## DDL-DATA DEFINATION LANGUAGE

1. **CREATE-**

CREATE TABLE TABLE\_NAME

(

CLOUMN\_NAME1 DATATYPE [CONSTRAINT],

CLOUMN\_NAME2 DATATYPE [CONSTRAINT],

CLOUMN\_NAME3 DATATYPE [CONSTRAINT],

.................

.................

CLOUMN\_NAMEn DATATYPE [CONSTRAINT]);

(OR)

CREATE TABLE TABLE\_NAME

(

CLOUMN\_NAME1 DATATYPE NOT NULL/NULL,

CLOUMN\_NAME2 DATATYPE NOT NULL/NULL,

CLOUMN\_NAME3 DATATYPE NOT NULL/NULL,

...............

...............

CLOUMN\_NAMEn DATATYPE NOT NULL/NULL,

CONSTRAINT CONS\_REF\_NAME CONSTRAINT[UNIQUE,PRIMARY KEY, CHECK](COLUMN\_NAME);

CONSTRAINT CONS\_REF\_NAME FOREGIN KEY(COLUMN\_NAME) REFERENCES PARENT\_TABLE\_NAME(COLUMN\_NAME));

1. **RENAME-**

RENAME TABLE\_NAME TO NEW\_TABLE\_NAME;

1. **ALTER-**

**TO ADD A COLUMN-**

ALTER TABLE TABLE\_NAME

ADD COLUMN COLUMN\_NAME DATATYPE [CONSTRAINT];

**TO RENAME A COLUMN-**

ALTER TABLE TABLE\_NAME

RENAME COLUMN COLUMN\_NAME TO NEW\_COLUMN\_NAME;

**TO DROP A COLUMN-**

ALTER TABLE TABLE\_NAME

DROP COLUMN COLUMN\_NAME;

**TO MODIFY THE DATA TYPE-**

ALTER TABLE TABLE\_NAME

MODIFY COLUMN\_NAME NEW\_DATATYPE;

**TO MODIFY NOT NULL CONSTRAINT-**

ALTER TABLE TABLE\_NAME

MODIFY COLUMN\_NAME EXISTING [NULL/NOT NULL];

**TO ADD CONSTRAINTS**

**TO ADD UNIQUE CONSTRAINTS-**

ALTER TABLE TABLE\_NAME

ADD CONSTRAINT CONS\_REF\_COLUMNAME UNIQUE(COLUMN\_NAME);

**TO ADD CHECK CONSTRAINTS-**

ALTER TABLE TABLE\_NAME

ADD CONSTRAINT CONS\_REF\_COLUMNAME CHECK(COLUMN\_NAME);

**TO ADD PRIMARY KEY CONSTRAINTS-**

ALTER TABLE TABLE\_NAME

ADD CONSTRAINT CONS\_REF\_COLUMNAME PRIMARY KEY(COLUMN\_NAME);

**TO DROP CONSTRAINTS-**

ALTER TABLE TABLE\_NAME

DROP CONSTRAINT CONSTRAINT\_REF\_NAME;

**TO ADD FOREIGN KEY CONSTRAINTS-**

ALTER TABLE TABLE\_NAME

ADD CONSTRAINT CONS\_REF\_COLUMNAME FOREGIN KEY(COLUMN\_NAME) REFERENCES PARENT\_TABLE\_NAME(COLUMN\_NAME);

1. **TRANCATE-**

TRANCATE TABLE TABLE\_NAME;

1. **DROP-**

DROP TABLE TABLE\_NAME;

**USER\_CONSTARINT-**

SELECT \*

FROM USER\_CONSTARINTS

[WHERE <FILTER\_TABLE>];

**DUPLICATE TABLE-(WITH RECORDS)**

CREATE TABLE\_NAME

AS

SELECT SELECT \* FROM TABLE\_NAME;

**DUPLICATE TABLE-(WITH OUT RECORDS)**

CREATE TABLE\_NAME

AS

SELECT SELECT \* FROM TABLE\_NAME

WHERE FALSE\_CONDITION;

**FLASHBACK-[RECOVER THE TABLE]**

FLASHBACK TABLE TABLE\_NAME TO BEFORE DROP;

**PURGE-[TO DELETE THE TABLE FROM BIN]**

PURGE TABLE TABLE\_NAME;

**TO DELECT COMPLETELY FROM BIN-**

DROP TABLE TABLE\_NAME PURGE;

# 2. DML-DATA MANIPULATION LANGUAGE

1. **INSERT-**

INSERT INTO TABLE\_NAME VALUES(VALUE1,VALUE2,...........,VALUEN);

**INSERTING MULTIPLE RECORDS IN THE TABLE-**

INSERT INTO TABLE\_NAME VALUES(&COLUMN1,&COLUMN2,..................,&COLUMNNTH);

**INSERT PARTICULAR RECORDS IN THE TABLE-**

INSERT INTO TABLE\_NAME VALUES([VALUE1/’’],......................... ,[VALUENTH/’’]);

(OR)

INSERT INTO TABLE\_NAME (&COLUMN1,........&COLUMNNTH)VALUES(VALUE1,..........,VALUENTH);

**NOTE-**

ACCORDING TO THE NO.OF COLUMN PRESENT IN TABLES THE INSERTION TO BE TAKE PLACE.

1. **UPDATE-**

UPDATE TABLE\_NAME

SET COLUMN\_NAME1=’VALUE’,.................... ,COLUMN\_NAMENTH=’VALUE’

[WHERE <FILTER\_CONDITION>];

1. **DELETE-**

DELETE FROM TABLE\_NAME

[WHERE <FILTER\_CONDITION>];

# DCL-DATA CONTROL LANGUAGE

1. **GRANT-**

GRANT SQL\_STATEMENTS[SELECT,PROJECTION,ETC]

ON TABLE\_NAME

TO USER\_NAME;

1. **REVOKE-**

REVOKE SQL\_STATEMENT[SELECT,UPDATE,ETC]

ON TABLE\_NAME

FROM USER\_NAME;

# TCL-TRANSACTION CONTROL LANGUAGE

1. **COMMIT-**

COMMIT;

1. **ROLLBACK-**

ROLLBACK;

**ROLLBACK TO SAVEPOINT-**

ROLLBACK TO SAVEPOINT\_NAME;

1. **SAVEPOINT-**

SAVEPOINT SAVEPOINT\_NAME;