

Activity 1: Enhancing a Weather Station

- **Type:** Group assignment
- **Due:** Next class day (check MyCourses Assignments for exact time)
- **Deliverables:** Zip file with the source code
- **CLO:** Part of CLO1 (3.75 pts/20 pts).

Introduction

You are working for a firm that is entering the weather station market. As its first offering, the firm has created a prototype for a text-based application that creates and starts an independent Java Thread to periodically read a temperature sensor and print the retrieved value in Celsius. The station is represented by a **WeatherStation** object, while the sensor is an object of class type **TemperatureSensor** which, for the demonstration purposes, simply simulates the behavior of a sensor available from a local electronics supply house.

The simulated sensor has a **reading()** method which returns a 16-bit integer, a value that ranges from 0 through 65535 (the range of values that can be held in 16 bits), representing the temperatures 0.00°K (absolute zero) through 655.35°K. The **WeatherStation** object converts the Kelvin temperature to Celsius, using the fact that 0°C is 273.15°K, and prints the result as a string to the output window or terminal. The current temperature is reported once per second on a separate line as follows:

Reading is *nnn.nn* degrees C

where *nnn.nn* is the actual numeric temperature. See the source in [WeatherStation.java](#) for details.

Support Files

In this zip you will find:

1. [TemperatureSensor.java](#)
The simulated sensor object class – do not change this!
2. [WeatherStation.java](#)
The **WeatherStation** class, which contains the

```
static public void main(String args[])  
  
method.
```

Compile and execute the Java source files to see the program in action:

```
javac WeatherStation.java  
java WeatherStation
```

Your Task

Update the program to add Kelvin temperature printing. Each reading still prints one line, but updated to the following form exactly:

Reading is *nnn.nn* degrees C and *mmm.mm* degrees K

where *nnn.nn* is as before and *mmm.mm* is the corresponding Kelvin temperature.

When you've completed the task:

- Create a zip archive named **activity1.zip** that contains:
 - Your updated **WeatherStation.java**.
 - The original **TemperatureSensor.java**.
- Deposit the archive in the *Activity #1 – Enhancing the Weather Station* assignments folder by the due date as specified on the assignments folder.

Grading

15 points. There is no partial credit – the solution is working (15 pts) OR not (0 pts).