CS410J Project 2: Storing Your Phone Bill in a Text File (9 points[1])

In this project you will write classes that store your phone bill and its phone calls in a text file using a format of your own design.

Goals: Learn how to use Java's I/O facilities and throw exceptions.

For this project you will have to implement the following three classes[2] in your `edu.pdx.cs410J.`*your-login-id* package:

- Class `TextDumper` that implements the `edu.pdx.cs410J.PhoneBillDumper` interface. A `TextDumper` dumps the contents of a phone bill[3] (including its calls) to a text file.
- Class `TextParser` that implements the `edu.pdx.cs410J.PhoneBillParser` interface. A `TextParser` reads the contents of a text file and from it creates a phone bill with phone calls.
- Class `Project2` that contains a `main` method that optionally reads a `PhoneBill` from the contents of a text file, creates a new `PhoneCall` as specified on the command line, adds the `PhoneCall` to the `PhoneBill`, and then optionally writes the `PhoneBill` back to the text file. `Project2` should have the following command line interface:

```
usage: java edu.pdx.cs410J.<login-id>.Project2 [options] <args>
  args are (in this order):
    customer               Person whose phone bill we're modeling
    callerNumber           Phone number of caller
    calleeNumber           Phone number of person who was called
    startTime              Date and time call began (24-hour time)
    endTime                Date and time call ended (24-hour time)
  options are (options may appear in any order):
    -textFile file         Where to read/write the phone bill
    -print                 Prints a description of the new phone call
    -README                Prints a README for this project and exits
  Dates and times should be in the format: mm/dd/yyyy hh:mm
```

If the text file does not exist, then the program should create an empty `PhoneBill`. It should **not** issue an error. Phone numbers, dates, and times are specified using the same format as in the previous assignment.

[1] 8 for code, 1 for POA

[2] Make sure to submit **all** of your classes, even if you haven't modified them since the previous project.

[3] Note that the `dump` method dumps an `AbstractPhoneBill`. You cannot change this fact. This is okay because your `PhoneBill` **is an** `AbstractPhoneBill`.

Error handling: Your program should exit "gracefully" with a user-friendly error message under all reasonable error conditions. Examples of such conditions include

- Something is missing from the command line or there are extraneous command line arguments
- The format of the day or time is incorrect
- The text file is malformatted
- The customer name given on the command line is different than the one found in the text file.