# Santiago Bermudez

# ID: 301118090

# CSC 134-02

# Assignment 3

# Due: 3/13/22

# Assignment 3 (100 points)

## Complex SQL Statements

**Due at 11:59 pm, Thursday, March 10, 2022**

**Question 1 (30 pts).** Consider the following relational schema and write SQL statements to create the related tables.

Assumption: one employee can work for multiple departments and one department may have multiple employees, of course.

**Emp (*eid:* integer, *ename:* string, *age:* integer, *salary:* real)**
**Dept (*did:* integer, *budget:* real, *manager_id:* integer)**
**Works (*eid:* integer, *did:* integer, *work_ time:* integer)**

Note that you should define all the primary keys and foreign keys and enforce the entity integrity constraints.

**CREATE TABLE** EMP

| | | |
|---|---|---|
| (Eid | INT | **NOT NULL**, |
| Ename | VARCHAR(36), | |
| Age | INT, | |
| Salary | REAL, | |
| **PRIMARY KEY** (Eid)); | | |

**CREATE TABLE** DEPT

| | | |
|---|---|---|
| (Did | INT | **NOT NULL**, |
| Budget | REAL, | |
| Manager_id | INT | **NOT NULL**, |
| **PRIMARY KEY** (Did), | | |

**FOREIGN KEY** (Manager_id) **REFERENCES** EMP(Eid));

**CREATE TABLE** WORKS
    (Eid               INT                           **NOT NULL**,
    Did               INT                           **NOT NULL**,
    Work_time     INT,
    **PRIMARY KEY** (Eid, Did),
    **FOREIGN KEY** (Eid) **REFERENCES** EMP(Eid),
    **FOREIGN KEY** (Did) **REFERENCES** DEPT(Did));

**Question 2 (60 pts).** The following relations keep track of information for an airline flight database:



- **Flight (*flno:* integer, *from_city:* string, *to_city:* string, *distance:* integer, *departs:* datetime, *arrives:* datetime, *price:* real)**
- **Aircraft (*aid:* integer, *aname:* string, *cruising_range:* integer)**
- **Certified (*eid:* integer, *aid:* integer)**
- **Employee (*eid:* integer, *ename:* string, *salary:* integer)**

Note that the Employee relation describes pilots and other kinds of employees as well; every pilot is certified for some aircraft, and only pilots are certified to fly. Write each of the following queries in SQL.

1. For each pilot who is certified for more than three aircrafts, find the *eid* and the maximum *cruising_range* of the aircraft for which she or he is certified.

| | |
|---|---|
| **SELECT** | P.Eid, **MAX**(A.Cruising_range) |
| **FROM** | CERTIFIED P, AIRCRAFT A |
| **WHERE** | P.Aid = A.Aid |
| **GROUP BY** | P.Eid |
| **HAVING** | **COUNT**(*) > 3; |

2. Find the names of employees whose salary is more than twice the price of the most expensive flight from 'Sacramento, CA' to 'Chicago, IL'.

| | |
|---|---|
| **SELECT** | **DISTINCT** E.Ename |
| **FROM** | EMPLOYEE E |
| **WHERE** | E.Salary > 2* (**SELECT**     **MAX**(F.Price) |

                                       **FROM**       Flight F
                                         **WHERE**      F.From_city = 'Sacramento, CA' **AND** F.To_city = 'Chicago, IL');

3. Find the names of pilots certified for some 'Boeing' aircraft. (hint: consider substring pattern matching)

| | |
|---|---|
| **SELECT** | **DISTINCT** E.Ename |
| **FROM** | EMPLOYEE E, CERTIFIED P, AIRCRAFT A |
| **WHERE** | E.Eid = P.Eid **AND** P.Aid = A.Aid **AND** A.Aname **Like** |

'%Boeing%';

4. Find the *aid*s of all aircrafts that can be used on ALL the routes from 'Miami, FL' to 'Sacramento, CA'.

| | |
|---|---|
| **SELECT** | A.Aid |
| **FROM** | AIRCRAFT A |
| **WHERE** | A.Cruising_range > (**SELECT**     **MIN**(F.Distance) |

                                             **FROM**       FLIGHT F
                                             **WHERE**      F.From_city = 'Miami, FL' **AND** F.To_city = 'Sacramento, CA');

```
SELECT       A.Aid
FROM         AIRCRAFT A
WHERE        A.Cruising_range > (SELECT       MAX(F.Distance)
                                 FROM         FLIGHT F
                                 WHERE        F.From_city = 'Miami,
             FL' AND F.To_city = 'Sacramento, CA');
```

5. Find the cheapest price, the highest price and the count of the flights that fly from 'Sacramento, CA' to either 'San Diego, CA' or 'Los Angeles, CA'.

```
SELECT       MIN(F.Price), MAX(F.Price), COUNT(*)
FROM         Flight F
WHERE        F.From_city = 'Sacramento, CA' AND (F.To_city = 'San
Diego OR F.To_city = 'Los Angeles, CA'));
```

6. Find the name of each employee who is not certified for any aircrafts.

```
SELECT       E.Ename
FROM         EMPLOYEE E
WHERE        E.Eid NOT IN      (SELECT       C.Eid
                               FROM          CERTIFIED C);
```

**Question 3 (10 pts).** Based on the schema in question 2, write a SQL statement to add a new attribute "age" into relation Employee.

**ALTER TABLE** EMPLOYEE **ADD** Age INT;

Or

**ALTER TABLE** EMPLOYEE **ADD** [column] Age INT;

**Deliverables**

   1. A doc or pdf file containing all your answers.

**Requirements on deliverables**

*1.* Your deliverable should be ***FLastname_A3.doc*** or ***FLastname_A3.pdf*** where *F* indicates first letter, in uppercase, of your firstname and *Lastname* indicates your last name where first letter is in uppercase. Please exactly follow the naming rule described above. You will be deducted 5 points for incorrect naming.

2. On the first page, clearly state your name, ID, course title, assignment number, and due date.

3. Submit your doc or pdf file via Canvas.

4. No late submission will be accepted.

5. When grades are returned to you on Canvas, you have 7 days to meet with the instructor for grade changes. Issues and/or disagreements concerning your grade must be resolved in such 7 days window. After 7 days, the grades are written in stone and can't be changed after that point, for whatever reason.