Privileges ====>

* A privilege is a right to execute a particular type of SQL statement.

* There are two types of privileges: SYSTEM PRIVILEGES and OBJECT PRIVILEGES.

* An example of a SYSTEM PRIVILEGE is the right to create a table or an index.

* A particular OBJECT PRIVILEGE allows us to access an individual object, such as the privilege to SELECT from the EMP table, to DELETE from the STUDENTS table, or to SELECT a number from a specific sequence.

System Privileges ===>

* To establish a connection to the database, a user must be granted certain SYSTEM PRIVILEGES.

* These privileges are granted either individually or in the form of roles.

* Remember that a role is a collection of privileges.

* Although the user INDIAN has been created, but it cannot start a session, as we see from the following error message.

* The user lacks the CREATE SESSION SYSTEM PRIVILEGE to log in to the database.

Object Privileges ===>

* Object privileges are granted for a particular object (for example, table, view, sequence).

GRANT Command ===>

* We give a system privilege or an object privilege to a user by using the GRANT command.

* The syntax for granting system privileges is as follows:
- GRANT <list_of_privileges> TO <user_name>;

grant create session, create table to indians;

* The following statement grants the CREATE SESSION and CREATE TABLE system privilege to the INDIAN user.

* This allows the INDIAN user to establish a session to the database and create tables.

Granting Object Privilege ===>

* Object privileges grant certain privileges on specific objects, such as tables, views, or sequences.

* We can grant object privileges to other users when we want them to have access to objects we created.

* We can also grant the user , access to objects , we do not own if the object's owner gives us permission to extend rights to others.

* The following is the general syntax for granting object privileges.
- GRANT <list_of_privileges>
  ON <resource_name>
  TO <user_name>
  [WITH GRANT OPTION];

* For example, the following statement connects as the ORACLEBATCH user account and grants the SELECT privilege on the STUDENTS table to the new user INDIAN.

- connect -> prashant/prashant
grant select on students to indians;

* In this case, the prashant user is the grantor, and indians is the grantee, the recipient of the privileges.

* Now the indians user can query the STUDENTS table.

connect indians/intelligent
select * from prashant.students;

```
ROLL_NO NAME              PER
------- ---------- ----------
    101 Prashant         81.5
    102 Santosh          79.3
    103 Shivam           78.9
```

* In addition to SELECT, other object privileges can be granted on a table, such as INSERT, UPDATE, DELETE, ALTER, INDEX, and REFERENCES .

grant update on students to indians;

* The ALTER privilege allows another user to change tabledefinitions with the ALTER table command, the INDEX privilege allows the creation of indexes on the table, and the REFERENCES privilege allows the table to be referenced with a foreign key constraint.

* We can also grant all object privileges at once by using the GRANT ALL command.

* We can grant UPDATE privilege on individual columns on a table.

* For example, to grant UPDATE on the columns ENAME and JOB of the EMP table, we can execute the following command.
# GRANT UPDATE (ename, job) ON EMP TO INDIAN;

Extending Privilege To Others ===>
==========

* We can extend or further grant the privilege to the resources on which we have got the privilege to other users.

* But for this we must have received the privilege to pass it on to others through WITH GRANT OPTION.

* The following SQL statement grants all object privileges on the EMP table to the INDIAN user.

* It also passes to the INDIAN user the ability to grant these privileges to yet other users, using the WITH GRANT OPTION.

GRANT ALL ON EMP TO INDIAN WITH GRANT OPTION

* Here, INDIAN is the grantee but can become a grantor if the privilege is passed on to another user.

* To allow users to pass on SYSTEM PRIVILEGES to other users, we must have been granted the SYSTEM PRIVILEGE with WITHADMIN OPTION or have been granted the GRANT ANY PRIVILEGE SYSTEM PRIVILEGE.

* For example, after execution of the following statement, the user INDIAN can grant the CREATE SESSION SYSTEM PRIVILEGE to other users.

GRANT CREATE SESSION TO INDIAN WITH ADMIN OPTION

REVOKE Command ===>
==========

* Privileges can be taken away with the REVOKE command.

* Syntax :-
# REVOKE <list_of_privileges>FROM <user_name>;

* Example :-
# revoke update on students from indians;

* The following statement revokes the CREATE SESSION SYSTEM PRIVILEGE from the INDIAN user.
# REVOKE CREATE SESSION FROM INDIAN;

* This prohibits the INDIAN user from establishing a session to the database.

* Object privileges can also be taken away with the REVOKE command.

* The syntax for revoking object privileges is as follows:
# REVOKE <list_of_privileges> ON <resource_name> FROM <user_name>;

An Important Point ==>
==========

* Object privileges granted using the WITH GRANT OPTION are revoked if the grantor’s object privilege is revoked.

* For example, assume that USER1 is granted SELECT privilege on the EMP table, using the WITH GRANT OPTION, and he grants the same privilege to USER2.

* If the SELECT privilege is revoked from USER1, then the revoke cascades to USER2.