Introduction to Java Streams

In this recitation you will gain experience using streams and other functional features introduced in Java 8.

In `StreamPractice.java`, we have provided many methods that are written without using Java streams. Your task is to re-write each method to be functionally equivalent to the method we provided, but using Java streams. For each problem we have provided an empty method stub, and we have included one example (#0: `lineCount`) to help you get started. Most of these methods can be rewritten using a single streams expression. Discuss the advantage and disadvantage of your new code compared with the implementation we have provided.

Infinite streams

Java streams are lazily evaluated: operations on a stream are not executed until the terminal result of the stream is used. Lazy evaluation has several advantages, including allowing the processing of infinitely long streams. Although some operations on an infinite stream would run forever, other operations can return a result without traversing an entire stream, and therefore still work on infinite streams.

Some of the problems in `StreamPractice.java` require you to generate and use an infinite stream to generate and process Fibonacci numbers.