4. SQL

Q.1 Explain data definition language (DDL)

⇒ i) To create database schema and database objects like table DDL is used.

ii) DDL statements are used to build and modify the structure of table and other objects in the database

iii) The set of DDL commands are -

a) CREATE
b) ALTER
c) DROP
d) RENAME
e) TRUNCATE

a) CREATE Statements

⇒ CREATE statements are used to create new database objects like table, index and others.

- CREATE TABLE is the command in database system is used to create a new table with unique name.
- It's syntax is -

```
CREATE TABLE <table_name>
(
    column1_name    datatype,
    column2_name    datatype,
    ....
    columnN_name    datatype
);
```

- For e.g.,

```
CREATE TABLE EMPYOEE (
    emp_id      int NOT NULL,
    emp_name    varchar2(20),
    emp_address varchar2(20),
    emp_age     int,
    salary      DECIMAL (18,2)
);
```

b) ALTER TABLE

⇒ Once object is created in database, we may require ALTER command to update structure of database object.

- ALTER statement can be used to add, delete, or modify columns in an existing table.
- It's syntax is -

```
ALTER TABLE <table_name>
  ADD column_name datatype;

ALTER TABLE <table_name>
  MODIFY column_name datatype;

ALTER TABLE <table_name>
  DROP column_name;
```

- For e.g.:

```
ALTER TABLE Employee
    ADD E_DOB Date ;
```

c) DROP TABLE

⇒ DROP command can be used to remove database objects from user database.

- The SQL DROP TABLE statement is used to remove a table definition and all related data like indexes, triggers, constraints.
- It's syntax is -

```
DROP TABLE <table_name> ;
```

- For e.g.;

```
DROP TABLE Employee;
```

d) RENAME TABLE

⇒ It is possible to change name of table with or without data in it using simple RENAME command.

- It's Syntax -

```
RENAME TABLE <table_name> to <new_name>
```

- For eg.;

```
RENAME TABLE EMPLOYEE TO EMP;
```

e) TRUNCATE TABLE

⇒ The Truncate table command is used to delete all the data from an existing table.

- It's syntax is -

```
TRUNCATE TABLE <table-name>;
```

- For e.g.:

```
TRUNCATE TABLE EMP;
```

Q.2 Explain Data Manipulation language (DML)

⇒ i) DML statements are used to manipulating or managing data in database.

ii) DML commands are not auto-committed like DDL.

iii) It means changes done by DML command can be rolled back.

iv) DML is set of command used to -

a) INSERT
b) DELETE
c) UPDATE

a) INSERT Statement

⇒ Insert statements are used to add record to the existing table.

- To insert data into a table, SQL INSERT INTO commands can be used.
- It's syntax is -

```
INSERT INTO <table_name> (column1, ...., columnN)
VALUES (column1, ...., columnN);
```

- For e.g.,

```
INSER INTO EMP (E_name, E_id)
VALUES ('Mahesh', 1001);
```

b) DELETE Statement

⇒ Delete statements is used to delete some or all records from the existing table.

- To delete data into a table, SQL DELETE command can be used.
- It's Syntax -

```
DELETE
FROM <table_name>;
```

- To ~~select~~ delete selected rows from table we can specify where condition

⇒

```
DELETE
FROM <table_name>
WHERE <condition>;
```

- For eg;

```
DELETE
FROM EMP
WHERE Eid is NULL;
```

c) UPDATE Statements

⇒ The UPDATE statements is used to modify the existing data present in a table.

- The UPDATE data in a table, SQL UPDATE command can be used.
- It's syntax -

```
UPDATE <table_name>
SET column1 = new_value;

OR

UPDATE <table_name>
SET Column1 = new_value;
WHERE condition;
```

- For e.g.:

```
UPDATE Employee
SET Eid = 1002
WHERE name = 'Suhas';
```

Q.3 Explain Data Control language (DCL)

⇒ i) DCL commands is used to control various user actions in database.

a) GRANT

⇒ SQL GRANT command is specifically used to provide privileges to database objects for an user.

- Syntax -

GRANT privilege_name ON object_name to user_name

- For e.g.,

GRANT ALL PRIVILEGES ON EMP TO U2

b) REVOKE

⇒ Revoke command withdraw user privileges on database objects if any granted.

- Syntax -

REVOKE privilege_name on object_name From user_name

- For eg.,

REVOKE INSERT ON Department From U1

Q.5 Explain Aggegrate function.

⇒ i) Sometimes for decision making we need summarize data from table like average, sum, average, etc.

ii) SQL provides aggegrate functions which can summarize data of given table.

iii) Such queries are generally used for producing reports and summary forms in an application

iv) Types of Aggegrate Functions:

a) Count ([DISTINCT] c)
b) SUM ([DISTINCT] c)
c) AVG ([DISTINCT] c)
d) MIN (c)
e) MAX (c)

a) COUNT ()

⇒ This function is used to calculate number of rows in a table selected by query.

- Count returns the no. of rows in the table when the column value is not NULL
- Column in the query must be numeric
- It's Syntax -

  COUNT (* OR column_name)

- For e.g.,

```
SELECT  COUNT (Sid)  as Count
FROM    exam_Marks;
```

b) SUM()

⇒ This function is used to calculate sum of the column values in a table selected by query.

- Column in the query must be numeric.
- It's syntax -

SUM (* | column-name)

- For e.g.:)

```
SELECT  SUM(Marks) as Sum
FROM    Exam_marks;
```

c) AVG()

⇒ This function is used to calculate average of the column values in a table selected by query.

- This function first calculates sum of column and then divide by total number of rows.
- AVG returns the average of all the values in the specified column.
- Column ~~must~~ in the query must be numeric.
- For e.g.:)

```
SELECT  AVG(Marks) as AVG
FROM    EXAM_MARKS;
```

d) MIN()

⇒ This function is used to ~~calculate~~ find the minimum value out of column values in the table selected by query.

- Column in the query need not be numeric data type.
- For e.g.;

```
SELECT  MIN(marks) as Min
FROM    Exam_marks;
```

e) MAX()

⇒ This function is used to find the maximum value out of column values in table selected by query.

- Column in the query need not be numeric data type.
- For eg.;

```
SELECT  MAX (Marks)  as Max
FROM    Exam_marks;
```

Q.6 Short note on : Trigger.

⇒ i) Trigger is a procedure that is automatically invoked by the DBMS in response to specific alteration to the database or table in database.

ii) Trigger are stored in database as a simple database object.

iii) A database that has a set of triggers is called an active database.

iv) A database trigger enables DBA to create relationship between separate database.

v) Components of Trigger (E-C-A) Model

⇒

a) Event (E) - SQL statements that cause the trigger to fire. This event may be insert, update or delete operation database table.

b) Condition (C) - A condition that must be satisfied for execution of trigger.

c) Action (A) - This is code or statement that executes when triggering condition is satisfied and trigger is activated on database table

vi) It's syntax is-

```
CREATE TRIGGER <trigger_name>
[<ENABLE | DISABLE]
<BEFORE | AFTER>
<INSERT | UPDATE | DELETE>
ON <table_name>
```

[FOR EACH ROW]

```
DECLARE
    <variable_definitions>
BEGIN
    <Trigger_code> ;
END;
```

vii) There are two types of triggers -

a) Row level trigger

⇒ A row level trigger is fired each time the table is affected by triggering statement.

- IF FOR EACH ROW clause is written that means trigger is row level trigger.

b) Statement level trigger

⇒ A statement level trigger is fired only once on behalf of the triggering statement, irrespective of number of rows in the table that are affected by triggering statement.

- It's default trigger occurs when FOR EACH ROW is not specified.

For e.g.)

```
CREATE TRIGGER student_marks
AFTER INSERT ON STUDENT
FOR EACH ROW
UPDATE STUDENT
```

```
SET TOTAL = sub1 + sub2 + sub3,
    per = (TOTAL * 100) /300;
```