

File I/O

Make a new Eclipse project and copy the enable1 word list from the file-io project to the top-level folder of your new project. Except for problem #3, you may assume that the enable1 word list is all in lower case. Recall from the lecture that the usual best real-life approach is to break your file reading into two pieces: one reusable method that takes a `Stream<String>` and a second method that reads the file and passes the `Stream` to the first method. But, for simple testing, you can just put everything in main:

```
public static void main(String[] args) throws Exception {
    String result = Files.lines(...)
        .map(...)
        .filter(...)
        .findFirst()
        .orElse("No result found");
    System.out.println(result);
}
```

1. Print out the first 10-letter word found in the file. Print out the first 6-letter word that contains “a”, “b”, and “c”.
2. Repeat the previous problem, but handle the possibility of mixed-case words in the file. Hint: do something shorter than merely modifying your filter tests (to include “A”, “B”, and “C”).
3. Define a static method `isOoWord` that returns true only for words that have at least two consecutive o’s. Given that method, print out the first word that has 6 or more letters, contains a “b”, and is an oo Word.
4. Make a file called “twitter-words.txt” that contains all words from the enable1 list that contain “wow” or “cool”. The words should be sorted, in uppercase, and have an exclamation point at the end. (E.g., “COOLER!”).
5. Print out the number of files in your project. Folders count as files.
6. Create a file containing 17 random doubles between 0 and 100, each with exactly three digits after the decimal point.