Mohammad Ali Jinnah University Karachi

Department of Computer Science

LAB MANUAL

CS2231: Database Management System

LAB 10

Instructor
Safiyah Batool

LAB 10

Schema

Doctors (docid, docname, docdept, docfee)
Patients (patid, patname, patage, pathistory)
Visits (docid, patid, no_of_visits, date_firstvisit)

PRE-STEPS

1- Create tables according to the schema provided above. Apply proper constraints.
2- Insert 10 rows in each of the doctors and patients tables and 20 rows in the visits table.

SQL AGGREGATE FUNCTIONS

SQL aggregate functions are used to have summarizations over large data.

- **COUNT()**

The SQL COUNT function returns the number of rows in a table satisfying the criteria specified in the WHERE clause.

Syntax:

// Display the count of rows in the doctors table.

```
SELECT    COUNT (*)
FROM      Doctors;
```

// Display the count of rows for a specific column. The result may be different from the above query, BECAUSE some column may have NULL value in some row.

INSERT INTO doctors (docid, docname, docfee) VALUES ('d14', 'hameed', 1200); //**insert data**

```
SELECT    COUNT (*)
FROM      Doctors;
```

```
SELECT    COUNT (docdept)
FROM      doctors;
```

// Display the count with an alias used as the column title

```
SELECT    COUNT (docdept)    AS    number_of_departments
FROM      doctors;
```

// you would surely like to remove the duplicate values☺, so use DISTINCT

```
SELECT    COUNT (DISTINCT  docdept)    AS    number_of_departments
FROM      doctors;
```

//you may like to see the entries for a specific department, so use WHERE Clause

```
SELECT    COUNT (docdept)    AS    cardio_entries
FROM      doctors
```

```
WHERE       docdept = 'cardio';
```

// you may wish to see the entries for each department separately☺, so use GROUP BY clause

```
SELECT      docdept, COUNT (docdept)        AS      number_of_doctors
FROM        doctors
GROUP BY    docdept;
```

// you may wish see the entries for those departments, who have more than 2 entries in the table.

```
SELECT      docdept, COUNT (docid)          AS      number_of_doctors
FROM        doctors
GROUP BY    docdept
HAVING      COUNT (docid) > 2;
```

QUESTION: What's the difference between the above two queries (at query level + at output level)

- **SUM()**

The SQL SUM function is used to return the sum of the values in a column or sum of an expression in a SELECT statement. [The column must contain numbers ☺]

Syntax:

```
SELECT      SUM (docfee)
FROM        doctors;
```

// an expression can also be used in the sum function

```
SELECT      SUM (docfee*2)
FROM        doctors;
```

// DISTICNT can be used to add only unique values

```
SELECT      SUM (DISTINCT  docfee)
FROM        doctors;
```

- **AVG()**

The SQL AVERAGE function returns the average of the column values.

```
SELECT      AVG (docfee)
FROM        doctors;
```

// you may wish to see the rounded value ☺

```
SELECT      ROUND (AVG (docfee))    AS      Rounded_Average
FROM        doctors;
```

- **MAX()**

The SQL MAX function returns the maximum of the column values.

```
SELECT      MAX (docfee)
```

```
FROM        doctors;
```

- **MIN ()**

The SQL MIN function returns the minimum of the column values.

```
SELECT      MIN (docfee)
FROM        doctors;
```

SQL ARITMETIC FUNCTIONS

A mathematical function executes a mathematical operation usually based on input values that are provided as arguments, and return a numeric value as the result of the operation. Mathematical functions operate on numeric data.

- **ABS ()**

This SQL ABS() returns the absolute value of a number passed as argument.

INSERT INTO doctors VALUES ('d16', 'farhan', 'TB', 1225); //**insert required data**

Syntax:

```
SELECT      ABS ( AVG (docfee))
FROM        doctors;
```

- **CEIL ()**

This SQL CEIL() will rounded up any positive or negative decimal value within the function upwards.

Syntax:

```
SELECT      CEIL ( AVG (docfee))
FROM        doctors;
```

- **FLOOR ()**

The SQL FLOOR() rounded up any positive or negative decimal value down to the next least integer value. SQL DISTINCT along with the SQL FLOOR() function is used to retrieve only unique value after rounded down to the next least integer value depending on the column specified.

Syntax:

```
SELECT      FLOOR ( AVG (docfee))
FROM        doctors;
```

- **MOD ()**

This SQL MOD() function returns the remainder from a division.

Syntax:

```
SELECT      d1.docid, d2.docid, d1.docfee, d2.docfee, MOD (d1.docfee, d2.docfee)
FROM        doctors  d1,  doctors        d2
WHERE       d1.docid < > d2.docid;
```

LAB ACTIVITY

1- Write a query to find the number of patients who have visited any doctor on 30-JAN-1981.
2- Write a query to find the number of patients who have been diagnosed by doctor ‘d01’.
3- Write a query to find the most elder patient.
4- Write a query to find the number of patients affected by each disease.

HOME TASK

1- Write a query to find the number of patients according to age group. Only display the age groups under 40.
2- Write a query to find the number of visits made by each patient. Display the data in sorted order of patients.
3- Write a query to find the average age of cardio patients.
4- Write a query to find the average age of patients affected by each disease. Display the rounded value of the average.
5- Write a query to find the sum of doctors’ fee for doctors in each department.
6- Write a query to find the most recent date on which a visit has been made by some patient.