

Exercises for Chapter 5 (Week 8)

Introductory Programming 2020

Exercises

5.3

Read through the full source code of the `AnimalMonitor` class to understand how the methods work. The class only uses techniques we have covered in previous chapters, so you should be able to work out the details.

5.4

Why does the `removeZeroCounts` method use a while loop with an iterator, instead of a for-each loop as used in the other methods? Could it be written with a for-each loop? Could the other methods be written using a while loop?

5.6

Modify your own `printList` method as discussed here: try out all the syntax variations for lambdas that we have shown. Recompile and test the project with each variation to check that you have the syntax correct.

5.11

Write a method in the `AnimalMonitor` class to print details of all the sightings recorded on a particular `dayID`, which is passed as a parameter to the method.

5.15

Write a method to print the counts of all sightings of a particular animal. Your method should use the `map` operation as part of the pipeline.

5.16

If a pipeline contains a `filter` operation and a `map` operation, does the order of the operations matter to the final result? Justify your answer.

5.17

Rewrite the `printEndangered` method in your project to use streams. Test. (To test this method, it may be easiest to write a test method that creates an `ArrayList` of animal names and calls the `printEndangered` method with it.)

5.20

Add a method to `AnimalMonitor` that takes three parameters: `animal`, `spotterID`, and `dayID`, and returns a count of how many sightings of the given animal were made by the spotter on a particular day.

5.23

Write method `removeSpotter` that removes all records reported by a given spotter.