Experiment 8

Name	Ameya S. Daddikar
College I.D.	161070015
Course	Btech. Computer Engineering

Aim

Create the environment for the database application. Perform database administration and performance of database Application.(grant, revoke).

Theory

Oracle Database (Oracle XE)

Oracle XE (eXpress Edition) is Oracle Corporation's free to use and distribute database edition. XE is available for Windows and Linux and can be downloaded free of charge from Oracle TechNet. Linux RPM's are also available for easy deployment on Linux servers.

Features

Multitenant: Get isolation, agility, and economies of scale by managing multiple Pluggable Databases inside your Oracle Multitenant Container Database

In-Memory: Support real-time analytics, business intelligence, and reports by keeping your important data in the Oracle Database In-Memory column store

Partitioning: Enhance performance, availability, and manageability of your database with data partitioning that meets diverse business requirements

Advanced Analytics: Get valuable insights and deliver predictions from your data using Data Mining SQL, R programming, and the Oracle Data Miner UI

Advanced Security: Protect your sensitive data at the source and build end-to-end encrypted apps with layers of security including Oracle Transparent Data Encryption and Data Redaction.

Required Resources

- Up to 12 GB of user data
- Up to 2 GB of database RAM
- Up to 2 CPU threads
- Up to 3 Pluggable Databases

User Privileges and Roles

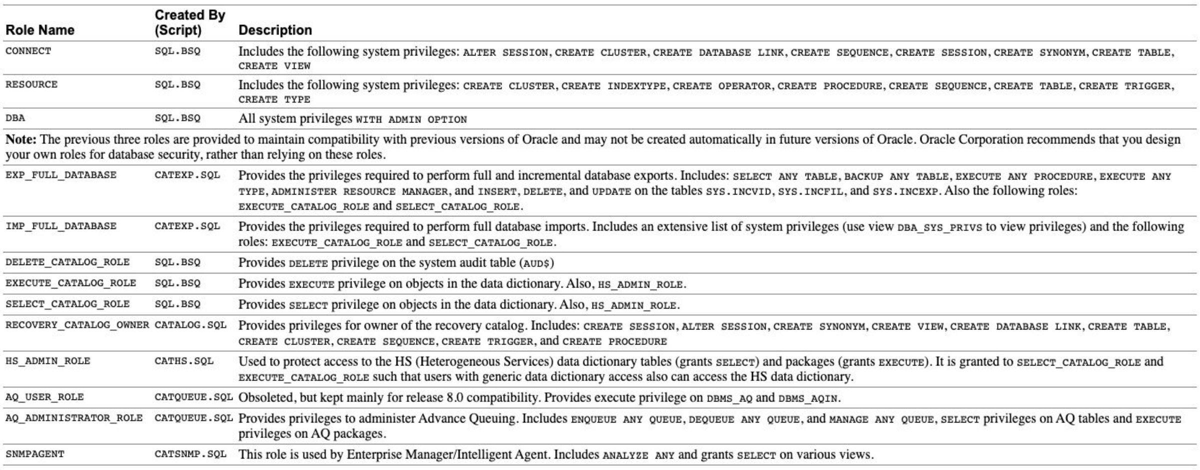
A user privilege is a right to execute a particular type of SQL statement, or a right to access another user's object. The types of privileges are defined by Oracle.

Roles, on the other hand, are created by users (usually administrators) and are used to group together privileges or other roles. They are a means of facilitating the granting of multiple privileges or roles to users.

User Roles

A role groups several privileges and roles, so that they can be granted to and revoked from users simultaneously. A role must be enabled for a user before it can be used by the user.

Oracle provides some predefined roles to help in database administration. These roles, listed in the table below, are automatically defined for Oracle databases when you run the standard scripts that are part of database creation. You can grant privileges and roles to, and revoke privileges and roles from, these predefined roles in the same way as you do with any role you define.



Role Name	Created By (Script)	Description
CONNECT	SQL.BSQ	Includes the following system privileges: ALTER SESSION, CREATE CLUSTER, CREATE DATABASE LINK, CREATE SEQUENCE, CREATE SESSION, CREATE SYNONYM, CREATE TABLE, CREATE VIEW
RESOURCE	SQL.BSQ	Includes the following system privileges: CREATE CLUSTER, CREATE INDEXTYPE, CREATE OPERATOR, CREATE PROCEDURE, CREATE SEQUENCE, CREATE TABLE, CREATE TRIGGER, CREATE TYPE
DBA	SQL.BSQ	All system privileges WITH ADMIN OPTION
Note: The previous three roles are provided to maintain compatibility with previous versions of Oracle and may not be created automatically in future versions of Oracle. Oracle Corporation recommends that you design your own roles for database security, rather than relying on these roles.		
EXP_FULL_DATABASE	CATEXP.SQL	Provides the privileges required to perform full and incremental database exports. Includes: SELECT ANY TABLE, BACKUP ANY TABLE, EXECUTE ANY PROCEDURE, EXECUTE ANY TYPE, ADMINISTER RESOURCE MANAGER, and INSERT, DELETE, and UPDATE on the tables SYS.INCVID, SYS.INCFIL, and SYS.INCEXP. Also the following roles: EXECUTE_CATALOG_ROLE and SELECT_CATALOG_ROLE.
IMP_FULL_DATABASE	CATEXP.SQL	Provides the privileges required to perform full database imports. Includes an extensive list of system privileges (use view DBA_SYS_PRIVS to view privileges) and the following roles: EXECUTE_CATALOG_ROLE and SELECT_CATALOG_ROLE.
DELETE_CATALOG_ROLE	SQL.BSQ	Provides DELETE privilege on the system audit table (AUD$)
EXECUTE_CATALOG_ROLE	SQL.BSQ	Provides EXECUTE privilege on objects in the data dictionary. Also, HS_ADMIN_ROLE.
SELECT_CATALOG_ROLE	SQL.BSQ	Provides SELECT privilege on objects in the data dictionary. Also, HS_ADMIN_ROLE.
RECOVERY_CATALOG_OWNER	CATALOG.SQL	Provides privileges for owner of the recovery catalog. Includes: CREATE SESSION, ALTER SESSION, CREATE SYNONYM, CREATE VIEW, CREATE DATABASE LINK, CREATE TABLE, CREATE CLUSTER, CREATE SEQUENCE, CREATE TRIGGER, and CREATE PROCEDURE
HS_ADMIN_ROLE	CATHS.SQL	Used to protect access to the HS (Heterogeneous Services) data dictionary tables (grants SELECT) and packages (grants EXECUTE). It is granted to SELECT_CATALOG_ROLE and EXECUTE_CATALOG_ROLE such that users with generic data dictionary access also can access the HS data dictionary.
AQ_USER_ROLE	CATQUEUE.SQL	Obsoleted, but kept mainly for release 8.0 compatibility. Provides execute privilege on DBMS_AQ and DBMS_AQIN.
AQ_ADMINISTRATOR_ROLE	CATQUEUE.SQL	Provides privileges to administer Advance Queuing. Includes ENQUEUE ANY QUEUE, DEQUEUE ANY QUEUE, and MANAGE ANY QUEUE, SELECT privileges on AQ tables and EXECUTE privileges on AQ packages.
SNMPAGENT	CATSNMP.SQL	This role is used by Enterprise Manager/Intelligent Agent. Includes ANALYZE ANY and grants SELECT on various views.

Creating Roles

You can create a role using the CREATE ROLE statement, but you must have the CREATE ROLE system privilege to do so. Typically, only security administrators have this system privilege.

Immediately after creation, a role has no privileges associated with it. To associate privileges with a new role, you must grant privileges or other roles to the new role.

You must give each role you create a unique name among existing usernames and role names of the database. Roles are not contained in the schema of any user. In a database that uses a multibyte character set, Oracle recommends that each role name contain at least one single-byte character. If a role name contains only multibyte characters, the encrypted role name/password combination is considerably less secure.

The following statement creates the clerk role, which is authorized by the database using the password bicentennial:

CREATE ROLE clerk IDENTIFIED BY bicentennial;

The IDENTIFIED BY clause specifies how the user must be authorized before the role can be enabled for use by a specific user to which it has been granted. If this clause is not specified, or NOT IDENTIFIED is specified, then no authorization is required when the role is enabled. Roles can be specified to be authorized by:

- The database using a password
- An application using a specified package
- Externally by the operating system, network, or other external source
- Globally by an enterprise directory service

Later, you can set or change the authorization method for a role using the ALTER ROLE statement. The following statement alters the clerk role to specify that the user must have been authorized by an external source before enabling the role:

ALTER ROLE clerk IDENTIFIED EXTERNALLY;

Dropping Roles

In some cases, it may be appropriate to drop a role from the database. The security domains of all users and roles granted a dropped role are immediately changed to reflect the absence of the dropped role's privileges. All indirectly granted roles of the dropped role are also removed from affected security domains. Dropping a role automatically removes the role from all users' default role lists.

Because the creation of objects is not dependent on the privileges received through a role, tables and other objects are not dropped when a role is dropped.

You can drop a role using the SQL statement DROP ROLE. To drop a role, you must have the DROP ANY ROLE system privilege or have been granted the role with the ADMIN OPTION.

The following statement drops the role CLERK:

DROP ROLE clerk;

Granting System Privileges and Roles

You can grant system privileges and roles to other users and roles using the GRANT statement. The following privileges are required:

To grant a system privilege, you must have been granted the system privilege with the ADMIN OPTION or have been granted the GRANT ANY PRIVILEGE system privilege.

To grant a role, you must have been granted the role with the ADMIN OPTION or have been granted the GRANT ANY ROLE system privilege.

Note: You cannot grant a roll that is IDENTIFIED GLOBALLY to anything. The granting (and revoking) of global roles is controlled entirely by the enterprise directory service.

The following statement grants the system privilege CREATE SESSION and the accts_pay role to the user jward:

GRANT CREATE SESSION, accts_pay TO jward;

A user or role that is granted a privilege or role specifying the WITH ADMIN OPTION clause has several expanded capabilities:

- The grantee can grant or revoke the system privilege or role to or from any user or other role in the database. Users cannot revoke a role from themselves.
- The grantee can further grant the system privilege or role with the ADMIN OPTION.
- The grantee of a role can alter or drop the role.

In the following statement, the security administrator grants the new_dba role to michael:

GRANT new_dba TO michael WITH ADMIN OPTION;

Oracle allows you to create a new user with the GRANT statement. If you specify a password using the IDENTIFIED BY clause, and the username/password does not exist in the database, a new user with that username and password is created. The following example creates ssmith as a new user while granting ssmith the CONNECT system privilege:

GRANT CONNECT TO ssmith IDENTIFIED BY p1q2r3;

Revoking User Privileges and Roles

You can revoke system privileges and roles using the SQL statement REVOKE.

Any user with the ADMIN OPTION for a system privilege or role can revoke the privilege or role from any other database user or role. The revoker does not have to be the user that originally granted the privilege or role. Users with GRANT ANY ROLE can revoke any role.

The following statement revokes the CREATE TABLE system privilege and the accts_rec role from tsmith:

REVOKE CREATE TABLE, accts_rec FROM tsmith;

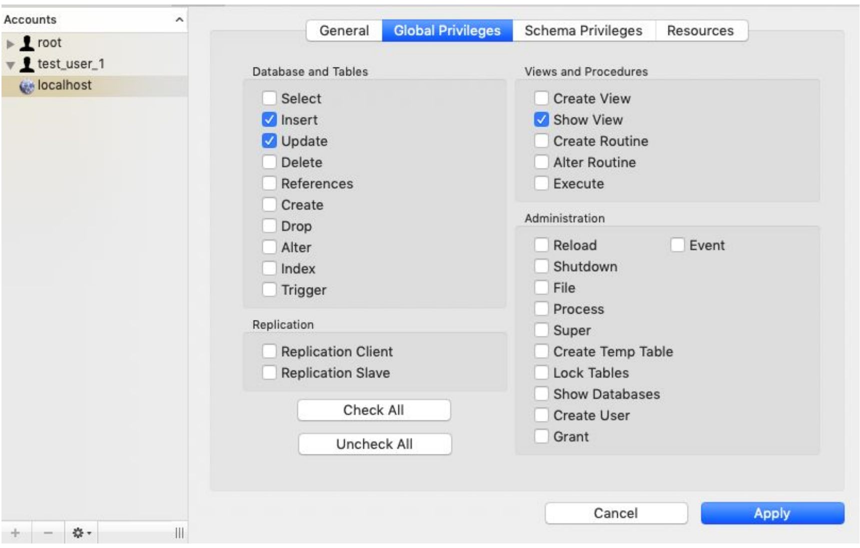
Depending on what is granted or revoked, a grant or revoke takes effect at different times:

When Do Grants and Revokes Take Effect?

All grants/revokes of system and object privileges to anything (users, roles, and PUBLIC) are immediately observed.

- All grants/revokes of roles to anything (users, other roles, PUBLIC) are only observed when a current user session issues a SET ROLE statement to re-enable the role after the grant/revoke, or when a new user session is created after the grant/revoke.
- You can see which roles are currently enabled by examining the SESSION_ROLES data dictionary view.

Output



Accounts: root; test_user_1 → localhost

Database and Tables

☐ Select
☑ Insert
☑ Update
☐ Delete
☐ References
☐ Create
☐ Drop
☐ Alter
☐ Index
☐ Trigger

Replication

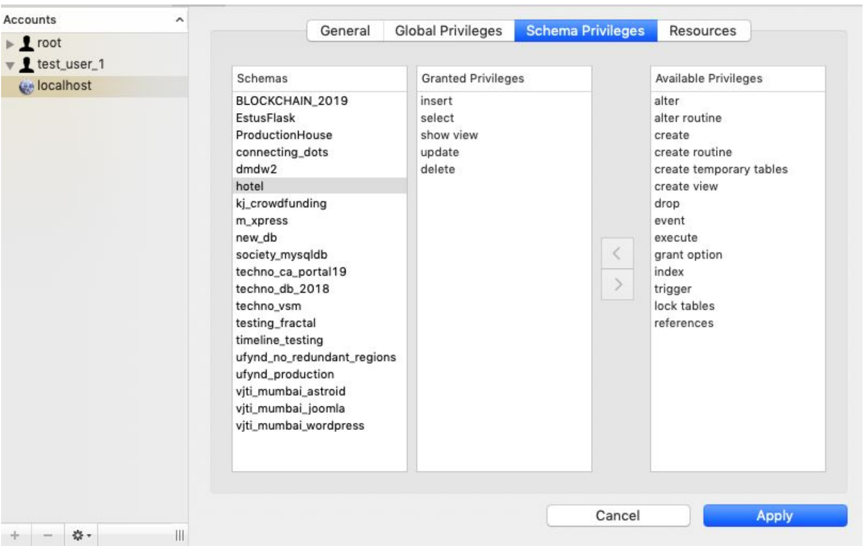
☐ Replication Client
☐ Replication Slave

Views and Procedures

☐ Create View
☑ Show View
☐ Create Routine
☐ Alter Routine
☐ Execute

Administration

☐ Reload ☐ Event
☐ Shutdown
☐ File
☐ Process
☐ Super
☐ Create Temp Table
☐ Lock Tables
☐ Show Databases
☐ Create User
☐ Grant



Accounts: root; test_user_1 → localhost

Schemas: BLOCKCHAIN_2019, EstusFlask, ProductionHouse, connecting_dots, dmdw2, hotel, kj_crowdfunding, m_xpress, new_db, society_mysqldb, techno_ca_portal19, techno_db_2018, techno_vsm, testing_fractal, timeline_testing, ufynd_no_redundant_regions, ufynd_production, vjti_mumbai_astroid, vjti_mumbai_joomla, vjti_mumbai_wordpress

Granted Privileges: insert, select, show view, update, delete

Available Privileges: alter, alter routine, create, create routine, create temporary tables, create view, drop, event, execute, grant option, index, trigger, lock tables, references

Conclusion

Thus, we reviewed the Privileges and Roles in an Oracle Database environment.