CS1013 Programming Project 2018

1. Goal

The goal of the project is to construct an application to explore data on customer reviews of businesses in processing. The data set we will use is a public dataset from the "yelp" website.

The application will read in data from a file "reviews.csv", render it, and allow the user to interact with it. The data will take the following format (comma separated values):

user_id, user_name, business ID, business name, stars, date, text, useful, funny, cool
For example:
"Kki2nwtP8U2qmWwRvPwLRA", "Mark", "u0LXt3Uea_GidxRW1xcsfg", " Farmer's Market","4","2011-10-14","
Fruits and veggies look so good! lots of varieties of apples!","0","0","1"

Each entry appears on a new line. A small test dataset is available at https://www.scss.tcd.ie/Gavin.Doherty/reviews.csv
This only contains 1500 reviews. Full datasets are available from links on Module webpage, but will need to be processed to fit into above format.

More ambitious groups should use a larger portion of the yelp_review.csv file and map the business ID (an alphanumeric code) to the business name and details contained separate files (yelp_business.csv). A separate file is also available for user details. You may add additional fields from the original data if you wish. A local version of the full 3GB dataset will be made available. You may also pull in other data (e.g. mapping).
All fields are surrounded by quotation marks. Commas may appear inside quotation marks so be careful of this when splitting the fields.

User ID – alphanumeric code uniquely identifying the user.
User_name – Screen name of the user, not unique.
Business ID – alphanumeric code uniquely identifying the business.
Business Name – Name of the business, not unique.
Stars – 1 to 5 (5 is better).
Date – in YYYY-MM-DD format.
Text – a free text field reviewing the business.
Useful,funny,cool – each is a number, zero or higher, rating the helpfulness of the review.

Fields are not expected to be empty, but it is best to program defensively.

2. Structure

The progam will contain the following components:

- code to read in the data from a file and place it in classes.
  - Processing provides both loadBytes and loadStrings commands
  - A simple (although not particularly efficient) solution would be to define a datapoint class which represents a single review. There would be one instance of the class for each entry in the input file.
- code to select a subset of this data.
  - Not all the data will be shown on the screen at one time, and so a set of queries must be defined in your code. At a minimum, the following queries should be implemented:
    - View the data for a particular business, or the businesses which match a search query (e.g. the same address or name).
    - Ranked listings (by review score).
    - Changes in scores over time.
- code to draw the data to the screen.
  - The results of each query will need to drawn on the screen.
  - You are encouraged to use graphical representations where appropriate (eg. the data could be on a map or barchart).
- code to handle user commands.
  - Selecting what data is to be displayed (the query), the address, date range, time of day, etc.
- code to put everything together
  - You are advised to have an outline of this as early as possible (first week of project).

3. Assessment

The project marks will be allocated according to both your individual effort and the effort you put into the group. This is a group project, and part of the project is to manage the group effort. Each individual will receive a mark will be based on their own contribution to both the individual and group components**. If you make no contribution to individual or group tasks, you will receive no marks**.

You will be required to submit and demonstrate the current status of your project **every week** until the end of the semester. **Half** of the project marks will be allocated for the **weekly submissions. DO NOT MISS THE LABS.**

Hence it is not possible to leave the project until the end of the semester.

Code is to be submitted via the subversion revision control system. All code must be commented, and the authors indicated. You **must** use comments to indicate revisions to the code (eg. " M. Jones, Added Graph class for displaying results, 8pm, 10/3/2018". "J. Smith, Updated to show the dates on the chart, 7pm 14/3/2018", "L. White, Fixed bug which stopped user from going back to homepage, 9pm, 15/3/2018". etc.). It is in your own best interest to get credit for the code you have written!

Attendance at the labs is **mandatory** and you will be expected to work in pairs at the labs. You should take turns "driving". The project will take more time than is available at the labs. Remember that CS1013 is marked **only on coursework**. You will need to schedule time in your groups outside of lab time to work on your assignment / divide up workload.

The assessment will focus on the demonstration of working code; a check on code authorship; a check on quality of code and documentation, progress towards the overall goal of the project, and the features implemented. Your code is to be accompanied by a short report of **maximum 5 pages** outlining your design and any ways in which your solution goes beyond the original project brief.

You will present your project at the lecture in the last week of term. The best team, as judged by the panel, will win a prize for Best JF Programming Project.

4. Assessment Schedule

The following is the schedule for demonstration:

Week 6	Project Handout	Plan project. Start thinking about it. Outline main program.
Week 8	File reading and code outline	Demonstrate reading of data from file and give outline of main program, including major classes, sketch of screens.
Week 9	Initial rendering of data	Graph and textual output of data.
Week 10	Data selection and rendering	Apply user queries to data set and present results on screen.
Week 11	User Interaction	Select different options and change screens
Week 12	(5th April) Demo of project (6th April) Final Submission	Full demo to panel and submission of report – **5 pages MAXIMUM**.

The project demo will be in Regent House from **16:00 to 19:00** on 5th of April. Report due on Friday 6th.

5. Other queries.

- You may use multiple tables.
- You may not use an alternative development environment (IDE) without permission. Such permission will only be granted if all team members indicate they are happy to do so, and I am convinced that all team members are fully engaged with the project.
- You must use Subversion rather than any other revision control system. It is used to track individual contributions.