` **Grant Command – Privilege like SELECT, INSERT, UPDATE, DELETE, ALTER**

**❖ GRANT ALL PERMISSION ON CUSTOMER TABLE TO EVERYONE**.

SQL> create view custview as select \* from customer;

View created.

SQL> grant all on custview to public;

Grant succeeded.

**❖ GRANT PERMISSION TO INSERT ONLY PARTICULAR COLUMNS IN CUSTVIEW;**

SQL> grant insert (customer\_name) on custview to rdbms;

Grant succeeded.

**❖ GRANT SELECT PERMISSION TO CUSTOMER\_NAME TABLE TO USER RDBMS AND ALLOW HIM TO GIVE FURTHER GRANTS**

SQL> create view custnameview as select \* from customer;

View created**.**

SQL> grant select on custnameview to rdbms with grant option;

Grant succeeded.

**REVOKE** – revoke database privileges from user.

SQL> revoke all on custview from public;

Revoke succeeded.

SQL> revoke select on custnameview from rdbms;

Revoke succeeded**.**

**APPLICATION PRIVILEGES MANAGEMENT – used to create ROLE –**

**collection of related system privileges.**

SQL> create role rkarthik;

Role created.

SQL> grant select, update on custview to rkarthik;

Grant succeeded.

SQL> grant rkarthik to rdbms, scott;

Grant succeeded.

**ENABLE ROLE**

SQL> set role rkarthik;

Role set.

**DROP ROLE**

SQL> drop role rkarthik;

Role dropped.

**INDEXES** –common way to enhance performance in retrieving information from

table.

SQL> create table branchv as select \* from branch;

Table created.

SQL> create index brindx on branchv(branch\_name);

Index created.

SQL> create unique index brrindx on branchv (branch\_name,branch\_city);

Index created.

**DROPPING INDEX**

SQL> drop index brindx;

Index dropped**.**

SQL> drop index brrindx;

Index dropped.

**SEQUENCES** – used to generate unique integers for use as primary key;

SQL> create sequence loanno increment by 1 start with 1000;

Sequence created.

**GENERATING SEQUENCE NUMBER WITH NEXTVAL**

SQL> select loanno.nextval from dual;

NEXTVAL

----------

1000

SQL> create view loanv as select \* from loan;

View created.

**GENERATING SEQUENCE NUMBERS WITH NEXTVAL**

SQL> insert into loanv (loan\_no, branch\_name, amount) values (loanno.nextval, 'sbi', 5000);

1 row created.

SQL> select loanno.currval from dual;

CURRVAL

----------

1001

**USING SEQUENCE NUMBERS WITH CURRVAL**

SQL> select loanno.currval from dual;

CURRVAL

----------

1001

**ALTERING A SEQUENCE:**

SQL> alter sequence loanno increment by 2;

Sequence altered.

**REMOVING SEQUENCE:**

SQL> drop sequence loanno;

Sequence dropped.

**VIEWING SEQUENCE CREATED:**

SQL> select \* from user\_sequences;

no rows selected

**COMMIT AND ROLLBACK** - commit used to make changes to data permanent,

rollback used to discard parts or all the work the user has done in the current

transaction.

SQL> set autocommit on;

SQL> set autocommit off;

SQL> commit;

Commit complete.

SQL> rollback;

Rollback complete.

**SAVEPOINTS:** - identifies a point in a transaction to which one can later

rollback.

SQL> insert into branch values ('kvb', 'usa', 1000);

1 row created.

SQL> savepoint a;

Savepoint created.

SQL> select \* from branch;

BRANCH\_NAME BRANCH\_CITY ASSETS

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

icici mardurai 40000

sbi trichy 80000

hdfc salem 40000

hsbc ahemdabad 40000

pnb mumbai 80000

iob chennai 79000

kvb usa 1000

7 rows selected.

SQL> update branch set assets = 10000 where branch\_name = 'kvb';

1 row updated.

SQL> select \* from branch;

BRANCH\_NAME BRANCH\_CITY ASSETS

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

icici mardurai 40000

sbi trichy 80000

hdfc salem 40000

hsbc ahemdabad 40000

pnb mumbai 80000

iob chennai 79000

kvb usa 10000

7 rows selected.

SQL> rollback to savepoint a;

Rollback complete.

SQL> select \* from branch;

BRANCH\_NAME BRANCH\_CITY ASSETS

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

icici mardurai 40000

sbi trichy 80000

hdfc salem 40000

hsbc ahemdabad 40000

pnb mumbai 80000

iob chennai 79000

kvb usa 1000

7 rows selected.

SQL> rollback to savepoint a;

Rollback complete.

SQL> select \* from branch;

BRANCH\_NAME BRANCH\_CITY ASSETS

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

icici mardurai 40000

sbi trichy 80000

hdfc salem 40000

hsbc ahemdabad 40000

pnb mumbai 80000

iob chennai 79000

kvb usa 1000

7 rows selected.

**TYPE OBJECT:** - handles multivalued attribute.

SQL> create type phno as object (land number(10),office number(10),mobile number(10));

2 /

Type created.

SQL> alter table customer add customer\_phone phno;

Table altered.

QL> update customer set customer\_phone = phno('9999999999','8888888888','7777777777')where customer\_name='kadam';

1 row updated.

SQL> select \* from customer where customer\_name='kadam';

CUSTOMER\_NAME CUSTOMER\_STREET

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

CUSTOMER\_CITY

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

CUSTOMER\_PHONE(LAND, OFFICE, MOBILE)

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

kadam sgroad

marduarai

PHNO(9999999999, 8888888888, 7777777777)