Python – Lists and Data Files

The **One Stop Insurance Company** needs a program to enter and calculate new insurance policy information for its customers. Create a data file called OSICDef.dat that contains the next policy number, the basic premium, the discount for additional cars, the cost of extra liability coverage, the cost of glass coverage, the cost for loaner car coverage, the HST rate, and the processing fee for monthly payments. The file will look as follows:

1944 869.00 .25 130.00 86.00 58.00 .15 39.99

As the program starts, read the values from the defaults file.

The user will input the *customer's* first and last name, address, city, province (validate using a list to make sure the province is valid), postal code, and phone number. They will also enter the number of cars being insured, and options for extra liability up to $1,000,000 (enter Y for Yes or N for No), optional glass coverage (Y or N), and optional loaner car (Y or N). Finally enter a value to indicate if they want to pay in full or monthly (Full or Monthly – use a list to validate). Convert the first and last name, the city, and the payment Method to title case and the Y/N values upper case. No validations required – other than those specified - but go for it if you want. Be careful when testing - enter values that are valid for each input.

Insurance premiums are calculated using a basic rate of $869.00 for the first automobile, with each additional automobile offered at a discount of 25%. If the user enters a Y for any of the options, the costs are $130.00 per car for extra liability, $86.00 per car for glass coverage, and $58.00 per car for the loaner car option. All these values are added together for the total extra costs. The total insurance premium is the premium plus the total extra costs. HST is calculated at 15% on the total insurance premium, and the total cost is the total insurance premium plus the HST. Customers can pay for their insurance in full or in 8 monthly payments. Calculate the monthly payment by adding a processing fee of $39.99 to the total cost and dividing the total cost by 8. The invoice date is the current date and the next payment date is the first day of the next month. Use the values from the defaults file in your processing.

Display all input and calculated values to the screen in a **well-designed receipt**.

Save the Policy number, all input values and the total insurance premium to a file called Policies.dat for future reference the invoice date will be the current date. Display the message "Policy information processed and saved." – you may also include a type of progress bar at this point. And increase the next policy number by 1. A sample line in the file might appear as follows:

1944, 2023-03-10, John, Smith, 12 Main St., St. John's, NL, A1A8H4, 123-123-1234, 2, Y, N, Y, Full, 1329.00

Allow the user to enter as many policies as they need. When the user is done entering policies and exits the program, write the current values back to the defaults file.

Create a second Python program that will allow the user to enter the total amount of sales for each month from January to December – note that if it not the end of the year some of the monthly values may be 0. Place the values in a list representing the y-axis. Values for the x-axis will be the months of the year in the form [“Jan”, “Feb”, “Mar”, ...]

Use Matplotlib to create a graph of the total sales ($) against the months using the two lists created. Add an appropriate title and other options that you see fit.