HW 09: Delimited Data

Release date: Friday, November 4th, 2016
Due date: Friday, November 11th, 2016 – 11:59 pm

Goals

- Learn to read files
- Learn to create, throw and handle exceptions
- Use String manipulation to parse file contents

Description

Consider a Database application program that accepts files from users that contain simplified SQL (Structured Query Language) queries and runs such queries on the database. The file should contain information about the user who wants to submit the queries and the actual query strings. Query strings start with either "SELECT", "UPDATE", "INSERT", or "DELETE". The file has a specific format which is described as follows.

```
C
<username>
c
N
<some number>
n
Q
<Query1>
<Query2>
...
<Queryn>
q
```

The username is specified between lines **C** and **c,** the number of queries is specified between lines **N** and **n** and the list of queries is specified between lines **Q** and **q:** one query per line. You should only care about the beginning of the String.

You are required to implement a parser for such file taking into account possible exceptions that could occur. You should implement and account for the following exceptions:

1. `WrongFileFormatException`: This exception is thrown when the format of the file is not correct. This is when the delimiters are not placed properly.
2. `InvalidInputException:` This exception is thrown when the data that is delimited in the second section of the file (between N and n) is not an integer number 1 or larger.
3. `WrongNumberOfQueriesException:` This exception is thrown when the number of queries in the third block does not match the number provided in the second block.
4. `MalformedQueryException:` This exception is thrown when the query is not a valid SQL command type (see above).
5. `IOException:` (**NOTE:** You do not need to implement this class. You should import it from the Java Library.) This exception is thrown if opening or reading from the file fails.

Implement the following methods in a class `Parser`:

1. `public void parse(String filePath) throws WrongFileFormatException, WrongNumberOfQueriesException, InvalidInputException, MalformedQueryException, IOException`
2. `public String getUserName()`
3. `public int getNumQueries()`

Hint: Create instance variables to store the username and queries during a call to `parse` and return them in the getter methods.

Submission

Submit all your files to Vocareum by following these instructions. Keep in mind that there is a limit of 20 submissions and only your last submission will be considered.

Rubric:

- getUsername() returns correct username after a call to parse: 15%
- getNumQueries() returns correct number after a call to parse: 15%
- parse throws `IOException:` 10%
- parse throws `WrongFileFormat:` 20%
- parse throws `MalformedQuery:` 10%
- parse throws `InvalidInput:` 10%
- parse throws `WrongNumberOfQueries:` 20%