**EX.NO:1a**

**DATA DEFINITION LANGUAGES (DDL) COMMANDS Of Base Tables and Views**

A Data Definition Language (DDL) statement is used to define the database structure or schema.

##### Aim:

To study and execute the DDL commands in RDBMS.

##### DDL commands:

✴ CREATE

✴ ALTER

✴ DROP

✴ RENAME

✴ TRUNCATE

#### SYNTAX’S OFCOMMANDS CREATE TABLE:

1. To make a new database, table, index, or stored query. A create statement in SQL creates inside of a relational database an management system (RDBMS).

CREATE TABLE <table\_name> (

Column\_name1 data\_type ([size]), Column\_name2 data\_type ([size]),

.

.

.

Column\_name-n data\_type ([size])

);

#### ALTER A TABLE:

To modify an existing database object. Alter the structure of the database. To add a column in a table

ALTER TABLE table\_name ADD column\_name datatype;

To delete a column in a table

ALTER TABLE table\_name DROP column column\_name;

#### DROP TABLE:

Delete Objects from the Database

DROP TABLE

table\_name;

**TRUNCATE TABLE:**

Remove all records from a table, including all spaces allocated for the records are removed.

TRUNCATE TABLE table\_name;

#### EXERCISE:

**Create Table**

SQL> create table employee

2 (

3 empid varchar(10) primary key, 4 empname varchar2(20) not null,

1. gender varchar2(7) not null,
2. age number(3) not null,
3. dept varchar2(15) not null,
4. dob date not null,
5. doj date not null 10);

Table created.

SQL> create table salary

2 (

1. empid varchar(10) references employee(empid),
2. salary number(10) not null,
3. dept varchar(15) not null,
4. branch varchar2(20) not null
5. );

Table created.

SQL> create table branchtable

2 (

1. branch varchar2(20) not null,
2. city varchar2(20) not null
3. );

Table created.

#### DESCRIBE TABLE

SQL> desc employee;

| Name | Null? | Type |
| --- | --- | --- |
| EMPID | NOT NULL | VARCHAR2(10) |
| EMPNAME | NOT NULL | VARCHAR2(20) |
| GENDER | NOT NULL | VARCHAR2(7) |
| AGE | NOT NULL | NUMBER(3) |
| DEPT | NOT NULL | VARCHAR2(15) |
| DOB | NOT NULL | DATE |
| DOJ | NOT NULL | DATE |

| SQL> desc salary; |  |  |
| --- | --- | --- |
| Name | Null? | Type |



| EMPID  SALARY | NOT NULL  NOT NULL | VARCHAR2 (10)  NUMBER (10) |
| --- | --- | --- |
| DEPT | NOT NULL | VARCHAR2 (15) |
| BRANCH | NOT NULL | VARCHAR(20) |

SQL> desc branchtable;

| Name | Null | Type |
| --- | --- | --- |
| Branch | Not Null | VARCHAR (20) |
| City | Not Null | VARCHAR (20) |

#### ALTER TABLE

**ADD:**

SQL> alter table employee add(designation varchar2(15)); Table altered.

SQL> alter table salary add(constraint nithi unique(empid)); Table altered.

#### MODIFY:

SQL> alter table employee modify (designation varchar2(20)); Table altered.

#### RENAME TABLE

SQL> create table emp 2 (

* 1. empid varchar2(10),
  2. empname varchar2(20),
  3. age number(3),
  4. sex char
  5. );

Table created.

SQL> rename emp to empl; Table renamed.

SQL> desc empl;

Name Null? Type



| EMPID EMPNAME AGE  SEX | VARCHAR2(10) VARCHAR2(20) NUMBER(3)  CHAR(1) |
| --- | --- |

SQL>desc emp; ERROR:

ORA-04043: object emp does not exist Table altered.

#### TRUNCATE TABLE DATA

SQL> insert into emp values(&no,'&name','&dept',&age,'&sex'); Enter value for no: 1

Enter value for name: arun Enter

value for dept: it Enter value for age: 22 Enter value for sex: m

old 1: insert into emp values(&no,'&name','&dept',&age,'&sex') new 1: insert into emp values(1,'arun','it',22,'m')1 row created.

SQL> insert into emp values(&no,'&name','&dept',&age,'&sex'); Enter value for no: 2

Enter value for name: bala Enter value for dept: service Enter value

for age: 26 Enter value for sex: m

old 1: insert into emp values(&no,'&name','&dept',&age,'&sex') new

arun it 22 m

1: insert into emp values(2,'bala','service',26,'m') 1 row created.

SQL> insert into emp values(&no,'&name','&dept',&age,'&sex'); Enter value for no: 3

Enter value for name: chitra Enter value for dept: sales Enter value

for age: 25

Enter value for sex: f

old 1: insert into emp values(&no,'&name','&dept',&age,'&sex') new 1: insert into emp values(3,'chitra','sales',25,'f') 1 row created.

SQL> select \* from emp;

EMPID EMPNAME DEPT AGE SEX

| 2 | bala | service | 26 | m |
| --- | --- | --- | --- | --- |
| 3 | chitra | sales | 25 | f |

SQL> commit; Commit complete.

SQL>

truncate table emp; Table

truncated.

SQL>

select \* from emp; no rows selected

SQL> commit; Commit complete. **DROP TABLE** SQL>

drop

table empl; Table droppe d.

SQL>

desc empl; ERROR:

ORA-04043: object empl does not exist

#### RESULT:

Thus executed the DDL commands in RDBMS

**EX.NO. 1b**

**DATA MANIPULATIONLANGUAGE (DML)OF BASE TABLE AND VIEWS**

Data manipulation language allows the users to query and manipulate data in existing schema in object. It allows following data to insert, delete, update and recovery data in schema object.

##### Aim:

To study DML commands in RDBMS.

#### DML COMMANDS:

* INSERT
* UPDATE
* DELETE
* SELECT

#### QUERY:

Query is a statement in the DML that request the retrieval of data from database.

* The portion of the DML used in a Query is called Query language. The SELECT statement is used to query a database

#### SYNTAX OF COMMANDS INSERT:

Values can be inserted into table using insert commands. There are two types of insert commands. They are multiple value insert commands (using ‘&’ symbol) single value insert command (without using ‘&’symbol) Syntax:

INSERT INTO table\_name VALUES (value1, value2, value3,…..); (OR)

INSERT INTO table\_name (column1, column2, column3,….) VALUES

(value1,value2,value3,…..);

#### UPDATE:

This allows the user to update the particular column value using the where clause condition.

Syntax:

UPDATE <table\_name> SET <col1=value> WHERE <column=value>;

#### DELETE:

This allows you to delete the particular column values using where clause condition.

Syntax:

DELETE FROM <table\_name> WHERE <condition>;

#### SELECT:

The select statement is used to query a database .This statement is used to retrieve the information from the database. The SELECT statement can be used in many ways. They are:

##### Selecting some columns :

To select specified number of columns from the table the Following command is used.

Syntax**:**

SELECT column\_name FROM table\_name;

##### Query All Columns:

To select all columns from the table \* is used instead of column names.

Syntax**:**

SELECT \* FROM table\_name;

##### Select using DISTINCT:

The DISTINCT keyword is used to return only different values (i.e. ) this

command does not select the duplicate values from the table.

Syntax**:**

SELECT DISTINCT column name(s) FROM table\_name;

##### Select using IN:

If you want to get the rows which contain certain values, the best way to do it is to use the IN conditional expression.

Syntax**:**

SELECT column name(s) FROM table\_name WHERE Column name IN (value1, value2,……,value-n);

##### Select using BETWEEN:

BETWEEN can be used to get those items that fall within a range.

Syntax**:**

SELECT column name FROM table\_name WHERE Column name BETWEEN value1 AND value2;

##### Renaming:

The select statement can be used to rename either a column or the entire table.

Syntax**:**

**Renaming a column:**

SELECT column name AS new name FROM table\_name;

**Renaming a table:**

SELECT column name FROM table\_name AS newname;

##### Sorting:

The select statement with the **order by Clause** is used to sort the contents Table either in ascending or descending order.

Syntax**:**

SELECT column name FROM table\_name WHERE Condition ORDER BY column name ASC/DESC;

##### To select by matching some patterns:

The select statement along with **like clause** I is used to match strings. The

##### like

condition is used to specify a search pattern in a column.

Syntax**:**

SELECT column name FROM table\_name WHERE Column name LIKE “% or-“;

**%**: Matches any sub string.

**-** : Matches a single character.

##### SELECT INTO statement:

The SELECT INTO statement is most often used to create backup copies of

tables or for archiving records.

Syntax**:**

SELECT Column\_name(s) INTO variable\_name(s) FROM table\_name

WHERE condition.

##### To Select NULL values:

We can use the SELECT statement to select the ‘null’ values also.

For retrieving roes where some of the columns have been defined as NULLs there is a special comparison operator of the form IS [NOT]NULL.

Syntax:

SELECT column name FROM table\_name WHERE Column name IS NULL;

##### Select using AND, OR, NOT:

We can combine one or more conditions in a SELECT statement using the

logical operators AND, OR, NOT.

**Syntax:**

#### EXERCISE:

SELECT column name FROM table\_name WHERE Condition1 LOGICAL OPERATOR condition2;

#### INSERT COMMAND

SQL> insert into employee values('&empid','&empname','&gender',&age,'&dept','&dob','&doj','&desig'); Enter value for empid: it9001

Enter value for empname: arunkumar Enter value for gender: male

Enter value for age: 22 Enter value for dept: it

Enter value for dob: 12-jan-1988

Enter value for doj: 23-oct-2006 Enter value for desig: manager old 1: insert into employee values('&empid','&empname','&gender',&age,'&dept','&dob','&doj','&des i new 1: insert into employee values('it9001','arunkumar','male',22,'it','12- jan-1988','23-oct- 2006'

1 row created.

SQL> insert into employee values('&empid','&empname','&gender',&age,'&dept','&dob','&doj','&desig'); Enter value for empid: it9001

Enter value for empname: arunkumar Enter value for gender: male

Enter value for age: 22 Enter value for dept: it

Enter value for dob: 12-jan-1988

Enter value for doj: 23-oct-2006

Enter value for desig: manager old 1: insert into employee

values('&empid','&empname','&gender',&age,'&dept','&dob','&doj','&des i new 1: insert into employee values('it9001','arunkumar','male',22,'it','12- jan-1988','23-oct- 2006'

1 row created.

SQL> insert into employee values('&empid','&empname','&gender',&age,'&dept','&dob','&doj','&desig'); Enter value for empid: it9002

Enter value for empname: balakrishnan Enter value for gender: male

Enter value for age: 27 Enter value for dept: it

Enter value for dob: 27-mar-1983 Enter value for doj: 02-dec-2008 Enter value for desig: coordinator old 1: insert into employee values('&empid','&empname','&gender',&age,'&dept','&dob','&doj','&d esi new 1: insert into employee values('it9002','balakrishnan','male',27,'it','27-mar-1983','02- dec-20

1 row created.

SQL> insert into employee values('&empid','&empname','&gender',&age,'&dept','&dob','&doj','&desig'); Enter value for empid: acc9001

Enter value for empname: kannan

Enter value for gender: male Enter value for age: 35

Enter value for dept: accounts

Enter value for dob: 28-dec-1975

Enter value for doj: 01-jan-1995 Enter value for desig: manager old 1: insert into employee values('&empid','&empname','&gender',&age,'&dept','&d ob','&doj','&desi

new 1: insert into employee values('acc9001','kannan','male',35,'accounts','28- dec-1975','01- jan-1

1 row created.

SQL> insert into employee values('&empid','&empname','&gender',&age,'&dept','&dob','&doj','&desig'); Enter value for empid: acc9002

Enter value for empname: magudeshwaran Enter value for gender: male

Enter value for age: 27

Enter value for dept: accounts Enter value for dob: 25-aug-1983 Enter value for doj:

12-apr-2000 Enter

value for desig: asst manager old 1: insert into employee

values('&empid','&empname','&gender',&age,'&dept','&dob','&doj','&desi new 1: insert into employee values('acc9002','magudeshwaran','male',27,'accounts','25-aug- 1983','1

1 row created.

SQL> insert into employee values('&empid','&empname','&gender',&age,'&dept','&dob','&doj','&desig'); Enter value for empid: ser9001

Enter value for empname: jagadheesh Enter value for gender: male

Enter value for age: 33 Enter value for dept: service

Enter value for dob: 31-mar-1877

Enter value for doj: 3-jun-1999 Enter value for desig: manager old 1: insert into employee values('&empid','&empname','&gender',&age,'&dept','&dob','&doj', '&desi new 1: insert into employee values('ser9001','jagadheesh','male',33,'service','31-mar- 1877','3-jun

1 row created.

SQL> insert into employee values('&empid','&empname','&gender',&age,'&dept','&dob','&doj','&desig'); Enter value for empid: ser9006

Enter value for empname: muruganandam Enter value for gender: male

Enter value for age: 35 Enter value for dept: service Enter value for

dob: 09-aug-1975 Enter value for doj: 02-mar-2000 Enter value for desig: painter old 1: insert into employee

values('&empid','&empname','&gender',&age,'&dept','&dob','&doj','&d esi new 1: insert into employee values('ser9006','muruganandam','male',35,'service','09-aug- 1975','02-

1 row created.

SQL> insert into employee values('&empid','&empname','&gender',&age,'&dept','&dob','&doj','&desig'); SQL> /

Enter value for empid: sal9001 Enter value for empname: suresh

Enter value for gender: male Enter value for age: 40 Enter value for dept: sales

Enter value for dob: 12-jul-1970

Enter value for doj: 01-apr-1996 Enter value for desig: manager old 1: insert into employee values('&empid','&empname','&gender',&age,'&dept','&dob','&doj','&des i new 1: insert into employee values('sal9001','suresh','male',40,'sales','12- jul-1970','01-apr- 1996

1 row created.

SQL> insert into employee values('&empid','&empname','&gender',&age,'&dept','&dob','&doj','&desig'); Enter value for empid: sal9006

Enter value for empname: sharmila Enter value for gender: female Enter value for age: 27

Enter value for dept: sales

Enter value for dob: 12-jan-1983 Enter value for

doj: 09-aug-2007

Enter value for desig: executive old 1: insert into employee

values('&empid','&empname','&gender',&age,'&dept','&dob','&doj','&desi new 1: insert into employee values('sal9006','sharmila','female',27,'sales','12- jan-1983','09- aug-

1 row created.

SQL> insert into salary values(‘&empid’,&salary,’&dept’,’&branch’); Enter value for empid: it9002

Enter value for salary: 18000 Enter value for dept: it

Enter value for branch: abt maruthi old 1: insert into salary

values('&empid',&salary,'&dept','&branch') new 1:

insert into salary values('it9002',18000,'it','abt maruthi') 1 row created.

SQL> insert into salary values(‘&empid’,&salary,’&dept’,’&branch’); Enter value for empid: acc9001

Enter value for salary: 35000 Enter value for

dept: accounts

Enter value for branch: cars india

old 1: insert into salary values('&empid',&salary,'&dept','&branch') new 1: insert into salary

values('acc9001',35000,'accounts','cars india') 1 row created.

SQL> insert into salary values(‘&empid’,&salary,’&dept’,’&branch’); Enter value for empid: acc9002

Enter value for salary: 26000 Enter value for

dept: accounts Enter value for branch: cars india

old 1: insert into salary values('&empid',&salary,'&dept','&branch') new 1: insert into salary values('acc9002',26000,'accounts','cars india') 1 row created.

SQL> insert into salary values(‘&empid’,&salary,’&dept’,’&branch’); Enter value for empid: ser9001

Enter value for salary: 35000 Enter value for

dept: service Enter value for branch:

chennai cars

old 1: insert into salary values('&empid',&salary,'&dept','&branch') new 1: insert into salary values('ser9001',35000,'service','chennai cars') 1 row created.

SQL> insert into salary values(‘&empid’,&salary,’&dept’,’&branch’); Enter value for empid: ser9006

Enter value for salary: 12000 Enter value for dept:

service Enter value for branch: greenland cars

old 1: insert into salary values('&empid',&salary,'&dept','&branch') new 1: insert into salary values('ser9006',12000,'service','greenland cars') 1 row created.

SQL> insert into salary values(‘&empid’,&salary,’&dept’,’&branch’); Enter value for empid: sal9001

Enter value for salary: 40000 Enter value for

dept: sales Enter value for branch:

abt maruthi

old 1: insert into salary values('&empid',&salary,'&dept','&branch') new 1: insert into salary values('sal9001',40000,'sales','abt maruthi') 1 row created.

SQL> insert into salary values(‘&empid’,&salary,’&dept’,’&branch’); Enter value for empid: sal9006

Enter value for salary: 17000

Enter value for dept: sales Enter value for branch:

abt maruthi

old 1: insert into salary values('&empid',&salary,'&dept','&branch') new 1: insert into salary values('sal9006',17000,'sales

','abt maruthi') 1 row created.

SQL> select \* from salary;

| EMPID | SALARY | DEPT | BRANCH |
| --- | --- | --- | --- |
| it9001 | 35000 | it | abt maruthi |
| it9002 | 18000 | it | abt maruthi |
| acc9001 | 35000 | accounts | cars india |
| acc9002 | 26000 | accounts | cars india |
| ser9001 | 35000 | service | chennai cars |
| ser9006 | 12000 | service | greenland cars |
| sal9001 | 40000 | sales | abt maruthi |
| sal9006 | 17000 | sales | abt maruthi |



8 rows selected.

SQL> select \* from employee;

EMPID EMPNAME GENDER AGE DEPT DOB



DOJ DESIGNATION



| it9001 arunkumar  23-OCT-06 manager | male | 22 it | 12-JAN-88 |
| --- | --- | --- | --- |
| it9002 balakrishnan  02-DEC-08 coordinator | male | 27 it | 27-MAR-83 |
| acc9001 kannan 01-JAN-95 manager | male | 35 accounts | 28-DEC-75 |
| EMPID EMPNAME |  | GENDER AGE DEPT DOB | |



DOJ DESIGNATION



acc9002 magudeshwaran male 27 accounts 25-AUG-83 12-APR-00 asst manager

ser9001 jagadheesh male 33 service 31-MAR-77 03-JUN-99 manager

ser9006 muruganandam male 35 service 09-AUG-75 02-MAR-00 painter

EMPID EMPNAME GENDER AGE DEPT DOB



DOJ DESIGNATION



sal9001 suresh male 40 sales 12-JUL-70 01-APR-96 manager

sal9006 sharmila female 27 sales 12-JAN-83 09-AUG-07

executive 8 rows selected.

SQL> insert into branchtable values('&branch','&city'); Enter value for branch:

abt maruthi Enter value for city: chennai

old 1: insert into branchtable values('&branch','&city') new 1: insert into branchtable values('abt maruthi','chennai') 1 row created.

SQL> select \* from salary;

EMPID SALARY DEPT BRANCH

| it9001 | 35000 | it |  | abt maruthi |
| --- | --- | --- | --- | --- |
| it9002 | 18000 | it |  | abt maruthi |
| acc9001 | 35000 | accounts |  | cars india |
| acc9002 | 26000 | accounts |  | cars india |
| ser9001 | 35000 | service |  | chennai cars |
| ser9006 | 12000 | service |  | greenland cars |

| sal9001 | 40000 | sales | abt maruthi |
| --- | --- | --- | --- |
| sal9006 | 17000 | sales | abt maruthi |

8 rows selected.

SQL> insert into branchtable values('&branch','&city');

Enter value for branch: cars india Enter value for city:

vellore

old 1: insert into branchtable values('&branch','&city') new 1: insert into branchtable values('cars india','vellore') 1 row created.

SQL> insert into branchtable values('&branch','&city'); Enter value for branch:

chennai cars Enter value for city: thambaram

old 1: insert into branchtable values('&branch','&city') new 1: insert into branchtable values('chennai cars','thambaram') 1 row created.

SQL> insert into branchtable values('&branch','&city'); Enter value for branch:

greenland cars Enter value for city: kanchipuram

old 1: insert into branchtable values('&branch','&city') new 1: insert into branchtable values('greenland cars','kanchipuram') 1 row created.

SQL> select \* from branchtable;

| BRANCH | CITY |
| --- | --- |
| abt maruthi | chennai |
| cars india | vellore |
| chennai cars | thambaram |
| greenland cars | kanchipuram |



#### UPDATE COMMAND

SQL> update employee set empname = 'arunprasanth' where empid='it9001'; 1 row updated.

SQL> update employee set designation='&designation' where empname='&empname';

Enter value for designation: supervisor Enter value for empname: muruganandam

old 1: update employee set designation='&designation' where empname='&empname' new 1: update employee set designation='supervisor' where empname='muruganandam' 1 row updated.

SQL> select empname,designation from employee;

| EMPNAME | DESIGNATION |
| --- | --- |
| arunprasanth | manager |
| balakrishnan | coordinator |
| kannan | manager |
| magudeshwaran | asst manager |
| jagadheesh | manager |



muruganandam supervisor suresh manager

sharmila executive 8 rows selected.

#### SELECT COMMAND

To retrieve particular column SQL> select empname from emp;

| EMPNAME  arun bala bakyaraj chitra  To retrieve all columns SQL> select \* from emp; | |  |  |  |
| --- | --- | --- | --- | --- |
| EMPID | EMPNAME | DEPT | AGE | S |
| 1 | arun | it | 22 | m |
| 2 | bala | accounts | 26 | m |
| 3 bakyaraj | | stores | 30 | m |
| 4 | chitra | sales | 24 | f |





#### DELETE COMMAND

**To delete particular record** SQL> delete emp where empid=1; 1 row deleted.

SQL> select \* from emp;

| EMPID | EMPNAME | DEPT | AGE | S |
| --- | --- | --- | --- | --- |
| 2 | bala | accounts | 26 | m |
| 3 | bakyaraj | stores | 30 | m |
| 4 | chitra | sales | 24 | f |



##### To delete all records SQL>

delete from emp; **3 rows deleted.**

SQL> create table student (idno number, name varchar(10),branch varchar(4)); Table created.

SQL> desc student;

| NAME | NULL? TYPE |
| --- | --- |
| IDNO | NUMBER |
| NAME | VARCHAR2(10) |
| BRANCH | VARCHAR2(4) |



SQL> alter table student add degree varchar(10); Table altered.

SQL> desc student;

NAME NULL? TYPE



IDNO NUMBER

| NAME | VARCHAR2 (10) |
| --- | --- |
| BRANCH | VARCHAR2 (4) |
| DEGREE | VARCHAR2 (10) |

SQL> alter table student modify degree varchar(6); Table altered.

SQL> desc student;

| NAME | NULL? TYPE |
| --- | --- |
| IDNO | NUMBER |
| NAME | VARCHAR2 (10) |
| BRANCH | VARCHAR2 (4) |
| DEGREE | VARCHAR2 (6) |



SQL> insert into student (name, degree, branch, idno) values('ASHOK','BE','CSE',01); 1 row created.

SQL> insert into student values(02,'BHAVANA','CSE','BE'); 1 row created.

SQL> insert into student values(&idno, &name, &branch, &degree); Enter value for idno: 03

Enter value for name: 'CAVIN'

Enter value for branch: 'CSE' Enter value for degree: 'BE' old 1: insert into student values(&idno,&name,&branch,&degree) new 1: insert into student values(03,'CAVIN','CSE','BE')

1 row created.

Enter value for idno: 04

Enter value for name: 'DANNY'

Enter value for branch: 'IT' Enter value for degree: 'BE' old 1: insert into student values(&idno,&name,&branch,&degree) new 1: insert into student values(04,'DANNY','IT','BE')

1 row created.

SQL> /

Enter value for idno: 05

Enter value for name: 'HARRY'

Enter value for branch: 'IT' Enter value for degree: 'BE' old 1: insert into student values(&idno,&name,&branch,&degree) new 1: insert into student values(05,'HARRY','IT','BE')

1 row created.

SQL> select \* from student;

| IDNO NAME | BRAN | DEGREE |
| --- | --- | --- |
| 1 ASHOK | CSE | BE |
| 2 BHAVANA | CSE | BE |
| 3 CAVIN | CSE | BE |
| 4 DANNY | IT | BE |
| 5 HARRY | IT | BE |



SQL> update student set degree='B.TECH' where branch='IT';

2 rows updated.

SQL> select \* from student;

| IDNO NAME BRAN | |  | DEGREE |
| --- | --- | --- | --- |
| 1 | ASHOK | CSE | BE |
| 2 | BHAVANA | CSE | BE |
| 3 | CAVIN | CSE | BE |
| 4 | DANNY | IT | B.TECH |
| 5 | HARRY | IT | B.TECH |



SQL> delete from student where idno=5; 1 row deleted.

#### CREATING TABLES WITH CONSTRAINTS: NOT NULL

SQL> select \* from student;

IDNO NAME BRAN DEGREE



| 1 ASHOK | CSE | BE |
| --- | --- | --- |
| 2 BHAVANA | CSE | BE |
| 3 CAVIN | CSE | BE |
| 4 DANNY | IT | B.TEC H |

SQL> create table staff (

idno number (4) not null,name varchar(10),branch varchar(6)

); Table created.

SQL> desc staff;

NAME NULL? TYPE



IDNO NOT NULL NUMBER(4) NAME VARCHAR2(10)

BRANCH VARCHAR2(6)

SQL> insert into staff values (&idno, &name, &branch); Enter value for idno: 1

Enter value for name: 'ABILASH' Enter value for branch: 'CSE' old 1: insert into staff values(&idno, &name, &branch) new 1: insert into staff values(1,'ABILASH','CSE')

1 row created.

SQL> /

Enter value for idno: 2

Enter value for name: 'ANTON' Enter value for branch: 'CSE' old 1: insert into staff values(&idno, &name, &branch) new 1: insert into staff values(2,'ANTON','CSE')

1 row created.

SQL> /

Enter value for idno:

Enter value for name: 'BENNY' Enter value for branch: 'IT'

old 1: insert into staff values(&idno,&name,&branch) new 1: insert into staff values(,'BENNY','IT')

insert into staff values(,'BENNY','IT') \* ERROR at line 1:

ORA-00936: missing expression

#### UNIQUE

SQL> create table employee (

rollno numb

er uniqu e, name varch ar(10)

,

salary numb er

)

;

Table created.

SQL> desc employee;

NAME NULL? TYPE



ROLLNO NUMBER NAME VARCHAR2(10) SALARY NUMBER

SQL> insert into employee values(&rollno,&name,&salary); Enter value for rollno: 1

Enter value for name: 'anton'

Enter value for salary: 10290

old 1: insert into employee values(&rollno,&name,&salary)

new 1: insert into employee values(1,'anton',10290) 1 row created.

SQL> /

Enter value for rollno: 2

Enter value for name: 'dharun' Enter value for salary: 23322 old 1: insert into employee

values(&rollno,&name,&salary) new 1: insert

into employee values(2,'dharun',23322)

1 row created.

SQL> /

Enter value for rollno: 1 Enter value for name: 'aaron' Enter value for salary: 32212

old 1: insert into employee values(&rollno,&name,&salary) new 1: insert into employee values(1,'aaron',32212)

insert into employee values(1,'aaron',32212)

\*

ERROR at line 1:

ORA-00001: unique constraint (SCOTT.SYS\_C001265) violated

#### PRIMARY KEY

SQL> create table cars (model

number primary key, name varchar(10),

cost number(6)

)

;

Table created.

SQL> desc cars;

| NAME | NULL? TYPE |
| --- | --- |
| MODEL | NOT NULL NUMBER |
| NAME | VARCHAR2(10) |
| COST | NUMBER(6) |



SQL> insert into cars values(&model,&name,&cost); Enter value for model: 1098

Enter value for name: 'omni' Enter value for cost: 200000

old 1: insert into cars values(&model,&name,&cost) new 1:

insert into cars values(1098,'omni',200000)

1 row created.

SQL> /

Enter value for model: 9087 Enter value for name: 'qualis' Enter value for cost: 500000

old 1: insert into cars values(&model,&name,&cost) new 1:

insert into cars values(9087,'qualis',500000)

1 row created.

SQL> /

Enter value for model: 1098 Enter value for name: 'innova'

Enter value for cost: 600000

old 1: insert into cars values(&model,&name,&cost) insert into cars values(1098,'innova',600000)

\*

ERROR at line 1:

ORA-00001: unique constraint (SCOTT.SYS\_C001266) violated

#### CHECK CONSTRAINT:

##### SQL> create table employ (

**rno numb er(5), name varch ar(10**

**),**

##### salary number(10) constraint no\_ck check(salary between 10000 and 30000)

**)**

**;**

##### Table created.

SQL> desc employ;

NAME NULL? TYPE



RNO NUMBER(5)

NAME VARCHAR2(10) SALARY NUMBER(10)

SQL> insert into employ values(&rno,&name,&salary); Enter value for rno: 1

Enter value for name: 'sachin' Enter value for salary: 29000

old 1: insert into employ values(&rno,&name,&salary) new 1:

insert into employ values(1,'sachin',29000) SQL> /

Enter value for rno: 20 Enter value for name: 'rohit'

Enter value for salary: 10000

old 1: insert into employ values(&rno, &name, &salary) new 1: insert into employ values(20,'rohit',10000)

1 row created.

SQL> /

Enter value for rno: 15

Enter value for name: 'dhoni'

Enter value for salary: 40000 old 1: insert into employ

values(&rno,&name,&salary) new 1:

insert into employ values(15,'dhoni',40000) insert into employ values(15,'dhoni',40000)

\*

ERROR at line 1:

ORA-02290: check constraint (SCOTT.NO\_CK) violated

#### FOREIGN KEY

SQL> create table admin (

stuid number constraint stuid\_pk primary key, name varchar(10),

permit number(6)

)

;

Table created.

SQL> desc admin;

| NAME | NULL? TYPE |
| --- | --- |
| STUID | NOT NULL NUMBER |
| NAME | VARCHAR2(10) |
| PERMIT | NUMBER(6) |



SQL> insert into admin values(&stuid, '&name', &permit); Enter value for stuid: 1

Enter value for name: ASWIN Enter value for permit: 80 old 1: insert into admin values(&stuid,'&name',&permit) new 1: insert into admin values(1,'ASWIN',80)

1 row created.

SQL> /

Enter value for stuid: 2

Enter value for name: ROHIT Enter value for permit: 67 old 1: insert into admin values(&stuid,'&name',&permit) new 1: insert into admin values(2,'ROHIT',67)

1 row created.

SQL> /

Enter value for stuid: 4

Enter value for name: SANJAY Enter value for permit: 45 old 1: insert into admin values(&stuid,'&name',&permit) new 1: insert into admin values(4,'SANJAY',45)

1 row created.

SQL> /

Enter value for stuid: 5

Enter value for name: KAMALINI Enter value for permit: 35 old 1: insert into admin values(&stuid,'&name',&permit) new 1: insert into admin values(5,'KAMALINI',35)

1 row created.

SQL> select \* from admin;

| STUID | NAME | PERMIT |
| --- | --- | --- |
| 1 | ASWIN | 80 |
| 2 | ROHIT | 67 |
| 4 | SANJAY | 45 |
| 5 | KAMALINI | 35 |



SQL> create table course (

stuid number constraint sid\_fk references admin(stuid), branch varchar(6),

sec varchar(2)

)

;

Table created.

SQL> insert into course values(&stuid,'&branch','&sec'); Enter value for stuid: 1

Enter value for branch: CSE Enter value for sec: A old 1: insert into course values(&stuid,'&branch','&sec') new 1: insert into course values(1,'CSE','A')

1 row created.

SQL> /

Enter value for stuid: 2

Enter value for branch: CSE Enter value for sec: A old 1: insert into course values(&stuid,'&branch','&sec') new 1: insert into course values(2,'CSE','A')

1 row created.

SQL> /

Enter value for stuid: 4

Enter value for branch: IT Enter value for sec: A old 1: insert into course values(&stuid,'&branch','&sec') new 1: insert into course values(4,'IT','A')

1 row created.

SQL> /

Enter value for stuid: 6

Enter value for branch: CSE Enter value for sec: A

old 1: insert into course values(&stuid,'&branch','&sec') new 1: insert into course values(6,'CSE','A')

insert into course values(6,'CSE','A')

\*

ERROR at line 1:

ORA-02291: integrity constraint (SCOTT.SID\_FK) violated - parent key not found

SQL> delete from admin where stuid=5; 1 row deleted.

SQL> delete from admin where stuid=1; delete from admin where stuid=1

\*

ERROR at line 1:

ORA-02292: integrity constraint (SCOTT.SID\_FK) violated - child record found

SQL> select \* from admin;

STUID NAME PERMIT



| 1 | ASWIN | 80 |
| --- | --- | --- |
| 2 | ROHIT | 67 |
| 4 | SANJAY | 45 |

SQL> select \* from course; STUID BRANCH SE



1 CSE A

2 CSE A

4 IT A

SQL> create table student (

idno varch ar(4), name varch ar(10

),

dept varch ar(4), degre e varch ar(6), year num ber(4

)

);

table created.

SQL> desc student;

NAME NULL? TYPE



IDNO VARCHAR2(4)

NAME VARCHAR2(10)

DEPT VARCHAR2(4) DEGREE VARCHAR2(6) YEAR NUMBER(4)

SQL> insert into student values('&idno', '&name', '&dept', '&degree', &year); Enter value for idno: A01

Enter value for name: AARON Enter value for dept: CSE

Enter value for degree: BE Enter value for year: 2 old 1: insert into student values('&idno','&name','&dept','&degree',&year) new 1: insert into student values('a01','aaron','cse','BE',2)

1 row created.

SQL> /

Enter value for idno: A02 Enter value for name: AKIL Enter value for dept: ECE

Enter value for degree: BE Enter value for year: 2 old 1: insert into student values('&idno','&name','&dept','&degree',&year) new 1: insert into student values('A02','AKIL','ECE','BE',2)

1 row created.

SQL> /

Enter value for idno: A03

Enter value for name: BENNY Enter value for dept: IT

Enter value for degree: B.TECH Enter value for year: 2 old 1: insert into student values('&idno','&name','&dept','&degree',&year) new 1: insert into student values('A03','BENNY','IT','B.TECH',2)

1 row created.

SQL> /

Enter value for idno: B01 Enter value for name: COOK Enter value for dept: CSE

Enter value for degree: BE Enter value for year: 1

old 1: insert into student values('&idno','&name','&dept','&degree',&year) new 1: insert into student values('B01','COOK','CSE','BE',1)

1 row created.

SQL> /

Enter value for idno: B02 Enter value for name: DANNY Enter value for dept: MECH

Enter value for degree: BE Enter value for year: 1 old 1: insert into student

values('&idno','&name','&dept','&degree',&year) new 1: insert into student values('B02','DANNY','MECH','BE',1)

1. row created.

SQL> /

Enter value for idno: B03 Enter value for name: ELAN Enter value for dept: IT

Enter value for degree: B.TECH Enter value for year: 1 old 1: insert into student values('&idno','&name','&dept','&degree',&year) new 1: insert into student values('B03','ELAN','IT','B.TECH',1)

1. row created.

SQL> SELECT \* FROM STUDENT;

IDNO NAME DEPT DEGREE YEAR



| A01 | AARON | CSE | BE | 2 |
| --- | --- | --- | --- | --- |
| A02 | AKIL | ECE | BE | 2 |
| A03 | BENNY | IT | B.TECH | 2 |
| B01 | COOK | CSE | BE | 1 |
| B02 | DANNY | MECH | BE | 1 |
| B03 | ELAN | IT | B.TECH | 1 |

6 rows selected.

#### DISTINCT

SQL> select distinct dept from student;

DEPT



CSE ECE IT MECH

SQL> select name from student; NAME



AARON AKIL

B E N N Y C O O K

DANNY ELAN

6 rows selected.

#### IN

SQL> select \* from student where year IN 2; IDNO NAME DEPT DEGREE YEAR



| A01 AARON | CSE | BE | 2 |
| --- | --- | --- | --- |
| A02 AKIL ECE | BE | 2 |  |

A03 BENNY IT B.TECH 2

SQL> select \* from student where name BETWEEN 'AARON' and 'COOK'; IDNO NAME DEPT DEGREE YEAR



| A01 | AARON | CSE BE | 2 |
| --- | --- | --- | --- |
| A02 | AKIL | ECE BE | 2 |
| A03 | BENNY | IT B.TECH | 2 |
| B01 | COOK | CSE BE | 1 |

#### AS

SQL> select IDNO as rollno from student; ROLLNO



A01 A02 A03 B01 B

0

2

B 0

3

6 rows selected.

#### SORT

SQL> select \* from student where year<3 order by name desc; IDNO NAME DEPT DEGREE YEAR

| B03 | ELAN | IT | B.TECH | 1 |
| --- | --- | --- | --- | --- |
| B02 | DANNY | MECH | BE | 1 |
| B01 | COOK | CSE | BE | 1 |
| A03 | BENNY | IT | B.TECH | 2 |
| A02 | AKIL | ECE | BE | 2 |
| A01 | AARON | CSE | BE | 2 |

6 rows selected.

SQL> select \* from student where year<3 order by dept asc; IDNO NAME DEPT DEGREE YEAR



| A01 | AARON | CSE | BE | 2 |
| --- | --- | --- | --- | --- |
| B01 | COOK | CSE | BE | 1 |
| A02 | AKIL | ECE | BE | 2 |
| A03 | BENNY | IT | B.TECH | 2 |
| B03 | ELAN | IT | B.TECH | 1 |
| B02 | DANNY | MECH | BE | 1 |

6 rows selected.

#### LIKE

SQL> select \* from student where name LIKE '%Y';

| IDNO NAME | DEPT DEGREE YEAR |
| --- | --- |
| A03 BENNY | IT B.TECH 2 |
| B02 DANNY | MECH BE 1 |
| SQL> select \* from st udent where name LIKE 'A%'; IDNO NAME DEPT DEGREE YEAR | |

| A01 AARON | CSE | BE | 2 |
| --- | --- | --- | --- |
| A02 AKIL ECE  **IS NULL** | BE | 2 |  |

SQL> select \* from student where IDNO IS NULL; no rows selected

#### LOGICAL OR

SQL> select \* from student where IDNO='A01' OR IDNO='B01'; IDNO NAME DEPT DEGREE YEAR



| A01 | AARON | CSE | BE | 2 |
| --- | --- | --- | --- | --- |
| B01 | COOK | CSE | BE | 1 |

#### RESULT:

Thus the data manipulation language (dml) of base tables and views are executed.

### 1.c Transaction Control Language ;

1. COMMIT: This command is used to end a transaction only with the help of the commit

command transaction changes can be made permanent to the database. Syntax: SQL> COMMIT;

Example: SQL> COMMIT;

1. SAVE POINT: Save points are like marks to divide a very lengthy transaction to smaller

once. They are used to identify a point in a transaction to which we can latter role back. Thus,

save point is used in conjunction with role back. Syntax: SQL> SAVE POINT ID;

Example: SQL> SAVE POINT xyz;

1. ROLLBACK: A role back command is used to undo the current transactions. We can role

back the entire transaction so that all changes made by SQL statements are undo (or) role

38

back a transaction to a save point so that the SQL statements after the save point are role

back.

Syntax: ROLLBACK (current transaction can be role back) ROLLBACK to save point ID;

Example: SQL> ROLLBACK;

SQL> ROLLBACK TO SAVE POINT xyz

#### RESULT:

Thus the data manipulation language (dml) of base tables and views are executed.