EL BACH Achraf Trinity College, Computer Science and Statistics

Supplemental Programming Project Report
CSU11013 – PROGRAMMING PROJECT

Summary:

Introduction

This project is a software used by store to keep track of every sale, whether the search revolves around a customer, a store, or a product. This a supplemental solo project with a “three weeks” working period.

Data Processing

The data file and data objects were processed using three core classes/tools:

Data Parsing:

The *ParsingTool*’s purpose is to transforms a data text file into a set of reusable Java objects. The parsing procedure is executed only once at the start of the program. The *ParsingTool* then acts as a storage unit containing the data set of sale items (represented by the *SaleInfo* class), as well as the lists of individual names of the customers, the stores, the products, and each store’s customers list.

Data Filtering:

The *FilteringTool*’s purpose is to return a subset of sale objects that verifies a specified set of predicates. Using the Java native class *Predicate*, we can create custom predicates specifically made for the sale objects named *ObjectFilter*(s). These custom filters allow us to facilitate the use of the Java Collections’ stream functionality. We can filter the dataset by customer, store, product, and date.

Data Sorting:

The *SortingTool*’s purpose is to create an efficient mean of comparison for a dataset, according to the user requests. Using the Java native class *Comparator*, we can create custom comparators specifically made for the sale objects. They are then all combined into a single comparator that will be used in the Java *Stream*’s sorted method. We can sort the dataset by customer, store, product, date, and price.

Avatar Class

The Avatar class is an esthetic plus that adds a visual representation of the customers, the stores, and the products. It comes in handy when we display the individual lists so that the user can select the entry he needs. The avatar is composed of its name, its type (which individual it represents) and its image, which is uniquely drawn for each entity.

Data Display

The graphical representation of the user interaction panels and of the query results is drawn to the screen using a screen abstraction. The only responsibility of the program is to draw the screen that has been provided whenever it changes. The screens return a command String on each user interaction which is processed in the main class in order to understand the user request and correspondingly change screens, which are:

Menu Screen:

The menu screen is the one appearing at the start of the program. It has a logo, a name, and three distinct options, one brings you to a store selection, one brings you to a customer selection and the other one brings you to a product selection.

Selection Screen:

As mentioned above, the user can select an entity which is used to process that use queries. On this screen there is always two options and the outcomes of these two buttons depend on the type of entity displayed. This screen is the only one that display the avatars' image.

Graph Screen:

The Graph screen is one of the two screens used to display queries' results. This one is composed of two display methods, a point graph or a list. The data represented is the amount of one specified attribute occurrences based on another attribute. The *GraphList* is a generic class using Java reflection API.

Sale History Screen:

The history screen displays the sale dataset in a simple and refined format. For each day of the dataset, a separate frame is formed containing each product bought with its discounted price next to it. This screen takes the most time at generation because the way I chose to create it was not the smartest.

Conclusion

I shouldn't have done this supplemental project considering the trouble I have had to experience during the last semester (which clearly is a mishandling of the administration) and considering that I have already done a full project (four full weeks) which I'm sure has probably not even been seen. I didn't give it my all due to the circumstances, but I did give a good part of my soul to the other project, and I would be really grateful if my previous project was taken in consideration for this project evaluation. This module has caused me a lot of trouble during my semester abroad, but it was the most entertaining. It's sad that my whole experience was based on a poor administrative work. Here is a link to my previous project:

https://github.com/AchrafEL-BACH/Programming-Project/settings

I wish you the best and I thank you for this semester.