Homework 5

CS430/CS630

100 points

INSTRUCTIONS:

Your solution must be submitted electronically, as follows.

- Write your answer to question 1 in a file called Q1.sql. The answers to questions 2 and 3 should be saved in files Q2.py and Q3.py respectively.
- Copy your solution files (Q1.sql, Q2.py and Q3.py) to a folder called hw5 under the cs630 folder in your Unix account. Follow the previously provided instructions (available on Canvas) on how to create folders, copy files and check/set the correct permissions on them.
- Copy a blank file under assignment 5 on gradescope so that the TA can share comments and grades to your solutions.
- You may submit/copy your solution files any number of times before the due date. Submitting/copying any new files after the due date invalidates your submission even if the content is unchanged.

Important Notes:

- SQL statements must run against the Oracle database we use in class. Make sure you run and test your queries against the dbs3 database in the CS' Unix lab. Create the tables, insert some data, and test your queries!
- SQL queries that do not run successfully will not receive credit even if they run successfully against other databases.
- SQL statements end with a semicolon.
- Write the part of the question that you are answering in the Q1.sql file in a comment line, e.g.: `-- Question 1 – Part c`.
- For Python Code, up to 20% of the total marks may be deducted for styling and formatting issues.
- Python code must be tested in your Unix account and run against the Oracle DB. Make sure you use version 3 (`Python3`, not `Python`) to run your program.
- Python files that do not run successfully (e.g., exit with an error) will receive at most 50% of the total mark.

Question 1 - (28 points)

Take the following DB schema.

```
Articles(aid: integer, title: string, author: string, pubyear: integer)
Students(sid: integer, name: string, city: string, state: string,
         age: real, gpa: real)
Reads(aid: integer, sid: integer, rday: date)
```

Primary keys are underlined in each relation. Each article is uniquely identified by aid. Every article has an id (aid), a title, an author, and a publication year (pubyear). Students are uniquely identified by their id (sid). In addition to an id, every student has the following attributes: name, city, state, age, and gpa. If a student reads an article, a record will be present in the Reads table, with the id of the student, and the id (aid) of the article they read, along with the date the article was read (the rday attribute).

Using the above schema, for each of the items below, write one or more SQL statements (in correct order, if it matters).

a) Create all three tables in the schema. Be sure to handle all key constraints. Additionally, make sure: students' GPAs are between 1 and 4 (inclusive), and no fields in any of the tables can be left without a value.

b) Insert 3 students and 2 articles into the database (choose reasonable values).

c) Based on the students and articles that you inserted in part (***b***), insert some records into the Reads table such that: 1) one of the students reads all articles and 2) another student reads one article, and the third student reads no article.

d) Create a View called MAStudents that contains all the information for Students from state of 'MA'.

e) Create a View called StudentsReads that contains information about the id, name and the city of students and the id and title of the articles they read. If a student has not read any article, it should not appear in this view.

f) Query the view you create in part (e) to extract the count of articles read by each student. You must use the StudentsReads view for this question.

g) Drop the two views you created in parts d and e.

Note: all views are non-materialized.

Question 2 - (32 points)

Using the schema from Question 1, write a Python script that uses the Pandas library and does the following. Be sure to have some data in your tables so that you can test your program properly and to make sure it produces the right results.

- Ask the users to enter the following information and use it to log into the user's Oracle Database: username, password, hostname, DB name.
- Connects to your Oracle DB schema.
- Uses PANDAS library to run a query against the DB and read all students records from the database and stores them pandas dataframe. Use the dataframe to do the following. For the last three items, be sure to use the dataframe aggregate.
    - Print out the name of the columns of that dataframe.
    - Print out the shape of the dataframe.
    - Print out the first two records from the dataframe.
    - Extract and print the average age of all students.
    - Extract and print the min and max gpa among all students.
    - Calculate and print the sum of gpa values.
- Runs a second query against the DB to extract information about the id, name and the state of students and the id and title of articles they read (the resulting relation will have the SID, NAME, STATE, AID, TITLE columns). Save the result in a Pandas dataframe. Use this dataframe to do the following.
    - Print the entire dataframe.
    - Print how many records are in the new dataframe.
    - Print how many columns are in the new dataframe.
    - Print the name of the columns from this new dataframe.
    - Filter this dataframe to keep only students from the state of MA. Saves the result into a third dataframe.
    - Extract how many articles each student from the state of MA read and prints the result. Use Pandas' group by feature.

Note: For all the above, you must use dataframe features (e.g., aggregate, filtering, groupby, etc.). Remember to close the connection after you're done reading from it.

Question 3 - (30 points)

Using the schema from Question 1, write a Python script that uses a `cursor` to execute queries against the DB. The program should do the following. Use cursor, not dataframe for this question.

- Ask the user to enter the following information and use it connect to your DB schema and acquire a cursor. username, password, hostname, db name (dbs3).
- Drop tables Students, Articles, Reads. Code must gracefully handle any exception. Before dropping any table, ensure that it exists in your database, so that the code doesn't generate any error if the table you're trying to drop doesn't exist.
- Recreate the 3 tables from Schema from Question 1.
- Insert two records in each table. Use reasonable values.
- Run a select query that extracts all articles and print them on the screen.
- Run a select query that extracts all students and print them on the screen.
- Run a select query that extracts all records from Reads and prints them on the screen.

Note: Remember to commit the transaction and close the connection.