

13

Controlling User Access

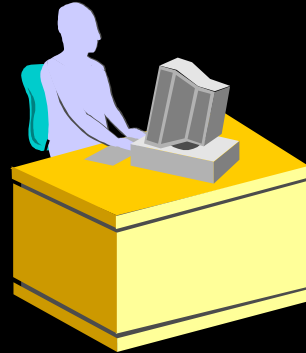
Objectives

After completing this lesson, you should be able to do the following:

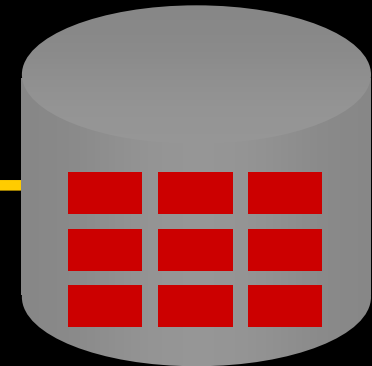
- **Create users**
- **Create roles to ease setup and maintenance of the security model**
- **Use the GRANT and REVOKE statements to grant and revoke object privileges**
- **Create and access database links**

Controlling User Access

**Database
administrator**



Username and password
Privileges



Users



Privileges

- **Database security:**
 - **System security**
 - **Data security**
- **System privileges: Gaining access to the database**
- **Object privileges: Manipulating the content of the database objects**
- **Schemas: Collections of objects, such as tables, views, and sequences**

System Privileges

- **More than 100 privileges are available.**
- **The database administrator has high-level system privileges for tasks such as:**
 - **Creating new users**
 - **Removing users**
 - **Removing tables**
 - **Backing up tables**

Creating Users

The DBA creates users by using the **CREATE USER** statement.

```
CREATE USER user  
IDENTIFIED BY password;
```

```
CREATE USER scott  
IDENTIFIED BY tiger;  
User created.
```

User System Privileges

- Once a user is created, the DBA can grant specific system privileges to a user.

```
GRANT privilege [, privilege...]  
TO user [, user| role, PUBLIC...];
```

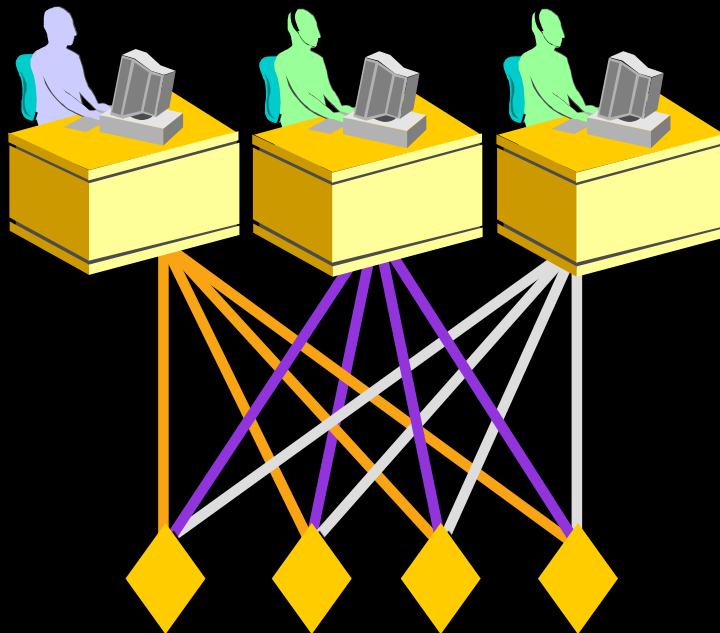
- An application developer, for example, may have the following system privileges:
 - CREATE SESSION
 - CREATE TABLE
 - CREATE SEQUENCE
 - CREATE VIEW
 - CREATE PROCEDURE

Granting System Privileges

The DBA can grant a user specific system privileges.

```
GRANT  create session, create table,  
       create sequence, create view  
TO     scott;  
Grant succeeded.
```

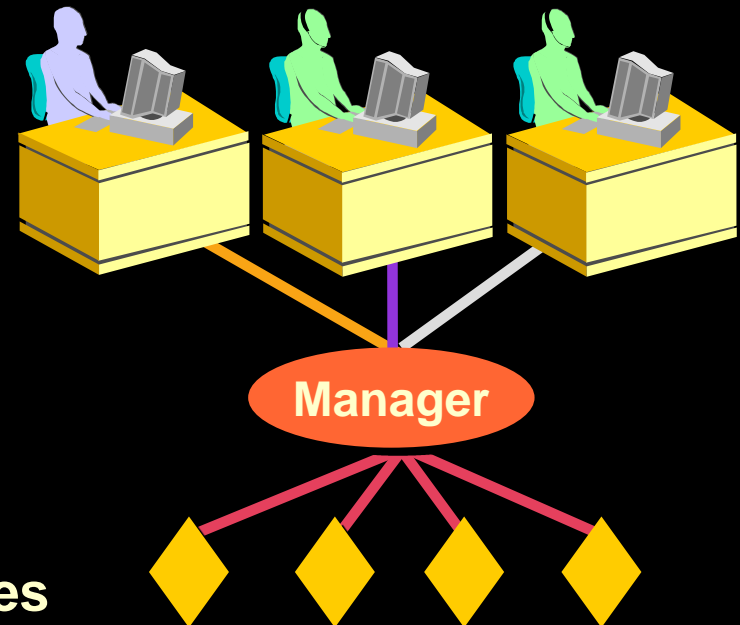

What is a Role?



**Allocating privileges
without a role**

Users

Privileges



**Allocating privileges
with a role**

Creating and Granting Privileges to a Role

- **Create a role**

```
CREATE ROLE manager;  
Role created.
```

- **Grant privileges to a role**

```
GRANT create table, create view  
TO manager;  
Grant succeeded.
```

- **Grant a role to users**

```
GRANT manager TO DEHAAN, KOCHHAR;  
Grant succeeded.
```

Changing Your Password

- The DBA creates your user account and initializes your password.
- You can change your password by using the `ALTER USER` statement.

```
ALTER USER scott  
IDENTIFIED BY lion;  
User altered.
```

Object Privileges

Object Privilege	Table	View	Sequence	Procedure
ALTER	√		√	
DELETE	√	√		
EXECUTE				√
INDEX	√			
INSERT	√	√		
REFERENCES	√	√		
SELECT	√	√	√	
UPDATE	√	√		

Object Privileges

- Object privileges vary from object to object.
- An owner has all the privileges on the object.
- An owner can give specific privileges on that owner's object.

```
GRANT      object_priv [(columns)]  
ON         object  
TO         {user|role|PUBLIC}  
[WITH GRANT OPTION];
```

Granting Object Privileges

- Grant query privileges on the EMPLOYEES table.

```
GRANT  select
ON      employees
TO      sue, rich;
Grant succeeded.
```

- Grant privileges to update specific columns to users and roles.

```
GRANT  update (department_name, location_id)
ON      departments
TO      scott, manager;
Grant succeeded.
```

Using the WITH GRANT OPTION and PUBLIC Keywords

- Give a user authority to pass along privileges.

```
GRANT  select, insert
ON     departments
TO     scott
WITH   GRANT OPTION;
Grant succeeded.
```

- Allow all users on the system to query data from Alice's DEPARTMENTS table.

```
GRANT  select
ON     alice.departments
TO     PUBLIC;
Grant succeeded.
```

Confirming Privileges Granted

Data Dictionary View	Description
<code>ROLE_SYS_PRIVS</code>	System privileges granted to roles
<code>ROLE_TAB_PRIVS</code>	Table privileges granted to roles
<code>USER_ROLE_PRIVS</code>	Roles accessible by the user
<code>USER_TAB_PRIVS_MADE</code>	Object privileges granted on the user's objects
<code>USER_TAB_PRIVS_RECD</code>	Object privileges granted to the user
<code>USER_COL_PRIVS_MADE</code>	Object privileges granted on the columns of the user's objects
<code>USER_COL_PRIVS_RECD</code>	Object privileges granted to the user on specific columns
<code>USER_SYS_PRIVS</code>	Lists system privileges granted to the user

How to Revoke Object Privileges

- You use the **REVOKE** statement to revoke privileges granted to other users.
- Privileges granted to others through the **WITH GRANT OPTION** clause are also revoked.

```
REVOKE {privilege [, privilege...]|ALL}  
ON      object  
FROM    {user[, user...]|role|PUBLIC}  
[CASCADE CONSTRAINTS];
```

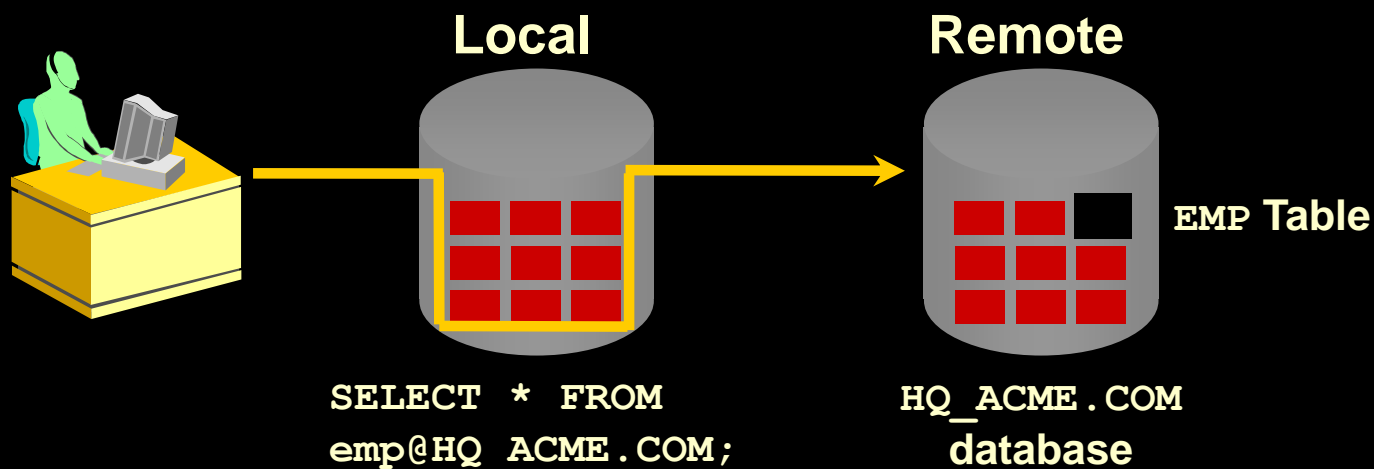
Revoking Object Privileges

As user Alice, revoke the **SELECT** and **INSERT** privileges given to user **Scott** on the **DEPARTMENTS** table.

```
REVOKE  select, insert
ON      departments
FROM    scott;
Revoke succeeded.
```

Database Links

A database link connection allows local users to access data on a remote database.



Database Links

- Create the database link.

```
CREATE PUBLIC DATABASE LINK hq.acme.com  
USING 'sales';  
Database link created.
```

- Write SQL statements that use the database link.

```
SELECT *  
FROM emp@HQ.ACME.COM;
```

Summary

In this lesson, you should have learned about DCL statements that control access to the database and database objects:

Statement	Action
CREATE USER	Creates a user (usually performed by a DBA)
GRANT	Gives other users privileges to access the your objects
CREATE ROLE	Creates a collection of privileges (usually performed by a DBA)
ALTER USER	Changes a user's password
REVOKE	Removes privileges on an object from users

Practice 13 Overview

This practice covers the following topics:

- **Granting other users privileges to your table**
- **Modifying another user's table through the privileges granted to you**
- **Creating a synonym**
- **Querying the data dictionary views related to privileges**