To practice implementing and testing a class.

Weather

Often when you go outside, it feels much warmer or colder than the weatherman predicted. This is not a conspiracy. It is simply that the predicted temperature is not always what the real world *feels like*. Most weather channels and meteorology apps have started using this term to give folks a better understanding of how they will actually feel out in their environment. They do this by collecting readings from *Stevenson Screen* shelters.

A Stevenson Screen is a standard shelter that protects meteorological instruments which measure things like temperature, dew point, wind speed, rain, etc. These readings are used to calculate the different temperatures that appear in your weather report. For instance:

* **Dew point** is the temperature at below which water droplets (or dew) begin to form. The relationship between dew point (D), temperature in degrees Celsius (T), and relative humidity in percentage between 0 and 100 (R) is expressed in the following formula:

D=T−100−R5D=T−100−R5

* **Heat index** is a measure of how hot it feels when relative humidity is factored in with the actual temperature. It is calculated in a number of different ways but for our example, consider the formula for heat index (HI):

HI=c1+c2T+c3R+c4TR+c5T2+c6R2+c7T2R+c8TR2+c9T2R2HI=c1+c2T+c3R+c4TR+c5T2+c6R2+c7T2R+c8TR2+c9T2R2

where T is the temperature in degrees Celsius, R is the relative humidity in percent, and the coefficients are: c1 = -8.78469475556, c2 = 1.61139411, c3 = 2.33854883889, c4 = -0.14611605, c5 = -0.012308094, c6 = -0.0164248277778, c7 = 0.002211732, c8 = 0.00072546, and c9 = -0.000003582.

* **Wind chill** is related to heat index and is used when the real-feel temperature is lower than the actual temperature. There is some variation on how it is calculated depending on where you are but here in the United States, it is calculated with this formula:

WC=35.74+0.6215T−35.75v+0.16+0.4275Tv+0.16WC=35.74+0.6215T−35.75v+0.16+0.4275Tv+0.16

where WC is the wind chill based on the air temperature in degrees Fahrenheit (T) and the wind speed in miles per hour (v).

What to do

**Package:** weather

Design and implement a class called WeatherReading that represents a single reading of a weather station in a Stevenson Station. Your constructor takes four parameters: the air temperature in Celsius, the dew point temperature in Celsius which cannot be greater than the air temperature, the non-negative wind speed in miles per hour, and the non-negative total rain received in the last 24 hours in millimeters. It should throw an IllegalArgumentException for invalid values.

Methods that get different values from an object are called *accessor methods* or *getters* because they are named starting with "get". Your implementation should include each of the following:

* getTemperature
* getDewPoint
* getWindSpeed
* getTotalRain
* getRelativeHumidity
* getHeatIndex
* getWindChill

Do not forget that every well-written class in Java should also include a toString method. Your implementation should create an output like this one:

Reading: T = 23, D = 12, v = 3, rain = 12

Testing

Whenever you write a class, you should also write tests for that class that proves not only that your code CAN work but that it WILL ALWAYS work. Your goal here is to write tests that achieve as close to 100% coverage as possible. To learn why test coverage is an important part of software development, you can read [this article by Hardik Shah (Links to an external site.)](https://www.simform.com/test-coverage/#:~:text=Test%20coverage%20is%20defined%20as,then%20test%20coverage%20is%2090%25.). In this class, we will measure coverage using Eclipse's "Coverage As ..." feature (right-click on the test/ directory to find this option).

Even more important than test coverage is that your tests should be written in such a way to ***convince someone else that your code works correctly.***