Assignment #2

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

Date Given: Jan 31, 2022
Due Date: Feb 13, 2022 at 11:59 pm

Late Submissions are not accepted and will result in a late penalty of 10% deductions / day in the assignment.

Disclaimer: This assignment requires students to work on research paper, open Datasets, and RDBMS 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 #2
Correctness	All parts of the given tasks are correct	Most of the given tasks are correct However, some portions need	Most of the given tasks are incorrect. The submission	Incorrect and unacceptable	Problem #2

		minor modifications	requires major modifications.		
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
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 #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 one reading task.

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

Y. Gao, X. Gao, X. Yang, J. Liu and G. Chen, "An Efficient Ring-Based Metadata Management Policy for Large-Scale Distributed File Systems," in *IEEE Transactions on Parallel and Distributed Systems*, vol. 30, no. 9, pp. 1962-1974, 1 Sept. 2019, doi: 10.1109/TPDS.2019.2901883.

→ Read the paper and perform the following:

- Write a summary (≅ 1.5 page) 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?

Problem #1 Submission Requirements:

Approx. 1.5-page Report (***problem1.pdf***) containing the summary and analysis

Problem #2: This problem contains series of tasks. You need to fill a worksheet and provide screenshots to support your worksheet

Research and Development: You need to create distributed DBMS infrastructure, and perform a comparative study of local and distributed transactions

Visit the website and extract the following datasets:
https://www.kaggle.com/olistbr/brazilian-ecommerce

Task 1: Building database from given dataset

- Download the CSV files given in the Kaggle link, and consider those as your data points.
    - Similar to previous assignment (Assignment 1), clean and format each csv file (replace **null** values with '0' if it is a number column || replace **null** values with ' ' if it is a text/string column) - Write a ½ page summary on what type of cleaning and formatting you have applied on the CSV files. A single ½ page summary for all CSV files is sufficient. Add summary to ***problem2.pdf***
    - **You do not need to remove any columns, keep all columns as it is in CSVs.**
- Create a single database (name of the database "***A2<your_netid>***") in your local MySQL RDBMS.
- Create **9 tables** within the newly created database. These tables will contain structure identical to the CSV files you cleaned, and you will populate the tables by importing data from the CSV files. (if files are large, you need to import using command prompt. Workbench import option may not work)
- Export the SQL (structure + value) of the database with all tables and data. Name the file "**MyDatabase.SQL**"
- Create a GCP MySQL Virtual Machine instance and use the **MyDatabase.SQL** to create the database, tables, and uploading data.
- If you have successfully completed the above steps - it means, at this point, you have two identical databases. One in your local machine and another in the cloud.
- In ***Problem2.pdf***, attach screenshots of your cloud instance, and local instance running mysql command "***SHOW TABLES FROM A2<your_netid>***;"

Task 2: Perform Transaction in your local machine

- In your local machine MySQL, you have 9 tables in the database ***A2<your_netid>.*** You need to write a **Transaction** with at least (2 insert, 2 delete, 2 update, 3 select operations) considering all the 9 tables. For insert, and update you can create your own set of dummy data. **Based on your transaction query formation, complexity, and uniqueness you will get points for Novelty**
- Capture the transaction execution time (i.e. execution time for all 9 or more statements

```
Begin Transaction
        -----
        -----
```

End Transaction

Task 3: Perform Transaction in the Remote machine

- In your GCP MySQL virtual machine MySQL, you have 9 tables in the database ***A2<your_netid>.*** You need to write a **Remote Transaction** with at least (2 insert, 2 delete, 2 update, 3 select operations) considering all the 9 tables. For insert, and update you can create your own set of dummy data. **Based on your transaction query formation, complexity, and uniqueness you will get points for Novelty. You should not use the same queries that you used in Task 2. Please do some major modifications in the SQL queries**
- Capture the transaction execution time (i.e. execution time for all 9 or more statements

```
Begin Transaction
        -----
        -----
End Transaction
```

Task 4: Perform Distributed Transaction

- In your GCP MySQL virtual machine, and in your local MySQL, you have 9 tables in the database ***A2<your_netid>.*** You need to write a **Distributed Transaction** with at least (2 insert, 2 delete, 2 update, 3 select operations) considering all the 9 tables.
- This time, from the 9 unique tables, you need to use any 5 tables (your choice) from your Local machine database, and the remaining 4 tables in the GCP instance for performing the distributed transaction.
- For insert, and update you can create your own set of dummy data. **Based on your transaction query formation, complexity, and uniqueness you will get points for Novelty. Please do some major modifications in the SQL queries.**
- You need to write a **Java program** to call the distributed transaction.

  Hint: Set connection.setAutoCommit(false); in your Java program to control the execution and database update

  - Run Java Program:
  - within Java program select the 1st database (Local machine)

```
Begin Transaction
        -----
        -----
select the 2nd database (Remote machine)
        -----
        -----
End Transaction
connection.setAutoCommit(true);
```

- Capture the distributed transaction execution time (i.e. execution time for all 9 or more statements
- Fill the worksheet and submit it as part of **Problem2.pdf**

Tasks	Execution Time	Transaction Query	Your Observations
Task 2		*Write the transaction query here for each transaction*	
Task 3			
Task 4			

Problem #2 Submission Requirements:

1. Upload your program code for Problem 2 (Task 4) to gitlab (https://git.cs.dal.ca).
2. Provide summary needed for Task 1, and screenshots
3. Provide Queries of all transactions you performed in Task2, Task3, and Task4.
4. Provide the filled worksheet