# Assignment #2

**CSCI 5408 (Data Management, Warehousing, Analytics)**
**Faculty of Computer Science, Dalhousie University**

**Date Given: Oct 13, 2021**
**Due Date: Oct 26, 2021 at 11:59 pm**

**Late Submissions are not accepted. 10% deduction per day will be applied for late submissions**

**Disclaimer**: This assignment requires students to work on various websites and open Datasets with appropriate citation. Submissions related to this assignment will not be used for commercial purposes.

---

## Objective:

- The objective of this assignment is to understand research and industry problems related to distributed database operations, and transactions management.

## Plagiarism Policy:

- This assignment is an individual task. Collaboration of any type amounts to a violation of the academic integrity policy and will be reported to the AIO.
- Content cannot be copied verbatim from any source(s). Please understand the concept and write in your own words. In addition, cite the actual source. Failing to do so will be considered as plagiarism and/or cheating.
- The Dalhousie Academic Integrity policy applies to all material submitted as part of this course. Please understand the policy, which is available at: https://www.dal.ca/dept/university_secretariat/academic-integrity.html

## Assignment Rubric

| | Excellent (25%) | Proficient (15%) | Marginal (5%) | Unacceptable (0%) | This Rubric Applied to |
|---|---|---|---|---|---|
| Completeness including Citation | All required tasks are completed | Submission highlights tasks completion. However, missed some tasks in between, which created a disconnection | Some tasks are completed, which are disjoint in nature. | Incorrect and irrelevant | Problem #1 |
| Correctness | All parts of the given tasks are correct | Most of the given tasks are correct However, some portions need minor modifications | Most of the given tasks are incorrect. The submission requires major modifications. | Incorrect and unacceptable | Problem #2 – Task 2 |

| | | | | | |
|---|---|---|---|---|---|
| Novelty | The submission contains novel contribution in key segments, which is a clear indication of application knowledge | The submission lacks novel contributions. There are some evidences of novelty, however, it is not significant | The submission does not contain novel contributions. However, there is an evidence of some effort | There is no novelty | Problem #2 – Task 2 |
| Clarity | The written or graphical materials, and developed applications provide a clear picture of the concept, and highlights the clarity | The written or graphical materials and developed applications do not show clear picture of the concept. There is room for improvement | The written or graphical materials, and developed applications fail to prove the clarity. Background knowledge is needed | Failed to prove the clarity. Need proper background knowledge to perform the tasks | Problem #2 – Task 1 |

**Citation:**
McKinney, B. (2018). The impact of program-wide discussion board grading rubrics on students' and faculty satisfaction. Online Learning, 22(2), 289-299.

**Problem #1: This problem contains two reading tasks.**

**Reading Material #1:** To retrieve the paper, visit IEEE database through libraries.dal.ca

M. Sharma and G. Singh, "Analysis of Joins and Semi-joins in Centralized and Distributed Database Queries," *2012 International Conference on Computing Sciences*, Phagwara, 2012, pp. 15-20, doi: 10.1109/ICCS.2012.15.

**Reading Material #2:** To retrieve the paper, visit IEEE database through libraries.dal.ca

V. Kate, A. Jaiswal and A. Gehlot, "A survey on distributed deadlock and distributed algorithms to detect and resolve deadlock," 2016 Symposium on Colossal Data Analysis and Networking (CDAN), Indore, 2016, pp. 1-6, doi: 10.1109/CDAN.2016.7570873.

**Read the papers and perform the following:**

- Write a summary (≅ 1 page/ paper) on the paper in your own words. (you do not need to add images/figures/tables from the paper. However, you can add your own block diagrams or flowcharts to support the summary you have written)
- What is the central idea of discussion?
- Did you find any topic of interest in this paper? If Yes, what are those, and why do you think those are interesting? If No, then as per you, what are the shortcomings of this paper?

**Submission Expectations:** 1 page Report for each paper (total 2 pages) containing the summary and analysis

**Problem #1 Submission Requirements:** A single PDF file (2 pages for two summaries)

**Problem #2: This problem contains two tasks. 1 logical task + 1 Programming task**

**Research and Development:** You need to simulate a distributed DBMS

Visit the website and extract the following datasets:
https://www.kaggle.com/olistbr/brazilian-ecommerce?select=olist_order_payments_dataset.csv

**Problem Scenario:** A company "Data5408" has two branches, ***VM1*** and ***VM2***. Assume that the datasets you received from Kaggle are data of "Data5408". In this question, you need to perform two tasks:

**Task 1: Build Distributed Database**

- If the datasets are converted to database tables, and database(s), how will it be placed, state the reasons? (E.g. why did you consider specific Fragmentation, transparency etc.)
- You need to create two MySQL instances in two GCP Virtual Machines {VM1, and VM2}. Your ***VM1*** site is responsible for storing customer, geolocation, user related information. ***VM2*** site is responsible for storing all remaining information such as, item, product, payments etc. [**Note**: If you experience issues in handling large datasets, then consider a random reasonable size (<1000 data points) subset of the given data.]
- If required, please perform data cleaning, decomposition of dataset etc. before creating the database and record your logic in the PDF. Cleaning using spreadsheet is sufficient
- Since "Data5408" implemented a distributed database, it should create and maintain a Global Data Catalogue or Global Data Dictionary. How will you create it? Where will it be placed? [**Hint**: Global data dictionary (GDD) is an additional component, which does not eliminate the need of local data dictionaries. GDD usually contains information on databases, tables that are located at different sites, and connected using the network.]
- You do not have to write SQL script for this part, you can use import statement to upload your clean table on VM1 and VM2 database.

**Problem #2 – Task 1 - Submission Requirements:**

- A single PDF file with data cleaning, formatting logic or screenshots
- Screenshots of VM1, VM2 MySQL instances
- SQL dump {structure and value} taken from VM1, and VM2

**Task 2: Perform Concurrent Remote Transactions (programming needed) on a single DBMS (VM1 MySQL)**

- Write a simple DBMS Transaction processing logic using Java program*, and run the program on your local machine (TP). This program will access VM1 MySQL instance (DP) and execute concurrent remote transactions.
- Your program will perform **three concurrent execution of transactions written in SQL.**
- Your program will also create a simple text file, which will act as a Transaction Log.
- The details of the transactions are given below:

**You must follow the sequence. Write your observation on how MySQL handled this particular case**

**Tab 1: The table shows the sequence of Transactions entering the system for the execution**

| | T1 | T2 | T3 |
|---|---|---|---|
| Sequence 1 | Read customers data where zip code = "01151" | Read customers data where zip code = "01151" | |
| Sequence 2 | Update retrieved customers' city to "T1 City" | | |
| Sequence 3 | | | Read customers data where zip code = "01151" |
| Sequence 4 | | Update retrieved customers' city to "T2 City" | |
| Sequence 5 | | | Update retrieved customers' city to "T3 City" |
| Sequence 6 | | | Commit |
| Sequence 7 | Commit | | |
| Sequence 8 | | Commit | |

**Note**: If you do not have "01151" in your dataset, you can randomly select any fixed zip code for all Transactions {T1, T2, T3}

- Modify your program and add one or two new method(s) to create exclusive locks for the data. A transaction must obtain a lock based on the sequence of arrival (Tab 1), and must release after operation.

  * **You can only use standard libraries.**

**Problem #2 - Task 2 - Submission Requirements:**

- Upload your program code before adding the Locking logic and after the modification with the locking logic to gitlab (https://git.cs.dal.ca).
- Provide screenshots of your concurrent transaction testing with locking and without locking logic

**Assignment Submission Instructions:**

- Two PDF files – Problem #1, Problem 2 (Task 1)
- Two SQL Dump Files related to Problem 2 (Task 1) – files with .SQL extension

Must be added to a single .zip file before uploading to Brightspace. Do not use any other compression format. rename the .zip file as **Your_FirstNameB00xxxxx.zip**

- Program code (before and after modification) for Problem #2 (Task 2) should be in gitlab.