Alex Smith

Professor Rivas

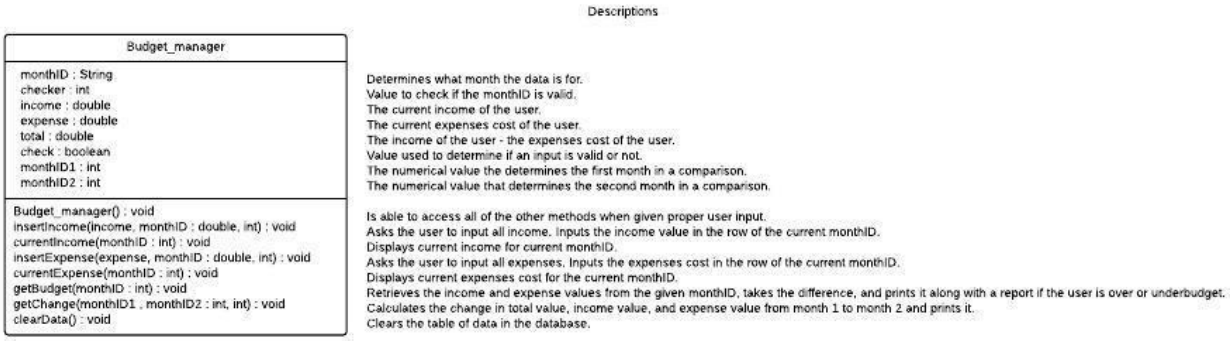
Software Development 1

5/6/17

Project 2 Writeup – Budget Manager

My project is a household budget manager that will utilize a database to store data input by the user regarding their income and expenses for a given month. This project writeup will explain my motivations for creating this project, give a detailed description of what the program can do including all the data fields and methods, briefly detail the system requirements for this program, compare this program to similar applications, and explain how to use the program. In addition, a UML diagram and ER diagram will be used to visually demonstrate some features of this program. This project was used to culminate my experience with Java gained throughout the Software Development 1 course and apply my knowledge to a practical program.

The conception of the original idea for my project began with a focus on database systems. The topic of database systems was originally chosen after realization that I would be required to take multiple courses about it later in my four years at college. I also have a potential internship involving a database project for a company, so completing this introductory database project would assist my understanding of database languages such as PostgreSQL or MySQL. Using PostgreSQL, the table for the program was created, storing the monthID for the information of the user, the income and expenses of the user for the given month, and the total value of the user had left over after subtracting the expenses from the income. The idea for a budget management system was chosen due to the practical use of this application for an average person. For these reasons, this concept was chosen as my final project for Software Development 1 and the creation of this program will serve as useful experience in the future.



The budget management program has a variety of data fields and methods detailed in the UML diagram (shown above). The monthID data field records which month the following data corresponds to, input by the user. The monthID is then converted to an Integer and stored in the value of the checker variable in order to catch any exceptions before continuing with the program. The income and expense variables store the user's income and expenses as a double when the respective input methods are accessed by the user. The total variable is a double that is calculated by taking the difference of the income and expense in the given month when the getBudget method is accessed. If the getBudget method is called before the user has input an income or expense value, both variables have a default value of 0. The Boolean variable check is used to determine if the user is inputting valid values for each variable. The last two variables, monthID1 and monthID2 are integers that are used with the getChange method to determine which two months will be compared. The variables are used to store the user's inputs before updating the database and to catch exceptions when attempting to input invalid values.

Including the main method, there are nine methods that interact with database to accomplish the goals of the management program. The main method establishes an initial connection to the database and starts the budget_manager() method, which can be used to call the remaining methods. The budget_manager method asks the user to input a monthID as an integer (and will reject non-integer inputs by catching the exception and running the method again) and then asks for a valid case-sensitive

command to be entered, which will access the respective method. The only command that is not a separate method is the 'help' command, which will print all the valid commands. The insertIncome method uses the SQL command UPDATE to update the income value in the given monthID row. The insertExpense method is identical to the insertIncome method but it updates the expense value of the row instead. The currentIncome and currentExpense methods use the SQL SELECT command to retrieve the income or expense respectively from the current monthID row and prints the value. The getBudget method uses the SELECT command to retrieve the income and expense for the current monthID, then finds the difference between the income and expense values. This value is stored in the total variable, which is then put into the database table using the UPDATE command. The total variable is then printed and the value is checked to see if it greater than zero to determine if the user is under or over budget for the current month. The getChange method requires two monthIDs and the SELECT command retrieves the total, income, and expense values from each row. The same values are compared by taking the difference of the first month and the second month and compares it to zero. The method will print if there is an increase, decrease, or no change between both months for each variable. The last method is clearData, which erases the all previously input data using the SQL TRUNCATE command. In the budget_manager method, all of the other methods besides the main method can be accessed by entering the name of the method and following the instructions. The methods and data fields of this program make up the features of the budget management program.

The budget management program is not very intensive on the system. The program requires the PostgreSQL driver for Java to run correctly. In addition, the program requires the database with the table and PostgreSQL to access the database locally. Due to the numerous try and catch commands in the code, the program does make more operations to insure there are no invalid inputs. Although more inputs are made, the program does not take long to execute commands and updates to the database

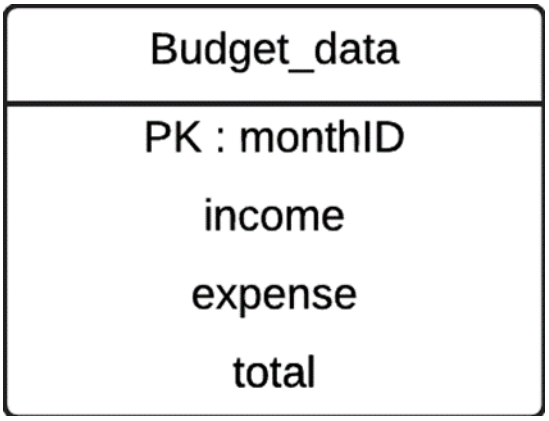
are quick. Overall, it is not a very intensive program although there are some programs that need to be installed before it can properly function.

There are many similar programs that can manage household budgets. These programs may have more depth features such as storing individual expenses such as rent, food costs, and more. This budget management program also allows the user to compare the inputs between multiple months using the getChange command. The program can also be used to store any values in the income and expense columns and use the getChange command for a comparison. The program is very similar to preexisting programs however it can be used in a variety of ways.

The budget management program is simple to use for the User. Once PostgreSQL, the Java PostgreSQL driver, the program, and database have all been downloaded and properly installed (including the database file in the correct location), the User can access the program by referencing the Java driver in using -cp and the location of the .jar file. An example is located as a comment at the top of the Java file. After the program is launched, the prompt tells the User to enter an integer for a monthID value. The user will be informed if an invalid input is entered and will be asked to enter another one. After, the User will be prompted to press enter to continue the program and then will be asked to enter a command. It will inform the User if they have entered a valid command and will suggest the 'help' command to list all valid commands. If an input is required for a command, it will inform the User if their input is invalid or not. The income and expense inputs are double values, the monthID values for the getChange method and budget_manager method are integers. If an invalid input type is entered, the User will be informed of the valid input type and the program will reset. The budget management system is simple for a User to use and provides instructions for how to properly use each command.

The budget manager project provides a practical program that can manage a household's budget over multiple months. This project has helped me learn how to use SQL languages such as

PostgreSQL and has helped me learn how to integrate SQL languages in Java. In addition, this project has helped me practice creating UML diagrams and describing the methods in the program. As detailed in this paper, other programs are similar in approach to this project and this project is very simple and straight forward to use. The budget management project is a practical program created using the knowledge of Java I have gained during Software Development 1.



Database table diagram

Works Cited / References

1. LIANG, Y. DANIEL. "Objects and Classses." INTRODUCTION TO JAVA PROGRAMMING. Vol. 10. S.l.: PRENTICE HALL, 2015. 322-64. Print.

2. "PostgreSQL - JAVA Interface." Www.tutorialspoint.com. N.p., n.d. Web. 28 Apr. 2017.