

Beveiliging





**Cursus
Boek**

cursusmateriaal

- cursus 'Databanken 1', hoofdstuk 4: blz. 69-73
- Deze slides

Agenda

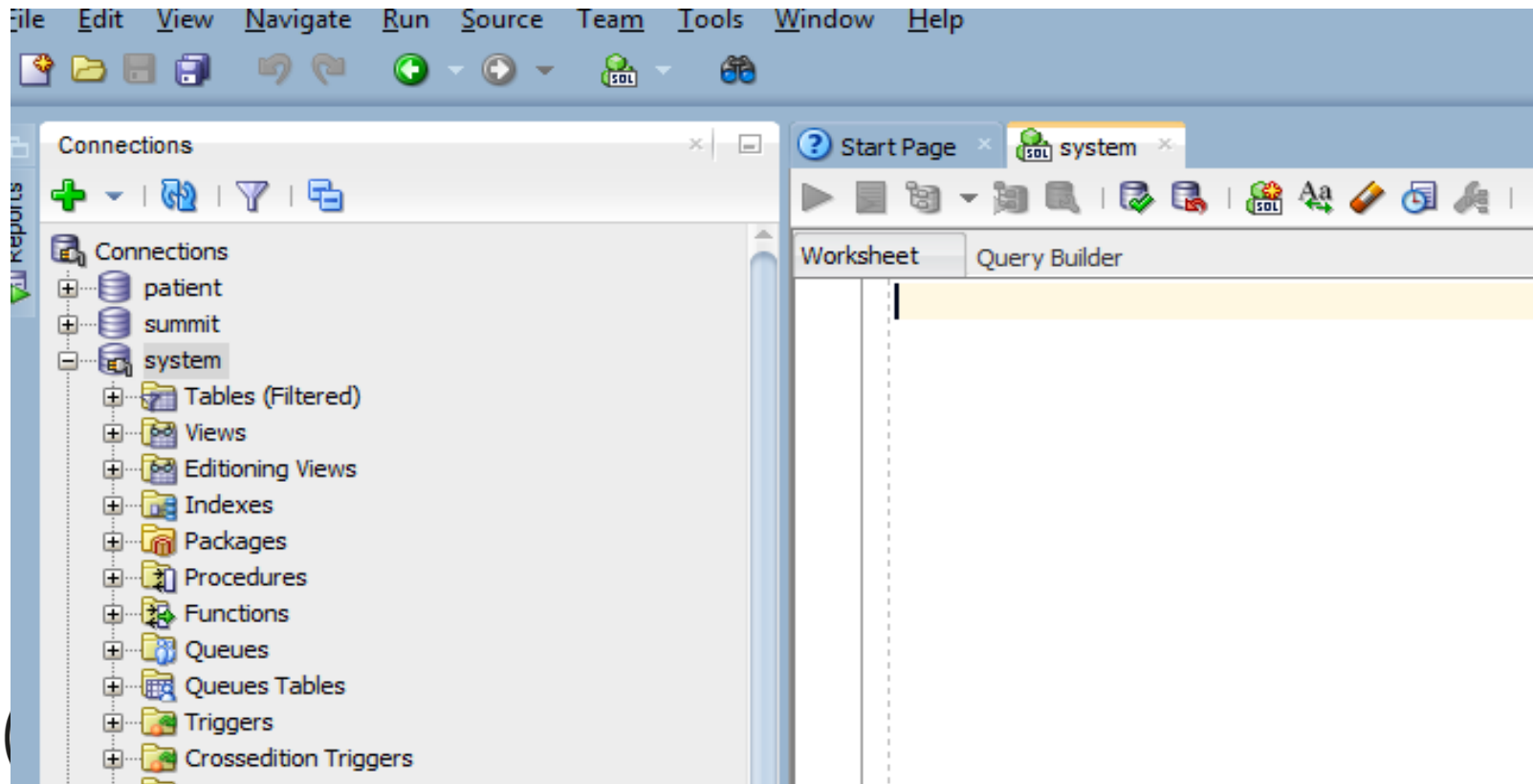


- Gebruikers
- Gegevensbibliotheek (data dictionary)
- Beheren van een gebruiker
- Systeem privileges
- Object privileges
- Rollen

Gebruikers

- Standaard heeft een ORACLE databank volgende gebruikers:
 - **PUBLIC:** eigenaar van alle objecten voor gemeenschappelijk gebruik
 - Je kan niet aanloggen als public
 - **SYS:** eigenaar van de gegevensbibliotheek (data dictionary)
 - **SYSTEM:** eigenaar van andere standaard objecten
 - Bij installatie van de databank gaf je SYS en SYSTEM een (zelfde) paswoord
- **gewone gebruikers** die met de CREATE USER instructie zullen aangemaakt worden.

System



De gegevensbibliotheek bestaat uit een reeks views. Om het de gebruiker gemakkelijk te maken zijn deze views onderverdeeld in 3 categorieën.

- ❑ views met voorvoegsel **USER_**

bevatten alle informatie over objecten waarvan een individuele gebruiker eigenaar is of privileges die hij kreeg.

- ❑ views met voorvoegsel **ALL_**

bevatten informatie over alle objecten waartoe de individuele gebruiker toegang heeft (inclusief zijn eigen objecten)

- ❑ views met voorvoegsel **DBA_**

bevatten informatie voor de database beheerder.

De DBA maakt gebruikers aan met de instructie:

```
CREATE USER username IDENTIFIED BY paswoord  
[DEFAULT TABLESPACE tablespace_name  
TEMPORARY TABLESPACE tablespace_name  
QUOTA n K/M ON tablespace_name] ;
```

- Username uniek binnen database
- Paswoord en username opgeslagen in data dictionary
- Username en paswoord zijn case sensitive vanaf versie 11g
- tablespace met beschikbare ruimte in kilobyte of megabyte.
Dit komt aan bod in het 2^e jaar.

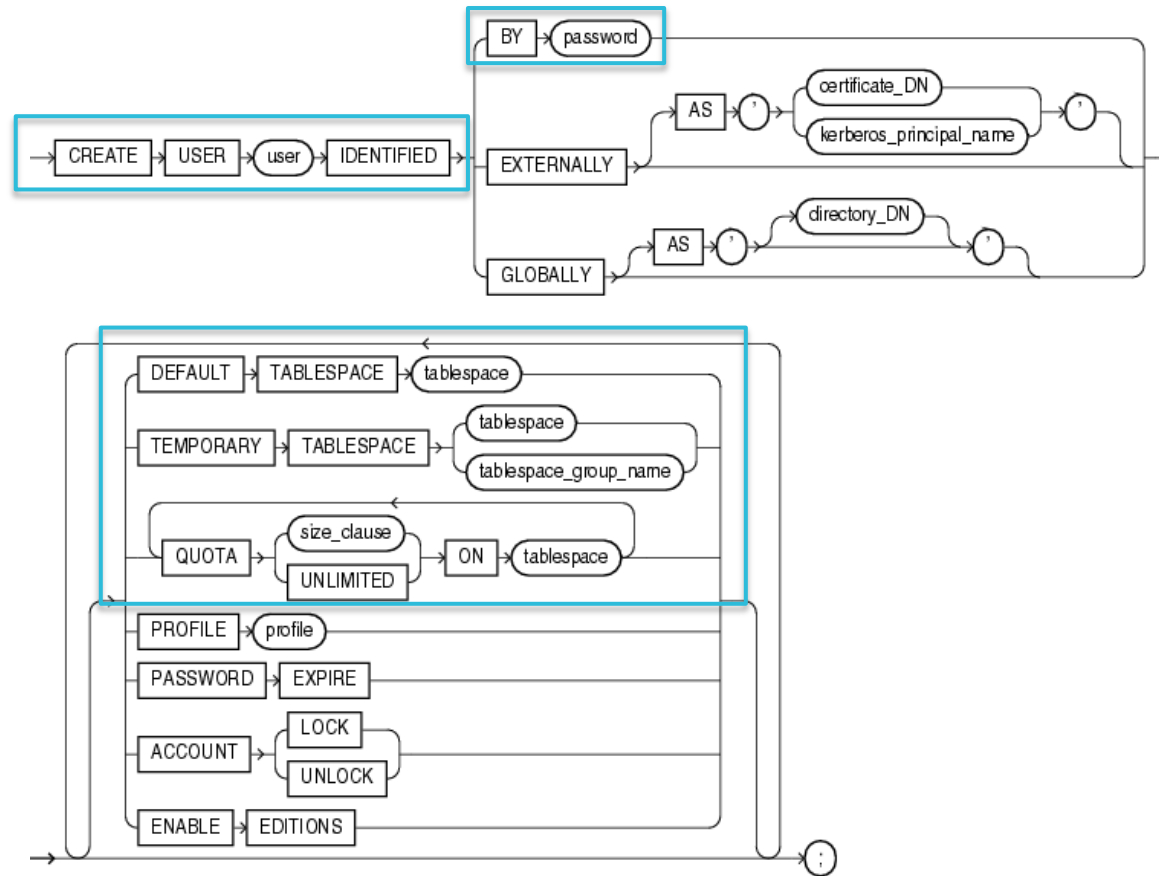
Aanmaken gebruiker

```
CREATE USER t191fi IDENTIFIED BY t191fi;
```



Definitie

SQL
Reference
p. 127




Aanmaken gebruiker



!! We gebruikten dit al uit het document:

Starten met SQL Developer

Bijgevoegde bestanden:  [Starten in SQL Developer.docx](#) |
kB)

```
CREATE USER theorie IDENTIFIED BY theorie  
DEFAULT TABLESPACE users  
QUOTA 2M ON users;
```

```
CREATE USER praktijk IDENTIFIED BY praktijk  
DEFAULT TABLESPACE users  
QUOTA 2M ON users;  
GRANT DBA TO theorie,praktijk;
```

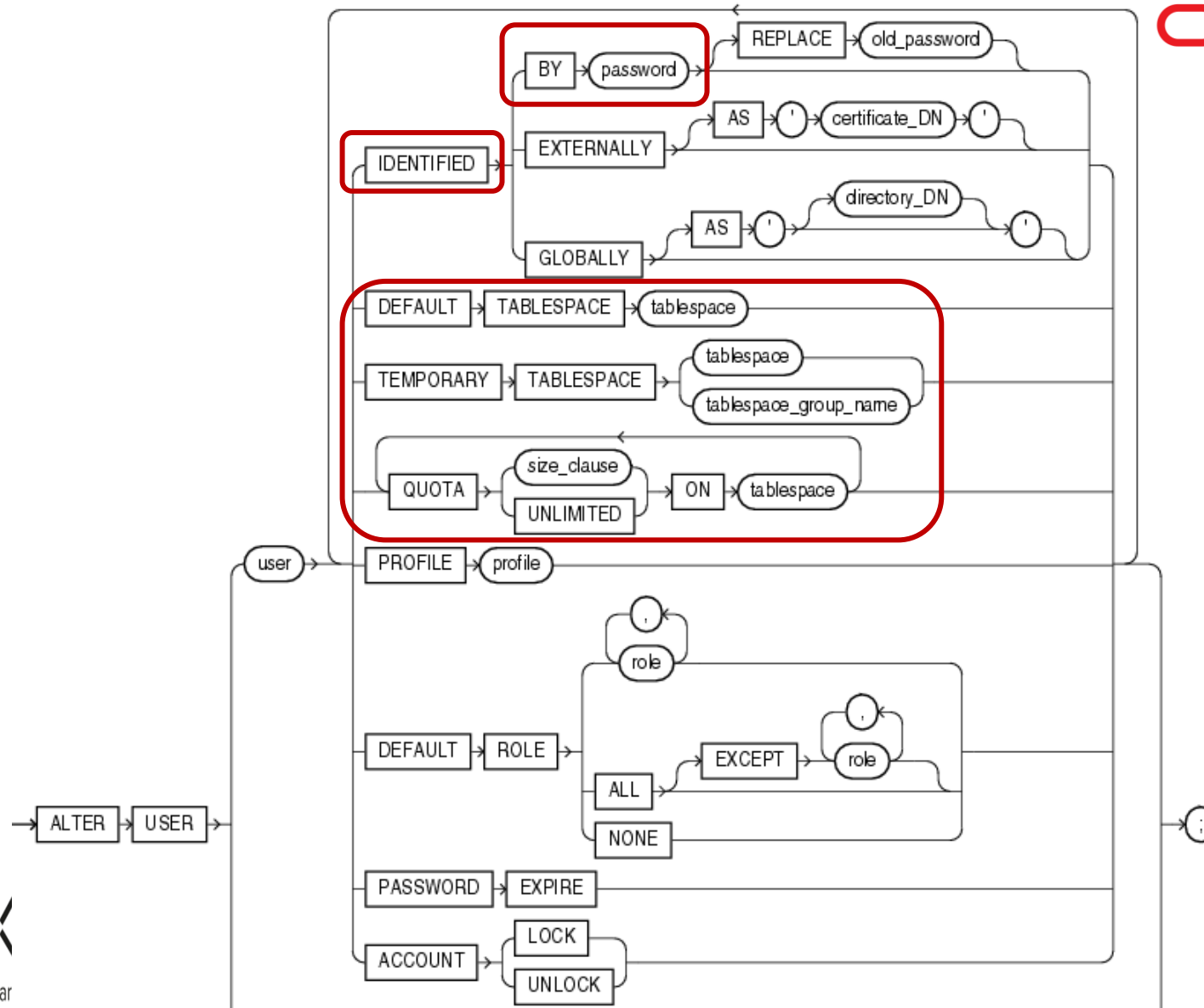
wijzigen gebruiker

ORACLE®



Definitie

SQL
Reference
p. 72



Een gebruiker kan zijn paswoord wijzigen via de instructie:

```
ALTER USER username IDENTIFIED BY  
nieuw_paswoord;
```

De nieuwe informatie wordt in de data dictionary opgeslagen.

```
ALTER USER t191fi IDENTIFIED BY tiger;
```

Verwijderen gebruiker

ORACLE®



Definitie

S Q L

Reference

p. 141



De DBA kan een gebruiker verwijderen.

➤ Indien de gebruiker geen eigenaar is van objecten volstaat de instructie:

DROP USER `username` ;

➤ Bovenstaande instructie faalt als de gebruiker wel objecten bezit. In dat geval kan men met de gebruiker EN zijn objecten verwijderen met:

DROP USER `username` **CASCADE** ;

*Hoe objecten behouden? Exporteren en daarna importeren onder andere user (=nieuwe eigenaar)

Verwijderen gebruiker

Stel dat gebruiker t191fi geen eigenaar is van objecten.

Dan zal de instructie **DROP USER** t191fi; hem verwijderen.

Stel dat gebruiker t191fi wel eigenaar is van objecten.

Dan zal de instructie **DROP USER** t191fi; volgende foutmelding geven:

```
SQL Error: ORA-01922: CASCADE must be specified to drop 'T191FI'
```

```
01922. 00000 - "CASCADE must be specified to drop '%s'"
```

```
*Cause: Cascade is required to remove this user from the system.  
The user own's object which will need to be dropped.
```

Willen we gebruiker + objecten weg dan geven we de instructie:

```
DROP USER t191fi CASCADE;
```

Privileges

- Na creatie van een gebruiker, kan die gebruiker nog niet aanloggen aan de databank of handelingen uitvoeren op de databank. Daarvoor moet de DBA hem systeemprivileges geven.
- **Systeemprivileges** zijn privileges die bepalen welke handelingen een gebruiker op de database mag uitvoeren en worden toegekend door de DBA.

Voorbeelden van systeemprivileges:

- **CREATE SESSION** nodig om te kunnen aanloggen aan de databank. Je hebt dan ook toegang tot alle objecten die PUBLIC zijn.
- **CREATE TABLE** nodig om tabellen te kunnen creëren. Je kan ze dan ook structureel wijzigen en verwijderen.
- **CREATE SEQUENCE** nodig om volgnummers te kunnen aanmaken.
- **CREATE ANY INDEX** je kan indexen op alle tabellen aanmaken
- **ALL PRIVILEGES** alle systeemprivileges

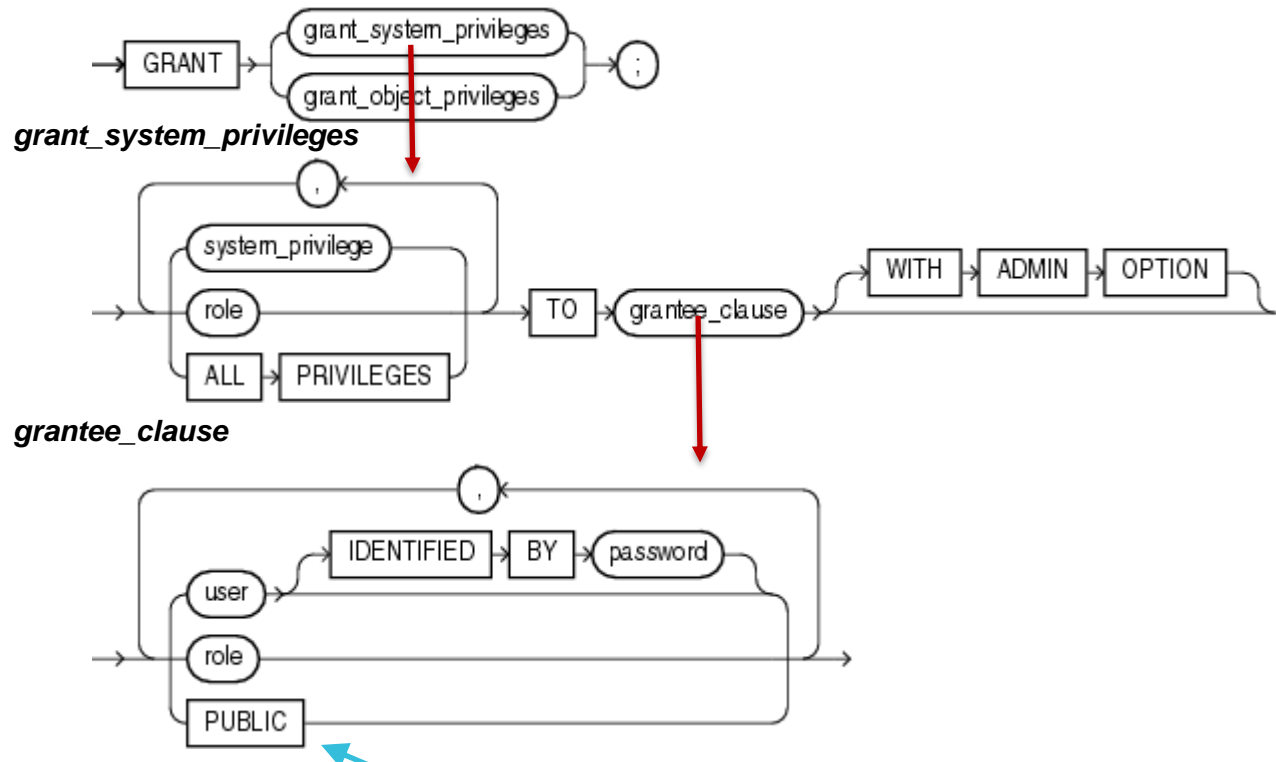
Systeemprivileges Toekennen

ORACLE®



Definitie

SQL
Reference
p. 144



Als een systeemprivilege wordt toegekend aan **PUBLIC**, is het van toepassing op alle huidige en toekomstige gebruikers van de databank

Systeemprivileges Toekennen

- De grantee (=hij die een privilege krijgt) kan elk systeemprivilege, waarvoor hij 'ADMIN OPTION' kreeg, **doorgeven (of ontnemen)** aan anderen. Hij kan hierbij al dan niet zelf WITH ADMIN OPTION gebruiken.
- Gebruiker t191fi moet kunnen aanloggen aan de databank en tabellen kunnen creëren.

```
GRANT CREATE SESSION, CREATE TABLE TO t191fi;
```

- Stel dat hij het CREATE TABLE privilege ook mag doorgeven aan andere gebruikers

```
GRANT CREATE SESSION TO t191fi;
```

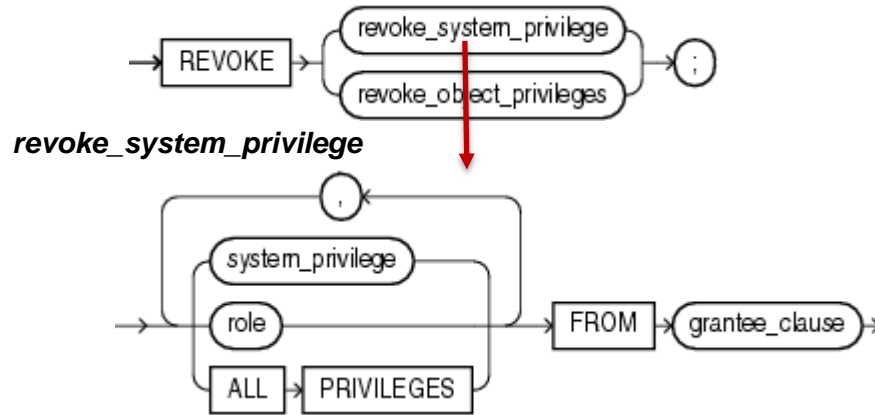
```
GRANT CREATE TABLE TO t191fi WITH ADMIN OPTION;
```

Systeemprivileges ontnemen



Definitie

SQL
Reference
p. 157



Bemerging:

Het ontnemen van een **systeemprivilege** gegeven met WITH ADMIN OPTION heeft **geen cascade effect**.

Systeemprivileges ontnemen



User1 -> t191fi -> t192fi

```
CREATE TABLE      CREATE TABLE  
WITH ADMIN OPTION
```

User1 geeft:

```
REVOKE CREATE TABLE FROM t191fi;
```

T191fi kreeg dit privilege met WITH ADMIN OPTION. Als hij ondertussen het privilege al doorgaf, behouden die gebruikers dat privilege, terwijl hij zelf zijn privilege kwijt is.

User1 -> t191fi -> t192fi

```
CREATE TABLE      CREATE TABLE  
WITH ADMIN OPTION
```

Waar vindt een gebruiker informatie over gekregen systeemprivileges?

USER_SYS_PRIVS

systeemprivileges van de huidige gebruiker

SESSION_PRIVS

privileges binnen een sessie (momenteel geldig)

SYSTEM_PRIVILEGE_MAP

overzicht systeemprivileges

Object privileges toekennen

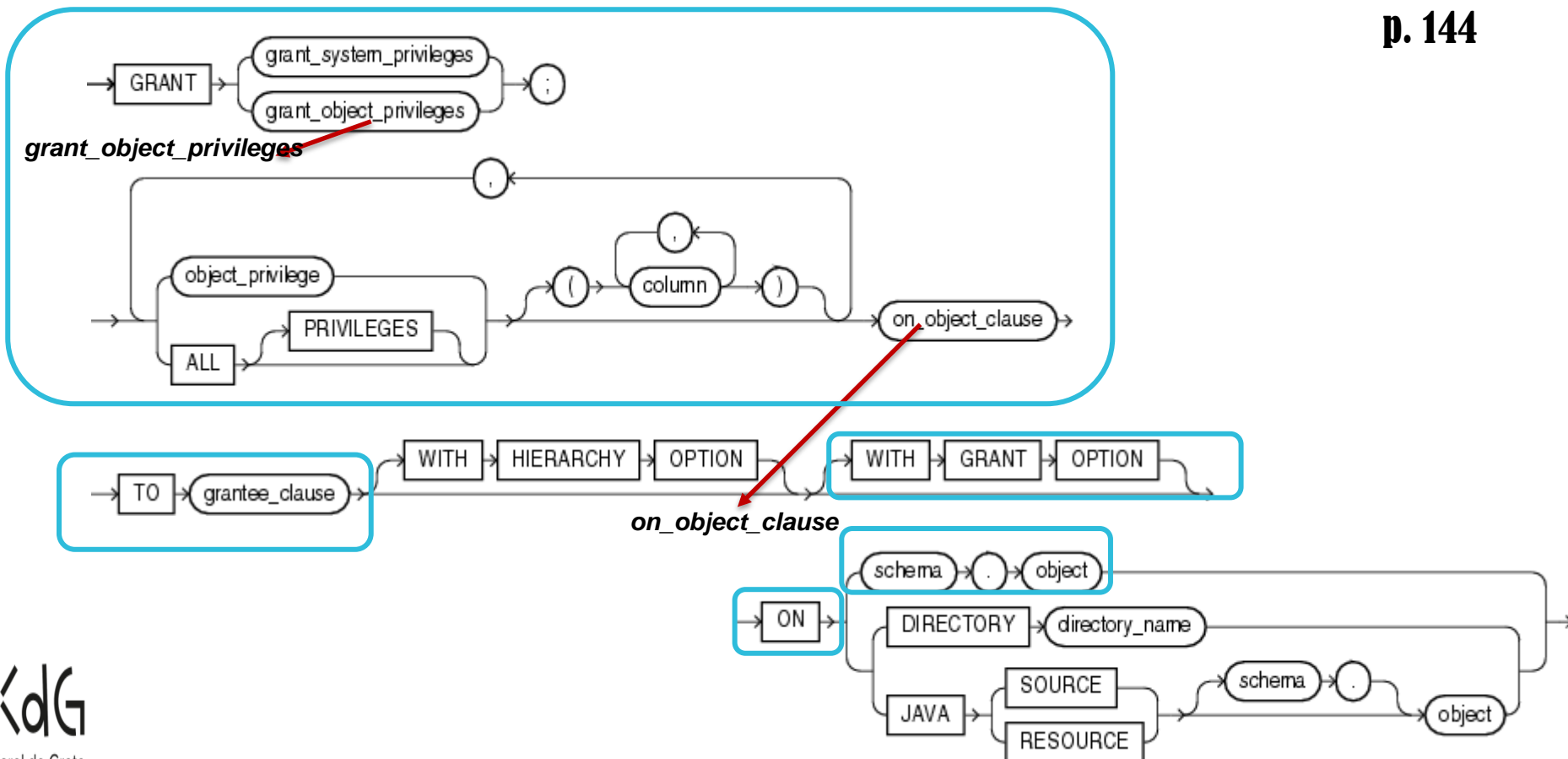


GRANT *object_privileges* **ON** *object* **TO** *grantees*
[WITH GRANT OPTION] ;



Definitie

SQL
Reference
p. 144



Object privileges toekennen



GRANT *object_privileges* **ON** *object* **TO** *grantees*
[WITH GRANT OPTION] ;

1 of meer van deze privileges
(gescheiden door een komma)
of
ALL:verzamelnaam voor alle
rechten die aan een object
toegekend kunnen worden

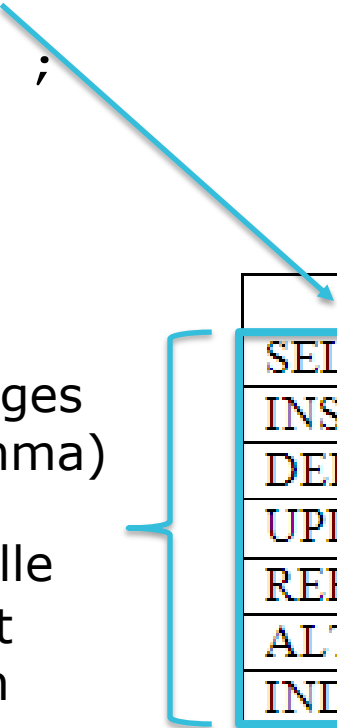


	Table	View	Sequence
SELECT	x	x	x
INSERT	x	x	
DELETE	x	x	
UPDATE	x	x	
REFERENCES	x		
ALTER	x		x
INDEX	x		

Object privileges toekennen



GRANT object_privilege ON *object* TO *grantees*
[WITH GRANT OPTION] ;

Naam van **exact één** object,
van één van deze types

	Table	View	Sequence
SELECT	x	x	x
INSERT	x	x	
DELETE	x	x	
UPDATE	x	x	
REFERENCES	x		
ALTER	x		x
INDEX	x		

- View zien we in een later hoofdstuk

Object privileges toekennen



```
GRANT object_privilege ON object TO grantees  
[WITH GRANT OPTION] ;
```

Naam van één of meer
gebruikers of rollen
(gescheiden door een komma).
Rollen zien we later in dit hoofdstuk.

Object privileges toekennen



GRANT *object_privilege* **ON** *object* **TO** *grantees*
[WITH GRANT OPTION] ;



- De eigenaar van een object kan aan andere gebruikers rechten toekennen of rechten ontnemen op zijn objecten.
- Wanneer een object privilege wordt gegeven **WITH GRANT OPTION**, kan de **ontvanger** van het object privilege het **privilege doorgeven aan andere gebruikers**.

Object privileges toekennen

Stel dat user THEORIE eigenaar is van de onderneming database. Hij kan bijvoorbeeld de volgende privileges geven:

```
GRANT SELECT ON medewerkers TO t191fi;
```

```
GRANT INSERT,UPDATE,DELETE ON afdelingen TO t191fi;
```

```
GRANT ALTER ON afdelingen TO t191fi;
```

=> **ALTER**: gebruiker t191fi kan structurele wijzigingen aanbrengen aan de tabel AFDELINGEN van user THEORIE

Object privileges toekennen



GRANT REFERENCES ON afdelingen **to** t191fi;

=> gebruiker t191fi mag in een tabel die hij aanmaakt, referentiële constraints maken **naar** de tabel afdeling van gebruiker THEORIE

GRANT SELECT ON seq_mednr **TO** t191fi;

=> t191fi mag gebruik maken van de sequence seq_mednr

GRANT ALTER ON seq_mednr **TO** t191fi;

=> t191fi mag de sequence structureel wijzigen

Object privileges toekennen



Stel dat gebruiker theorie eigenaar is van de tabel MEDEWERKERS.

Via de instructie:

```
GRANT SELECT, INSERT ON medewerkers TO t191fi WITH  
GRANT OPTION;
```

geeft hij gebruiker t191fi de toelating om te selecteren op en rijen toe te voegen aan de tabel MEDEWERKERS én om deze privileges door te geven aan andere gebruikers.

Bemerking: een gebruiker verwijst als volgt naar de tabel waarvan hij geen eigenaar is : *naam_eigenaar.naam_tabel*

(=schema)

Object privileges toekennen

De object privileges **INSERT, UPDATE en REFERENCES** kunnen **selectief toegekend** worden: de grants zijn dan niet op alle kolommen van het object (=tabel) van toepassing.

```
GRANT UPDATE (salaris) ON medewerkers  
TO t191fi;
```

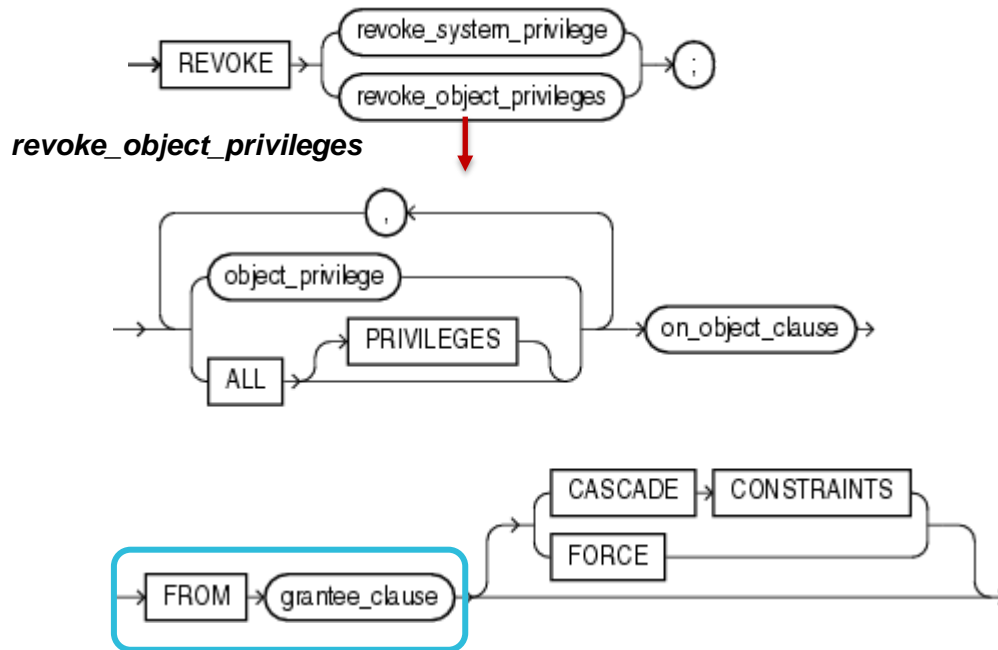
=> Gebruiker t191fi kan enkel het attribuut salaris uit de tabel medewerker wijzigen.

Object privileges ontnemen



Definitie

SQL
Reference
p. 157



Opmerkingen:

- Het ontnemen van een **object privilege** gegeven met WITH GRANT OPTION heeft **wel het cascade effect**.
- Object privileges kunnen **niet selectief ontnomen** worden;

Object privileges ontnemen



cascade effect:

Stel dat gebruiker theorie het volgende privilege toekende:

```
GRANT SELECT ON medewerkers TO t191fi WITH GRANT  
OPTION;
```

Stel dat t191fi dit privilege doorgaf aan gebruiker t192fi.

Dan zal, wanneer theorie het privilege terug ontnemt van
t191fi (**REVOKE SELECT ON medewerkers FROM t191fi;**)
het privilege ook van t192fi ontnomen worden.

theorie - > t191fi - > t192fi

~~SELECT ON medewerkers~~

~~SELECT ON medewerkers~~

~~WITH GRANT option~~

Object privileges ontnemen



test selectief ontnemen privileges:

Stel dat gebruiker theorie het volgende privilege toekende:

```
GRANT UPDATE (parkeerplaats, salaris) ON medewerkers TO  
t191fi;
```

Achteraf wil theorie het aan t191fi toegekende privilege aanpassen. t191fi mag enkel nog het attribuut salaris aanpassen.

```
REVOKE UPDATE (parkeerplaats) ON medewerkers FROM  
t191fi;
```

Wél: **REVOKE UPDATE ON** medewerkers **FROM** t191fi;

```
GRANT UPDATE (salaris) ON medewerkers TO t191fi;
```

Besluit: je kan object privileges NIET selectief ontnemen.

Object privileges ontnemen



Voorbeeld

Een **REVOKE REFERENCES** (of impliciet een **REVOKE ALL**) kan een fout geven als er foreign key constraints naar deze tabel verwijzen:

```
REVOKE ALL ON medewerkers FROM theorie;
```

```
SQL Error: ORA-01981: CASCADE CONSTRAINTS  
must be specified to perform this revoke
```

```
REVOKE ALL ON medewerkers FROM theorie  
CASCADE CONSTRAINTS ;
```

CASCADE CONSTRAINTS zal ook de foreign key constraints verwijderen die de gebruiker (theorie) maakte.



USER_TAB_PRIVS

USER_TAB_PRIVS_MADE

USER_TAB_PRIVS_RECD

USER_COL_PRIVS

USER_COL_PRIVS_MADE

USER_COL_PRIVS_RECD



voor selectieve object
privileges

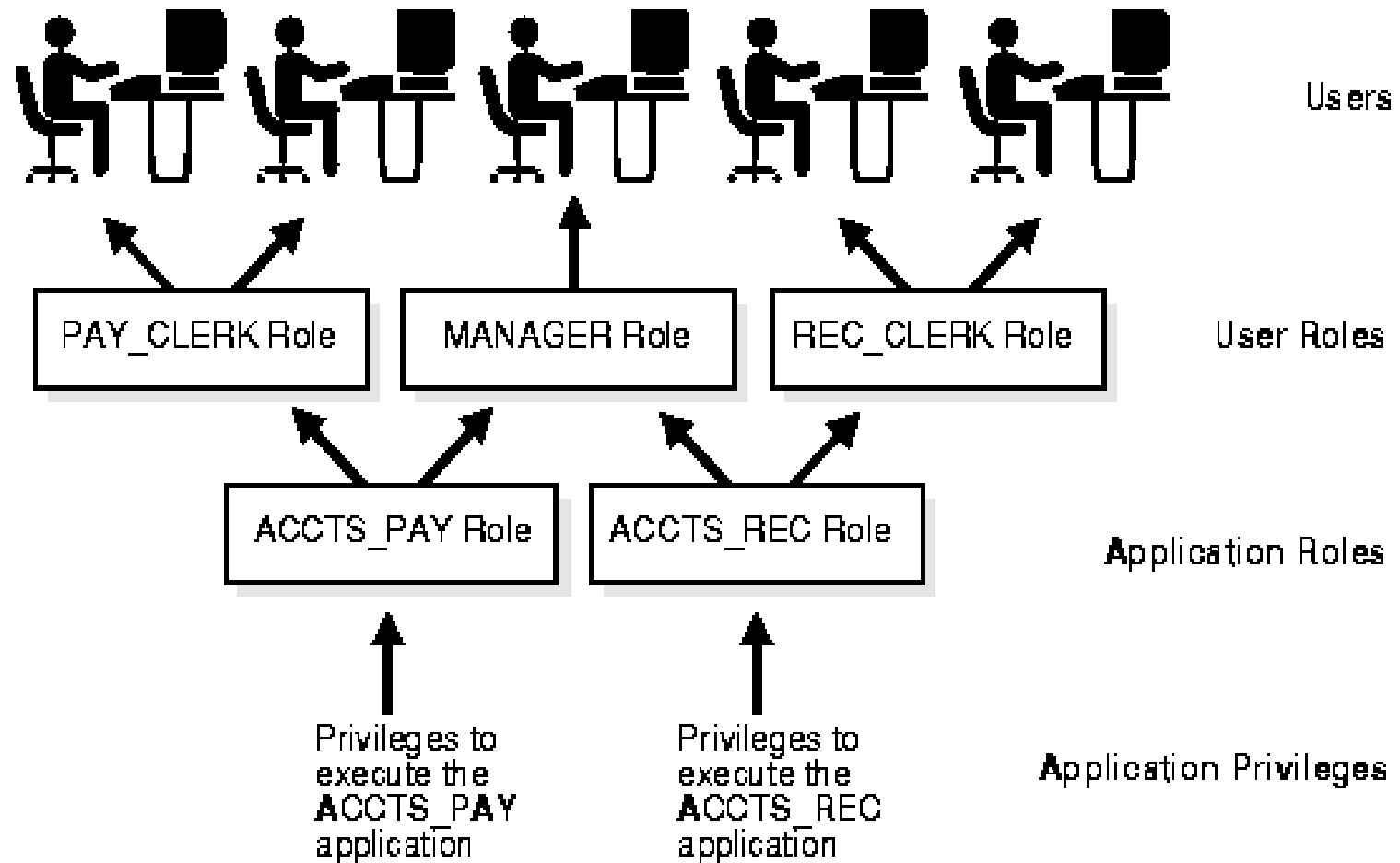
Rollen

Gebruikers van de database kunnen gegroepeerd worden al naargelang hun taken.

De database beheerder (of een gebruiker met het CREATE ROLE systeemprivilege) kan aan elke groep (=role) een verzameling van systeem- en object privileges aanbieden.

- Men kan gebruik maken van de voorgedefiniëerde rollen **CONNECT**, **RESOURCE** of **DBA**
- Men kan zelf een rol samenstellen

Rollen



Aanmaken van een rol

ORACLE®



Definitie

SQL
Reference
p. 106



rolename moet een unieke naam zijn binnen de database,
mag ook niet hetzelfde zijn als een gebruikersnaam



Voorbeeld

```
CREATE ROLE rmed;
```

Toekennen van privileges aan rol

- Je kan een privilege toekennen aan een rol (= grantee)
- Net zoals bij gebruikers kunnen systeemprivileges met `WITH ADMIN OPTION` en objectprivileges met `WITH GRANT OPTION` toegekend worden aan een rol
- Alleen dan kunnen gebruikers in die rol het privilege doorgeven

Toekennen van privileges aan rol



```
GRANT CREATE SESSION TO rmed;
```

```
GRANT CREATE TABLE, CREATE SEQUENCE TO rmed;
```

```
GRANT SELECT,UPDATE(salaris,afd_nr),DELETE ON  
medewerkers TO rmed;
```

```
GRANT SELECT ON afdelingen TO rmed;
```

Ontnemen van privileges van een rol



```
REVOKE DELETE ON medewerkers FROM rmed;
```


Toekennen aan en ontnemen van een rol



- Rollen worden toegekend/ontnomen zoals system privileges

```
GRANT rmed TO gebruiker1;
```

```
GRANT rmed TO PUBLIC;
```

```
REVOKE rmed FROM gebruiker1;
```

Toekennen aan en ontnemen van een rol

Je kan rollen nesten = een rol toekennen aan een andere rol.

```
CREATE ROLE rproj;  
GRANT SELECT, UPDATE,DELETE ON projecten to rproj;  
GRANT SELECT, UPDATE,DELETE ON opdrachten to rproj;  
GRANT rmed TO rproj;
```

Iemand die een rol toegewezen kreeg met WITH ADMIN OPTION kan:


- Die rol wijzigen
 - extra rechten eraan toekennen
 - extra users toewijzen aan de rol
- Die rol droppen

Toekennen aan en ontnemen van een rol



We gebruikten dit in de script voor het aanmaken van de gebruikte databanken

Starten met SQL Developer

Bijgevoegde bestanden:  [Starten in SQL Developer.docx](#) |
kB)

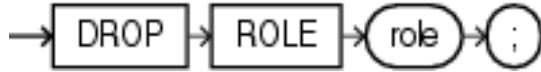
```
CREATE USER theorie IDENTIFIED BY theorie  
DEFAULT TABLESPACE users  
QUOTA 2M ON users;  
CREATE USER praktijk IDENTIFIED BY praktijk  
DEFAULT TABLESPACE users  
QUOTA 2M ON users;  
GRANT DBA TO theorie,praktijk;
```

Verwijderen van een rol



Definitie

SQL
Reference
p. 138



Voorbeeld

DROP ROLE rmed;

- De privileges in rollen worden aan een gebruiker toegekend bij het begin van een sessie.
- Veranderingen aan rollen *tijdens* de sessie hebben geen impact op de privileges van de gebruiker in die sessie.

Rollen en dictionary tabellen



USER_ROLE_PRIVS:

Welke roles kreeg de huidige user toegewezen?

ROLE_ROLE_PRIVS

Geeft informatie over geneste roles

ROLE_SYS_PRIVS

Welke systeemprivileges omvat de role?

ROLE_TAB_PRIVS

Welke object privileges omvat de role?

SESSION_ROLES

Welke rollen zijn actief voor de ingelogde gebruiker?