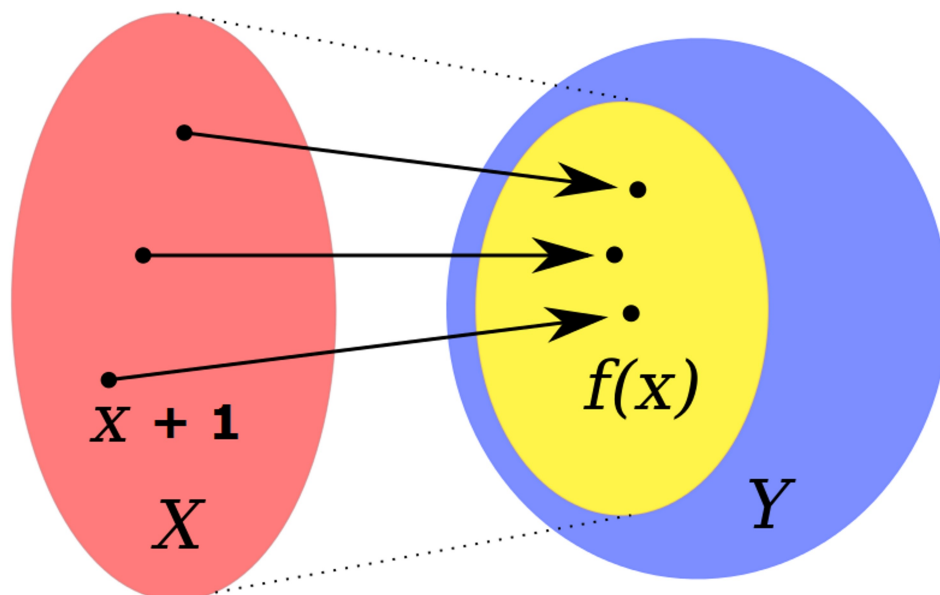
- 정의역 X : 함수를 정의해주는 집합
- 공역 Y : 정의역의 원소에 대응되는 원소가 모인 집합
- 치역 : 공역의 원소들 중에서 정의역 원소에 대응되는 원소만 모아놓은 집합

• 함수 정의하기

$$y = x+1$$

$$f(x) = x+1(\text{함수의 정의})$$

$$f : x+1 \rightarrow y$$



$$f : X \rightarrow Y$$
