



# Highnote

## Live Coding Prompt: Transaction Processor

Thank you for your interest in Highnote! We are looking forward to working with you and are excited about the opportunity for you to join Highnote. At Highnote, when you interview with our team, our goal is to support you throughout the process and set you up for success. Please do not hesitate to reach out to your recruiter regarding preparation, questions, or concerns.

**Here are the set up directions for your 1 hour Live Coding Session.** You will be given the prompt by a Highnote Engineer at the beginning of your session and will have some time to plan your approach. **At a high level, the exercise will require you to parse lines of transaction messages from an input file and perform some business logic operations based on each message.**

### Setup Instructions:

- During your interview, you can be expected to utilize your favorite search engine and libraries (within reason) for help. If you're unsure about something, please ask your interviewer for clarification.
- **You must implement this test in Java.** Please come to the interview with your favorite IDE already set up and ready to run unit/integration tests in Java. eg. JUnit tests.
- **Please have a new Java project already set up, so you can start filling in the implementation right away.** Your environment should be ready to run unit tests. It may be helpful to have a method to load input text files for integration testing.
  - For example, please have something like this scaffolded out:

Java

```
// Main Class  
public class TransactionProcessorApp {
```



# Highnote

```
public void processTransactions(String[] transactions) {
    System.out.println("Starting Transaction Processor:");
    System.out.println("Start Inputs:");
    for (String transaction: transactions){
        System.out.println(transaction);
    }
    System.out.println("End Inputs:");

    // TODO: Implement
}

// Test Class
class TransactionProcessorAppTest {

    @Test
    void testProcessor() {
        String[] inputFileLines = loadFileLines();
        TransactionProcessorApp app = new TransactionProcessorApp();
        app.processTransactions(inputFileLines);
    }

    private static String[] loadFileLines() {
        try {
            BufferedReader in = new BufferedReader(
                new FileReader("src/main/resources/input.txt"));

            String str;

            List<String> list = new ArrayList<String>();
            while((str = in.readLine()) != null){
                list.add(str);
            }
            String[] stringArr = list.toArray(new String[0]);
        }
    }
}
```



# Highnote

```
        return stringArr;
    } catch (IOException e) {
        System.out.println("Could not load file." + e.getMessage());
    }
    return null;
}
}
```

## Topics for Discussion and Evaluation for the Coding Session:

In order to prepare for your interviews, we wanted to share some topics you should be ready to discuss and what we will be assessing:

- **Code cleanliness**
  - As you make changes to your code to satisfy requirements, we will look into how you scaffold your code, name components, and whether you do proper refactoring. Remember, we are simulating a working environment, so we are looking for production-like code.
- **Testing**
  - Consider how you might design your code to be testable, as you update your code to satisfy the requirements, we will expect you to test and debug your code. Ensure that you prepare your coding environment (IDEs, etc) so that you can easily add new tests and debug your code.
- **Functionality**
  - Your implementation should be correct and the design should be demonstrated by testing your code. Any assumptions you make that impact the result should be shared with the interviewers.
- **Language Proficiency**
  - We expect that you can fluently code in this session. You should be able to use basic language features to complete the work. During the session, if you need to look up additional information on the internet, you will have the opportunity to do so.
- **Performance**
  - We need you to be aware of the performance on the code you are working on and may be asked performance related questions.