

# BCIS 5110

## Programming Languages for Business Analytics

### Project 2

Project 2 includes 2 steps as follows. For each step follow the description and requirements of the step and provide your answers/results in the report.

- Use Jupyter Notebook as the platform to write your report.
- The entire report should be prepared in a Jupyter Notebook
- Make sure to document every step you take and explain what you have done.
- Apply Headings and text formatting using Markdown for Jupyter Notebooks ([Ultimate Markdown Guide for Jupyter Notebook](#)).

### Objectives & Deliverables

In this project you will implement and test the algorithm you designed in the first project.

Given the information in the next steps, design an algorithm that does the followings:

1. Generates 20,000 customers using the given characteristics.
2. Generates accounts information and activity for the customers using the information given.
3. Stores customers' and accounts information and activity in the right format for further analysis.

In your submission, you need to provide the following items:

1. A Jupyter Notebook in `.ipynb` and `.html` formats containing:
  - a. Your Algorithm coded in Python
    - Note that your code needs to be self-explanatory, so please add comments to explain how it works.
  - b. Your simulation run outputs and logs
  - c. Samples of the results generated by the program
2. Samples of output files generated by your code.
  - Note that the total size of the files should not exceed 1 MB.
3. Log file generated by your code.

Please submit these files separately and not as a zip file.

Please note that the evaluation of this project will be based on the numbers you calculate in Step 2 of this project. Make sure your calculations are correct for every bullet point.

## Step 1 – Writing Python Program

As the first step of this project, you need to convert the flowchart or pseudocode from your Project 1 to Python codes. In this step you will write a Python program that performs the simulation described in the first project.

Please consider the followings when writing your codes:

- Make sure you follow the Python Enhancement Proposals ([PEPs](#)) when writing your codes.
- Use Jupyter Notebook to organize your codes and for documentation purposes.
- Your codes and your Jupyter Notebook should be self-explanatory, meaning use Jupyter Notebook as a platform for writing your report along with your codes.
- Use Markdown to format your report along with your codes.

Please make sure your program performs the following tasks along with the tasks described in the first project:

1. Stores the results of the simulation on disk.
2. Provides complete logging information.
  - a. The logging information should include:
    - i. Timestamp (Date and Time) information
    - ii. Source information (where the log is coming from, e.g. name of the component, module, function, class, ...)
    - iii. Content or Message of the log (description of the event happening)
    - iv. Contextual information (depending on the situation, you may need to include additional information, e.g. the value of the loop iterator: `i=4`)
    - v. Performance metrics (including memory usage and CPU usage)
  - b. The logging information should be provided in the following ways:
    - i. Printed into the console in Jupyter Notebook
    - ii. Printed to a file stored on the disk.
3. Provides the total time and memory used for execution.
4. Provides samples of the results in the Jupyter Notebook (at the end)

## Step 2 – Functional Testing

In this step you will perform functional testing which in simple words means checking if your program is doing exactly what it is meant to. All the functional or requirements specification of your program is already described in Project 1. In this step of the project you will test your program to verify if your program behaves according to the specified functional requirements described before.

Please note that in this step you will perform the tests and sanity checks and provide the test results but the verifications will be performed later by the professor.

In your Jupyter Notebook, provide the following information (use the results generated by your program to get the following information)

Please organize this information in a readable format preferably a table.

### A. Cohort information

---

1. Number of customers in your results
2. The minimum and maximum Customer ID in your results
3. Number of unique Customer IDs
4. Min, P25, Median, P75, Max, Mean, and Standard Deviation for the age of customers.
5. Frequency table for Gender of the customers (category, number of observations as frequency, percent frequency, cumulative frequency, cumulative percent frequency)
6. Frequency table for Marital Status of the customers (category, number of observations as frequency, percent frequency, cumulative frequency, cumulative percent frequency)

7. Provide percent frequency of Marital Status for the following categories separately:
  - a. For customers with age in [20, 30]
  - b. For customers with age in (30, 60]
  - c. For customers with age in (60, 80]
8. Frequency table for Number of Children of the customers (category, number of observations as frequency, percent frequency, cumulative frequency, cumulative percent frequency)
9. Provide percent frequency of Number of Children for the following categories separately:
  - a. For customers with age in [20, 40]
  - b. For customers with age in (40, 80]
10. Frequency table for Education Level of the customers (category, number of observations as frequency, percent frequency, cumulative frequency, cumulative percent frequency)
11. Provide percent frequency of Education Level for the following categories separately:
  - a. For customers with age in [20, 25]
  - b. For customers with age in (25, 35]
  - c. For customers with age in (35, 80]
12. Min, P25, Median, P75, Max, Mean, and Standard Deviation for the Annual Income of customers.
13. Frequency table for Number of Accounts of the customers (category, number of observations as frequency, percent frequency, cumulative frequency, cumulative percent frequency)
14. Min, P25, Median, P75, Max, Mean, and Standard Deviation for the Total Credit Line of customers.

## **B. Accounts information**

---

1. The minimum and maximum for the Date Opened across the entire cohort
2. Min, P25, Median, P75, Max, Mean, and Standard Deviation for the Age of the accounts (as of January 1<sup>st</sup>, 2022, calculated in years).
3. Frequency table for Account Age Flag (category, number of observations as frequency, percent frequency, cumulative frequency, cumulative percent frequency) (Account Age Flag is column in your data with values either True or False; the condition evaluated to get the Boolean is:  $(\text{Customer's Age} - \text{Account's Age}) \geq 20$ )
4. The minimum and maximum Account Number in your results
5. Frequency table for last digit of the Account Number (category, number of observations as frequency, percent frequency, cumulative frequency, cumulative percent frequency)
6. Min, P25, Median, P75, Max, Mean, and Standard Deviation for the Account Credit Line.
7. Frequency table for Account Credit Line Flag (category, number of observations as frequency, percent frequency, cumulative frequency, cumulative percent frequency) (Account Credit Line Flag is column in your data with values either True or False; the condition evaluated to get the Boolean is:  $(\text{sum}(\text{Account Credit Lines}) == \text{Total Credit Line})$ )
8. Min, P25, Median, P75, Max, Mean, and Standard Deviation for the Annual Fee.
9. Frequency table for Annual Fee Flag (category, number of observations as frequency, percent frequency, cumulative frequency, cumulative percent frequency) (Account Fee Flag is column in your data with values either True or False; the condition evaluated to get the Boolean is:  $(\text{Annual Fee} == \text{Account Credit Line} \times 0.01)$ )
10. Min, P25, Median, P75, Max, Mean, and Standard Deviation for the Annual Interest Rate.

## **C. Account Activity information**

---

1. Min, P25, Median, P75, Max, Mean, and Standard Deviation for the Number of Transactions (Purchase or Cash Advance) for each card during the activity period.
2. Min, P25, Median, P75, Max, Mean, and Standard Deviation for the Number of Purchases.
3. Min, P25, Median, P75, Max, Mean, and Standard Deviation for the Number of Cash Advances.

4. Min, P25, Median, P75, Max, Mean, and Standard Deviation for all the Purchase Amounts.
5. Min, P25, Median, P75, Max, Mean, and Standard Deviation for all the Cash Advance Amounts.
6. Min, P25, Median, P75, Max, Mean, and Standard Deviation for all the Payments Amounts.
7. Min, P25, Median, P75, Max, Mean, and Standard Deviation for all the Closing Balances.
8. Min, P25, Median, P75, Max, Mean, and Standard Deviation for all the Minimum Amounts Due.
9. Min, P25, Median, P75, Max, Mean, and Standard Deviation for all the Total Purchase Amounts of the month.
10. Min, P25, Median, P75, Max, Mean, and Standard Deviation for all the Total Cash Advance Amounts of the month.
11. Min, P25, Median, P75, Max, Mean, and Standard Deviation for all the Payment Amounts of the month.
12. Min, P25, Median, P75, Max, Mean, and Standard Deviation for all the Total Interests of the month.
13. Frequency table for Delinquency Counter (category, number of observations as frequency, percent frequency, cumulative frequency, cumulative percent frequency)
14. Frequency table for Annual Active Flag (category, number of observations as frequency, percent frequency, cumulative frequency, cumulative percent frequency) (Account Active Flag is column in your data with values either True or False; the condition evaluated to get the Boolean is: `(Delinquency Counter < 3)`)