

4.1)  
**SELECT** \*  
**FROM** tablename;

- a) Show all data from **countries** table.
- b) Show all data from **employees** table.
- c) Show all data from **departments** table.
- d) Show all data from **job\_history** table.

4.2)  
**SELECT** \*  
**FROM** tablename  
**WHERE** condition;

**Operator>> LIKE / NOT LIKE**

- a) Show those employees details whose first name starts with the letter **s**
- b) Show those employees details whose first name doesn't starts with the letter **s**
- c) Show those employees details whose first name ends with the letter **a**
- d) Show those employees details whose first name contains the letter **d**
- e) Show those employees details whose first name contains **da** as substring
- f) Show those employees details whose first name starts with **s** and ends with **a**
- g) Show those employees details whose first name either starts with **s** or starts with **m**
- h) Show those employees details whose first name contains the letter **o** and **a**
- i) Show those employees details whose first name contains a letter **o** followed by a letter **a**
- j) Show those employees details whose first name consists of exactly 3 chars
- k) Show those employees details whose first name consists of minimum 3 characters
- l) Show those employees details whose first name contains the letter **a** from the second last position

**Operator>> IN() / NOT IN()**

- a) Show those countries details whose country\_id is AU/BR/CN/JP
- b) Show those departments details whose manager\_id is not 204/100/145
- c) Show those employees details whose job\_id is ST\_MAN/IT\_PROG
- d) Show those employees details who is not assigned to the department\_id 100/30/90
- e) Show those locations details whose postal\_code is either 2901/50090
- f) Show those locations details where the city is either Roma/Venice/Tokyo

**Operator>> BETWEEN ... AND... / NOT BETWEEN ... AND ...**

- a) Show those departments details whose location\_id is within the range 1000 to 2000 inclusive.
- b) Show those employees details whose salary is within the range 10000 to 20000 inclusive.
- c) Show those employees details whose hire\_date is within the range '1987-01-01' to '1987-06-30' inclusive.
- d) Show those jobs details where the difference between max\_salary and min\_salary is within the range 5000 to 10000 inclusive.
- e) Show those job\_history details where the end\_date is within '1998-12-01' to '1998-12-31' inclusive.
- f) Show those employees details whose department\_id is within the range 50 to 60 inclusive.

4.3)  
**SELECT** col1, col2\*5, col3+col4, **function**(col5), ... ..  
**FROM** tablename  
**[WHERE** condition];

**Mathematical Functions>>**

- a) Show each employees employee\_id, first\_name and his salary in "10 thousands 5 hundreds and 12 taka only" format. [use CONCAT function]

- b) For each jobs, show the job\_id, job\_title and how much greater the max\_salary then min\_salary in percentage. [show the output in 2 decimal points] [here, %greater=(max\_salary-min\_salary)\*100/min\_salary]
- c) Show each employees employee\_id, and yearly total salary with commission he receives. [show the floor value of the total salary] [here, yearly total salary = salary \* 12\* ( 1+(commission\_pct/100)) ]

**Date and Time Functions>>**

- a) Show each employees email, hire\_date in “January 4<sup>th</sup>, 1987” format.
- b) Show each employees email, hire\_date in “Jan 1987, 04” format.
- c) Show each employees email, hire\_date in “1<sup>st</sup> Aug, 87 05:10 PM” format.
- d) Show each employees email, hire\_date in “15 Jan, 2019 Tuesday 14:10” format.
  
- e) Show those employees first\_name, email, phone\_number, address who is hired after the date “05 May, 1987 00:00 AM”.
- f) Show those employees first\_name, email, phone\_number, address details who is hired before the date “1<sup>st</sup> June 1987 12:01 PM”.
  
- g) Show each employees employee\_id and his total current experience(upto present date) in number of years format. [show the no of years in 3 decimal points.]
- h) Show each employees employee\_id, email and his total current experience in “10 years, 6 months and 15 days” format.
- i) Show those employee\_id s from the job\_history table whose total experience is greater than 5 years.
  
- j) For each job\_history, show how many days an employee have served during his last month of retirement.
- k) Show how many days an employee have served during his first month of joining.
  
- l) Show those employees details who have been hired in the leap day(19<sup>th</sup> Feb) of any leap year.
- m) Show those employees details whose hiring month is either 2/4/6/8.
  
- n) Show each employees first\_name, department\_id, manager\_id and his modified join\_date that is one week before the his real join\_date.
- o) From the employees table, show each employees employee\_id, join\_date and estimated retirement date that is 30 years after his join\_date.

**String Functions>>**

- a) Show each employees short name that is first 3 letters from the first name followed by an underscore and then followed by the first 3 letters from last\_name.
- b) Show those employees names that is a palindrome.
- c) Show each employees employee\_id and email (join '@gmail.com' in each email).
- d) Show each employees first\_name and phone\_number(format: 515.xxx.xxx7).
- e) Show each employees employee\_id, email and full\_name (format: first\_name<SPACE>last\_name and right pad the string if total no of characters is less than 20)

4.4)

```
SELECT *|col1, col2*5, col3+col4, function(col5), ... .. .
FROM tablename
[WHERE condition]
ORDER BY col1 [ASC|DESC], col2 [ASC|DESC], ... .;
```

- a) Show employees first\_name, last\_name, email, hire\_date, salary in descending order of salary and ascending order of first\_name.
- b) Show all the locations details in ascending order of postal\_code.

4.5)

```
SELECT [DISTINCT] col1, col2*5, col3+col4, function(col5), ... ..  
FROM tablename  
[WHERE condition]  
[ORDER BY col1 [ASC|DESC], col2 [ASC|DESC], ... ..]
```

- a) Show all the distinct manager\_id s from employees table.
- b) Show all the distinct job\_id s from the employees table.
- c) Show all the distinct country\_id s from the locations table.
- d) Show all the distinct country\_id s and department\_id s from employees table.

4.7)

```
SELECT [DISTINCT] col1, col2*5 [AS 'newcol2'], col3+col4 [AS 'newcol3'], function(col5)  
[AS 'newcol4'], ... ..  
FROM tablename  
[WHERE condition]  
[ORDER BY col1 [ASC|DESC], col2 [ASC|DESC], ... .. ]  
LIMIT [offset,] rowcount;
```

- a) Show the highest salary holder employee details from employees table.
- b) Show the top 10 experienced employee details from employees table.
- c) Show the 2<sup>nd</sup> highest salary range(max\_salary-min\_salary) job details from the jobs table. (suppose all the ranges are unique).