Written Assignment #1
CAS CS 660: Graduate Introduction to Database Systems
Fall 2021

Due: Thursday, Sep 23, 2021, using Gradescope at 11:59 PM.

Problem 1. (SQL) (35 points)

Consider a (simplified) database of a Real Estate company containing the following tables (primary keys are underlined):

Agent(AID: integer, name : varchar(50), address : varchar(100), phone: char(10))
Here we store the real estate agents in the company and their information.
Property (PID: interger, location: varchar(100), long: real, lat: real, type: varchar(50), size: integer)
Here we store the properties that are managed by the company. A property can be of given type and can have a location and/or longitude and latitude information and size (in sq feet).
Customer(CID: integer, name : varchar(50), address : varchar(100), phone: char(10))
Here we store information about the customers.
Listing(LID: integer, PID: integer, selling_AID: integer, asking_price: real, listing_date: date)
Here we store information about the listings. A listing associates, a property, a selling agent (only one), an asking price, and the listing date.
Offer(OID: integer, LID: integer, CID: integer, offer_date: date, amount: real, buyer_AID: integer)
Here we store offers to a property from a customer on a specific date and we store the offer amount and the buyer's agent.

Write SQL queries for the following questions. Notice that there may be multiple ways the express the same question, you need to write one of them. You can break a large query into smaller ones using the SELECT INTO statement.

1) Print the property location of the properties that have size larger than 1000 sq feet and are of the type "Residential Home".
2) Find the properties that were listed in the last 10 days and print the location and the size of each such property. Each property must be printed only once.
3) For each agent, find the property with the largest asking price where this agent is a selling agent, and print the agent name and the property location.
4) Find the agent(s) that have listed (as a selling agent) property of all the types and print the agent name(s).
5) Find the offers with the maximum offer amount for a property such that the offer amount was larger than the asking price and print the customer name for this offer and the location of the property.

Problem 2 (Based on Exercise 5.10) (Constraints) (30 points)

Consider the following database schema:

Emp(eid: integer, ename: string, age: integer, salary: real)
Works(eid: integer, did: integer, pcttime: integer)
Dept(did: integer, dname: string, budget: real, managerid: integer)

The Emp and Dept relations represent employee entities and department entities with their respective attributes. The domain of each field is listed after the field name. The Works relation represents the relationships of an employee working in a department. In this schema, an employee can work in more than one department; the pcttime field shows the percentage of time that a given employee works in a given department. We assume that pct time is always greater than 0.

Write SQL integrity constraints (domain, key, foreign key, or CHECK constraints; or assertions) or SQL triggers to ensure each of the following requirements, considered independently. You have to give the CREATE TABLE statements for each relation and you can modify it to add constraints if needed.

1) Employees must make a minimum salary of $2000.
2) The total percentage of all appointments for an employee must be less than 100%.
3) The budget of each department should not exceed the 20% of the total budget of the company.
4) When an employee who is a manager for a department is fired, the employee with the highest salary for this department should become the new (temporary) manager.

Problem 3 (SQL) (35 points)

Consider again the following database schema:

Emp(eid: integer, ename: string, age: integer, salary: real)
Works(eid: integer, did: integer, pcttime: integer)
Dept(did: integer, dname: string, budget: real, managerid: integer)

Write the SQL queries for the following questions:

1) Print the name and age of each employee who works in the ‘Legal’ department. The output should list the name first, then the age, and order the results first by name and then by age.
2) Print the names of the employees who work in the ‘Software’ department more than 60% of time. Print the output in sorted order of name.

3) Print the did(s) of the departments with at least two employees. Print the output in sorted order of did. Do not output duplicates (every did in the result should appear once).
4) Print the names of manager(s) who manage the department(s) with the largest budget.
5) Print the names of managers who manage only departments with budgets larger than 1 million, but at least one department with budget less than 5 million.