**SQL STATEMENTS**

**I.DATA DEFINITION LANGUAGE:**

**1.CREATE :**

SYNTAX:

CREATE TABLE table\_name

(

COLUMN\_NAME1 DATATYPE NOT NULL / NULL, COLUMN\_NAME2 DATATYPE NOT NULL / NULL, .

.

COLUMN\_NAMEn DATATYPE NOT NULL / NULL, CONSTRAINT constraint\_ref\_name UNIQUE(COLUMN\_NAME), CONSTRAINT constraint\_ref\_name CHECK(CONDITION), CONSTRAINT constraint\_ref\_name PRIMARY KEY(COLUMN\_NAME), CONSTRAINT constraint\_ref\_name FOREIGN KEY(COLUMN\_NAME) REFERENCES parent\_table\_name (COLUMN\_NAME)

);

**2.RENAME:**

SYNTAX:

RENAME current\_table\_name TO New\_name;

**3.ALTER:**

SYNTAX:

1.TO ADD A COL :

ALTER TABLE table\_name

ADD COLUMN\_NAME DATATYPE[NULL/NOT NULL];

2.TO DROP A COL :

ALTER TABLE table\_name

DROP COLUMN COLUMN\_NAME ;

3.TO CHANGE THE DATATYPE:

ALTER TABLE table\_name

MODIFY COLUMN\_NAME new\_datatype;

4.TO CHANGE THE NOT NULL CONSTRAINT:

ALTER TABLE table\_name

MODIFY COLUMN\_NAME exixting\_datatype NULL/NOTNULL; 5.TO RENAME THE COLUMN:

ALTER TABLE table\_name

RENAME COLUMN current\_name TO new\_name;

6.TO MODIFY CONSTRAINTS:

a) ALTER TABLE table\_name

ADD CONSTRAINT constraint\_ref\_name UNIQUE(column\_name); b) ALTER TABLE table\_name

ADD CONSTRAINT constraint\_ref\_name CHECK(condition);

c) ALTER TABLE table\_name

ADD CONSTRAINT constraint\_ref\_name PRIMARY KEY(column\_name); d) ALTER TABLE table\_name

ADD CONSTRAINT constraint\_ref\_name FOREIGN KEY(column\_name) REFERENCES parent\_table\_name (column\_name);

7.TO DROP/DISABLE/ENABLE A CONSTRAINT:

ALTER TABLE table\_name

DROP/DISABLE/ENABLE CONSTRAINT constraint\_ref\_name;

**4. TRUNCATE:**

SYNTAX: TRUNCATE TABLE table\_name; **5. DROP:**

SYNTAX: DROP TABLE table\_name; TO RECOVER THE TABLE:(only in oracle) SYNTAX: FLASHBACK TABLE table\_name TO BEFORE DROP

[RENAME TO new\_name] TO DROP THE TABLE FROM RECYCLE BIN SYNTAX: PURGE TABLE table\_name;

**II.DATA MANIPULATION LANGUAGE 1.INSERT:**

SYNTAX 1: INSERT INTO table\_name VALUES (V1,V2,....,Vn); 2: INSERT INTO table\_name (COL1,COL2,....COLn) VALUES(V1,V2,....,Vn);

Or

INSERT INTO table\_name (COL1,COL2,....COLn) VALUES(&COL1,&COL2,...&COLn) 3. INSERT INTO table\_name

SELECT statement;

**2.UPDATE:**

SYNTAX: UPDATE table\_name

SET COL1=V1,COL2=V2,......,COLn=Vn [WHERE <filter\_condition>];

**3.DELETE:**

SYNATX : DELETE

FROM table\_name

[WHERE <filter\_condition>];

**III.TRANSACTION CONTROL LANGUAGE 1.COMMIT:**

SYNATX: COMMIT;

**2.SAVEPOINT:**

SYNATX: SAVEPOINT savepoint\_name;

**3.ROLLBACK:**

SYNATX: ROLLBACK;

ROLLBACK TO SAVEPOINT

SYNATX: ROLLBACK TO savepoint\_name;

**IV.DATA CONTROL LANGUAGE: 1.GRANT:**

SYNATX: GRANT sql\_statement ON table\_name TO user\_name;

**2.REVOKE:** :

SYNATX: REVOKE sql\_statement ON table\_name FROM user\_name;

**V.DATA QUERY LANGUAGE:**

**1.SELECT:**

SELECT \*/[DISTINCT] column\_name/Expression [ALIAS] **2.PROJECTION:**

SYNTAX: SELECT \*/[DISTINCT] column\_name/Expression [ALIAS] FROM table\_name ;

**3.SELECTION:**

SYNTAX: SELECT \*/[DISTINCT] column\_name/Expression [ALIAS] FROM table\_name

WHERE <filter\_condition> ;

**4.JOIN**

**1.CARTESIAN JOIN/CROSS JOIN**

SYNTAX:ANSI->

SELECT col\_name

FROM table\_name1 CROSS JOIN table\_name2; SYNTAX:ORACLE->

SELECT col\_name

FROM table\_name1, table\_name2;

**2.INNER JOIN/EQUI JOIN**

SYNTAX:ANSI->

SELECT col\_name

FROM table\_name1 INNER JOIN table\_name2 ON table\_name1.col\_name=table\_name2.col\_name; SYNTAX:ORACLE->

SELECT col\_name

FROM table\_name1, table\_name2

WHERE table\_name1.col\_name=table\_name2.col\_name;

**3.OUTER JOIN**

I. LEFT OUTER JOIN

SYNTAX:ANSI->

SELECT col\_name

FROM table\_name1 LEFT [OUTER] JOIN table\_name2 ON table\_name1.col\_name=table\_name2.col\_name;

SYNTAX:ORACLE->

SELECT col\_name

FROM table\_name1, table\_name2

WHERE table\_name1.col\_name=table\_name2.col\_name(+);

II. RIGHT OUTER JOIN

SYNTAX:ANSI->

SELECT col\_name

FROM table\_name1 RIGHT [OUTER] JOIN table\_name2 ON table\_name1.col\_name=table\_name2.col\_name;

SYNTAX:ORACLE->

SELECT col\_name

FROM table\_name1, table\_name2

WHERE table\_name1.col\_name(+)=table\_name2.col\_name;

III. FULL OUTER JOIN

SYNTAX:ANSI->

SELECT col\_name

FROM table\_name1 FULL [OUTER] JOIN table\_name2 ON table\_name1.col\_name=table\_name2.col\_name; NOTE: NO ORACLE SYNTAX FOR FULL OUTER JOIN

**4.SELF JOIN**

SYNTAX:ANSI->

SELECT col\_name

FROM table\_name1 T1 JOIN table\_name1 T2 ON T1.col\_name=T2.col\_name;

SYNTAX:ORACLE->

SELECT col\_name

FROM table\_name1 T1, table\_name1 T2 WHERE T1.col\_name=T2.col\_name;

**5.NATURAL JOIN**

SYNTAX:ANSI->

SELECT col\_name

FROM table\_name1 NATURAL JOIN table\_name2 ; NOTE: NO ORACLE SYNTAX FOR NATURAL JOIN