

CS430 Homework 5

Brendan Nguyen

Due: Wednesday, Apr. 13, 2023

Question 1 (40 points)

Given the following DB schema:

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

For this schema:

- Write the SQL statement to create the table *Articles*. Do not forget about the key constraints.
Write the SQL statement to create table *Students*. Add the constraint that *gpa* should be between 1 and 4 (including 1 and 4). Do not forget about the key constraints. Write the SQL statement to create table *Reads*. Add the constraint that no attribute can be null. Do not forget about the key constraints.
- Write the SQL statement to create an index on column *rday*. Explain when such an index will be useful.
- Write the *INSERT* statements to insert 3 students. Write the *INSERT* statements to insert 2 articles.
- Write the *INSERT* statements to insert some records into *Reads* following these conditions: one of the students from (b) read all articles inserted for (b). Another student from (b) read one article inserted for (b). One student from (b) read no article.
- Write the SQL statement to create a View called *MAStudents* that contains all the information for *Students* from MA.
- Write the SQL statement to create a View called *StudentsReads* that contains information about the id, name, and city of students and the id and title of the article they read.
- Write an SQL query that uses the view from (f) (view *StudentsReads*) to extract the count of articles read by each student. Queries that do not use the view *StudentsReads* are given no credit.
- Write the SQL statements to drop the 2 views: *StudentsReads*, *MAStudents*.

Question 2 (30 points)

Using the schema from Question 1, write a Python file that uses the PANDAS library and does the following:

- Reads from the input: an Oracle username, Oracle password, Oracle hostname, Oracle DB name.
- Connects to our Oracle DB.
- Uses PANDAS library to run a query against the DB that extracts information about all *Students*. Saves the results in a PANDAS dataframe.
- Prints out the name of the columns of that dataframe.
- Prints out the shape of the dataframe.

- Prints out the first 3 records from the dataframe. Uses PANDAS aggregates to extract the average and min age of students. Prints the value.
- Uses PANDAS aggregates to get the minimum and maximum gpa for students. Prints the result.
- Uses PANDAS aggregates to get the sum of gpa values. Prints that result.
- Runs a second query against the DB to extract information about the id, name, and 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.
- Prints this new dataframe.
- Prints how many records are in the new dataframe.
- Prints how many columns are in the new dataframe.
- Prints the name of the columns from this new dataframe.

Note: please remember to close the connection.

Question 3 (30 points)

Using the schema from Question 1, write a Python file that uses the `connection.cursor()` to execute queries against the DB. The program should do the following:

- Reads from the input: an Oracle username, Oracle password, Oracle hostname, Oracle DB name.
- Connects to our Oracle DB.
- Uses the cursor to drop tables *Students*, *Articles*, *Reads*. Code must gracefully handle any exception, for the case these tables it tries to drop were not in the DB. Uses cursor to re-create the 3 tables from Schema from Question 1.
- Uses the cursor to insert two records in each table.
- Uses the cursor to run a `SELECT` query that extracts all articles. Prints all records extracted.
- Uses the cursor to run a `SELECT` query that extracts all students. Prints all records extracted.
- Uses the cursor to run a `SELECT` query that extracts all records from Reads. Prints all records extracted.

Note: please remember to commit the transaction and close the connection.