**CSCE 4523 Introduction to Database Management Systems**

**Spring 2024**

**HW 2**

**Due: Friday, March 1, 2024, 11:59 PM**

**Tutorial/Example**

Open a web browser and visit: [RelaX : Relational Algebra Calculator](https://dbis-uibk.github.io/relax/landing)A screenshot of a computer

Description automatically generated

Click on Get Started.

Select the Wikipedia data set.

You can see the database contents and a good description of relational algebra in: [Wikipedia Relational Algebra](https://en.wikipedia.org/wiki/Relational_algebra)

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

You may only have 1 query in the window at a time. Test out an easy query using project to see the contents of the Employee Relation. Use your mouse to copy/paste the π. When you are done entering the query, click on “execute query” to see the results.

A screenshot of a computer

Description automatically generated

Delete that query and create queries for the following questions. When you are done, add a screenshot of your final query and results (similar to above) to a Word document or pdf to turn in for each of the following questions.

**Question 1**

Create a relation that shows the Employee.Name, Dept.DeptName, and Dept.Manager name for all employees in the Sales department.

A screenshot of a computer

Description automatically generated

**Question 2**

Create a relation that contains, for each Dept, the DeptName and count of employees that work in that dept.

A screenshot of a computer

Description automatically generated

**Question 3**

Create a relation that contains, for each Employee, their Name, their DeptName, and their Manager’s name (or null if they have no manager)

A screenshot of a computer

Description automatically generated

**Questions from the textbook: Brief, point form answers are fine**

1.4 *Describe the main characteristics of the database approach and contrast it with the file-based approach.*

Focus is now on the data first, and then the applications. The structure of the data is now kept separate from the programs that operate on the data. This is held in the system catalog or data dictionary. Programs can now share data, which is no longer fragmented. There is also a reduction in redundancy, and achievement of program-data independence. See also Section 1.3.

2.1 *Explain the concept of database schema and discuss the three types of schema in a database. (Also known as ANSI-SPARC Three-Level Architecture)*

The overall description of the database is called the database schema. External schemas (also called subschemas) that correspond to different views of the data. Conceptual schema describes all the entities, attributes, and relationships together with integrity constraints. internal schema, which is a complete description of the internal model, containing the definitions of stored records, the methods of representation, the data fields, and the indexes and storage structures used.

3.11 *Discuss the five essential characteristics of cloud computing.*

The essential characteristics are:

* *On-demand self-service:* consumers can obtain, configure and deploy cloud services themselves using cloud service catalogues, without requiring the assistance of anyone from the cloud provider.
* *Broad network access:* it is network based, and accessible from anywhere, from any standardized platform (e.g. desktop computers, laptops, mobile devices).
* *Resource pooling.* The cloud provider’s computing resources are pooled to serve multiple consumers, with different physical and virtual resources dynamically assigned and reassigned according to consumer demand. Examples of resources include storage, processing, memory, and network bandwidth.
* *Rapid elasticity.* Resource pooling avoids the capital expenditure required for the establishment of network and computing infrastructure. By outsourcing to a cloud, consumers can cater for the spikes in demand for their services by using the cloud provider’s computing capacity and the risk of outages and service interruptions are significantly reduced. Moreover, capabilities can be elastically provisioned and released, in some cases automatically, to scale rapidly based on demand. To the consumer, the capabilities available for provisioning often appear to be unlimited and can be called on in any quantity at any time.
* *Measured service.* Cloud systems automatically control and optimize resource use by leveraging a metering capability at some level of abstraction appropriate to the type of service (e.g., storage, processing, bandwidth, and active user accounts). Resource usage can be monitored, controlled, and charged for.

4.1 *Discuss each of the following concepts in the context of the relational data model:*

*(a) Relation:* A table with columns and rows.

*(b) Attribute:* A named column of a relation.

*(c) Domain:* The set of allowable values for one or more attributes.

*(d) Tuple:* A row of a relation.

*(e) Degree:* The number of attributes in a relation.

*(f) Cardinality:* The number of tuples in a relation.

4.5 For the schema below, list the primary key, candidate keys, and foreign keys (if any):

**Student (ID, StudentFirstName, StudentLastName, StudentAddress, AdvisorID)**

Primary Key: ID

Candidate Key(s): {ID}, {StudentFirstName, StudentLastName, StudentAddress, AdvisorID}

Foreign Key(s): {AdvisorID}

Advisor (AdvisorID, AdvisorFirstName, AdvisorLastName, AdvisorAddress)

Primary Key: AdvisorID

Candidate Key(s): {AdvisorID}, {AdvisorFirstName, AdvisorLastName, AdvisorAddress}

Foreign Key(s): None

* 1. *Consider the relations that would be produced by the following relational algebra operations using the schema from chapter 4 (also shown below). Which attributes would the resulting tuples have? What would an informal interpretation of the relation contents be?*

Hotel (hotelNo, hotelName, city)

Room (roomNo, hotelNo, type, price)

Booking (hotelNo, guestNo, dateFrom, dateTo, roomNo)

Guest (guestNo, guestName, guestAddress)

b) Hotel.hotelNo  Room.hotelNo(Hotel  Room)

This will produce a join of the Hotel and Room relations containing all the attributes of both Hotel and Room (there will be two copies of the hotelNo attribute). Essentially this will produce a relation containing all rooms at all hotels.

d) Guest (dateTo ≥ ‘1-Jan-2007’ (Booking))

This will produce a (left outer) join of Guest and those tuples of Booking with an end date (dateTo) greater than or equal to 1-Jan-2007. All guests who don’t have a booking with such a date will still be included in the join. Essentially this will produce a relation containing all guests and show the details of any bookings they have beyond 1-Jan-2002.

e) Hotel Hotel.hotelNo  Room.hotelNo (price  50 (Room)) )

This will produce a (semi) join of Hotel and those tuples of Room with a price greater than £50. Only those Hotel attributes will be listed. Essentially this will produce a relation containing all the details of all hotels with a room price above £50.

*5.12 Using the same schema as above, generate the relational algebra expressions for the following queries:*

b) List all single rooms with a price below £20 per night.

σtype=‘S’ ∧ price < 20(Room)

d) List the price and type of all rooms at the Grosvenor Hotel.

Πprice, type(Room hotelNo (σhotelName = ‘Grosvenor Hotel’(Hotel)))

g) List the guest details (guestNo, guestName, and guestAddress) of all guests staying at the Grosvenor Hotel on February 14, 2024.

ΠguestNo, guestName, guestAddress(Guest guestNo (σdateFrom ≤ ‘01-01-15’ ∧ dateTo ≥ ‘01-01-15’ (

Booking hotelNo (σhotelName=‘Grosvenor Hotel’(Hotel)))))