

Capitolul 6

Gestiune utilizatori, profile, privilegii, roluri

Setari pentru useri

- ◆ Mecanismul de autentificare
- ◆ Cota pe diverse tablespace-uri
- ◆ Tablespace implicit (default)
- ◆ Tablespace temporar
- ◆ Blocare cont
- ◆ Limitari de resurse (profiluri)
- ◆ Privilegii user
- ◆ Roluri

Mecanismul de autentificare

- ◆ Autentificarea userului se poate face in mai multe feluri:
 1. Prin sistemul de operare (operating system authentication) – Oracle foloseste informatiile despre user aflate in sistemul de operare si il autentifica, nemaifiind necesara introducerea unui username si a unei parole.
 2. Prin retea (network authentication) – folosind servicii de autentificare third-party, ca de exemplu: Distributed Computing Environment (DCE), Kerberos, public key infrastructure, the Remote Authentication Dial-In User Service (RADIUS), SSL, etc
 3. De catre serverul de BD (database authentication) – pe baza unui username si a unei parole (cum lucrati de obicei la laborator).

Cota pe diverse tablespace-uri

- ◆ La crearea unui nou user se poate specifica spatiul pe care acel user il poate 'consuma' din diversele tablespace-uri care exista la acel moment in sistem.
- ◆ Nu se pot asocia cote pe tablespace-urile temporare
- ◆ Implicit userii nu au cote asociate cu nici un tablespace

Cota - cont

- ◆ Assignarea unei cote pentru un user intr-un tablespace are urmatoarele efecte:
 - Userii care au privilegiul de a crea obiecte pot crea acele obiecte in tablespace-ul respectiv.
 - Oracle limiteaza spatiul pe care acele obiecte il pot ocupa in tablespace-ul specificat la cat spune cota alocata.
- ◆ Se poate inhiba pentru un user posibilitatea de creare de noi obiecte intr-un anumit tablespace prin setarea unei cote egale cu 0

Cota - cont

- ◆ Cand cota unui user este modificata la o valoare mai mica decat spatiul ocupat la acel moment de acel user in acel tablespace (inclusiv la setarea unei cote egala cu 0) obiectele existente nu se sterg dar:
 - nu se mai pot crea noi obiecte
 - Obiectele existente nu mai pot creste in dimensiune (dar pot scadea)

Tablespace implicit (default)

- ◆ Orice user are un tablespace implicit (default).
- ◆ Acest tablespace definește locația unde sunt create obiectele (segmentele) userului în absența specificării tablespace-ului în momentul creării.
- ◆ În Oracle valoarea de default este tablespace-ul SYSTEM, ceea ce nu este foarte bine în cazul în care userul creează noi obiecte.
- ◆ Este bine ca să se creeze un alt tablespace și userii uzuali să-l aibă pe acesta ca default.
- ◆ Userii de sistem (SYS, SYSTEM) trebuie însă să rămână cu defaultul SYSTEM.
- ◆ Acest tablespace se poate schimba și după crearea userului, cu ALTER USER

Tablespace temporar

- ◆ În cazul în care sunt folosite segmente temporare (de exemplu sunt executate cereri care implică sortări de date voluminoase), acestea sunt stocate:
 - ♦ În tablespace-ul implicit (default) dacă nu s-a specificat un tablespace temporar la crearea userului
 - ♦ În tablespace-ul temporar dacă acesta a fost specificat
- ◆ Acest tablespace se poate specifica și ulterior, prin ALTER USER

Blocare cont

- ◆ Un cont poate fi configurat să se blocheze după un anumit număr de încercări de intrare fără succes.
- ◆ Contul se poate debloca după un anumit interval de timp, specificat, sau de către administratorul bazei de date.
- ◆ De asemenea, parola de la creare se poate seta ca expirată, forțând astfel schimbarea parolei (de către user sau de către administratorul bazei de date) înainte de a putea intra în sistem.

Obiectele unui user

- ◆ Ele formează 'schema' aceluși user
- ◆ Pot fi:
 - ♦ Tabele (cu declanșatori și constrângeri asociate)
 - ♦ Indeksi
 - ♦ Vederi
 - ♦ Secvențe
 - ♦ Subprograme stocate
 - ♦ Sinonime
 - ♦ Tipuri definite de user
 - ♦ Legături (database links – prin ele se pot accesa obiecte din alte baze de date)

Crearea unui nou user

- ◆ La crearea unui nou user trebuie mai întâi stabilite următoarele:
 - ♦ Numele și parola și metoda de autentificare pentru acel user
 - ♦ Tablespace-urile care pot fi utilizate de către acesta
 - ♦ Se stabilește cota alocată userului pentru fiecare în parte
 - ♦ Se stabilește tablespace-ul implicit și cel temporar
- ◆ Se emite comanda CREATE USER care folosește informațiile de mai sus
- ◆ Se adaugă apoi privilegii și roluri pentru user.

Sintaxa

```
CREATE USER username
IDENTIFIED {BY password
            | EXTERNALLY
            | GLOBALLY AS 'external_name'}
[ DEFAULT TABLESPACE tablespace ]
[ TEMPORARY TABLESPACE tablespace ]
[ QUOTA int {K | M} ON tablespace ]
[ QUOTA UNLIMITED ON tablespace ]
[ PROFILE { profile_name | DEFAULT } ]
[ PASSWORD EXPIRE ]
[ ACCOUNT {LOCK|UNLOCK} ]
```

Detalii

```
IDENTIFIED {BY password
            | EXTERNALLY
            | GLOBALLY AS 'external_name'}
```

- ◆ Aceasta clauza spune modul de autentificare pentru acest user:
- ◆ BY password arata ca este un user local care trebuie sa specifice username si parola la login,
- ◆ EXTERNALLY indica un user extern, autentificat fie prin sistemul de operare fie prin servicii third party
- ◆ GLOBALY arata ca este un user global, autentificat prin 'directory services'

Detalii

```
[ DEFAULT TABLESPACE tablespace ]
```

- ◆ Aceasta clauza specifica tablespace-ul default (implicit)

```
[ TEMPORARY TABLESPACE tablespace ]
```

- ◆ Aceasta clauza specifica tablespace-ul pentru segmente temporare

```
[ QUOTA int {K | M} ON tablespace ]
```

- ◆ Aceasta clauza specifica valoarea cotei pe un anumit tablespace in bytes / KB / MB.

```
[ QUOTA UNLIMITED ON tablespace ]
```

- ◆ Aceasta clauza specifica faptul ca nu este fixata o limita superioara pentru cota pe acel tablespace (bineinteles segmentele userului nu pot depasi spatiul existent acolo)

Detalii

```
[ PROFILE { profile_name | DEFAULT } ]
```

- ◆ Specifica profilul asociat cu acel user, acesta aratand limitarile privind resursele pe care le poate consuma userul. Daca nu se specifica, va fi asociat un profil implicit.

```
[ PASSWORD EXPIRE ]
```

- ◆ Specifica faptul ca parola este 'pre-expirata', decu DBA sau userul trebuie sa o schimbe inainte de a se putea intra in acel cont

```
[ ACCOUNT {LOCK|UNLOCK} ]
```

- ◆ Specifica faptul ca acel cont este blocat (LOCK), deci necesita deblocare inainte de a fi utilizat. Implicit contul este deblocat (UNLOCK) si se poate lucra.

Exemplu

- ◆ User autentificat prin parola:

```
CREATE USER mimi
IDENTIFIED BY EC004abc
DEFAULT TABLESPACE date
QUOTA 100M ON test
QUOTA 500K ON date
TEMPORARY TABLESPACE temp
PROFILE clerk;
```

- ◆ Se adauga si niste privilegii:

```
GRANT create session TO mimi;
```

Modificare date user

- ◆ Datele privind autentificarea userului:

```
ALTER USER username
IDENTIFIED {BY password
            | EXTERNALLY
            | GLOBALLY AS 'external_name'}
[ PASSWORD EXPIRE ]
[ ACCOUNT {LOCK|UNLOCK} ]
```

- ◆ In momentul blocarii unui cont (LOCK), daca userul e logat la acel moment nu va fi afectat. Modificarile date de comanda de mai sus sunt valabile incepand cu urmatoarea sesiune de lucru.

Modificare date user - cont

- ◆ Datele privind tablespace si cote:

```
ALTER USER username
[ DEFAULT TABLESPACE tablespace ]
[ TEMPORARY TABLESPACE tablespace ]
[ QUOTA int {K | M} ON tablespace ]
[ QUOTA UNLIMITED ON tablespace ]
```

- ◆ La trecerea pe 0 a cotei nu se mai pot crea obiecte si cele existente nu mai pot creste.

Exemplu:

```
ALTER USER mimi
QUOTA 0 ON date;
```

Stergere user

- ◆ Stergerea unui user se face cu comanda DROP USER:

DROP USER nume [CASCADE]

- ◆ Optiunea CASCADE sterge intai toate obiectele din schema userului respectiv (altfel se obtine un mesaj de eroare).
- ◆ Fara CASCADE se pot sterge useri care nu detin nici un obiect in schema proprie.

Vederi care se pot utiliza

View	Description
DBA_USERS	DBA view describes all users of the database.
ALL_USERS	ALL view lists users visible to the current user, but does not describe them.
USER_USERS	USER view describes only the current user.
DBA_TS_QUOTAS, USER_TS_QUOTAS	Describes tablespace quotas for users.
USER_PASSWORD_LIMITS	Describes the password profile parameters that are assigned to the user.
USER_RESOURCE_LIMITS	Displays the resource limits for the current user.
DBA_PROFILES	Displays all profiles and their limits.
RESOURCE_COST	Lists the cost for each resource.
V\$SESSION	Lists session information for each current session. Includes user name.
V\$SESSTAT	Lists user session statistics.
V\$STATNAME	Displays decoded statistic names for the statistics shown in the V\$SESSTAT view.
PROXY_USERS	Describes users who can assume the identity of other users.

Exemplu

```
SELECT TABLESPACE_NAME, BLOCKS, MAX_BLOCKS,
       BYTES, MAX_BYTES
FROM DBA_TS_QUOTAS
WHERE USERNAME = 'SCOTT';
```

- ◆ Se obtine un rezultat care contine date despre cota userului:

TABLESPACE_NAME	BLOCKS	MAX_BLOCKS	BYTES	MAX_BYTES
DATE	10	-1	20480	-1

- ◆ Valoarea -1 reprezinta cota nelimitata. Restul valorilor reprezinta spatiul ocupat la acel moment.

Alt exemplu

```
SELECT USERNAME, ACCOUNT_STATUS,
       TEMPORARY_TABLESPACE
FROM DBA_USERS
```

- ◆ Se obtine o lista cu starea fiecarui cont (si alte date):

USERNAME	ACCOUNT_STATUS	TEMPORARY_TABLESPACE
SYS	OPEN	TEMP
SYSTEM	OPEN	TEMP
DBSNMP	OPEN	TEMP
SCOTT	OPEN	TEMP

PROFIL

- ◆ Profilurile sunt o modalitate prin care se pot limita resursele care pot fi utilizate de un utilizator.
- ◆ Un profil se creaza cu CREATE PROFILE si se asigneaza userului la creare sau ulterior prin comanda ALTER USER.
- ◆ Exista un profil DEFAULT care se asociaza implicit la userii pentru care la creare nu s-a specificat un profil.

Ce se limiteaza?

- ◆ Resurse ale sistemului (pentru ca aceste limitari de sistem sa fie active trebuie ca parametrul de initializare RESOURCE_LIMIT sa fie setat pe True):
 - ◆ Numarul maxim de sesiuni concurente pentru user (sessions_per_user)
 - ◆ Timp CPU per sesiune (cpu_per_session) – masurat in sute de secunde.
 - ◆ Timp CPU per operatie (cpu_per_call) – masurat in sute de secunde. O operatie este un ciclu parse, execute, fetch.

Ce se limiteaza?

◆ Resurse ale sistemului - cont:

- Timpul maxim de conectare (connect_time) – masurat in minute.
Sesiunile userului sunt inchise de Oracle dupa expirarea acestui timp.
- Timp maxim de asteptare (idle_time) – masurat in minute.
Sesiunile vor fi inchise de Oracle dupa expirarea perioadei specificate daca in sesiunea respectiva nu s-a facut nimic (e 'idle'). Atentie: cererile a caror executie este lunga nu intra in aceasta categorie!

Ce se limiteaza?

◆ Resurse ale sistemului - cont:

- Numar maxim de blocuri citite per sesiune. Este vorba aici de numarul de blocuri citite de pe disc **sau** din memorie. Acest parametru este gandit pentru a limita cererile care fac citiri intensive. (logical_reads_per_session)
- Numarul maxim de blocuri citite per operatie (call) (logical_reads_per_call)
- Dimensiunea maxima de memorie ocupata (private_sga)

Ce se limiteaza?

◆ Resurse legate de parola:

- Numarul maxim de incercari eronate de login (failed_login_attempts)
- Timpul maxim cat parola este valida (password_life_time)
- Numarul minim de parole diferite utilizate pana cand o parola poate fi reutilizata (password_reuse_max)
- Numarul minim de zile dupa care o parola poate fi utilizata (password_reuse_time)

Ce se limiteaza?

◆ Resurse legate de parola - cont:

- Numarul de zile cat contul este blocat dupa incercari repetate de login (password_lock_time)
- Daca parola este sau nu verificata ca lungime, continut si complexitate (password_verify_function)
- ◆ Lista de mai sus nu este exhaustiva. Un tabel cu limitarile care se pot folosi in Oracle 11.1 se gaseste in pagina:
<http://www.psoug.org/reference/profiles.html>
- ◆ Am dat numele parametrilor (in paranteza) pentru ca fiecare in parte se poate modifica ulterior prin comenzi ALTER PROFILE.

Ce se intampla?

◆ Daca este atinsa o limita la nivel de sesiune atunci:

- Fie se afiseaza un mesaj de eroare (de exemplu cand se incearca deschiderea unei noi sesiuni si se depaseste sessions_per_user)
- Fie Oracle deconecteaza userul (sesiunea), de exemplu cand s-a atins durata ei maxima.

Ce se intampla?

◆ Daca este atinsa o limita la nivel de operatie (call) atunci:

- Procesarea cererii curente este oprita
- Cererea curenta este revocata (rollback)
- Efectul cererilor anterioare persista
- Userul ramane conectat.
- ◆ In continuare vom discuta despre cum se creaza si se modifica un profil si despre cum este asignat un profil la un user

Creare profil

```
CREATE PROFILE profile
LIMIT
[SESSIONS_PER_USER {integer | UNLIMITED | DEFAULT}]
[CPU_PER_SESSION {integer | UNLIMITED | DEFAULT}]
[CPU_PER_CALL {integer | UNLIMITED | DEFAULT}]
[CONNECT_TIME {integer | UNLIMITED | DEFAULT}]
[IDLE_TIME {integer | UNLIMITED | DEFAULT}]
[LOGICAL_READS_PER_SESSION {integer | UNLIMITED |
DEFAULT}]
[LOGICAL_READS_PER_CALL {integer | UNLIMITED | DEFAULT}]
[COMPOSITE_LIMIT {integer | UNLIMITED | DEFAULT}]
[PRIVATE_SGA {integer [K|M] | UNLIMITED | DEFAULT}]
```

- ◆ Pentru a executa aceasta operatie trebuie privilegiul CREATE PROFILE.

Creare profil – cont.

- ◆ UNLIMITED: arata faptul ca pentru acel profil resursa respectiva poate fi folosita in cota nelimitata
- ◆ DEFAULT: arata faptul ca resursa respectiva poate fi folosita limitat, valoarea fiind aceeaasi cu a profilului DEFAULT
- ◆ COMPOSITE_LIMIT limiteaza costul total al resurselor pentru o sesiune in unitati de servire. Oracle calculeaza acest cost ca o suma ponderata intre:
 - ◆ CPU_PER_SESSION
 - ◆ CONNECT_TIME
 - ◆ LOGICAL_READS_PER_SESSION
 - ◆ PRIVATE_SGA

Exemplu

```
create profile exemplu limit
sessions_per_user 1
idle_time 1
failed_login_attempts 3;
```

- ◆ Pentru a vizualiza restrictiile aferente profilului creat cu cererea de mai sus se poate executa cererea:

```
select resource_name, limit
from dba_profiles
where profile = 'EXAMPLU';
```

Exemplu - cont

- ◆ Rezultatul va contine lista urmatoare:

COMPOSITE_LIMIT	DEFAULT
SESSIONS_PER_USER	1
CPU_PER_SESSION	DEFAULT
CPU_PER_CALL	DEFAULT
LOGICAL_READS_PER_SESSION	DEFAULT
LOGICAL_READS_PER_CALL	DEFAULT
IDLE_TIME	1
CONNECT_TIME	DEFAULT
PRIVATE_SGA	DEFAULT
FAILED_LOGIN_ATTEMPTS	3
PASSWORD_LIFE_TIME	DEFAULT
PASSWORD_REUSE_TIME	DEFAULT
PASSWORD_REUSE_MAX	DEFAULT
PASSWORD_VERIFY_FUNCTION	DEFAULT
PASSWORD_LOCK_TIME	DEFAULT
PASSWORD_GRACE_TIME	DEFAULT

Asignarea unui profil

Asignarea unui profil la un user se poate face:

- ◆ La crearea userului (CREATE USER) exista clauza PROFILE care specifica un profil asociat acelui user (in lipsa se ia profilul DEFAULT).
- ◆ Ulterior se poate schimba profilul cu ALTER USER:

```
ALTER USER scott
PROFILE exemplu;
```

RESOURCE_LIMIT

- ◆ Asa cum s-a specificat, parametrul de initializare RESOURCE_LIMIT trebuie sa fie TRUE
- ◆ Pentru a vedea care este valoarea curenta a acestui parametru se poate folosi in SQL*Plus comanda SHOW PARAMETER:
- ◆ SQL> show parameter resource_limit
resource_limit boolean FALSE

RESOURCE_LIMIT - cont

- ◆ Putem schimba valoarea curenta cu ALTER SYSTEM:

```
ALTER SYSTEM
SET RESOURCE_LIMIT=TRUE
```

- ◆ Efectul acestei comenzi dureaza pana la o noua schimbare a valorii cu ALTER SYSTEM sau pana cand se opreste baza de date (la repornire va citi din nou valoarea din fisierul de parametri)

Modificare profil

- ◆ Modificarea unui profil se poate face cu ALTER PROFILE (similara cu CREATE):

```
ALTER PROFILE profile
LIMIT
[SESSIONS_PER_USER {integer | UNLIMITED | DEFAULT}]
[CPU_PER_SESSION {integer | UNLIMITED | DEFAULT}]
[CPU_PER_CALL {integer | UNLIMITED | DEFAULT}]
[CONNECT_TIME {integer | UNLIMITED | DEFAULT}]
[IDLE_TIME {integer | UNLIMITED | DEFAULT}]
[LOGICAL_READS_PER_SESSION {integer | UNLIMITED |
DEFAULT}]
[LOGICAL_READS_PER_CALL {integer | UNLIMITED |
DEFAULT}]
[COMPOSITE_LIMIT {integer | UNLIMITED | DEFAULT}]
[PRIVATE_SGA {integer [K|M] | UNLIMITED | DEFAULT}]
```

Modificare profil - cont

- ◆ Modificarile de profil nu afecteaza sesiunile curente ci doar pe cele deschise dupa modificare
- ◆ Pentru executia comenzii trebuie privilegiul ALTER PROFILE.

Stergerea unui profil

- ◆ Se face cu DROP PROFILE:
- ```
DROP PROFILE nume [CASCADE]
```
- ◆ Profilul DEFAULT nu se poate sterge
  - ◆ Stergerea unui profil nu afecteaza sesiunile curente
  - ◆ Optiunea CASCADE revoca acest profil de la userii care il au
  - ◆ Userii care au profilul sters vor trece automat pe profilul DEFAULT
  - ◆ Pentru executia operatiei trebuie privilegiul DROP PROFILE

## Vizualizari

- ◆ Pentru a vedea informatii despre stare cont, blocare, data expirarii parolei si alte limitari ale acestora se poate folosi vederea DBA\_USERS sau vederea DBA\_PROFILES:

```
SELECT username, password,
account_status, lock_date, expiry_date
FROM dba_users;
```

- ◆ Se va obtine un rezultat despre fiecare user, de tipul:

| USERNAME | PASSWORD        | ACCOUNT_ST | EXPIRY_DA |
|----------|-----------------|------------|-----------|
| SYS      | D4C5123456B4DC8 | OPEN       | 19-DEC-08 |

## Vizualizari - cont

- ◆ Pentru limitari asupra parolei, in cererile asupra lui DBA\_PROFILES se poate folosi in clauza WHERE conditia

```
WHERE resource_type= 'PASSWORD'
```

- ◆ Exemplu:

```
SELECT profile, resource_name, limit
FROM dba_profiles
WHERE resource_type= 'PASSWORD'
```

## PRIVILEGII

### ◆Exista doua tipuri de privilegii:

- ◆ Privilegii sistem: ele permit userilor sa execute operatii pe baza de date: creare, stergere, modificare pe tabele, vederi, segmente de rollback si proceduri
- ◆ Privilegii obiect: permit userilor sa efectueze anumite operatii pe un obiect: tabela, secventa, vedere, procedura, functie sau pachet

## Privilegii sistem

### ◆Exista un numar mare de privilegii sistem.

([http://www.psoug.org/reference/system\\_privs.html](http://www.psoug.org/reference/system_privs.html))

### ◆Privilegiile de sistem pot fi clasificate in:

- ◆ Privilegii pentru operatii care afecteaza intreg sistemul, ca de exemplu CREATE SESSION, CREATE TABLESPACE
- ◆ Privilegii care afecteaza obiectele din schema proprie, de ex. CREATE TABLE
- ◆ Privilegii care afecteaza obiectele din orice schema, de ex. CREATE ANY TABLE

## Privilegii sistem – cont.

- ◆In numele lor, particula ANY arata ca userul are acel privilegiu in orice schema.
- ◆Pentru a adauga privilegii la un user se foloseste cererea GRANT.
- ◆Pentru a inlatura privilegii de la un user se foloseste REVOKE.

## Exemple de privilegii - CREATE

- ◆ Create Any Index
- ◆ Create Any Indextype
- ◆ Create Any Materialized View
- ◆ Create Any Measure Folder
- ◆ Create Any Operator
- ◆ Create Any Outline
- ◆ Create Any Procedure
- ◆ Create Any Rule
- ◆ Create Any Rule Set
- ◆ Create Any Sequence
- ◆ Create Any SQL Profile
- ◆ Create Any Synonym
- ◆ Create Any Table
- ◆ Create Any Trigger
- ◆ Create Any Type
- ◆ Create Any View

◆ Cele mai multe dintre acestea au si varianta fara ANY.

## Observatii

- ◆Nu exista privilegiul CREATE INDEX. El este inclus in CREATE TABLE
- ◆Privilegiile CREATE TABLE, CREATE PROCEDURE si CREATE CLUSTER include si dreptul de a sterge aceste obiecte
- ◆Privilegiul UNLIMITEDTABLESPACE nu poate fi asignat unui rol ci doar userilor particulari.
- ◆Pentru trunchierea unei tabele este necesar privilegiul DROP ANY TABLE

## GRANT

### ◆ Sintaxa pentru comanda GRANT este:

```
GRANT { priv_sistem | rol },
 { priv_sistem | rol }...
TO { user | rol | PUBLIC },
 { user | rol | PUBLIC }...
[WITH ADMIN OPTION]
```

- ◆ Se asigneaza lista de privilegii si/sau roluri unui user sau unui rol
- ◆ In cazul in care se specifica PUBLIC privilegiile respective sunt asignat tuturor userilor.



## GRANT – cont.

- ◆ In cazul specificarii WITH ADMIN OPTION cel care primește privilegiul îl poate și el asigna mai departe, inclusiv cu ADMIN OPTION.
- ◆ Userii care au privilegiul GRANT ANY ROLE pot să asigneze orice rol în sistem.
- ◆ Cel care primește un privilegiu cu ADMIN OPTION îl poate de asemenea revoca de la orice user sau rol din sistem (nu numai de la cei cărora el l-a asignat).
- ◆ În general privilegiile SYSDBA și SYSOPER (s-a discutat despre ele anterior) trebuie asignate doar userilor de tip administrator pentru că ele dau acces la orice operație în baza de date (SYSDBA).

## Vizualizare privilegii

- ◆ Există vederile DBA\_SYS\_PRIVS și SESSION\_PRIVS care pot fi interogate pentru a vedea privilegiile asociate fiecărui user (DBA\_SYS\_PRIVS) și sesiunii curente.
- ◆ Privilegiile afișate provin atât din privilegii asignate individual cât și din privilegii asociate cu rolurile asignate userilor.

## REVOKE

- ◆ Sintaxa este:  

```
REVOKE { priv_sistem | rol },
 { priv_sistem | rol } ...
FROM { user | rol | PUBLIC },
 { user | rol | PUBLIC } ...
```
- ◆ Revoca acele privilegii care au fost asignate cu GRANT.
- ◆ Revocarea unor privilegii poate face ca anumite proceduri sau vederi care aveau nevoie de acel privilegiu să devină invalide.
- ◆ Revocarea unui privilegiu de la un user nu afectează userii cărora acesta le-a transmis privilegiul – deci REVOKE nu are efect în cascada.

## Privilegii obiect

- ◆ Sintaxa:  

```
GRANT {object_priv | ALL [PRIVILEGES]}
[(column [, column] ...)]
[, {object_priv | ALL [PRIVILEGES]}
[(column [, column] ...)]] ...
ON [schema.]object
TO {user | role | PUBLIC}
[, {user | role | PUBLIC}] ...
[WITH GRANT OPTION]
```

## Privilegii obiect

- ◆ Pe post de object\_priv poate fi:
- ◆ ALTER
- ◆ DELETE
- ◆ EXECUTE
- ◆ INDEX
- ◆ INSERT
- ◆ REFERENCES
- ◆ SELECT
- ◆ UPDATE

## Revocare privilegii obiect

- ◆ Sintaxa:  

```
REVOKE {object_priv | ALL [PRIVILEGES]}
[(column [, column] ...)]
[, {object_priv | ALL [PRIVILEGES]}
[(column [, column] ...)]] ...
ON [schema.]object
FROM {user | role | PUBLIC}
[, {user | role | PUBLIC}] ...
[CASCADE CONSTRAINTS]
```
- ◆ Ultima opțiune elimină constrangerile referențiale afectate.

## ROLURI

- ◆ Rolurile sunt colectii de privilegii (ca un cos de privilegii) care pot fi asignate si revocate impreuna.
- ◆ Un rol poate fi creat, i se pot asocia privilegii (umplem cosul) dupa care el se poate signa cu GRANT unui user sau unui alt rol, asa cum am vazut anterior.
- ◆ Exista o serie de roluri predefinite (CONNECT, RESOURCE, DBA, etc).
- ◆ Mai multe despre roluri: la laborator.

## Lecturi obligatorii

1. Oracle Database Security Guide – Cap 10:  
Administering User Privileges, Roles, and Profiles  
[http://download.oracle.com/docs/cd/B14117\\_01/network.101/b10773/admusers.htm](http://download.oracle.com/docs/cd/B14117_01/network.101/b10773/admusers.htm)

## Sfârșitul capitolului 6