



Gestió d'usuaris i permisos





- Definir i eliminar comptes d'usuari.
- Crear sinònims de taules i vistes.
- Identificar els privilegis sobre les bases de dades i els seus elements.
- Agrupar i desagrupar privilegis.
- Assignar i eliminar privilegis a usuaris.
- Assignar i eliminar grups de privilegis a usuaris.
- Crear vistes personalitzades per a cada tipus d'usuari.
- Garantir el compliment dels requisits de seguretat.



Usuaris
Permisos
Grups / Rols
Quotes
Perfils
Vistes
Esquemes / Sinònims



Seguretat

Control d'accés

Identificació

Autenticació

Permisos



Autenticitat

No repudi

+

Autorització

Comptabilitat



Usuaris

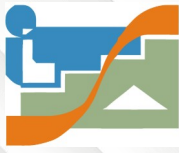
Identifica a un actor físic o digital que ha de realitzar accions en la BBDD.

Un usuari necessita una contrasenya per garantir **autenticació**

Accions amb usuaris

- ✓ Alta d'usuaris
- ✓ Modificació d'usuaris
- ✓ Consulta d'usuaris
- ✓ Baixa d'usuaris
- ✓ Boquejar / desbloquejar usuaris
- ✓ Forçar a usuari a canviar contrasenya

També es gestiona:
Comptabilitat i
No repudi



Permisos / privilegis

- ✓ Permisos de sistema
- ✓ Permisos sobre objectes

Defineixen que pot
fer o que no pot fer
un usuari

S'utilitzen per
garantir
confidencialitat

De sistema:

Connectar
Crear objectes
Consultar
Realitzar tasques d'Administració

Sobre objectes:

SELECT, INSERT, UPDATE, DELETE
ALTER
EXECUTE
INDEX



Permisos / privilegis

- ✓ Permisos de sistema
- ✓ Permisos sobre objectes

Accions amb privilegis

- ✓ Donar privilegi a usuari
- ✓ Llevar privilegi a usuari
- ✓ Permetre propagar privilegi a usuari



Rols

Un rol de treball defineix la funció en el negoci d'un usuari com, per exemple, Vicepresident de vendes, Analista de recursos humans o Responsable de compres

També podem definir un rol com un conjunt de privilegis

Un rol pot abastar diversos usuaris

Un usuari pot tindre diversos rols

Un usuari pot canviar de rol

Els rols faciliten molt el treball del DBA

e.g.





Rols

Accions amb rols

- ✓ Crear ROL
- ✓ Esborrar ROL
- ✓ Assignar privilegi a ROL / Llevar privilegi a ROL
- ✓ Assignar ROL a usuari / Llevar ROL a usuari
- ✓ Assignar ROL a altre ROL / Llevar ROL a altre ROL
- ✓ Activar / desactivar ROL
- ✓ Consultar ROL
- ✓ Consultar assignacions de ROLs (privilegis i rols)



Rols

Per facilitar l'administració, els SGBD solen disposar d'alguns rols predefinits

- DBA
- Manteniment
- Gestor BD
- Backup
- etc

No confondre amb els privilegis SYSDBA i SYSOPER, que no son ROLS

En **ORACLE** estan els rols CONNECT , RESOURCE i DBA



Usuaris, Permisos, Rols, Perfils, Vistes

Tots els usuaris, rols i permisos, quotes, vistes, etc. que creem seran objectes del SGBD pel que s'emmagatzemaran en el **diccionari de dades** i podran ser consultats amb posterioritat per a la correcta administració del SGBD, comprovació, operació, etc.



Quotes i Perfils

Els **perfils** permeten especificar dos tipus de restriccions sobre usuaris

- Limitacions d'ús de recursos (cpu, **quotes** de disc, connexions, etc.)
- Característiques que deuen complir les contrasenyes per garantir la seguretat

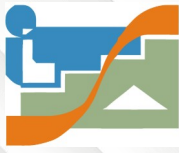


Vistes

Una vista és una consulta emmagatzemada a la qual se li assigna un nom a fi d'utilitzar-la tantes vegades com es desitge. Una vista no conté dades sinó la instrucció SELECT necessària per a crear la vista, això assegura que les dades siguen coherents en utilitzar les dades emmagatzemades en les taules.

Les vistes s'empren per a:

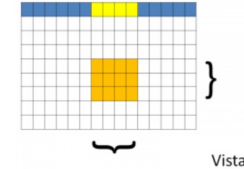
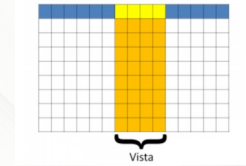
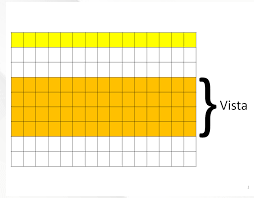
- Realitzar consultes complexes més fàcilment,
- Proporcionar taules amb dades resultants de formatar o realitzar càlculs sobre les dades originals
- Proporcionar formes personalitzades i més comprensibles de les dades
- Ocultar l'emmagatzematge intrínsec de la base de dades i aconseguir una major independència de les dades respecte a la resta d'elements de la base de dades.
- Ser utilitzades com a cursors de dades en els llenguatges procedimentals (com PL/SQL)
- Restringir l'accés a les dades originals**



Vistes

Tipus de vistes

- Vistes horitzontals
- Vistes verticals
- Vistes mixtes



WHERE

SELECT
camp1,camp2

```
CREATE [OR REPLACE] VIEW nom_vista AS SELECT ..... ;  
DROP VIEW nom_vista ;
```


Usuaris i permisos en ORACLE



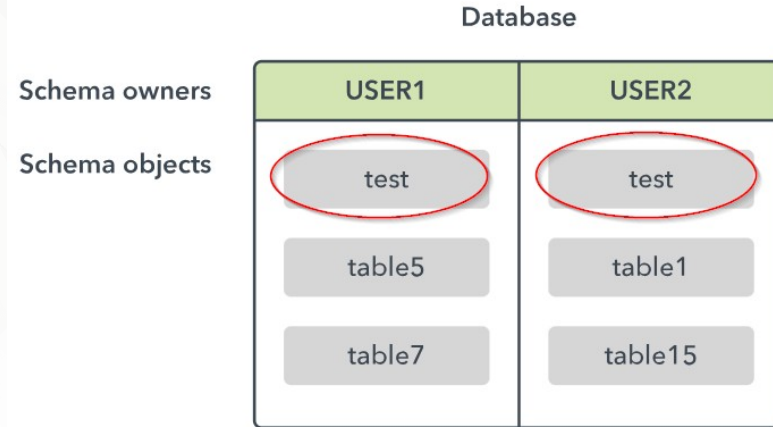
Usuaris en ORACLE / Esquemes

Esquemes

L'esquema d'un usuari en Oracle és una col·lecció d'objectes de base de dades com: taules, vistes, stored procedures, triggers, jobs, seqüències, índexs, etc.

Un usuari de base de dades posseeix un esquema de base de dades, que té el mateix nom que el nom d'usuari.

L'usuari és l'amo d'aquests objectes i podrà assignar permisos a altres usuaris per a accedir als seus objectes



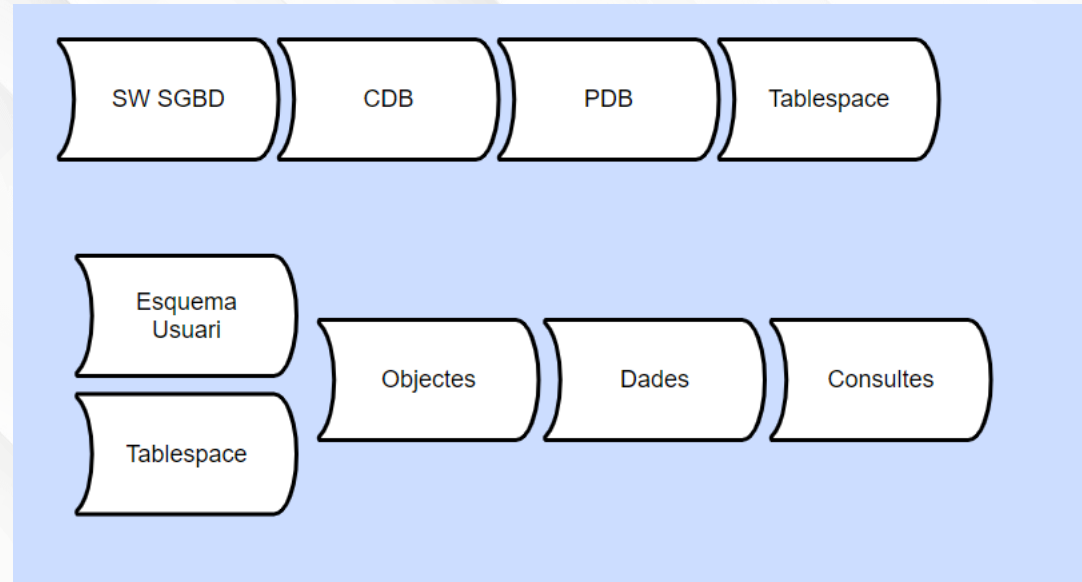
```
CREATE TABLE "USUARI1"."ALUMNES"
```



Usuaris en ORACLE / Esquemes

Hi ha tablespaces
predefinit

Hi ha usuaris
predefinit





Sols deuen
ser utilitzats
pels DBA

Usuaris en ORACLE

Usuaris predefinitos . Oracle es crea automàticament quan crea la BD. Oracle21c crea 37 usuaris predefinitos. Els principals són:

SYS -> Pot fer-ho tot. No es recomana utilitzar-lo si no és necessari i no es recomana crear objectes en el seu esquema.

SYSTEM -> Pot fer-ho tot menys Actualitzar BD i funcions de Backup i Restore. No es recomana utilitzar-lo si no és necessari i no es recomana crear objectes en el seu esquema.

SYSBACKUP -> Usuari amb permisos de Backup i Restore i altres.....



La vista **v\$pwfile_users** ens permet veure els usuaris administratius

Prova-ho en
SQL Developer



Usuaris en ORACLE

Oracle permet **crear** USUARIS propis amb credencials emmagatzemades en el SGBD o **utilitzar** USUARIS externs (S.O. LDAP, Kerberos o Radius).
Diferència: Autenticació.

Arquitectura Multitenant de Oracle.
Contenidor principal CDB que conté:

CDB\$ROOT

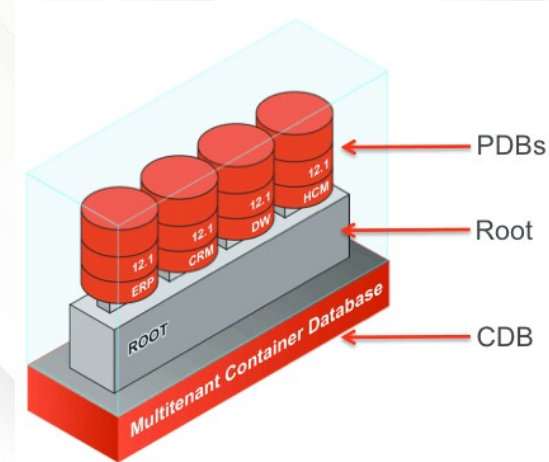
PDB\$SEED

PDB1 (Pluggable Data Base)

PDB2 (Pluggable Data Base)

PDB3 (Pluggable Data Base)

.....





Usuaris en ORACLE

Primera connexió

C:\> sqlplus / as sysdba

No demana contrasenya

o

sqlplus sys as sysdba

Demana contrasenya

Connecta com usuari **SYS** al Contenidor principal CDB , CDB\$ROOT
Amb privilegi de **sysdba**

*** SYS no pot connectar sense privilegi de sysdba

C:\> sqlplus system

Demana contrasenya

Connecta com usuari **SYSTEM** al Contenidor principal CDB , CDB\$ROOT
Sense privilegi de **sysdba**

SYSTEM pot connectar sense i amb privilegi de sysdba (però s'ha d'indicar)

C:\> sqlplus system as sysdba

Demana contrasenya

Prova-ho en sql*plus
Observa com funciona



Usuaris en ORACLE

Mètodes d'autenticació

*Creating Users Who Are Authenticated by the Database

CREATE USER sebastian IDENTIFIED BY password;

Exemple d'accés: **C:\> sqlplus system o sqlplus sys as sysdba**

*Schema-Only Accounts

CREATE USER psmith NO AUTHENTICATION;

*Operating System Authentication of Users

CREATE USER sebastian IDENTIFIED EXTERNALLY;

Exemple d'accés: **C:\> sqlplus / as sysdba**

* Authenticate Oracle Database Enterprise User Security users

CREATE USER sebastian IDENTIFIED GLOBALLY;

* Network Authentication of Users (TLS, Kerberos, Radius, LDAP, PKI,)



Usuaris

No es poden crear usuaris en el
contenidor principal CDB !!

Alta Usuaris

```
CREATE USER svf IDENTIFIED BY password ;
```

en oracle
(els usuaris es creen sense privilegis.....)

Modificació Usuaris

```
ALTER USER svf IDENTIFIED BY password ;
```

en oracle

Baixa Usuaris

```
DROP USER svf ;
```

en oracle

Consulta Usuaris (DD)

```
select * from DBA_USERS;
```



dba_users

Sols poden crear
usuaris els
administradors

O un usuari
amb permisos
especials

Prova-ho en SQL Developer
Observa com funciona



Usuaris en ORACLE

Dins de les BD hi han dos tipus d'usuaris

- Usuaris **locals** (a les respectives PDBs)

```
SQL> CONNECT system@localhost/PDB1
```

```
SQL> create user usuari1 identified by '1234' ;
```

```
SQL> CONNECT system@localhost/PDB2
```

```
SQL> create user usuari1 identified by '2345' ;
```

usuari1 de PDB1 és
diferent a usuari1
de PDB2 !!!

- Usuaris **comuns** (a totes les BD's) built-in sys, system ...

```
sqlplus system@localhost/ORCL
```

Es poden crear usuaris comuns a totes les pdbc amb el prefix C##

```
SQL> create user C##<nom-usuari-comu> identified by '1234' CONTAINER=ALL;
```

oracle recomana
NO utilitzar-los en
producció!!

Com saber el prefixe de l'usuari comú (estant connectats al contenidor principal CDB)

```
SQL> show parameter common
```



Usuaris en ORACLE

Connexió d'usuaris nous a la bbdd

Abans de connectar s'ha de donar permís de connexió a l'usuari

Des de system en pdb1 -> `SQL> grant create session to usuari1;`

usuari1
ha d'estar
creat i
existir

Connectar Indicant la contrasenya

```
C:\Users\usuari1> sqlplus usuari1/1234@localhost/pdb1
```

```
SQL> CONNECT usuari1/1234@localhost/PDB1 ;
```

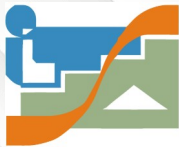
Connectar No indicant la contrasenya, després la demana

```
C:\Users\usuari1> sqlplus usuari1@localhost/pdb1
```

```
SQL> CONNECT usuari1@localhost/PDB1 ; ***
```

Prova-ho en sql*plus
Observa com funciona

L'usuari ja pot connectar, però de moment, res més ...



Usuaris en ORACLE

Sintaxis de create user

```
SQL> CREATE USER josep  
IDENTIFIED BY contrasenya  
DEFAULT TABLESPACE example  
QUOTA 10M ON example  
TEMPORARY TABLESPACE temp  
QUOTA 5M ON system  
PROFILE app_user ;
```

**

```
ALTER USER josep  
IDENTIFIED BY second_2nd_pwd  
DEFAULT TABLESPACE example2  
QUOTA UNLIMITED ON example2;
```

```
DROP USER josep [CASCADE]; * oracle no esborra usuaris connectats
```

Un esquema "schema" d'Oracle conté tots els objectes creats per un usuari de base de dades específic. Quan es crea un usuari, es crea automàticament un esquema amb el seu nom

Si no s'especifica un tablespace, oracle li assignarà el tablespace USERS

**Sense
quota !!**



Usuaris en ORACLE

Crear un usuari **operatiu**

Des de system

```
SQL> CREATE USER enric IDENTIFIED BY secret;
```

```
SQL> GRANT CONNECT, RESOURCE TO enric;
```

=====

```
SQL> ALTER USER enric QUOTA 20M ON users;
```

o

```
SQL> ALTER USER enric QUOTA UNLIMITED ON users;
```

==> L'usuari està preparat per connectar, crear objectes en el seu esquema i afegir informació dins dels objectes

RESOURCE:

CREATE TYPE
CREATE TABLE
CREATE CLUSTER
CREATE TRIGGER
CREATE OPERATOR
CREATE SEQUENCE
CREATE INDEXTYPE
CREATE PROCEDURE

És el TABLESPACE per defecte



Usuaris en ORACLE

Crear un usuari **operatiu**, recomanable utilitzar una **tablespace** diferent.

Des de system

```
SQL> CREATE USER enric IDENTIFIED BY secret  
DEFAULT TABLESPACE tab_app  
QUOTA UNLIMITED ON tab_app  
PASSWORD EXPIRE ;
```

Utilitzar un (o més)
TABLESPACE per cada
aplicació

```
SQL> GRANT CONNECT, RESOURCE TO enric;  
SQL> GRANT CREATE SESSION, RESOURCE TO enric;
```

Forçar a l'usuari a
canviar la password en
la primera connexió

CONNEXIÓ

OPERACIÓ

OPERACIÓ

OPERACIÓ

DESCONNEXIÓ

Prova-ho en SQL Developer
Observa com funciona



Usuaris en ORACLE

Altres possibilitats

Des de l'usuari vicent

```
ALTER USER vicent IDENTIFIED BY novapass;  
PASSWORD (<= comando sql*plus o SQL Developer)
```

```
SQL> password  
Cambiando la contraseña para PRUEBA  
Contraseña antigua:  
Contraseña nueva:  
Confirmar contraseña nueva:  
Contraseña cambiada  
SQL>
```

Des de l'usuari system

```
ALTER USER vicent PASSWORD EXPIRE ;
```

```
ALTER USER vicent ACCOUNT LOCK;
```

```
ALTER USER vicent ACCOUNT UNLOCK;
```

Un compte es bloqueja, be perquè es superen els intents d'accés erronis, , be perquè l'admin força el bloqueig

```
SELECT USERNAME, ACCOUNT_STATUS FROM DBA_USERS WHERE USERNAME = 'VICENT';  
o  
SELECT * FROM DBA_USERS WHERE USERNAME = 'VICENT';
```

dba_users





Usuaris en ORACLE

Des de l'usuari system

```
ALTER USER vicent PASSWORD EXPIRE ;  
ALTER USER vicent ACCOUNT LOCK;  
ALTER USER vicent ACCOUNT UNLOCK;
```

Un compte expira, be perquè supera el temps màxim , be perquè l'admin força l'expiració. L'objectiu és que l'usuari canvie la password obligatòriament

```
SELECT USERNAME, ACCOUNT_STATUS FROM DBA_USERS WHERE USERNAME = 'VICENT';
```

Possibles estats dels comptes d'usuaris:

Obert

Bloquejat

Expirat

Expirat i en període de gràcia

Expirat i bloquejat

..... (i més) ...

Quan un compte expira (per ell mateix), se li concedeix un període de gràcia perquè pugui canviar el passw ell/a mateix

dba_users





Usuaris en ORACLE

PASSWORD EXPIRED ==>

Des de Sql Developer

Conectar
Desconectar
Volver a Conectar
Restablecer Contraseña...

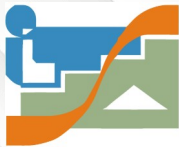
Des de sql*plus

```
C:\Users\admin>sqlplus prova1/123456@localhost/santvicent
SQL*Plus: Release 21.0.0.0.0 - Production on S8b Abr 27
Version 21.3.0.0.0

Copyright (c) 1982, 2021, Oracle. All rights reserved.

ERROR:
ORA-28001: the password has expired

Cambiando la contraseña para prova1
Contraseña nueva:
```



Usuaris en ORACLE

DD en Usuaris



```
describe dba_users
```

```
select * from dba_users;          *vore usuaris
```

```
select user from dual;
```

```
select username, password, default_tablespace, created from dba_users;
```

La vista v\$pwfile_users ens permet examinar als usuaris administratius.

Prova-ho en SQL Developer
Observa com funciona

dba_users
Dual
v\$pwfile_users



Permisos / privilegis

- ✓ Permisos de sistema
- ✓ Permisos sobre objectes

Concedir permisos / privilegis

(de sistema) `GRANT privilegi TO user [WITH ADMIN OPTION];` **en oracle**

(sobre objectes) `GRANT privilegi ON prop.objecte TO user [WITH GRANT OPTION];`

Revocar permisos

`REVOKE privilegi FROM user;` **en oracle**

`REVOKE privilegi ON prop.objecte FROM user;`

`[WITH ADMIN OPTION]` i `[WITH GRANT OPTION]`

vol dir que el permís que rep l'usuari, eixe usuari pot concedir-lo a un altre usuari

El `[WITH ADMIN OPTION]` sols s'utilitza per a privilegis de sistema (no de objectes)



PRIVILEGES en
ORACLE

Permisos en ORACLE

- ✓ Permisos de sistema
- ✓ Permisos sobre objectes

Permisos de sistema

```
CREATE SESSION (connectar)
CREATE TABLE, CREATE ANY TABLE
= amb ALTER... DROP...
CREATE USER DROP USER
SELECT ANY TABLE, UPDATE ANY TABLE,
INSERT ANY TABLE .....
CREATE TABLESPACES
ALL PRIVILEGES..... més
```

```
[usuario | rol | PUBLIC]
[WITH ADMIN OPTION]
```

Exemple:

```
grant create table, create view
to joan, anna;
```

```
GRANT privilegio1 [,privilegio2[,
...]] TO [usuario | rol | PUBLIC]
[WITH ADMIN OPTION];
```

```
REVOKE privilegio1 [,privilegio2 [,
...]] FROM usuario;
```



PRIVILEGES en
ORACLE

Permisos en ORACLE

- ✓ Permisos de sistema
- ✓ Permisos sobre objectes

Exemple:

```
create user joan identified by 1234 quota unlimited on users;  
grant connect to joan;  
grant create table to joan;
```

- Joan pot crear taules en el seu SCHEMA
- A més a més, joan podrà inserir, modificar, esborrar files de dades de les seues taules.
- A més a més, joan podrà modificar la estructura de les seues taules
- Joan també pot esborrar la taula (drop)

Però Joan no pot ..crear vistes,(ni modificar ni esborrar),
Ni crear index, ni roles, ni usuaris, etc...

Quan es
crea un usuari,
al mateix temps
es crea un
SCHEMA amb el
seu nom



Permisos en ORACLE

- ✓ Permisos de sistema
- ✓ Permisos sobre objectes

Permisos sobre objectes

SELECT
INSERT
UPDATE
DELETE
EXECUTE
INDEX
ALL.....[més](#)
[WITH GRANT OPTION]

Exemple

```
SQL> GRANT SELECT,INSERT,UPDATE,DELETE  
ON venta TO mgarcia WITH GRANT OPTION;
```

```
GRANT {privilegio [(listaColumnas)] [,privilegio  
[(listaColumnas)] [,...]] |  
ALL [PRIVILEGES]} ON [esquema.]objeto  
TO {usuario | rol | PUBLIC} [, {usuario | rol |  
PUBLIC} [,...]] [WITH GRANT OPTION]
```

Igual que els usuaris, els permisos poden ser comuns i locals, segons ens connectem i ho indiquem amb CONTAINER=ALL, o CONTAINER=CURRENT (<-per defecte)



Permisos en ORACLE

- ✓ Permisos de sistema
- ✓ Permisos sobre objectes

Exemple (estem connectats com joan)

```
create table exemple ( id number primary key, nom varchar2(40) );
```

La taula exemple pertany a joan

I joan li pot donar permís a pere per vore, inserir, modificar o eborrar files.

```
grant select,insert,update,delete on exemple to pere;
```

O si estem connectats com System o SYS podria fer:

```
grant select,insert,update,delete on joan.exemple to pere;
```

****Ara connectem amb l'usuari pere i provem a actualitzar o esborrar registres**

Prova-ho en sql-developer
Observa com funciona



Permisos en ORACLE

- ✓ Permisos de sistema
- ✓ Permisos sobre objectes

DD en Permisos

`select * from dba_sys_privs where grantee = 'ANNA';` *vore privilegis d'un usuari

`select * FROM USER_SYS_PRIVS;` *de sistema
`SELECT * FROM USER_TAB_PRIVS;` *sobre objectes

Vore privilegis de l'usuari concret 'ANNA'

`select grantee, privilege from dba_sys_privs where GRANTEE='ANNA';`

Vore privilegis de l'usuari connectat:

`select * from session_privs;`

permisos



dba_sys_privs
user_sys_privs
user_tab_privs



Rols

Crear rol

```
CREATE ROLE nom_rol ; en oracle
```

Esborrar rol

```
DROP ROLE nom_rol ;
```

Activar / Desactivar rols

```
SET ROLE {nom_rol} | ALL [EXCEPT {rol}] | NONE ;
```

Consultar Rols definit (DD)

```
select role from dba_roles;
```



dba_roles



Rols

Assignar privilegis a un rol

```
GRANT privilegis [ON obj] TO nom_rol ;    en oracle  
[WITH GRANT OPTION]
```

Revocar privilegis a un rol

```
REVOKE privilegis FROM nom_rol;
```

Assignar un rol a un usuari o a altre rol

```
GRANT nom_rol TO { usuari | nom_rol };
```

```
REVOKE DROP ANY TABLE FROM hr, oe;
```

Els usuaris hr i oe no poden borrar taules en esquemes que no siguen els seus.



Rols

DD en Rols

```
Select * from dba_roles;           *vore rols
SELECT * FROM USER_ROLE_PRIVS;    *rols assignats a usuari
      select * from dba_role_privs where granted_role='DBA'
```

```
select * from dba_sys_privs where grantee = 'RESOURCE'; *vore privilegis d'un rol
select * from role_sys_privs where role = 'RESOURCE';  *vore privilegis d'un rol
```

Consultar els rols assignats a un rol

```
select role, granted_role from role_role_privs;
```



```
dba_roles
dba_role_privs
dba_sys_privs
role_sys_privs
role_role_privs
```



Rols en ORACLE

Dins del CDB hi han dos tipus de rols

- Rols locals (als respectius PDBs)

```
SQL> create rol <nomrol> [CONTAINER=CURRENT];
```

- Rols comuns (a totes les BD's)

```
create rol C##<nomrol> CONTAINER=ALL;  
grant <priv> to C##<nomrol> CONTAINER=ALL;  
revoke <priv> from C##<nomrol>;  
grant <nomrol> to [usuario | nomrol2];  
revoke <nomrol> from [usuario | nomrol2];  
drop role <nomrol>;
```

DD:

```
select * from dba_roles ;
```

Use the DROP ROLE statement to remove a role from the database. When you drop a role, Oracle Database revokes it from all users and roles to whom it has been granted and removes it from the database. User sessions in which the role is already enabled are not affected



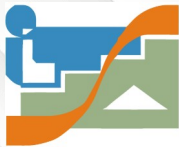
Rols en ORACLE

VISTA

DBA_ROLES
DBA_ROLE_PRIVS
DBA_SYS_PRIVS
DBA_TAB_PRIVS
ROLE_SYS_PRIVS
SESSION_ROLES

Que conté

Mostra tots les rols de la base de dades
Role assignats als usuaris o a altres rols
Privilegis de sistema assignats a usuaris i rols
Privilegis de objecte concedits a usuaris i rols
Privilegis de sistema assignats a rols
Rols en actiu per a l'usuari actual



Perfils en ORACLE

Quan es crea un usuari, se li assigna un perfil per defecte -> DEFAULT

```
SQL> select * from dba_profiles where profile='DEFAULT';
```

```
CREATE PROFILE nom_perfil LIMIT { } ... ;
```

```
ALTER USER usuari PROFILE nom_perfil;
```

```
ALTER USER usuari PROFILE DEFAULT;
```

```
ALTER PROFILE nom_perfil LIMIT {parametro [valor |UNLIMITED] } ;
```

```
DROP PROFILE nom_perfil [CASCADE];
```

Specify CASCADE to deassign the profile from any users to whom it is assigned. Oracle Database automatically assigns the DEFAULT profile to such users. You must specify this clause to drop a profile that is currently assigned to users



Perfils en ORACLE

Exemple creació d'un nou perfil

```
SQL> CREATE PROFILE perf-adm LIMIT  
      SESSIONS_PER_USER 5  
      CONNECT_TIME 120  
      IDLE_TIME 30  
      FAILED_LOGIN_ATTEMPTS 4  
      PASSWORD_LIFE_TIME 165;
```

Els paràmetres no definits en un perfil, agafaran el valor definit en el perfil DEFAULT

```
SQL> ALTER PROFILE perf-adm LIMIT PASSWORD_LOCK_TIME 5;
```

```
SQL> ALTER USER pere PROFILE perf-adm;  *assignació de perfil a usuari
```

Prova-ho en SQL Developer
Observa com funciona



Perfils en ORACLE

Paràmetres de limitacions de recursos

Oracle no porta els límits dels recursos activats per defecte.

Hem de activar-los amb: **ALTER SYSTEM SET RESOURCE_LIMIT=TRUE;**

SESSIONS PER USER	(per defecte unlimited en el perfil DEFAULT)
CONNECT_TIME	(per defecte unlimited , en minuts en el perfil DEFAULT)
IDLE_TIME	(per defecte unlimited , en minuts en el perfil DEFAULT)
CPU_PER_SESSION	, en centèsimes de segon en el perfil DEFAULT)
CPU_PER_CALL	, en centèsimes de segon en el perfil DEFAULT)
LOGICAL_READS_PER_SESSION	, en blocs en el perfil DEFAULT)
LOGICAL_READS_PER_CALL	, en blocs en el perfil DEFAULT)
PRIVATE_SGA	
COMPOSITE_LIMIT	

```
SQL> select * from dba_profiles where resource_type='KERNEL' ;
```

dba_profiles



Perfils en ORACLE

Paràmetres de limitacions de contrasenyes

Sols tenen efecte sobre usuaris validats per el sistema gestor (els externs no estan afectats)

FAILED_LOGIN_ATTEMPTS	(per defecte 10 en el perfil DEFAULT)
PASSWORD_LIFE_TIME	(per defecte 180 dies en el perfil DEFAULT)
PASSWORD_REUSE_TIME	(per defecte unlimited en el perfil DEFAULT)
PASSWORD_REUSE_MAX	(per defecte unlimited en el perfil DEFAULT)
PASSWORD_LOCK_TIME	(per defecte 1 dia en el perfil DEFAULT)
PASSWORD_GRACE_TIME	(per defecte 7 dies en el perfil DEFAULT)
PASSWORD_VERIFY_FUNCTION	(per defecte NULL en el perfil DEFAULT)

```
SQL> select distinct profile from dba_profiles;  
SQL> select * from dba_profiles order by profile;  
SQL> select * from dba_profiles where resource_type='PASSWORD';  
SQL> select * from dba_users where username='USU1';
```



dba_users
dba_profiles



Perfils en ORACLE

Paràmetres de limitacions de contrasenyes. *Exemple*

Abans (paràmetre per defecte)

Des de l'usuari vicent

```
ALTER USER vicent IDENTIFIED BY novapass;    (per defecte, es pot fer )
```

PASSWORD_VERIFY_FUNCTION (per defecte NULL en el perfil DEFAULT)

```
ALTER PROFILE DEFAULT LIMIT PASSWORD_VERIFY_FUNCTION "ORA12C_VERIFY_FUNCTION";
```

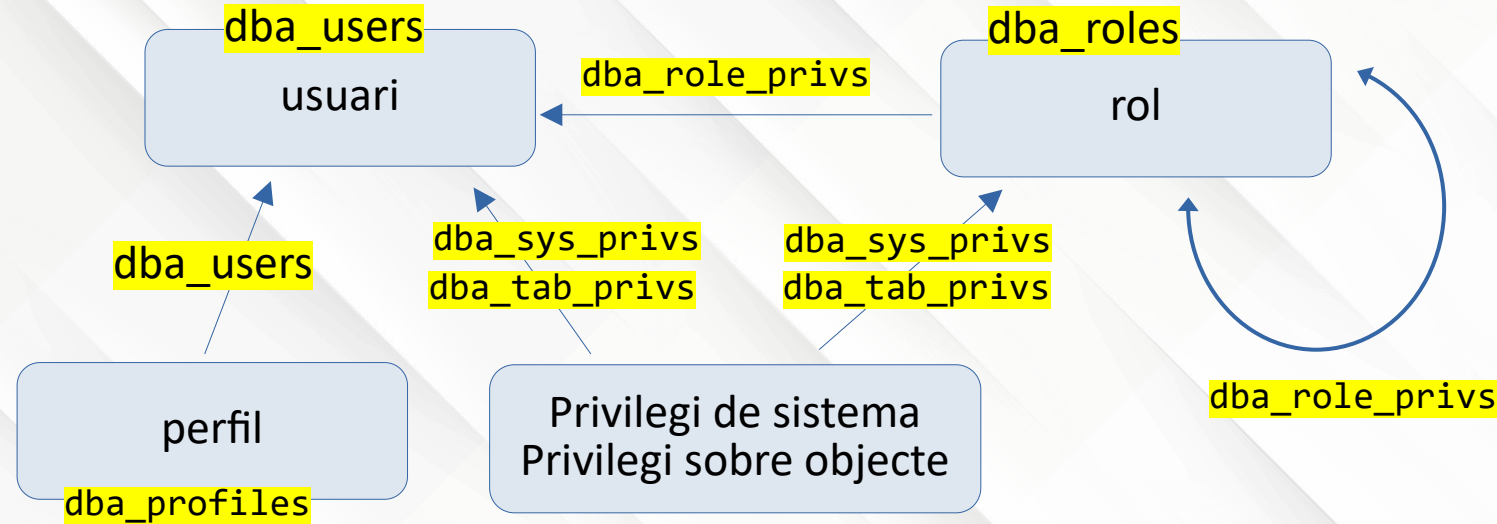
Després

Des de l'usuari vicent

```
ALTER USER vicent IDENTIFIED BY novapass;    <= JA NO DEIXA
```

```
ALTER USER vicent IDENTIFIED BY novapass REPLACE passvella;
```

<= deixa amb el REPLACE i pass complexa (may+min+num+sim)



resource_type='PASSWORD'
resource_type='KERNEL'

Un usuari sols pot tindre un perfil
Un usuari pot tindre molts privilegis
Un usuari pot tindre molts rols



Esquemes externs

És més potent assignar permisos sobre vistes (usuaris o rols sobre vistes)
I limitar(canviar) més tard les vistes

Quan necessitem crear una BBDD, el més habitual serà

- Crear un usuari propietari
- Crear un usuari per cada aplicació que haja d'accedir a les dades (convidats)

**No s'han de confondre usuaris de bases de dades amb
usuaris d'aplicació**

Si una organització té 100 empleats, no és necessari crear 100 usuaris de base de dades, sinó un usuari que usará l'aplicació per a connectar-se a la base de dades.



Esquemes externs

Usarem vistes i sinònims.

Avantatges

- Seguretat
- Facilitat d'us
- Homogeneïtat

```
CREATE VIEW vista_dept_201
AS (SELECT emp_id,name,department,hire_date)
FROM empleats
WHERE department = 201;
```

```
CREATE [OR REPLACE] SYNONYM nom_sinonim FOR esquema.vista ;
```

dba_views



Resum Diccionari de Dades (DD)



Usuaris **xxx_users**
Permisos **xxx_sys_privs**
Permisos **xxx_tab_privs**

Rols **xxx_roles**
Rols a usu **xxx_role_privs**

Perfiles **xxx_profiles**

Vistes **xxx_views**

Rols de la sessió actual
session_roles

Vista auxiliar **dual**
dictionary

v\$database
i database_properties
v\$pdbs = show pdbs
v\$parameter
v\$instance
v\$session
v\$sga = show sga

xxx_tablespace
o **v\$tablespace**
xxx_data_files
o **v\$datafile**
xxx_free_space

v\$log
v\$logfile

tabs = xxx_tables
xxx_tab_columns

On
xxx =

dba
all
user
role