Information and Database Management Systems I

(CIS 4301 UF Online)

Fall 2024

Instructor: Alexander Webber

TA: Kyuseo Park

Homework 2

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 *zipped* along with your source code and work for Question 5. 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.

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 (Relational Algebra – Queries (I)) [25 points]

Consider the following relation schemas. Primary keys are underlined. All attributes are of type *string* if not indicated otherwise. Use multi-step and multi-line queries to ease the formulation of queries. Use the rename operator ρ to give intermediate query results a name (some textbooks use the equivalent '$\leftarrow$' notation). Aggregate functions as you can find them defined in some textbooks are not allowed, neither in homework assignments nor in exams, since they are problematic extensions of the classical Relational Algebra.

- Patients(PatientID, FirstName, LastName, BirthDate, Gender, PrimaryPhysicianID)
- Physicians(PhysicianID, FirstName, LastName, Specialty, DepartmentID, HireDate)
- Departments(DepartmentID, Name, Location, StaffNumber)
- Nurses(NurseID, FirstName, LastName, Shift, DepartmentID, SupervisorID)
- Treatments(TreatmentID, Name, Description, Cost)
- PatientTreatments(PatientID, TreatmentID, Date, PhysicianID, NurseID)
- Appointments(AppointmentID, PatientID, PhysicianID, Date, Time, Purpose)
- MedicalRecords(RecordID, PatientID, Date, Summary, Diagnosis, TreatmentPlan)

(a) [5 points] Find the names of patients who have received treatment in more than one department, and list these departments along with the names of the patients.

(b) [5 points] Find the names of physicians who have treated patients whose first names are PatientA or PatientB on Nov 11.

(c) [5 points] Find the names of physicians in the general surgery department who have treated patients assisted by NurseA (unique first name).

(d) [5 points] Find the names of patients who have had appointments with physicianA (unique first name) but have not received any treatment (no show).

(e) [5 points] Identify the patients who have incurred the highest treatment costs. List their names along with the respective costs.

Question 2 (Relational Algebra – Queries (II)) [25 points]

Consider the following relation schemas. Primary keys are underlined. All attributes are of type *string* if not indicated otherwise. Use multi-step and multi-line queries to ease the formulation of queries. Use the rename operator ρ to give intermediate query results a name (some textbooks use the equivalent '$\leftarrow$' notation). Aggregate functions as you can find them defined in some textbooks are not allowed, neither in homework assignments nor in exams, since they are problematic extensions of the classical Relational Algebra.

- Passengers(PassengerID, FirstName, LastName, PassportNumber, Nationality)
- Flights(FlightID, Origin, Destination, DepartureTime, ArrivalTime, Date, AircraftID)
- Tickets(TicketID, PassengerID, FlightID, Class, Price, DateOfPurchase)
- Aircraft(AircraftID, ModelName, Manufacturer, Capacity)
- CrewMembers(CrewID, FirstName, LastName, Role, YearsOfExperience)
- FlightCrewAssignments(FlightID, CrewID)
- FlightPilotAssignments(FlightID, PilotID)
- Accidents(AccidentID, FlightID, Date, Location, Description)
- Pilots(PilotID, Name, Age)

(a) [5 points] Identify all accidents (date, location and description) that involved pilotA (unique name).

(b) [5 points] Identify crew members (find their first and last name) who have served on all flights piloted by PilotA (unique pilot name).

(c) [5 points] Find the aircrafts (Model name and manufacturer) that have never had any accidents.

(d) [5 points] Find the names of crew members whose experience is less than 5 years and have ever flown with pilotA (unique name).

(e) [5 points] Identify the names of passengers who are from the UK or the USA and who were on the flight from New York to London on December 25.

Question 3 (Relational Algebra – Thinking Deeper) [16 points]

This question lets you think deeper about the concepts of Relational Algebra. Relational Algebra can be regarded as an abstract query language. At this level, the performance of query execution does not play a role since the problem of query optimization is handled by the DBMS. But Relational Algebra is also used as a very important tool for query optimization to improve performance. This question is supposed to illustrate this.

(a) [4 points] Let $\mathscr{R}(A,B,C)$ be the schema of a relation R with r tuples. Let further $\sigma_{F_1}(\sigma_{F_2}(\sigma_{F_3}(\sigma_{F_4}(R))))$ be a valid Relational Algebra expression on R. Provide an equivalent Relational Algebra expression that always produces the same result as the given expression but is optimal in terms of performance if R had an implementation in a database. Argue why your expression is optimal. For this, think about how the given expression would be evaluated in an implementation and count how many read and write accesses of tuples this would take. Think how many read and write accesses of tuples your optimized expression would take. Describe what the worst case scenario for the performance is. Remember that Relational Algebra is a descriptive query language. An expression prescribes step by step which operations have to be executed and in which order.

(b) [4 points] Let $\mathscr{R}(A_1,\ldots,A_n)$ be the schema of a relation R for some $n \in \mathbb{N}$. Let $\pi_{L_1}(\pi_{L_2}(\pi_{L_3}(R)))$ with $L_1, L_2, L_3 \subseteq \mathscr{R}$ be a Relational Algebra expression on R. Determine the condition that has to be fulfilled such that the expression is valid. Argue if and how the expression can be optimized.

(c) [4 points] Let $\mathscr{R}(A,B,C)$ be the schema of a relation R. Let $\pi_{\mathrm{A}}(\sigma_{\mathrm{F}}(R))$ be a valid Relational Algebra expression. Determine if the projection operation and the selection operation in this expression can be swapped. That is, the question is if $\pi_{\mathrm{A}}(\sigma_{\mathrm{F}}(R))$ is equivalent to $\sigma_{\mathrm{F}}(\pi_{\mathrm{A}}(R))$.

(d) [4 points] Let R_1 and R_2 be two relations with schemas $\mathscr{R}_1(A,B,C)$ and $\mathscr{R}_2(D,E,F)$, respectively. Consider the Relational Algebra expression $\pi_{\mathrm{A,\,B}}(R_1) - \pi_{\mathrm{D,\,E}}(R_2)$. Is the expression a valid operation in the Relational Algebra? If yes, explain the meaning and the result of this operation. If not, argue why not.

Question 4 (Relational Algebra – Minimum and Maximum Numbers of Tuples) [14 points]

Let $\mathscr{R}(A,B,C)$, $\mathscr{S}(B,D)$, and $\mathscr{T}(A,C)$ be relation schemas such that relation R of schema $\mathscr{R}$ has r tuples, relation S of schema $\mathscr{S}$ has s tuples, and relation T of schema $\mathscr{T}$ has t tuples with $r > 0$, $s > 0$, and $t > 0$. The attributes A, B, C, and D are supposed to have the same numerical data type. Consider the given Relational Algebra expressions below and determine the *minimum* number and the *maximum* number of possible tuples of each result relation. Explain your answers.

(a) [3 points] $\pi_{\mathrm{A,\,C}}(R) \cap T$

(b) [3 points] $R \bowtie_{R.C=T.C} T$

(c) [5 points] $\pi_{\mathrm{B}}(\pi_{\mathrm{B}}(R) \cup \pi_{\mathrm{B}}(S))$ (For this question, also consider three other *possible* distinctions of numbers of tuples beside the *minimum* and *maximum* number of tuples.)

(d) [3 points] $R \div T$

Question 5 (About the Difficulties of Data Preparation for Database Upload) [20 points]

This practical question is supposed to demonstrate the time-consuming, troublesome, and often difficult task of preparing real-world data for database upload. Real world data can be found in all kinds of formats, and building a database from them requires their cleaning and restructuring that can be very tedious, time-consuming, and error-prone. A popular format for data is the CSV (comma-separated value) format. A CSV file is a plain text file. It stores tabular data as a list of lines. Each line corresponds to a record and consists of the same number of field values that are separated by commas. Usually a header line in such a file describes the meaning of each field by providing a field name. Another benefit of the CSV format is that a CSV file can be directly imported into and exported from spreadsheet programs such as *Microsoft Excel* on Microsoft Windows and *LibreOffice Calc* on Linux, Microsoft Windows, and macOS for data manipulation and analysis.

The task of this question is to take a given CSV file and transform it into a text file with a list of syntactically correct SQL `insert` commands. The `insert` commands can then be executed in the Oracle SQL Developer software or the DBeaver software, for example, with the purpose of uploading the specified records in the `insert` commands into an SQL table with a correspondingly predefined schema.

The basis of this question is again the data set about a company's processed orders and sales of toy cars from the Kaggle web page that you already know from Question 4 of Homework 1. Perform the following tasks and follow the instructions in every detail.

Task 1 (1 point) Create the following table schema `ToyCarOrdersAndSales` in CISE Oracle.

```
CREATE TABLE ToyCarOrdersAndSales
(
 OrderNumber         INT NOT NULL,
 QuantityOrdered     INT NOT NULL,
 PriceEach           FLOAT NOT NULL,
 OrderLineNumber     INT NOT NULL,
 Sales               DOUBLE PRECISION,
 OrderDate           DATE NOT NULL,
 DaysSinceLastOrder  INT,
 ProductLine         VARCHAR2(20),
 CustomerName        VARCHAR2(40) NOT NULL,
 AddressLine1        VARCHAR2(50) NOT NULL,
 City                VARCHAR2(20) NOT NULL,
 PostalCode          VARCHAR2(15) NOT NULL,
 Country             VARCHAR2(15) NOT NULL,
 ContactLastName     VARCHAR2(20),
 ContactFirstName    VARCHAR2(15),
 DealSize            VARCHAR2(10),
 PRIMARY KEY(OrderNumber, OrderLineNumber)
);
```

This is the target schema copied from our solutions to Homework 1. That is, data from the real-world data source in Kaggle are supposed to be stored according to this table schema. Similarly to Homework 1, provide a screenshot, and show that the table is empty.

Task 2 (0 points) Download the data set from the Kaggle web page by pressing the black download button on that page. You get a zip file named *archive.zip*. Extract the CSV file *Auto Sales data.csv* from it.

Task 3 (17 points) Transform (a copy of) the CSV file *Auto Sales data.csv* by applying appropriate methods into the text file *ToyCarOrdersAndSales Insert Commands.sql* that contains an SQL `insert` command for each line of the CSV file. Precisely describe your strategy and methods to achieve this goal.

Task 4 (1 point)	Copy and paste the `insert` commands from the file *ToyCarOrdersAndSales Insert Commands.sql* in the upper window of Oracle SQL Developer and execute them. Show screenshots of part of the `insert` commands (upper window) and the statements that rows have been inserted (lower window).
Task 5 (1 point)	Write an SQL query that returns the tuples in the table. Provide a screenshot that shows part of the first fifteen tuples. Write another SQL query that returns the number of tuples in the table. Provide a screenshot.

The difficulty of this question is in Task 3. Several string transformation (or manipulation) operations must be applied to each line of the CSV file, and their order is important. Your transformations must comply with the syntactical rules for SQL `insert` commands; otherwise, you will get syntax errors when you try to insert the tuples with Oracle SQL Developer into the empty table. Further, it is a question which tools to use to perform the string manipulations. Some proposed options are (1) the use of a text editor that supports *regular expressions*, (2) the execution of a program written in your preferred programming language that serially traverses the CSV and makes the needed modifications "manually", and (3) the execution of a program written in your preferred programming language that makes use of regular expressions if the programming language of your choice supports them. Other options could be possible and are allowed too.

Develop and well describe your strategy to perform these transformations and structure your description by the individual tasks.

Put all relevant documentation (for example, PDF file that describes your strategy and provides the required screenshots, ASCII file that contains your source code in your preferred program-ming language to perform the transformations) into a zip file with your answers to Questions 1-4.