**Faculty of Applied Science & Technology**

**School of Mechanical and Electrical Engineering Technology**

**Sheridan College**

**COURSE: ENGI55612-Data Analytics**

**LECTURER: Dr. Ameera Al-Karkhi**

**ASSIGNMENT PROJECT NUMBER: 1 ASSIGNMENT TYPE: (Group - 4 members)**

**ISSUE DATE: 12th September 2023**

**SUBMISSION DATE:** **2nd October 2023 @ 10:00 AM**

**PENALTIES FOR LATE HAND-IN:**

**(After which maximum obtainable mark is 50%)**

**ABSOLUTE DEADLINE: 3th October 2023 @ 10:00 AM**

**(After which mark is 0%)**

**PLAGIARISM: Students are reminded that plagiarism (copying) is a serious disciplinary matter. Checks are regularly made for misuse of the web and other existing materials.**

**Project 1- Data Exploration Methods and Cleansing Process**

**Introduction**

This project aims to help the student in practicing the fundamental of data analytics (data exploration and data cleaning processes). The given data bases for this project that you will be working on are four dataset that contain information on traffic accidents of Toronto from 2015- 2018. To learn more about this dataset, download the data set from the Databases folder-Project1 in Slate platform.

In this assignment you will get an introduction to some of the most common tasks that Data Analyst/Scientists have to perform before any analysis can be done such as gathering data, joins, column merging and creating **new data**. In addition to applying cleansing techniques.

**Objectives:**

* Apply principles and practices of data exploration, manipulation and preprocessing
* Program at a basic level using R programming language
* Explore data analytics techniques such as correlation, visualization such as scatter plots, pie chart, table, distribution of data, and outlier analysis.

**Requirements:**

1. Create a new single data set that can be used to output a table that lists the **number of persons injured** or killed in traffic accidents in each **neighbourhood** of Toronto in the last 4 **years**. (You will need to extract the following information: Year, vehicles in the street, district, and Neighbourhood)
2. Write a sequence of code that shows the total vehicles in the street for each district for the last 4 years during the accidents.
3. Write a sequence of code that lists the top 5 neighbourhoods with the highest average number of vehicles in the streets.

**Notes**:

* To remove comma from the variables, you will find 'gsub()' function very useful. See the following resources about using 'gsub()' function:<https://www.journaldev.com/43690/sub-and-gsub-function-r>

**Submission**

Please provide the following:

* + - Submit your assessment electronically to the appropriate folder in **Slate**
    - One file of Python code with detailed comments on each step.
    - The Pdf, word file of your code with a detailed results analysis that shows thorough understanding of the results (graphs/plots interpretations). In addition to your conclusion/summary section for the results interpretation.

**Project 1 Rubric**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **Unsatisfactory (0-40%)** | **Satisfactory (40-60%)** | **Good (80%)** | **Excellent (100%)** |
| **Requirements/ Correct run**  **50%** | * **Similar solution is found** **on the internet or submitted by another student** * Completed less than 50% of the requirements. * Does not execute due to errors. * No testing has been completed. | * Completed most of the requirements. * Executes without errors. * Some testing has been completed, but did not cover all cases. | * Completed all the requirements, but solution can be improved * Codes execute without errors * Thorough testing has been completed; all possible test inputs were examined | * Executes without errors * **Excellent** representation of input and output data. * **Thorough** **testing** has been completed and output from all test cases is included. |
| **Solution efficiency**  **20%** | * A difficult, non-easily-comprehendible, or inefficient solution. * Prompts/hints/messages to user are misleading or non-existent. | * A logical solution that is easy to follow but it is not the most efficient * Prompts/hints/messages to user contain little information, poor design. | * Solution is efficient and easy to follow * Hints, and messages to user are understandable, proper use of symbols or spacing in output | * Solution is efficient, easy to understand, and maintain * Excellent prompts/ hints/ messages to user * All aspects of solution were well considered, nothing to improve |
| **Coding Standards**  **10%** | * Poor use of indentation * Disorganized code * Poor use of variables (global variables, ambiguous naming) | * Indentation/White spacing makes program readable. * Organized work. * Good use of variables | * Good use of indentation * Organized work. * Good use of variables (no global variables, unambiguous naming) | * Excellent use of indentation * Creatively organized work. * Excellent use of variables (no global variables, unambiguous naming). |
| **Clarity/ documentation/ Presentation**  **20%** | * No code documentation included. * No comments on main sections/subsections of code | * Basic documentation has been completed * Purpose is noted for each function/method | * Clearly documented. * Specific purpose is noted for each function and control structure. | * Clearly and effectively documented including descriptions of all variables. * Specific purpose is noted for each section, control structure, I/O code |