

**Aim:- To implement various DDL,DML commands and constraints.**

**THEORY:**

**DDL(Data Definition Language):** DDL or Data Definition Language actually consists of the SQL commands that can be used to define the database schema. It simply deals with descriptions of the database schema and is used to create and modify the structure of database objects in database.

Examples of DDL commands:

- **CREATE** – is used to create the database or its objects (like table, index, function, views, store procedure and triggers).
- **DROP** – is used to delete objects from the database.
- **ALTER**–is used to alter the structure of the database.
- **TRUNCATE**–is used to remove all records from a table, including all spaces allocated for the records are removed.
- **COMMENT** –is used to add comments to the data dictionary.
- **RENAME** –is used to rename an object existing in the database.

**DML(Data Manipulation Language):** The SQL commands that deals with the manipulation of data present in database belong to DML or Data Manipulation Language and this includes most of the SQL statements. Examples of DML:

- **SELECT** – is used to retrieve data from the a database.
- **INSERT** – is used to insert data into a table.
- **UPDATE** – is used to update existing data within a table.
- **DELETE** – is used to delete records from a database table.

**SQL constraints:** It are used to specify rules for the data in a table. Constraints are used to limit the type of data that can go into a table. This ensures the accuracy and reliability of the data in the table. If there is any violation between the constraint and the data action, the action is aborted.

Constraints can be column level or table level. Column level constraints apply to a column, and table level constraints apply to the whole table. The following constraints are commonly used in SQL:

- **NOT NULL** - Ensures that a column cannot have a NULL value
- **UNIQUE** - Ensures that all values in a column are different
- **PRIMARY KEY** - A combination of a NOT NULL and UNIQUE. Uniquely identifies each row in a table
- **FOREIGN KEY** - Uniquely identifies a row/record in another table
- **CHECK** - Ensures that all values in a column satisfies a specific condition
- **DEFAULT** - Sets a default value for a column when no value is specified

A) DDL(Data Definition Language):

1) CREATE:

SOURCE CODE:

```
1 CREATE TABLE student_narender(  
2   id NUMBER(5) PRIMARY KEY,  
3   Fname VARCHAR2(15),  
4   Lname VARCHAR2(20),  
5   Address VARCHAR2(50),  
6   DOB DATE);
```

OUTPUT:

```
Table created.
```

2) DROP:

SOURCE CODE:

```
DROP TABLE student_narender;
```

OUTPUT:

```
Table dropped.
```

3) ALTER:

SOURCE CODE:

```
ALTER TABLE student_narender ADD Email varchar2(55);  
desc student_narender;
```

OUTPUT:

```
Table altered.
```

TABLE STUDENT\_NARENDER

Column	Null?	Type
ID	NOT NULL	NUMBER(5,0)
FNAME	-	VARCHAR2(15)
LNAME	-	VARCHAR2(20)
ADDRESS	-	VARCHAR2(50)
DOB	-	DATE
EMAIL	-	VARCHAR2(55)

[Download CSV](#)

6 rows selected.

4) TRUNCATE:

SOURCE CODE:

```
TRUNCATE TABLE student_narender;
```

OUTPUT:

```
Table truncated.
```

5) RENAME:

SOURCE CODE:

```
RENAME student_narender TO student_vesit_narender;
```

OUTPUT:

```
Statement processed.
```

B) DML(Data Manipulation Language):

1) SELECT:

SOURCE CODE:

```
select * from student_narender;
```

OUTPUT:

ID	FNAME	LNAME	ADDRESS	DOB
1	narender	keswani	ulhasnagar	10-NOV-99
2	neel	deshmukh	vasai	31-JAN-00
3	hassan	haque	mahalaxmi	26-AUG-00
4	ronak	karia	majiwada	15-OCT-00
5	wilson	rao	thane	01-JAN-72

Download CSV

5 rows selected.

2) INSERT:

SOURCE CODE:

```
insert into student_narender values(1,'narender','keswani','ulhasnagar','10-NOV-1999');
insert into student_narender values(2,'neel','deshmukh','vasai','31-JAN-2000');
insert into student_narender values(3,'hassan','haque','mahalaxmi','26-AUG-2000');
insert into student_narender values(4,'ronak','karia','majiwada','15-OCT-2000');
insert into student_narender values(5,'wilson','rao','thane','01-JAN-1972');
```

OUTPUT:

1 row(s) inserted.

1 row(s) inserted.

1 row(s) inserted.

1 row(s) inserted.

1 row(s) inserted.

3) UPDATE:

SOURCE CODE:

```
UPDATE student_narender SET Lname = 'yadav' WHERE id = 4;
```

OUTPUT:

1 row(s) updated.

ID	FNAME	LNAME	ADDRESS	DOB
1	narender	keswani	ulhasnagar	10-NOV-99
2	neel	deshmukh	vasai	31-JAN-00
3	hassan	haque	mahalaxmi	26-AUG-00
4	ronak	yadav	majiwada	15-OCT-00
5	wilson	rao	thane	01-JAN-72

Download CSV

5 rows selected.

4) DELETE:

SOURCE CODE:

```
DELETE FROM student_narender WHERE id = 4;
```

OUTPUT:

1 row(s) deleted.

ID	FNAME	LNAME	ADDRESS	DOB
1	narender	keswani	ulhasnagar	10-NOV-99
2	neel	deshmukh	vasai	31-JAN-00
3	hassan	haque	mahalaxmi	26-AUG-00
5	wilson	rao	thane	01-JAN-72

[Download CSV](#)

4 rows selected.

C) SQL constraints:

1) PRIMARY KEY:

SOURCE CODE:

```

1 CREATE TABLE student_narender(
2   id NUMBER(5) PRIMARY KEY,
3   Fname VARCHAR2(15),
4   Lname VARCHAR2(20),
5   Address VARCHAR2(50),
6   DOB DATE);
7
8
9 desc student_narender;
10 insert into student_narender values(1,'narender','keswani','ulhasnagar','10-NOV-1999');
11 insert into student_narender values(2,'neel','deshmukh','vasai','31-JAN-2000');
12 insert into student_narender values(3,'hassan','haque','mahalaxmi','26-AUG-2000');
13 insert into student_narender values(4,'ronak','karia','majiwada','15-OCT-2000');
14 insert into student_narender values(5,'wilson','rao','thane','01-JAN-1972');
15
16 select * from student_narender;
17

```

OUTPUT:

Table created.

TABLE STUDENT\_NARENDER

Column	Null?	Type
ID	NOT NULL	NUMBER(5,0)
FNAME	-	VARCHAR2(15)
LNAME	-	VARCHAR2(20)
ADDRESS	-	VARCHAR2(50)
DOB	-	DATE

[Download CSV](#)

5 rows selected.

1 row(s) inserted.

1 row(s) inserted.

1 row(s) inserted.

1 row(s) inserted.

1 row(s) inserted.

ID	FNAME	LNAME	ADDRESS	DOB
1	narender	keswani	ulhasnagar	10-NOV-99
2	neel	deshmukh	vasai	31-JAN-00
3	hassan	haque	mahalaxmi	26-AUG-00
4	ronak	karia	majiwada	15-OCT-00
5	wilson	rao	thane	01-JAN-72

[Download CSV](#)

5 rows selected.

2) FOREIGN KEY:

**SOURCE CODE:**

```

20 CREATE TABLE supplier_narender (
21   supplier_id numeric(10) not null,
22   supplier_name varchar2(50) not null,
23   contact_name varchar2(50),
24   CONSTRAINT supplier_narender_pk PRIMARY KEY (supplier_id)
25 );
26
27 CREATE TABLE products_narender (
28   product_id numeric(10) not null,
29   supplier_id numeric(10) not null,
30   CONSTRAINT fk_supplier
31   FOREIGN KEY (supplier_id)
32   REFERENCES supplier_narender (supplier_id)
33 );
34
35
36 desc supplier_narender;
37 desc products_narender;
38
39 insert into supplier_narender values(1,'datta supplier','narender');
40 insert into supplier_narender values(2,'sai supplier','neel');
41 insert into supplier_narender values(3,'shiv supplier','hassan');
42 insert into supplier_narender values(4,'swami supplier','ronak');
43 insert into supplier_narender values(5,'vishnu supplier','wilson');
44
45 insert into products_narender values(1,2);
46 insert into products_narender values(2,5);
47 insert into products_narender values(3,1);
48 insert into products_narender values(4,3);
49 insert into products_narender values(5,4);
50
51 select * from supplier_narender;
52 select * from products_narender;
53

```

**OUTPUT:**

Table created.

Table created.

TABLE SUPPLIER\_NARENDER

Column	Null?	Type
SUPPLIER_ID	NOT NULL	NUMBER(10,0)
SUPPLIER_NAME	NOT NULL	VARCHAR2(50)
CONTACT_NAME	-	VARCHAR2(50)

[Download CSV](#)

3 rows selected.



TABLE PRODUCTS\_NARENDER

Column	Null?	Type
PRODUCT_ID	NOT NULL	NUMBER(10,0)
SUPPLIER_ID	NOT NULL	NUMBER(10,0)

[Download CSV](#)

2 rows selected.

1 row(s) inserted.

1 row(s) inserted.

1 row(s) inserted.

1 row(s) inserted.

1 row(s) inserted.

1 row(s) inserted.

1 row(s) inserted.

1 row(s) inserted.

1 row(s) inserted.

SUPPLIER_ID	SUPPLIER_NAME	CONTACT_NAME
1	datta supplier	narender
2	sai supplier	neel
3	shiv supplier	hassan
4	swami supplier	ronak
5	vishnu supplier	wilson

[Download CSV](#)

5 rows selected.

PRODUCT_ID	SUPPLIER_ID
1	2
2	5
3	1
4	3
5	4

Download CSV

5 rows selected.

### 3) UNIQUE CONSTRAINT:

#### SOURCE CODE:

```

55 CREATE TABLE employee_narender (
56   employee_id numeric (10) not null,
57   employee_name varchar2(50) not null,
58   CONSTRAINT employee_id_unique UNIQUE (employee_id)
59 );
60
61
62 desc employee_narender;
63 insert into employee_narender values(1,'narender');
64 insert into employee_narender values(1,'neel');
65
66 select * from employee_narender;
```

#### OUTPUT:

Unique Constraint is used to set unique value of the particular field

In this example, the e\_id is set as unique value

If same value of e\_id is inserted again then it will create an error.

Table created.

TABLE EMPLOYEE\_NARENDER

Column	Null?	Type
EMPLOYEE_ID	NOT NULL	NUMBER(10,0)
EMPLOYEE_NAME	NOT NULL	VARCHAR2(50)

Download CSV

2 rows selected.

1 row(s) inserted.

ORA-00001: unique constraint (SQL\_QAPUDNWFQHBNSQD3HLIYTXCB.EMPLOYEE\_ID\_UNIQUE) violated ORA-06512: at "SYS.DBMS\_SQL", line 1721

EMPLOYEE_ID	EMPLOYEE_NAME
1	narender

#### 4) DEFAULT CONSTRAINT:

##### SOURCE CODE:

```
CREATE TABLE customers_narender (
  c_id numeric(10) not null,
  c_name varchar2(55) not null,
  c_country varchar2(55) DEFAULT 'INDIA'
);

desc customers_narender;

insert into customers_narender(c_id,c_name) values(1,'narender');
insert into customers_narender(c_id,c_name,c_country) values(2,'neel','germany');

select * from customers_narender;
```

##### OUTPUT:

Default constraint is used to set the default value for particular field.

In this example, the c\_country is set to the default value of 'INDIA'

If we do not specify the value of the c\_country then the default value will be 'INDIA'.

Table created.

TABLE CUSTOMERS\_NARENDER

Column	Null?	Type
C_ID	NOT NULL	NUMBER(10,0)
C_NAME	NOT NULL	VARCHAR2(55)
C_COUNTRY	-	VARCHAR2(55)

[Download CSV](#)

3 rows selected.

1 row(s) inserted.

1 row(s) inserted.

C_ID	C_NAME	C_COUNTRY
1	narender	INDIA
2	neel	germany

[Download CSV](#)

2 rows selected.

5) CHECK CONSTRAINT:

SOURCE CODE:

```
CREATE TABLE elections_narender (
  e_id numeric(10) not null,
  e_name varchar2(55) not null,
  e_age numeric(10),
  CONSTRAINT check_age CHECK (e_age >= 18)
);

desc elections_narender;

insert into elections_narender values(1,'narender',22);
insert into elections_narender values(2,'ganesh',15);

select * from elections_narender;
```

OUTPUT:

The check constraint is used to check the condition  
If the condition is true then only allow to insert the values, else it will throw error.  
In this example, we have checked the age of person who are eligible for voting  
So, if the person's age is less than 18 then it will not insert the value in the table.

Table created.

TABLE ELECTIONS\_NARENDER

Column	Null?	Type
E_ID	NOT NULL	NUMBER(10,0)
E_NAME	NOT NULL	VARCHAR2(55)
E_AGE	-	NUMBER(10,0)

[Download CSV](#)

3 rows selected.

1 row(s) inserted.

ORA-02290: check constraint (SQL\_I2GZXKBRMSEBAZVRAPTMQIIV.CHECK\_AGE) violated ORA-06512: at "SYS.DBMS\_SQL", line 1721

E_ID	E_NAME	E_AGE
1	narender	22

[Download CSV](#)

## 6) NOT NULL:

### SOURCE CODE:

```
CREATE TABLE student_narender(
id NUMBER(5) PRIMARY KEY,
Fname VARCHAR2(15) NOT NULL,
Lname VARCHAR2(20),
Address VARCHAR2(50),
DOB DATE);

desc student_narender;

insert into student_narender values(1,'narender','keswani','ulhasnagar','10-NOV-1999');
insert into student_narender values(2,'','deshmukh','vasat','31-JAN-2000');

select * from student_narender;
```

### OUTPUT:

If we do not pass the value in the not null field, it will throw the error.

Table created.

TABLE STUDENT\_NARENDER

Column	Null?	Type
ID	NOT NULL	NUMBER(5,0)
FNAME	NOT NULL	VARCHAR2(15)
LNAME	-	VARCHAR2(20)
ADDRESS	-	VARCHAR2(50)
DOB	-	DATE

[Download CSV](#)

5 rows selected.

1 row(s) inserted.

ORA-01400: cannot insert NULL into ("SQL\_VZSRJRMVGKNFOUEAEIESNPKR"."STUDENT\_NARENDER"."FNAME") ORA-06512: at "SYS.DBMS\_SQL", line 1721

ID	FNAME	LNAME	ADDRESS	DOB
1	narender	keswani	ulhasnagar	10-NOV-99

[Download CSV](#)

### CONCLUSION:

I have learned the basics of DML, DDL, SQL Constraints from this assignment/tutorial.