**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 CalculatorA screenshot of a computer

Description automatically generated](https://dbis-uibk.github.io/relax/landing)

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 . When you are done entering the query, click on “execute query” to see the results.

A screenshot of a computer

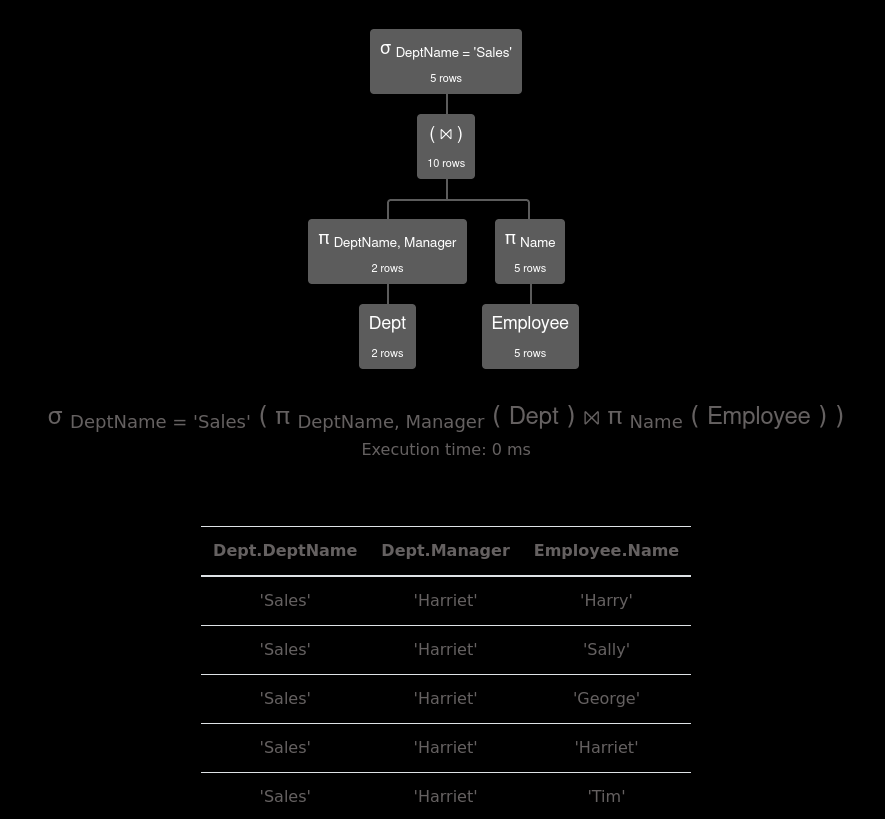
Description automatically generated

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**

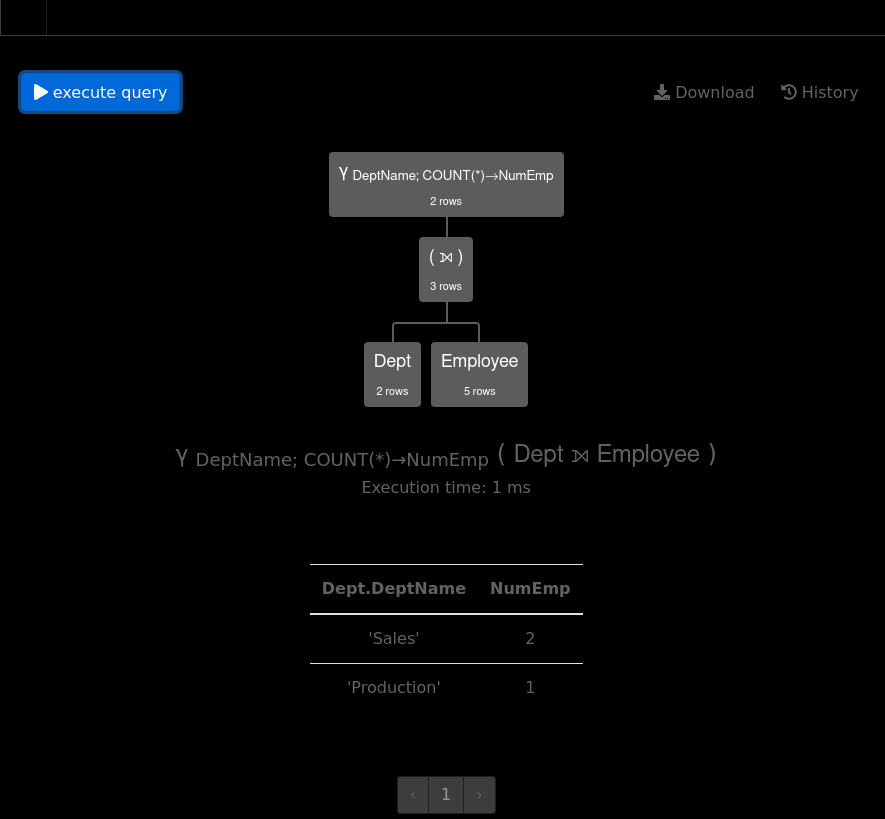
Create a relation that shows the Employee.Name, Dept.DeptName, and Dept.Manager name for all employees in the Sales department.

σ DeptName = 'Sales'(π DeptName,Manager(Dept)⨝ π Name (Employee))



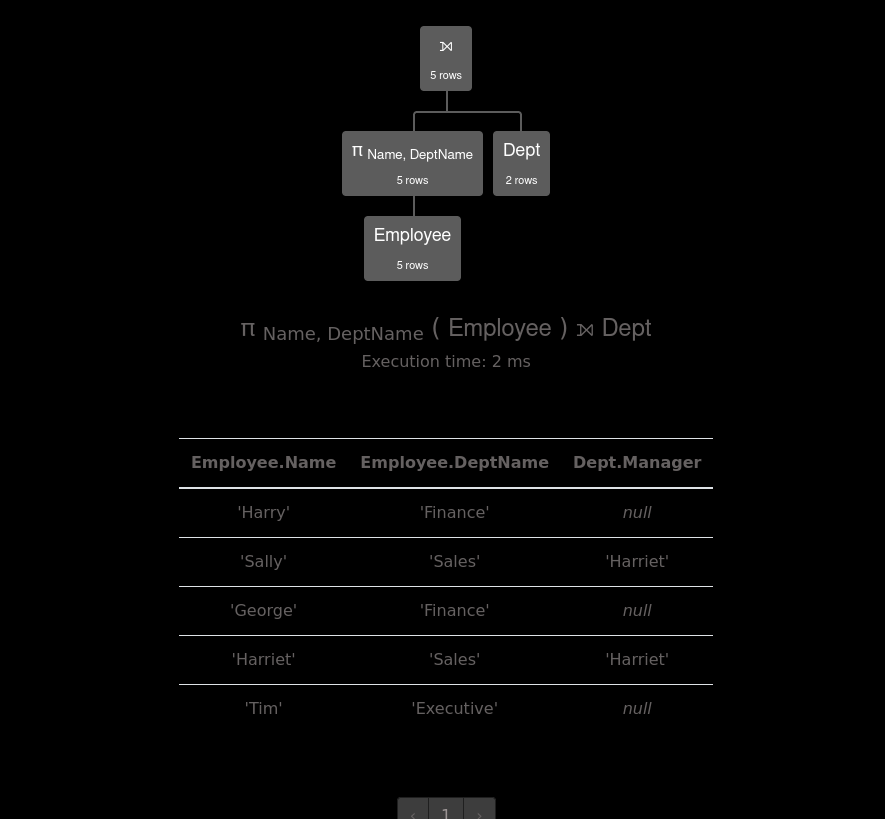
**Question 2**

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

γ DeptName; count(\*) -> NumEmp(Dept ⟕ Employee)

**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)

π Name,DeptName (Employee) ⟕ Dept

**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.

Databases can be used with multiple types of files while file based systems are constrained to specific files. Databases allow for fast query of data while file based do not have built in functions for that.

**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)

- External Schema

This provides a personalized view of the database to each user or group of users. It allows different users to view data in relation to there needs or tasks.

- Conceptual Schema

- Defines the abstract structure of the entire database for all users. Has all the logical costraints that apply to the data and the relationships among different parts of data.

- Internal Schema

- Describes how the data is physically stored in the database.

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

***-*** On-Demand Self-Service

- Users can provide computing resources like processing power, storage, and network connectivity automatically without depending on human interaction.

-Broad Network Access

- Cloud services are avaliable over the netwrok. This means services can be accessed from anywhere in the world.

- Rapid Elasticity

- Capabilities can be elastically provisioned and released. This is to allow rapid scalability outward and inward to follow demand.

- Resource Pooling

- Computing resources are pooled to server multiple conusmers using a multi-tenant model.

- Measured Service

- Cloud systems automatically monitor data usage and allow for a pay-as-you-go modle to ensure users only pay for what they use, which leads to large cost savings compared to alternative models.

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

**(a) Relation: Name that is made up of columns and rows which are both named**

**(b) Attribute: Column of a relation**

**(c) Domain: Set of allowable values from one or more attributes**

**(d) Tuple: Rows of a realation**

**(e) Degree: Number of attributes a relation contains**

**(f) Cardinality: 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 Keys:

StudentFirstName, StudentLastName, StudentAddress

Foreign Key(s):

AdvisorID

Advisor (AdvisorID, AdvisorFirstName, AdvisorLastName, AdvisorAddress)

Primary Key:

AdvisorID

Candidate Keys:

AdvisorFirstName, AdvisorLastName, AdvisorAddress

Foreign Key(s):

null

* 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)

A relation that shows all atributes in Hotel and Room

Makes a relation of Hotel Nums that are in both the Hotel and Room table

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

- Has Attributes guestNo, guestName, guestAddress

A relation that shows All attributes of guest that have stayed From 1-Jan-2007 onward

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

- All attributes from the hotel

- Everything from hotel where the hotel.hotelNo and Room.botelNo are price greater than 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.

**σ(type = 'single' and price < 20)π roomNum,type,price(Room)**

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

**σ(hotelName = 'Grosvenor Hotel') π hotelName (Hotel) ⨝ π price, type(Room)**

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

**(Guest) ⨝ σ(hotelName = 'Grosvenor Hotel') π hotelName (Hotel) ⨝ σ (dateFrom = 'Feb 14,2024) π dateFrom (Booking)**