Information and Database Management Systems I

(CIS 4301 UF Online)

Fall 2024

Instructor: Mr. Alexander R. Webber

TA: Kyuseo Park

Homework 3

Printed Name:	
UFID:	
Email Address:	

Instructions: Please provide your answers to the questions of the following pages in Word or handwritten on separate sheets of paper. Mark clearly to which question each answer belongs. Then convert or scan your work into PDF (the latter by using either a scanner or a suitable scanner app on your smartphone). Note that *only the PDF format* is allowed and that your submission must be a *single PDF file*. Finally, upload your PDF file into *Canvas* and follow the instructions there. In order to enable the graders to fast find the solutions to your questions, it is important that you correctly specify the location of your answer for each question in Camvas, as it is described there. Otherwise, 0.5 points will be deducted for each answer.

Note: All homework assignments are designed for a period of two, three, or even four weeks (see course deadline sheet). This means they cannot be solved in two or three hours but require a considerable amount of time and effort. Therefore, the first recommendation is to start with them as soon as they are posted. The second recommendation is to distribute the work on a homework assignment over the entire available period. The third recommendation is to submit the homework solutions *on time before the deadline*.

Pledge (Must be signed[1] according to the UF Honor Code):

On my honor, I have neither given nor received unauthorized aid in doing this assignment.

_____

Student signature

[1] Each student is obliged to print out this page, fill in the requested information in a handwritten and readable manner, make the *handwritten* signature, scan this page into PDF, and put this page as the first page of the PDF submission.

Question 1 (SQL Queries) [66 points]

We are given a geostatistical database about countries, continents, rivers, etc. The following information is available in Canvas together with data for download:

- An ER diagram of the geostatistical database in PDF format (*HW3Ex1- geostatistical-database-ER-diagram.pdf*).
- An informal description of the database schema in PDF format (*HW3Ex1- geostatistical-database-schema-explanation.pdf*).
- A text file that contains `create table` commands to create the database schema (*HW3Ex1-geostatistical-database-schema.sql*).
- A text file hat contains `insert` commands for about 47,800 tuples to fill the database tables (*HW3Ex1-geostatistical-database-input-data.sql*).
- A text file that contains `drop table` commands to delete the database schema and the data in the database (*HW3Ex1-geostatistical-database-drop-tables.sql*).

Use the CISE Oracle DBMS and the Oracle SQL Developer software to create the database schema and fill the database with data. This will also help you learn about the system environment for your group project. In particular, the use of MySQL, PostgreSQL, and other database systems is not allowed.

(a) [10 points] Look at the database schema in the file *HW3Ex1-geostatistical-database- schema.sql*. You will find the following SQL statements from line 38 to line 52:

```
ALTER TABLE Country
  ADD CONSTRAINT FK_CountryREFCity
  FOREIGN KEY (Code, Capital, Province)
  REFERENCES City(Country, Name, Province)
  INITIALLY DEFERRED DEFERRABLE;

ALTER TABLE City
  ADD CONSTRAINT FK_CityREFProvince
  FOREIGN KEY (Country, Province)
  REFERENCES Province(Country, Name)
  INITIALLY DEFERRED DEFERRABLE;

ALTER TABLE Province
  ADD CONSTRAINT FK_ProvinceREFCountry
  FOREIGN KEY (Country)
  REFERENCES Country(Code)
  INITIALLY DEFERRED DEFERRABLE;

ALTER TABLE Province
  ADD CONSTRAINT FK_ProvinceREFCity
  FOREIGN KEY (Capital, Country, CapProv)
  REFERENCES City(Name, Country, Province)
  INITIALLY DEFERRED DEFERRABLE;
```

Your task is to explore this scenario by using the Internet. The keywords `INITIALLY DEFERRED DEFERRABLE` are non-standard SQL. They are supported by several database systems such as Oracle and PostgreSQL.

Answer the following questions.

(1) [4 points] What is the meaning of these keywords?

(2) [6 points] Why is the action indicated by the keyword `INITIALLY DEFERRED DEFERRABLE` needed in the scenario above? What is the problem? How is the problem solved?

(b) [56 points] Write SQL queries for the colloquial queries below and **show the results by providing screenshots for both your SQL queries and query results**. The screenshots must be embedded into the PDF file that contains your solutions to this whole assignment. In order to increase readability, the SQL queries should be written in a structured manner, all SQL keywords should be fully capitalized, and the table names and attribute names should be written in the same way as in the schema file.

If not explicitly forbidden, the use of the Oracle specific "row limiting functions" such as the "FETCH" clause and the "ROWNUM" function is allowed. The following two examples illustrate their syntax and use:

```
SELECT <column_name(s)>
FROM <table_name>
ORDER BY <column_name(s)>
FETCH FIRST <number_of_rows> ROWS ONLY;

SELECT <column_name(s)>
FROM <table_name>
ORDER BY <column_name(s)>
WHERE ROWNUM <= <number_of_rows>;
```

Write SQL queries for the following questions and provide screenshots as described above.

(1) [5 points] Classify the continents (attribute *continent*) by total population (attribute *totalPopulation*) in decreasing order. We assume due to a lack of detailed data that the population of a country is evenly distributed over its area.

(2) [4 points] Determine the number of cities that are located in countries that belong to two continents. Output the country name (attribute name *Country*), the country code (attribute name *Country Code*), the two continents (attribute names *Continent 1* and *Continent 2*), and the number of cities (attribute name *Number of Cities*). Order the data by the first continent, the second continent, and the country name. [Do not use `WITH` clauses.]

(3) [3 points] Classify the continents (attribute *continent*) by *average population density* (attribute *avgPopulationDensity*, round by two digits with the function `ROUND`) in decreasing order. We assume due to a lack of detailed data that the population of a country is evenly distributed over its area. Further, output the *average areal concentration* (attribute *avgArealConcentration*, round by four digits), which is the inverse of average population density, of each continent.

(4) [2 points] Find the top 5 countries with the highest average elevation of their airports. For each country, display the country name, the average airport elevation, and the number of airports. Order the results by the average elevation in descending order.

(5) [3 points] Find the countries that share borders with more than three countries having a higher GDP than theirs. List these countries along with their GDP, the number of such neighboring countries, and the average GDP of these neighbors. Order the results by the number of wealthier neighbors in descending order.

(6) [3 points] Identify the top 10 countries with a population greater than 50 million, where the capital city has more than 1 million residents and the country's GDP exceeds 100 billion USD, ranked by the percentage of the country's population living in the capital city. Additionally, for each of these countries, list the country name, the capital city name, the population of the capital, the total population of the country, the GDP, and the percentage of the population living in the capital. Order the results by the percentage of the population living in the capital in descending order.

(7) [3 points] List the languages spoken by at least 10% of the population in countries where more than three languages are each spoken by at least 10% of the population. For each of these languages, display the country name, the language name, and the percentage of the population that

speaks it. Order the results first by the country's name alphabetically and then by the percentage of the population that speaks the language in descending order.

(8) [3 points] Find countries that have both mountains with elevations greater than 5,000 meters and rivers longer than 3,000 kilometers. For each of these countries, list the country's name, the elevation of the highest mountain, the length of the longest river, and the total number of distinct mountains and rivers meeting these criteria. Order the results by the total number of qualifying mountains and rivers in descending order, considering only countries with at least one mountain over 5,000 meters and one river over 3,000 kilometers in length.

(9) [4 points] Identify provinces that feature a significant elevation range, where the highest peak is classified as a mountain, and the lowest elevation point within the same province is represented by a lake. The elevation difference between the province's highest mountain and its lowest lake should exceed 4,500 meters. For each of these provinces, provide the province's name, the associated country's name, the highest mountain's elevation, the lowest lake's elevation, and the calculated elevation difference. The results should be ordered by the elevation difference, from the largest to the smallest.

(10) [4 points] Identify the country (or countries) with the highest number of international organization memberships among those that have experienced a population growth rate of more than 2%. For each such country, display the country name and its population growth rate. Ensure the results are ordered by the population growth rate in descending order. For this question, do not use "row limiting functions" such as the "ROWNUM" function or the "FETCH" clause.

(11) [3 points] Identify the longest river on each island within every continent and list the continent, island name, river name, and river length. For islands with multiple rivers, select only the longest river. Order the results by continent and the length of the river in descending order.

(12) [3 points] Identify the largest desert within each province of countries with a population greater than 500 million, where the desert area is greater than 1000. For each province, list the province name, the country name, and the name of the largest desert. Order the results by province and country name.

(13) [3 points] Identify the predominant ethnic group for each country with a population greater than 50 million, where the ethnic group constitutes more than 55% of the country's population. For each qualifying country, list the country name, the ethnic group's name, and its percentage of the total population. In cases where multiple ethnic groups are tied for the highest percentage, include all such groups. Order the results by the country and ethnic group's name.

(14) [3 points] Determine which countries have a highly diverse linguistic landscape, characterized by having four or more languages spoken by at least 5% of the population each. For these countries, provide the country name, the total count of such languages, and the name of the most spoken language among them, including its percentage of the population speaking it. Rank the countries by the total number of languages (where at least 5% of the population speaks each language) in descending order, followed by the country's name alphabetically.

(15) [2 points] What are the names of the countries where the proportion of Christians is greater than the proportion of Muslims? Exclude the countries where either religion is not present.

(16) [2 points] What are the names of the provinces that have a population greater than 10 million and are located in a country that is a member of the European Union? Do not hard-code the abbreviation 'EU'. Display the results in a table showing the names of the provinces, their populations, and the names of the countries they are located in.

(17) [3 points] Determine the countries that have a higher average city population than the population of their capital city. List the country name, the average population of its cities (excluding the capital), and the population of its capital city. Only consider countries with more than one city and where the capital's population is known. Order the results by the difference between

the country's average city population (excluding the capital) and the capital's population, in descending order.

(18) [3 points] Find countries that have both the highest mountain and the longest river on their continent. Only list the names of these countries.

Question 2 (SQL Queries – Interpreting Data) [25 points]

This question refers to the database from Question 1. Its task is to explore countries by three different criteria: (1) gross domestic product (GDP), (2) gross domestic product per capita (GDPPC), and (3) industrial GDP (IGDP). The goal is to investigate if the intuitive expectation that industrial countries tend to have the highest values regarding these three categories is correct. The solution must fulfill the following requirements:

(a) All countries with GDP data in the database have to be ranked regarding each of the three aforementioned criteria. You may use the Oracle SQL function `RANK` for this purpose.

(b) The overall solution has to be arranged as a *multi-step query*. This means that an intermediate computed result of a step has to be stored in a permanent, that is, *persistent*, table for later reuse. This requires finding out how temporary tables in Oracle are stored persistently on disk in the database. Note that this cannot be achieved by the `WITH` clause that is only able to create temporary tables in the same query but cannot make them persistent. Within a single query of the multi-step query, it *is* allowed to use the `WITH` clause.

(c) The first main query must calculate all the needed data and rankings and put them into the persistent table *GDPResults*.

(d) The next three main queries rearrange the data of the table *GDPResults* according to one of the three kinds of GDP (GDP, GDPPC, IGDP). This means each query must focus on one of the three exploration criteria and enable a comparison of all three rankings. Each query must store its data in a new persistent table with the names *GDPOfCountry*, *GDPPerCapita*, and *GDPByIndustryPercentage* respectively. Your screenshots of these tables should only show the first 30 tuples. Note that this does not mean that you should apply the FETCH command.

(e) In order to achieve uniformity of student solutions, the schemas of the tables *GDPOfCountry*, *GDPPerCapita*, and *GDPByIndustryPercentage* must be very similar and contain the following attributes:
- Attribute 1: "Country"
- Attribute 2: the superior rank with the name "GDP Rank", "GDPPC Rank", or "IGDP Rank"
- Attribute 3: one of the other two subordinate ranks with the name "GDP Rank", "GDPPC Rank", or "IGDP Rank"
- Attribute 4: the remaining subordinate rank with the name "GDP Rank", "GDPPC Rank", or "IGDP Rank"
- Attribute 5:the superior kind of GDP with the name "GDP (in millions)", "GDPPC", or "IGDP (in millions)"
- Attribute 6: one of the other subordinate kinds of GDP with the name "GDP (in millions)", "GDPPC", or "IGDP (in millions)"
- Attribute 7: the remaining subordinate kind of GDP with the name "GDP (in millions)", "GDPPC", or "IGDP (in millions)"
- Attribute 8: "Population"

(f) In the tables *GDPOfCountry*, *GDPPerCapita*, and *GDPByIndustryPercentage*, all values of the attributes "GDP (in millions)", "GDPPC", and "IGDP (in millions)" must be rounded to two decimal places, and the periods of all numbers must be vertically aligned to increase readability (trailing zeros!). You may use the Oracle SQL function `TO_CHAR` for this purpose.

(g) At the end, it is required to write a short text passage that provides a short interpretation and comparison of the results from the three tables.

Note that full documentation (descriptions, explanations, screenshots of queries and query results, etc.) of all steps, queries, and observations is required.

Question 3 (SQL and Relational Algebra) [9 points]

For each of the following SQL statements, give an equivalent Relational Algebra expression if possible. If such an expression cannot be given, explain why.

(a) [5 points]

```
SELECT DISTINCT enum
FROM            employee
WHERE           NOT EXISTS
                (
                  SELECT *
                  FROM   projects
                  WHERE  NOT EXISTS
                         (
                          SELECT *
                          FROM   works
                          WHERE  employee.enum = works.enum AND
                                 works.pnum = projects.pnum
                         )
                );
```

(b) [4 points]

```
SELECT   deptId, MAX(salary)
FROM     employee
WHERE    rank = 'manager'
GROUP BY deptId;
```