**INTRODUCTION TO ORACLE**

**1. DDL:** **Data Definition Language (DDL)** statements are used to define the database structure or schema.

DDL Commands:  Create, Alter, Drop, Rename, Truncate

CREATE - to create objects in the database

ALTER - alters the structure of the database

DROP - delete objects from the database

TRUNCATE - remove all records from a table, including all spaces allocated for the records are removed

RENAME - rename an object

**2. DML:** **Data Manipulation Language (DML)** statements are used for managing data within schema objects and to manipulate data of a database objects.

DML Commands:   Insert, Update, Delete, Select

INSERT - insert data into a table

UPDATE - updates existing data within a table

DELETE - deletes all records from a table, the space for the records remain

SELECT - retrieve data from the a database

**3. DCL:** **Data Control Language (DCL)** statements are used to create roles, permissions, and referential integrity as well it is used to control access to database by securing it. To control the data of a database.

DCL Commands:   Grant, Revoke

GRANT - gives user's access privileges to database

REVOKE -withdraw access privileges given with the GRANT command

**4. TCL:** **Transaction Control (TCL)** statements are used to manage the changes made by DML statements. It allows statements to be grouped together into logical transactions.

TCL Commands:  Commit, Rollback, Save point

COMMIT - save work done

SAVEPOINT - identify a point in a transaction to which you can later roll back

ROLLBACK - restore database to original since the last COMMIT

**Syntax with examples**

**1. DDL (Data Definition Language) Commands: CREATE, ALTER and DROP.**

**CREATE**: This command useful for creating creating table.

Syntax:

create table [table name] (column1 datatype[size], column 2 datatype[size],… column n datatype[size] );

**Ex:**

SQL >create table student (s\_rollno number(10) primary key,s\_name varchar2(10), gender varchar2(5),dob date,addr1 varchar2(10),addr2 varchar2(10),city varchar2(10), percentage number(4));

SQL> DESC STUDENT;

Name Null? Type

----------------- -------------- -------------------

S\_ROLLNO NOT NULL NUMBER(10)

S\_NAME VARCHAR2(10)

GENDER VARCHAR2(5)

DOB DATE

ADDR1 VARCHAR2(10)

ADDR2 VARCHAR2(10)

CITY VARCHAR2(10)

PERCENTAGE NUMBER(4)

SQL > select s\_rollno,s\_name from student;

no rows selected.

**Create table by using Constraints:**

Constraints are two types:

1. Table Level Constraints.
2. Column Level Constraints.

**1. NOT NULL**:

**a) *Not null constraint at column level.***

**Syntax:**

<col><datatype>(size)not null

SQL > create table emp(e\_id varchar(5) NOT NULL,e\_name varchar(10), e\_design varchar(10),dept varchar(10),mgr varchar(10),salary number(10));

**2. UNIQUE** :

***Unique constraint at column level.***

**Syntax:** <col><datatype>(size)unique

**Ex:-**

SQL > create table depositor(customer\_name varchar(10),acc\_no number(15) UNIQUE, brach\_name varchar(10));

***Unique constraint at table level:***

**Syntax:**

Create table tablename(col=format,col=format,unique(<col1>,<col2>));

**Ex:-**

SQL > create table depositor1(customer\_name varchar(10),acc\_no number(15), brach\_name varchar(10),UNIQUE(acc\_no));

**3. PRIMARY KEY**:

***Primary key constraint at column level***

**Syntax:**

<col><datatype>(size)primary key;

**Ex:-**

SQL> create table customer(customer\_id number (5) PRIMARY KEY, customer\_name varchar(10),customer\_street varchar(10),brach\_name varchar(10));

***Primary key constraint at table level****.*

**Syntax:**

Create table tablename(col=format,col=format primary key(col1>,<col2>);

**Ex:-**

SQL > create table customer1(customer\_id number (5),customer\_name varchar(10),customer\_street varchar(10),brach\_name varchar(10),PRIMARY KEY(customer\_id));

**4. CHECK**:

***Check constraint constraint at column level.***

**Syntax:** <col><datatype>(size) check(<logical expression>)

**Ex:-**create table loan(loan\_no varchar(10),customer\_name varchar(10), balance number (10) CHECK(balance>1000));

***Check constraint constraint at table level.***

**Syntax:** check(<logical expression>)

**Ex:-** create table loan1(loan\_no varchar(10),customer\_name varchar(10), balance number (10), CHECK(balance>1000));

**5. FOREIGN KEY**:

***Foreign key constraint at column level.***

**Syntax:**

Column\_name Datatype(size) REFERENCES parent\_table\_name (parent\_column\_name)

**Ex:-**CREATE TABLE books (book\_id NUMBER(3), book\_title VARCHAR2(30), book\_price          
  
NUMBER(3),  book\_author\_id NUMBER(3) REFERENCES author(author\_id ) );      

***Foreign key constraint at table level***

**Syntax:**

CONSTRAINT constraint\_name FOREIGN KEY(child\_table\_column) REFERENCES Parent\_table\_name(parent\_table\_column)

**Ex:-**CREATE TABLE books (book\_id  NUMBER(3)  CONSTRAINT  bok\_bi\_pk  PRIMARY KEY, book\_title   VARCHAR2(30), book\_price  NUMBER(3), book\_author\_id   NUMBER(3),CONSTRAINT  bok\_ai\_fk  FOREIGN KEY  (book\_author\_id) REFERENCES  author(author\_id) );

**WEEK-1**

**CREATION OF TABLES**

**1) Create a table called Employee with the following structure.**

|  |  |
| --- | --- |
| **Name** | **Type** |
| Empno | Number |
| Ename | Varchar2(10) |
| Job | Varchar2(10) |
| Mgr | Number |
| Sal | Number |

1. Add a column commission with domain to the Employee table.
2. Insert any five records into the table.
3. Update the column details of job
4. Rename the column of Employ table using alter command.
5. Delete the employee whose Empno is 105.

**SOLUTION:**

SQL> create table employee(empno number,ename varchar2(10),job varchar2(10),mgr number,sal number);

Table created.

SQL> desc employee;

Name Null? Type

--------------------------- -------- ----------------------------

EMPNO NUMBER

ENAME VARCHAR2(10)

JOB VARCHAR2(10)

MGR NUMBER

SAL NUMBER

1. **Add a column commission with domain to the Employee table.**

SQL> alter table employee add(commission number);

Table altered.

SQL> desc employee;

Name Null? Type

----------------------------------------- -------- ----------------------------

EMPNO NUMBER

ENAME VARCHAR2(10)

JOB VARCHAR2(10)

MGR NUMBER

SAL NUMBER

COMMISSION NUMBER

1. **Insert any five records into the table.**

SQL> insert into employee values(&empno,'&ename','&job',&mgr,&sal,'&commission');

Enter value for empno: 101

Enter value for ename: abhi

Enter value for job: manager

Enter value for mgr: 1234

Enter value for sal: 10000

Enter value for commission: 70

old 1: insert into employee values(&empno,'&ename','&job',&mgr,&sal,'&commission')

new 1: insert into employee values(101,'abhi','manager',1234,10000,'70')

1 row created.

SQL> /

Enter value for empno: 102

Enter value for ename: rohith

Enter value for job: analyst

Enter value for mgr: 2345

Enter value for sal: 9000

Enter value for commission: 65

old 1: insert into employee values(&empno,'&ename','&job',&mgr,&sal,'&commission')

new 1: insert into employee values(102,'rohith','analyst',2345,9000,'65')

1 row created.

SQL> /

Enter value for empno: 103

Enter value for ename: david

Enter value for job: analyst

Enter value for mgr: 3456

Enter value for sal: 9000

Enter value for commission: 65

old 1: insert into employee values(&empno,'&ename','&job',&mgr,&sal,'&commission')

new 1: insert into employee values(103,'david','analyst',3456,9000,'65')

1 row created.

SQL> /

Enter value for empno: 104

Enter value for ename: rahul

Enter value for job: clerk

Enter value for mgr: 4567

Enter value for sal: 7000

Enter value for commission: 55

old 1: insert into employee values(&empno,'&ename','&job',&mgr,&sal,'&commission')

new 1: insert into employee values(104,'rahul','clerk',4567,7000,'55')

1 row created.

SQL> /

Enter value for empno: 105

Enter value for ename: pramod

Enter value for job: salesman

Enter value for mgr: 5678

Enter value for sal: 5000

Enter value for commission: 50

old 1: insert into employee values(&empno,'&ename','&job',&mgr,&sal,'&commission')

new 1: insert into employee values(105,'pramod','salesman',5678,5000,'50')

1 row created.

SQL> select \* from employee;

EMPNO ENAME JOB MGR SAL COMMISSION

---------- ---------- ---------- ---------- ---------- ----------

101 abhi manager 1234 10000 70

102 rohith analyst 2345 9000 65

103 david analyst 3456 9000 65

104 rahul clerk 4567 7000 55

105 pramod salesman 5678 5000 50

1. **Update the column details of job**

SQL> update employee set job='trainee' where empno=103;

1 row updated.

SQL> select \* from employee;

EMPNO ENAME JOB MGR SAL COMMISSION

---------- ---------- ---------- ---------- ---------- ----------

101 abhi manager 1234 10000 70

102 rohith analyst 2345 9000 65

103 david trainee 3456 9000 65

104 rahul clerk 4567 7000 55

105 pramod salesman 5678 5000 50

1. **Rename the column of Employ table using alter command.**

SQL> alter table employee rename column mgr to manager\_no;

Table altered.

SQL> desc employee;

Name Null? Type

----------------------------------------- -------- ----------------------------

EMPNO NUMBER

ENAME VARCHAR2(10)

JOB VARCHAR2(10)

MANAGER\_NO NUMBER

SAL NUMBER

COMMISSION NUMBER

1. **Delete the employee whose Empno is 105.**

SQL> delete employee where empno=105;

1 row deleted.

SQL> select \* from employee;

EMPNO ENAME JOB MANAGER\_NO SAL COMMISSION

---------- ---------- ---------- ---------- ---------- ----------

101 abhi manager 1234 10000 70

102 rohith analyst 2345 9000 65

103 david trainee 3456 9000 65

104 rahul clerk 4567 7000 55

**2) Create department table with the following structure.**

|  |  |
| --- | --- |
| **Name** | **Type** |
| Deptno | Number |
| Deptname | Varchar2(10) |
| location | Varchar2(10) |

1. Add column designation to the department table.
2. Insert values into the table.
3. List the records of dept table grouped by deptno.
4. Update the record where deptno is 9.
5. Delete any column data from the table.

**SOLUTION:**

SQL> create table department(deptno number,deptname varchar2(10),location varchar2(10));

Table created.

SQL> desc department;

Name Null? Type

----------------------------------------- -------- ----------------------------

DEPTNO NUMBER

DEPTNAME VARCHAR2(10)

LOCATION VARCHAR2(10)

1. **Add column designation to the department table.**

SQL> alter table department add(designation varchar2(10));

Table altered.

SQL> desc department;

Name Null? Type

----------------------------------------- -------- ----------------------------

DEPTNO NUMBER

DEPTNAME VARCHAR2(10)

LOCATION VARCHAR2(10)

DESIGNATION VARCHAR2(10)

1. **Insert values into the table.**

SQL> insert into department values(&deptno,'&deptname','&location','&designation');

Enter value for deptno: 9

Enter value for deptname: accounting

Enter value for location: hyderabad

Enter value for designation: manager

old 1: insert into department values(&deptno,'&deptname','&location','&designation')

new 1: insert into department values(9,'accounting','hyderabad','manager')

1 row created.

SQL> /

Enter value for deptno: 10

Enter value for deptname: research

Enter value for location: chennai

Enter value for designation: professor

old 1: insert into department values(&deptno,'&deptname','&location','&designation')

new 1: insert into department values(10,'research','chennai','professor')

1 row created.

SQL> /

Enter value for deptno: 11

Enter value for deptname: sales

Enter value for location: banglore

Enter value for designation: salesman

old 1: insert into department values(&deptno,'&deptname','&location','&designation')

new 1: insert into department values(11,'sales','banglore','salesman')

1 row created.

SQL> /

Enter value for deptno: 12

Enter value for deptname: operations

Enter value for location: mumbai

Enter value for designation: operator

old 1: insert into department values(&deptno,'&deptname','&location','&designation')

new 1: insert into department values(12,'operations','mumbai','operator')

1 row created.

SQL> insert into department values(&deptno,'&deptname','&location','&designation');

Enter value for deptno: 9

Enter value for deptname: accounting

Enter value for location: chennai

Enter value for designation: manager

old 1: insert into department values(&deptno,'&deptname','&location','&designation')

new 1: insert into department values(9,'accounting','chennai','manager')

1 row created.

SQL> select \* from department ;

DEPTNO DEPTNAME LOCATION DESIGNATION

---------------- ------------------- ----------------- -------------------

9 accounting hyderabad manager

10 research chennai professor

11 sales banglore salesman

12 operations mumbai operator

9 accounting chennai manager

1. **List the records of dept table grouped by deptno.**

SQL> select deptno,deptname from department group by deptno,deptname;

DEPTNO DEPTNAME

---------- ----------

9 accounting

12 operations

10 research

11 sales

1. **Update the record where deptno is 9.**

SQL> update department set designation='accountant' where deptno=9;

2 rows updated.

SQL> select \* from department;

DEPTNO DEPTNAME LOCATION DESIGNATION

---------------- ------------------- ----------------- -------------------

9 accounting hyderabad accountant

10 research chennai professor

11 sales banglore salesman

12 operations mumbai operator

9 accounting chennai accountant

1. **Delete any column data from the table.**

SQL> alter table department drop(designation);

Table altered.

SQL> select \* from department;

DEPTNO DEPTNAME LOCATION

---------- --------------- ----------

9 accounting hyderabad

10 research chennai

11 sales banglore

12 operations mumbai

9 accounting Chennai

**LAB ASSIGNMENT:**

1. Create a table called Customer table

|  |  |
| --- | --- |
| **Name** | **Type** |
| Cust name | Varchar2(20) |
| Cust street | Varchar2(20) |
| Cust city | Varchar2(20) |

1. Insert records into the table.
2. Add salary column to the table.
3. Alter the table column domain.
4. Drop salary column of the customer table.
5. Delete the rows of customer table whose cust\_city is ‘hyd’.

2. Create a table called branch table.

|  |  |
| --- | --- |
| **Name** | **Type** |
| Branch name | Varchar2(20) |
| Branch city | Varchar2(20) |
| asserts | Number |

1. Increase the size of data type for asserts to the branch.
2. Add and drop a column to the branch table.
3. Insert values to the table.
4. Update the branch name column
5. Delete any two columns from the table

3. Create a table called sailor table

|  |  |
| --- | --- |
| **Name** | **Type** |
| Sid | Number |
| Sname | Varchar2(20) |
| rating | Varchar2(20) |

1. Add column age to the sailor table.
2. Insert values into the sailor table.
3. Delete the row with rating >8.
4. Update the column details of sailor.
5. Insert null values into the table.

4. Create a table called reserves table

|  |  |
| --- | --- |
| **Name** | **Type** |
| Boat id | Integer |
| sid | Integer |
| day | Integer |

* 1. Insert values into the reserves table.
  2. Add column time to the reserves table.
  3. Alter the column day data type to date.
  4. Drop the column time in the table.
  5. Delete the row of the table with some condition.

**WEEK -2**

**QUERIES USING DDL AND DML**

1. **a. Create a user and grant all permissions to the user.**

**b. Insert the any three records in the employee table and use rollback. Check the result.**

**c. Add primary key constraint and not null constraint to the employee table.**

**d. Insert null values to the employee table and verify the result.**

**SOLUTION:**

**a) create a user and grant all permissions to the user.**

CONNECT <USER-NAME>/<PASSWORD>@<DATABASE NAME>;  
  
--Create user query  
  
CREATE USER <USER NAME> IDENTIFIED BY <PASSWORD>;  
  
--Provide roles  
  
GRANT CONNECT,RESOURCE,DBA TO <USER NAME>;  
  
--Assigning privileges  
  
GRANT CREATE SESSION GRANT ANY PRIVILEGE TO <USER NAME>;

GRANT UNLIMITED TABLESPACE TO <USER NAME>;  
  
--Provide access to tables.  
  
GRANT SELECT, UPDATE, INSERT, DELETE ON <TABLE NAME> TO <USER NAME>;

**b) Insert the any three records in the employee table and use rollback. Check the result.**

SQL> SELECT \* FROM EMPLOYEE;

EMPNO ENAME JOB MANAGER\_NO SAL COMMISSION

---------------- ---------- ---------- --------------------- ------ -------------------

101 abhi manager 1234 1100 70

102 rohith analyst 2345 9000 65

103 david trainee 3456 9000 65

104 rahul clerk 4567 7000 55

SQL> insert into employee values(&empno,'&ename','&job',&manager\_no,&sal,&commission);

Enter value for empno: 105

Enter value for ename: aravind

Enter value for job: salesman

Enter value for manager\_no: 5678

Enter value for sal: 5000

Enter value for commission: 50

old 1: insert into employee values(&empno,'&ename','&job',&manager\_no,&sal,&commission)

new 1: insert into employee values(105,'aravind','salesman',5678,5000,50)

1 row created.

SQL> rollback;

Rollback complete.

SQL> SELECT \* FROM EMPLOYEE;

EMPNO ENAME JOB MANAGER\_NO SAL COMMISSION

---------------- ---------- ---------- --------------------- ------ -------------------

101 abhi manager 1234 1100 70

102 rohith analyst 2345 9000 65

103 david trainee 3456 9000 65

104 rahul clerk 4567 7000 55

**c) Add primary key constraint and not null constraint to the employee table.**

SQL> alter table employee modify(empno number primary key, ename varchar2(10) not null);

Table altered.

SQL> desc employee;

Name Null? Type

-------------------------- -------- -------------------

EMPNO NOT NULL NUMBER

ENAME NOT NULL VARCHAR2(10)

JOB VARCHAR2(10)

MANAGER\_NO NUMBER

SAL NUMBER

COMMISSION NUMBER

**d) Insert null values to the employee table and verify the result.**

SQL> desc employee;

Name Null? Type

-------------------- ------------------ ----------------------------

EMPNO NOT NULL NUMBER

ENAME NOT NULL VARCHAR2(10)

JOB NOT NULL VARCHAR2(10)

MANAGER\_NO NUMBER

SAL NOT NULL NUMBER

COMMISSION NUMBER

SQL> insert into employee values(&empno,'&ename','&job',&manager\_no,&sal,&commission);

Enter value for empno: 105

Enter value for ename: mohith

Enter value for job: salesman

Enter value for manager\_no: 5678

Enter value for sal: null

Enter value for commission: 50

old 1: insert into employee values(&empno,'&ename','&job',&manager\_no,&sal,&commission)

new 1: insert into employee values(105,'mohith','salesman',5678,null,50)

insert into employee values(105,'mohith','salesman',5678,null,50)

\*

**2. a. create a user and grant all permissions to the user.**

**b. Insert values in the department table and use commit.**

**c. Add constraints like unique and not null to the department table.**

**d. Insert repeated values and null values into the table.**

**SOLUTION:**

**a) create a user and grant all permissions to the user.**

CONNECT <USER-NAME>/<PASSWORD>@<DATABASE NAME>;  
  
--Create user query  
  
CREATE USER <USER NAME> IDENTIFIED BY <PASSWORD>;  
  
--Provide roles  
  
GRANT CONNECT,RESOURCE,DBA TO <USER NAME>;  
  
--Assigning privileges  
  
GRANT CREATE SESSION GRANT ANY PRIVILEGE TO <USER NAME>;

GRANT UNLIMITED TABLESPACE TO <USER NAME>;  
  
--Provide access to tables.  
  
GRANT SELECT, UPDATE, INSERT, DELETE ON <TABLE NAME> TO <USER NAME>;

**b) Insert values in the department table and use commit.**

SQL> insert into department values(&deptno,'&deptname','&location');

Enter value for deptno: 13

Enter value for deptname: sales

Enter value for location: delhi

old 1: insert into department values(&deptno,'&deptname','&location')

new 1: insert into department values(13,'sales','delhi')

1 row created.

SQL> commit;

Commit complete.

SQL> select \* from department;

DEPTNO DEPTNAME LOCATION

---------- -------------- ----------------

9 accounting hyderabad

10 research chennai

11 sales banglore

12 operations mumbai

9 accounting chennai

13 sales delhi

6 rows selected.

**c) Add constraints like unique and not null to the department table.**

SQL> alter table department modify(deptno number unique);

Table altered.

SQL> alter table department modify(location varchar2(10) not null);

Table altered.

SQL> DESC DEPARTMENT;

Name Null? Type

-------------------- ---------------- -------------------

DEPTNO NUMBER

DEPTNAME VARCHAR2(10)

LOCATION NOT NULL VARCHAR2(10)

**d) Insert repeated values and null values into the table.**

SQL> insert into department values(&deptno,'&deptname','&location');

Enter value for deptno: 10

Enter value for deptname: research

Enter value for location:

old 1: insert into department values(&deptno,'&deptname','&location')

new 1: insert into department values(10,'research','')

insert into department values(10,'research','')

SQL> insert into department values(&deptno,'&deptname','&location');

Enter value for deptno: 10

Enter value for deptname: research

Enter value for location: hyderabad

old 1: insert into department values(&deptno,'&deptname','&location')

new 1: insert into department values(10,'research','hyderabad')

insert into department values(10,'research','hyderabad')

**LAB ASSIGNMENT:**

1 a. create a user and grant all permissions to the user.

b. Insert values into the table and use commit.

c. Delete any three records in the department table and use rollback.

d. Add constraint primary key and foreign key to the table.

2 a. create a user and grant all permissions to the user.

b. Insert records in the sailor table and use commit.

c. Add save point after insertion of records and verify save point.

d. Add constraints not null and primary key to the sailor table.

3 a. create a user and grant all permissions to the user.

b. Use revoke command to remove user permissions.

c. Change password of the user created.

d. Add constraint foreign key and not null.

4 a. create a user and grant all permissions to the user.

b. Update the table reserves and use savepoint and rollback.

c. Add constraint primary key, foreign key and not null to the reserves table

d. Delete constraint not null to the table column.