# 제10주 실습과제, 람다

1. [java.util.function.Consumer](https://docs.oracle.com/en/java/javase/17/docs/api/java.base/java/util/function/Consumer.html) 인터페이스에 대한 학습입니다.

우선 아래 api 문서를 잘 읽어 보세요.

Graphical user interface, text

Description automatically generated with medium confidence

T – 연산을 위한 입력의 타입.

Consumer은 입력 T를 받아(accept), 그것을 가지고 뭔가 연산을 하고, 아무것도 반환하지 않는 놈이다.

Graphical user interface, text, application

Description automatically generated

“이 Consumer의 연산에 이어 after 연산을 수행하는” 복합 Consumer를 반환한다.

주어진 인자 t를 받아 연산을 수행한다.

[아래 프로그램](src/ConsumerDemo.java)은 java.util.function.Consumer의 사용법을 보여줍니다. 주석 형태로 설명이 적혀 있습니다. 설명과 코드를 잘 읽어 보고 프로그램을 실행하면 어떤 출력이 나올지 예측해 보고, 실행하여 확인해 보세요.

Text

Description automatically generated

ConsumerTest라는 이름의 클래스를 만들고 main 메소드에 아래 코드를 적으세요.

* 키보드로 입력한 여러 개의 String들을 리스트에 저장한다. 사용자는 한 줄에 하나의 String을 입력하며 마지막에는 EOF(윈도의 경우 control-z)로 입력을 마치거나, 빈 줄을 입력하여 입력을 마친다.
* 주어진 String의 첫 글자를 화면에 출력하는 Consumer 람다식을 작성한다.
* 주어진 String의 마지막 글자를 화면에 출력하는 Consumer 람다식을 작성한다.
* 주어진 String의 첫 글자를 화면에 출력한 후, 마지막 글자를 화면에 출력하는 복합 Consumer 람다식을 위 두 Consumer 람다식을 조합하여 작성한다.
* 첫 consumer 람다식을 이용하여 리스트의 첫 String을 처리한다. 그리고 나서 새줄문자를 줄력한다.
* 두 번째 consumer 람다식을 이용하여 리스트의 첫 String을 처리한다. 그리고 나서 새줄문자를 줄력한다.
* 세 번째 consumer 람다식을 이용하여 리스트의 첫 String을 처리한다. 그리고 나서 새줄문자를 줄력한다.
* 리스트의 각 String에 위에서 작성한 첫 번째 Consumer 람다식을 적용한다.

출력은 아래와 같게 하세요.

A screenshot of a computer

Description automatically generated A screenshot of a computer

Description automatically generated

1. [java.util.function.Function](https://docs.oracle.com/en/java/javase/17/docs/api/java.base/java/util/function/Function.html) 인터페이스에 대한 학습입니다. 우선 아래 설명서를 잘 읽어보세요.

Graphical user interface, text, application

Description automatically generated

Graphical user interface, text, application

Description automatically generated

[아래 프로그램](src/FunctionDemo.java)은 java.util.function.Function의 사용법을 보여줍니다. 주석 형태로 설명이 적혀 있습니다. 설명과 코드를 잘 읽어 보고 프로그램을 실행하면 어떤 출력이 나올지 예측해 보고, 실행하여 확인해 보세요.

A screenshot of a computer program

Description automatically generated

FunctionTest라는 이름의 클래스를 만들고 main 메소드에 아래 코드를 적으세요.

* 주어진 double 값의 제곱근을 구하는 Function 랍다식을 작성한다. 제곱근을 구할 때는 Math.sqrt(double) 함수를 사용한다.
* 주어진 double 값을 반올림한 정수를 구하는 Function 람다식을 작성한다. 반올림 값을 구할 때는 Math.round(double) 함수를 사용한다.
* 주어진 double 값의 제곱근을 반올림한 값을 구하는 복합 Function 람다식을 위 두 Function 람다식을 조합하여 작성한다.
* 키보드로 0.0 이상 소수 하나를 double 타입으로 입력받는다.
* 복합 function 람다식을 이용하여 입력받은 숫자의 제곱근을 반올림한 값을 구하여 출력한다.

출력은 아래와 같게 하세요.









위 마지막 예의 입력은 Double.MAX\_VALUE로서 1.7976931348623157E308입니다.

1. 주어진 [Lambdas.java](src/Lambdas.java)와 [Comparators.java](src/Comparators.java)의 코드 중 미완성인 부분을 채워 넣는 과제입니다.

이 실습과제에는 자기 테스트 기능이 들어 있어 코드 작성 후 자동으로 테스트를 할 수 있습니다.

Lambdas.java에는 22개, Comparators.java에는 10개의 문제가 있고 각 문제에는 테스트 코드가 포함되어 있습니다.

각 문제는 하나의 메소드로 만들어져 있고 메소드 바로 위에 “@Test” 아노테이션(표식)이 붙어 있습니다.

테스트는 언제든 실행할 수 있고, 반복해 실행할 수 있습니다.

각 문제에 포함되어 있는 테스트 코드를 통과하면 문제를 제대로 푼 것으로 일단 간주합니다. 그러나 테스트 코드가 아주 단순한 몇 가지 경우만 테스트하도록 작성되어 있기 때문에 테스트를 통과한다고 해서 문제를 제대로 푼 것으로 확신할 수는 없습니다.

테스트를 실행하기 위해서 JUnit을 사용합니다. 만약, 이클립스에 JUnit이 설치되어 있지 않다면 이 문서의 맨 아래에 있는 절차를 따라 JUnit을 이클립스에 설치하세요.

프로젝트 아래에 w10 패키지를 만들고 [Lambdas.java](src/Lambdas.java)와 [Comparators.java](src/Comparators.java), [Person.java](src/Person.java) 파일을 w10 패키지로 복사해 넣으세요. Person은 Lambda와 Comparators에서 사용하는 클래스입니다.

파일의 각 문제는 영어와 한글 두 가지 언어로 적혀 있습니다. 각 문제의 지시에 따라 아래 그림의 null 부분에 코드를 적고 // TODO는 지우세요.



언제든지 패키지 익스플로러 창의 Lambdas, 혹은 Comparators를 마우스 오른쪽 클릭하여 아래 그림처럼 JUnit Test를 실행할 수 있습니다.



문제를 풀기 전에 Junit을 실행하면 22개의 문제에 해당하는 각 메소드 이름이 보이고 테스트 실패를 의미하는 빨간 x가 보입니다.

Table

Description automatically generated

문제를 제대로 다 풀고 나면 아래처럼 됩니다.

Graphical user interface, text, application, table

Description automatically generated

실습을 마친 후에는 Lambdas와 Comparators를 강온실 w10p3\_Lambdas와 w10p4\_Comparators에 제출합니다.

강온실 w10p3\_Lambdas 문제에는 Lambdas 클래스를 제출하세요.

Lambdas 클래스를 제출할 때는 아래 import 문장들만 포함시켜 제출하세요.

import java.util.ArrayList;

import java.util.Arrays;

import java.util.List;

import java.util.function.BiFunction;

import java.util.function.Consumer;

import java.util.function.Function;

import java.util.function.Predicate;

import java.util.function.Supplier;

강온실 w10p4\_Comparators 문제에는 Comparators 클래스만 제출하면 됩니다.

Comparators 클래스를 제출할 때는 아래 import 문장들만 포함시켜 제출하세요.

import java.util.Comparator;

import java.util.function.IntBinaryOperator;

**<JUnit 설치 방법>**

패키지 익스플로러 창에서 프로젝트 이름을 마우스 오른쪽 버튼 클릭하여 아래와 같이 선택

Graphical user interface, text, application

Description automatically generated

나타나는 창에서 JUnit 선택하고 추가로 나타나는 창에서 JUnit5를 선택하고 Finish.

아래와 같이 보이면 설치가 된 것입니다.

Graphical user interface, text, application

Description automatically generated

끝.