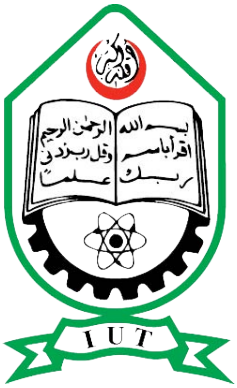
Islamic University of Technology



Database Management Systems Lab

CSE 4308 / CSE 4174

Lab 4

Author:
Ishmam Tashdeed
Md. Tariquzzaman
Zannatun Naim Sristy
CSE, IUT

Contents

1 JOIN operator **2**

2 Aggregation Function **2**

2.1 Group By 3

2.2 Having 3

3 Lab Task **5**

3.1 Database Schema 5

3.1.1 `vehicle` Table 5

3.1.2 `owner` Table 5

3.2 Data Insertion 6

3.2.1 `vehicle` Table Data 6

3.2.2 `owner` Table Data 6

3.2.3 `owner_vehicle` Table Data 7

3.3 Tasks 7

1 JOIN operator

In this lab, we will only be working with "Inner Joins". Inner Join is the simplest and most common type of join. It returns all rows from multiple tables where the join condition is met. The statement for declaring a inner join is as follows:

```
SELECT column1, column2, ...
FROM table1 INNER JOIN table2
ON table1.column = table2.column;
```

For example, joining `STUDENT` table and `DEPARTMENT` table based on a common column named `DEPT_NAME`.

```
SELECT *
FROM STUDENT INNER JOIN DEPARTMENT
ON STUDENT.DEPT_NAME = DEPARTMENT.DEPT_NAME;
```

In this case, we are just taking a Cartesian product between both tables and only showing the entries that match their common columns. `INNER JOIN` can be written by:

```
SELECT *
FROM STUDENT, DEPARTMENT
WHERE STUDENT.DEPT_NAME = DEPARTMENT.DEPT_NAME;
```

2 Aggregation Function

Aggregate functions are functions that take a collection (a set or multi-set) of values as input and return a single value. SQL offers five built-in aggregate functions:

- Average: `AVG`
- Minimum: `MIN`
- Maximum: `MAX`
- Total: `SUM`
- Count: `COUNT`

The input to `AVG` and `SUM` must be a collection of numbers, but the other operators can operate on collections of non-numeric data types, such as strings, as well. The syntax is following:

```
SELECT [AVG/MIN/MAX/SUM/COUNT](attribute_1)
FROM EMPLOYEE;
```

For example,

```
SELECT AVG(SALARY)
FROM EMPLOYEE;
```

2.1 Group By

There are circumstances where we would like to apply the aggregate function not only to a single set of tuples but also to a group of sets of tuples; we specify this wish in SQL using the `GROUP BY` clause. The attribute or attributes given in the group by clause are used to form groups. Let's say, now we want to know the average salary of each department. So, the statement will be:

```
SELECT DEPT, AVG(SALARY)
FROM EMPLOYEE
GROUP BY DEPT;
```

All selected attributes must be present either in aggregate functions or in group by clause, otherwise the query will be considered erroneous. For example, the following query is incorrect as `ID` appears in the `SELECT` statement, but not in the `GROUP BY` clause:

```
SELECT DEPT, ID, AVG(SALARY)
FROM EMPLOYEE
GROUP BY DEPT;
```

2.2 Having

The `HAVING` clause is used to specify conditions on groups rather than on tuples. For example, we might be interested in only those departments where the average salary of the employees is more than $5000. We cannot use the `WHERE` clause in this case because `WHERE` is used to specify conditions on a single set of tuples, whereas in this case, we are interested in finding a group of departments where the average salary is more than $5000.

```sql
SELECT DEPT, AVG(SALARY)
FROM EMPLOYEE
GROUP BY DEPT
HAVING AVG(SALARY)>5000;
```

As was the case for the SELECT statement, any attribute that is present in the HAVING clause without being aggregated must appear in the GROUP BY clause, otherwise the query is treated as erroneous.

3 Lab Task

3.1 Database Schema

3.1.1 `vehicle` Table

Column	Data Type	Description
`vehicle_id`	`INT`	Primary key
`model`	`VARCHAR(50)`	model of the vehicle
`type`	`VARCHAR(20)`	Type of the vehicle (e.g., Sedan)
`horsepower`	`INT`	Horsepower of the vehicle
`torque`	`INT`	Torque of the vehicle
`weight`	`INT`	Weight of the vehicle
`speed`	`INT`	Top speed of the vehicle

Table 1: `vehicle` Table Schema

3.1.2 `owner` Table

Column	Data Type	Description
`owner_id`	`INT`	Primary key
`first_name`	`VARCHAR(30)`	First name of the owner
`last_name`	`VARCHAR(30)`	Last name of the owner
`city`	`VARCHAR(30)`	City where the owner resides

Table 2: `owner` Table Schema

3.2 Data Insertion

3.2.1 `vehicle` Table Data

vehicle_id	model	type	horsepower	torque	weight	speed
1	Toyota Corolla	Sedan	132	128	2800	112
2	Honda Civic	Sedan	158	138	2760	125
3	Ford Fusion	Sedan	175	175	3431	125
4	Ford Mustang	Sports	310	350	3532	155
5	Chevrolet Camaro	Sports	275	295	3354	155
6	Dodge Challenger	Sports	305	268	3894	155
7	Honda CR-V	SUV	190	179	3337	112
8	Toyota RAV4	SUV	203	184	3370	114
9	Ford Explorer	SUV	300	310	4345	143
10	Tesla Model 3	Electric	283	307	3554	140
11	Tesla Model S	Electric	762	723	4647	155

Table 3: `vehicle` Table Data

3.2.2 `owner` Table Data

owner_id	first_name	last_name	city
1	John	Smith	New York
2	Sarah	Johnson	Los Angeles
3	Michael	Williams	Chicago
4	Emily	Brown	New York
5	David	Jones	Houston

Table 4: `owner` Table Data

3.2.3 `owner_vehicle` Table Data

owner_id	vehicle_id
1	4
1	5
1	7
1	10
2	8
2	9
3	4
4	1
4	5
6	8
7	4
8	5

Table 5: `owner_vehicle` Table Data (Adjusted)

3.3 Tasks

1. Create and insert into the tables.
2. Find the average `horsepower` of the cars.
3. Find the vehicle `type` with the highest average horsepower.
4. Find which `type` has the most number of cars.
5. Display the vehicles that are heavier than the average `weight`.
6. List the `first_names` of customers who bought a "Tesla".
7. Show the **full names** of owners and the `model` of their vehicles, separated by the word 'owns'. E.g. "John Smith owns Ford Mustang."
8. For each vehicle `type`, find the average `torque` and list only those types where the average `torque` is greater than 250.
9. List all `cities` that has more than one "Sedans" or "Sports cars".

10. Sort the list of vehicles made by "Ford" by their `speed` in descending order.

11. List the name of the owner who has the slowest car.

12. Find an "SUV" that is faster than **any** "Sedan".

13. Identify all owners who possess "Electric" `type` vehicles.

14. List owners who own the same vehicle `model`.

15. Show which owner has the slowest "Sports" car.

16. Bonus:

    (a) Find the total number of vehicles owned by owners from "New York".

    (b) List the names of people who own more vehicles than the average number of vehicles per owner.