COSC 1P03 Assignment 5

"Time to Pay the Piper"

Due: Apr. 7, 2016 @ 4:00 pm ***(late date Apr. 10 @ 4:00 pm)***

In preparation for this assignment, create a folder called `Assign_5` for the DrJava project for the assignment. The objective of this assignment is to apply the Software Development Process described in class to produce a system for the following problem.

Problem

Sharkey's Loans is a "lender of last resort" specializing in high risk loans. Sharkey is a member of a business consortium which enforces the prompt payment of the minimum balance each month for each loan customer. Due to changing market conditions, the business people in the consortium frequently change the interest rates applied to the loans. Sharkey has franchised their business model to like minded organizations in other cities.

Sharkey has decided to hire you to automate the production of his loan tracking system.

Throughout the month, clients may borrow additional funds (i.e. debits) and pay off some of the existing loan (credits). The business people have noticed that loans with a high outstanding balance tend to require enforcement of payment. For this reason, the interest rate for the next month is calculated using rate table consisting of 3 tiers, based on the new balance from the current month. A low interest rate is used on the first tier amount, a middle interest rate on the second tier amount, and a high interest rate on the third tier amount. The minimum payment each month can also vary, but is calculated as a straight percentage of the new balance.

Interest is owed on the previous balance, plus debits, minus credits. For example, suppose that the consortium has decided to charge 10% monthly interest on the first $1000 (first tier = $1000), 20% interest on the amount between $1000 and $6000 (second tier = $6000) and 30% interest on the remaining amount. Then the interest this month for a loan with a previous balance of $9000, debits this month of $3500 and credits this month of $2500 would be $2300 computed as:

```
0.10*1000 + 0.20*(6000-1000) + 0.30*(10000-6000)
```

giving a new balance of $12,300 ($10,000 + $2,300).

The new system should track the following loan information: loan number, client name, previous balance), amount borrowed by the customer this month (debits), and amount paid by the customer this month (credits), interest charged this month, new balance and minimum payment for next month. This information needs to be tracked from month to month implying persistent data.

Since the automated system does not yet exist, a text file (`ASCIIDataFile`) has been prepared that consists of ole line for each loan giving: loan number, client name, current (new) balance. As a starting point, the remaining loan values are 0.0. A second text file provides the rate table: low rate, first tier amount, middle rate, second tier amount, high rate, and minimum payment rate. Note that all rates are given as monthly percentages.

Once during the month, the loan information is updated based on the transactions (debits/credits) through the month. A form such is to input the total credits and total debits for each client for the month. Once this data is entered, the previous balance is the new balance, and the interest, new balance and minimum payment are reset to 0.0.

At some time later in the month (after the credits and debits have been entered), a month-end billing program is run which calculates the interest, new balance (prev+debits-credits+interest) and minimum payment for the month.

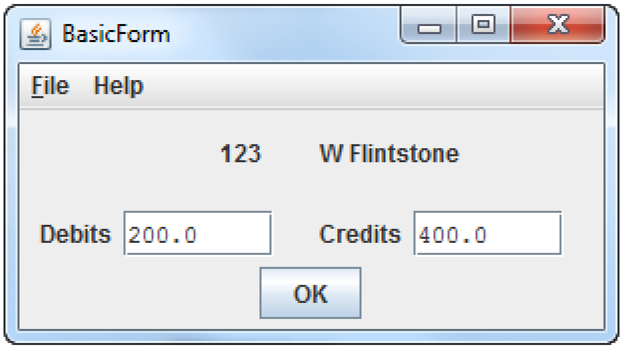
Advice

This assignment can be fairly lengthy. You should start on the problem early. Come to the tutorial prepared. Good design will greatly reduce the time needed in front of the keyboard. Starting this assignment shortly before it is due, is a recipe for disaster.

Inputs & Outputs

Two files are provided for initialization (bootstrap) of the system: `LoanData.txt` (loan number (string), name (string) and balance(double) and `RateData.txt` (low rate (double), first tier amount (double), middle rate (double), second tier amount (double), high rate (double), and minimum payment rate (double)). Note that all rates are given as monthly percentages.

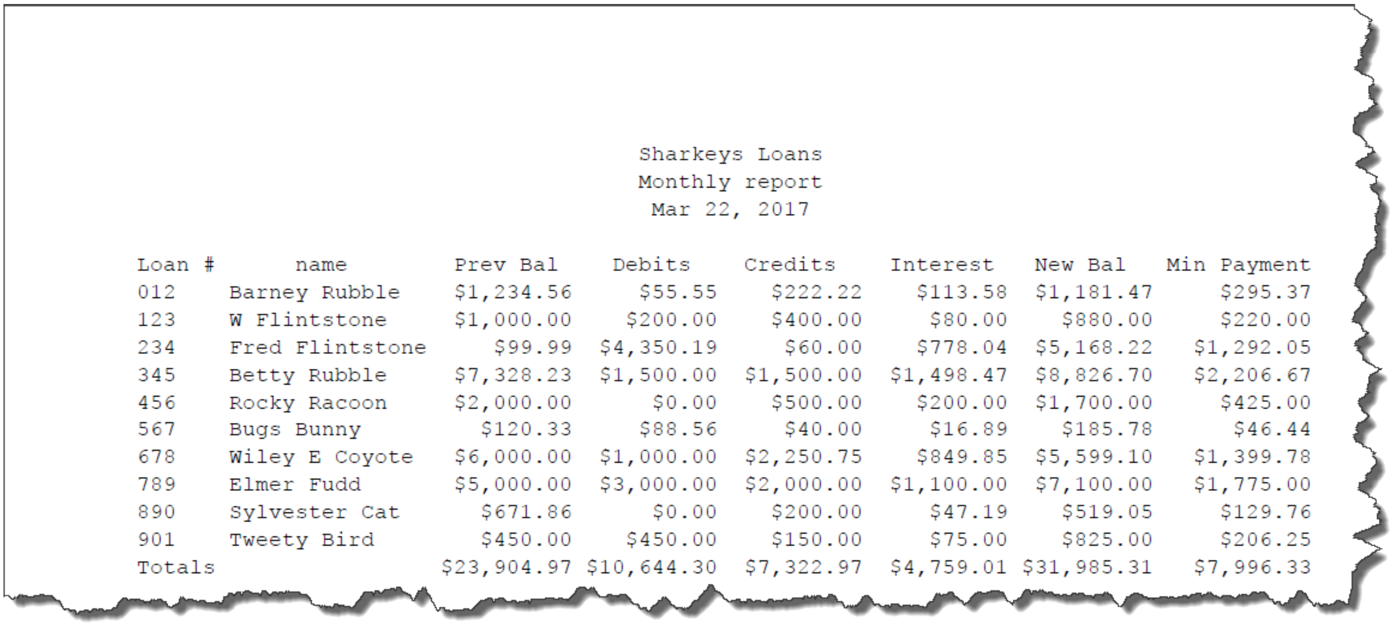
Entry of client credit and debits should be facilitated by a form such as:



123 W Flintstone

Debits 200.0 Credits 400.0

A report can be produced at any time to track the current state of the loans. The report might look like the following (in this case run after the month-end billing program has been run.



```
                                      Sharkeys Loans
                                      Monthly report
                                       Mar 22, 2017

Loan #      name         Prev Bal    Debits    Credits    Interest   New Bal   Min Payment
012    Barney Rubble     $1,234.56     $55.55    $222.22    $113.58  $1,181.47      $295.37
123    W Flintstone      $1,000.00    $200.00    $400.00     $80.00    $880.00      $220.00
234    Fred Flintstone      $99.99  $4,350.19     $60.00    $778.04  $5,168.22    $1,292.05
345    Betty Rubble      $7,328.23  $1,500.00  $1,500.00  $1,498.47  $8,826.70    $2,206.67
456    Rocky Racoon      $2,000.00      $0.00    $500.00    $200.00  $1,700.00      $425.00
567    Bugs Bunny          $120.33     $88.56     $40.00     $16.89    $185.78       $46.44
678    Wiley E Coyote    $6,000.00  $1,000.00  $2,250.75    $849.85  $5,599.10    $1,399.78
789    Elmer Fudd        $5,000.00  $3,000.00  $2,000.00  $1,100.00  $7,100.00    $1,775.00
890    Sylvester Cat       $671.86      $0.00    $200.00     $47.19    $519.05      $129.76
901    Tweety Bird         $450.00    $450.00    $150.00     $75.00    $825.00      $206.25
Totals                  $23,904.97 $10,644.30  $7,322.97  $4,759.01 $31,985.31    $7,996.33
```

Assumptions

You may assume that the loan shark currently has no more than than 100 clients and all data entry is error free.

Requirements

1. Perform analysis as discussed in class to get the primary set of classes. Draw the analysis diagram including relationships and multiplicities.
2. Perform design using CRC cards completing responsibilities for knowing, doing and collaborators.

Steps 1 & 2 can be done in groups. There will be discussion (and exercise) of analysis & top-level design during the tutorial on Mar. 29. Steps 3 through 6 ***must*** *be individual work.*

3. Perform detailed design to develop a complete (and compilable) set of Java interfaces for the identified classes.
4. Perform coding of implementations for the identified classes which may result in discovery of additional support classes.
5. Perform testing of the system.
6. For production (submission). Create the system. Enter the transactions to match the Debits and Credits columns in the report above. Run the month-end computations and Finally list the loan information (generating a report as above).

Suggestions

- The loan information is persistent. The number of loans does not currently change (i.e. none added or deleted).
- At implementation, use separate classes for GUIs and reports.
- You will likely end up with between 10 and 15 classes including classes/interfaces identified during design, main classes and support classes (e.g. GUIs and reports) identified during coding. There should be interfaces for the classes identified during design, but it is not necessary to have interfaces for support classes (although it doesn’t hurt).
- Marks will be assigned to both the quality of the design and the quality of the implementation.

Submission:

Details regarding preparation and submission of assignments in COSC 1P03 are found on the COSC 1P03 Sakai Site as `Assignment Guidelines` under `Course Documents`. This document includes a discussion of assignment preparation, programming standards, evaluation criteria and academic conduct (including styles for citation) in addition to the detailed assignment submission process copied below.

To prepare and submit the assignment electronically from the lab, follow the procedure below:

1. Ensure your folder (say `Assign_5`) containing the DrJava project is stored on your `Z:` drive.

2. Using DrJava, print (to `CutePDF Writer`) the `.java` files of your assignment (using the name *`ClassName`*`.pdf` where *`ClassName`* is the class name (i.e. same name as the `.java` file)) and save the `.pdf` file at the **top level** of the project folder (i.e. directly within `Assign_5`).
3. Create the loan information file. Enter the transaction information (as above). Perform the month-end computation and print the resulting loan report to `CutePDF Writer`. Save the pdf file at the **top level** of the project folder (i.e. directly within `Assign_5`).
4. Create a `.zip` file of your submission by right-clicking on the top level folder (i.e. `Assign_5`) and selecting `Send to/Compressed (zipped) folder`. A zipped version of the folder will be created. Use the default name (`Assign_5.zip`).
5. Log on to Sakai and select the COSC 1P03 site.
6. On the `Assignments` page select `Assignment 5`. Attach your `.zip` file (i.e. `Assign_5.zip`) to the assignment submission (use the `Add Attachments` button and select `Browse`). Navigate to where you stored your assignment on the `Z:` drive and select the `.zip` file (i.e. `Assign_5.zip`). The file will be added to your submission. Be sure to check the `Honor Pledge` checkbox. Press `Submit` to submit the assignment. You should receive a confirmation email.
7. Assignments incorrectly submitted will lose marks. Assignments without the required files may not be marked.

DrJava

The `.zip` folder you submit should contain the project folder including all files relevant to the project—the `.drjava`, `.java` and `.class` files for the assignment and `.pdf` files for program listings and output.

Other Platforms

If you are using an IDE other than DrJava to prepare your assignment, you must include the `.java` source files and the `.pdf` files described above as well as a file (likely `.class` or `.jar`) that will execute on the lab machines.