

---

## Sean Dustine M. Genove

seanygenove@gmail.com  
0935-375-7297

# Thermometer Application

21<sup>st</sup> June 2020

## OVERVIEW

This application simulates a simple thermometer. The temperature is read from the user's input as well as thresholds such as the boiling point, freezing point, and insignificant fluctuation value. Notifications are logged in the console whenever the temperature reaches the boiling or freezing point thresholds.

The code base is written in Java and guided by the SOLID principles. The application is implemented using the Observer design pattern. JavaDoc was utilized to generate HTML documentation files and Junit as the testing framework for the application.

## DESIGN PATTERN

The Observer pattern is used for a one-to-many relationship between objects such as if one object is modified, its dependent objects are to be notified automatically. It uses two actor classes. The *Subject* and *Observer*. Subject is an object having methods to attach and detach observers to a client object (observers).

The application implements the same concept. Having the Thermometer as the Subject and two observers for monitoring whether the current temperature has reached the boiling/freezing point thresholds.

## ASSUMPTIONS

### Thermometer class

1. Implements the Subject interface
2. Object instances are able to register observers.
3. For every property change, the thermometer object instance sends its latest properties to its observers.

### Boiling Point and Freezing Point Observers

1. Indicator for when the current temperature has reached its threshold.
2. Two types of observers.
  - a. *BoilingPointObserver*
  - b. *FreezingPointObserver*

## CONSIDERATIONS

Notifications are only sent if conditions below are satisfied.

1. Current temperature is at or beyond threshold.
2. Previous temperature was NOT at or beyond/below the thresholds.
3. Current temperature did not fluctuate from a previous temperature within the thresholds insignificant fluctuation range.

\* *Threshold pertains to boiling point or freezing point*

\* *Insignificant fluctuation range is from the threshold and its +/- from the given insignificant fluctuation value*

## LIMITATIONS

1. All thermometer property values are in Java's *double* primitive data type.
2. Thermometer observers are given an immutable hashmap containing the latest thermometer properties.
3. User input is collected in the main method.

## CODE DOCUMENTATION

Code documentation is provided in the code base. Alternatively, a web version of the documentation is provided in the *doc* directory of the application. JavaDoc was utilized in creating the documentation.

## TESTING

JUnit was utilized as the testing framework for the application. Tests are located in the *test* directory of the application and are composed of the following files

TEST FILE NAME	NO. OF TESTS
BoilingPointObserverTest	9
FreezingPointObserverTest	9
ThermometerTest	17

All test cases have 100% coverage for their respective class files.

## FUTURE IMPROVEMENTS

1. Observer classes can later be refactored into concrete objects. Examples of these would be a Kettle class for observing the boiling point and a Refrigerator class for observing the freezing point.
2. User interface portraying a virtual thermometer simulating readings on the user's inputted temperature.