

Creatie en Ontwerp

wim.bertels@ucll.be

Naamsvermelding-NietCommercieel-GelijkDelen 4.0 Unported
Licentie

INSERT

```
insert into recreanten (spelersnr, naam, plaats)
select  spelersnr, naam, plaats
from    spelers
where   bondsnr is null;
```

UPDATE

```
update wedstrijden  
set      gewonnen = 0  
where    spelersnr in  
         (select  spelersnr  
          from    spelers  
          where   plaats = 'Den Haag');
```

DELETE

```
delete  
from    spelers  
where   jaartoe >  
        (select  avg(jaartoe)  
        from     spelers);
```

Tijdelijke tabellen

```
create temporary table SOMBOETES  
  (totaal decimal(10,2));
```

```
insert into somboetes  
  select  sum(bedrag)  
  from    boetes;
```

Kopie tabel

```
create table teams_kopie as  
  (select  *  
   from    teams);
```

Default waarden

```
create table boetes (  
    bnr      integer not null    primary key,  
    snr      integer not null,  
    datum   date      not null   default '1990-01-01',  
    bedrag  int        not null   default 50);
```

Tabel beperkingen

```
[ CONSTRAINT constraint_name ]  
{ CHECK ( expression ) [ NO INHERIT ] |  
  UNIQUE [ NULLS [ NOT ] DISTINCT ] ( column_name [, ... ] ) index_parameters |  
  PRIMARY KEY ( column_name [, ... ] ) index_parameters |  
  EXCLUDE [ USING index_method ] ( exclude_element WITH operator [, ... ] ) index_parameters  
[ WHERE ( predicate ) ] |  
  FOREIGN KEY ( column_name [, ... ] ) REFERENCES reftable [ ( refcolumn [, ... ] ) ]  
    [ MATCH FULL | MATCH PARTIAL | MATCH SIMPLE ] [ ON DELETE referential_action ] [ ON  
UPDATE referential_action ] }  
[ DEFERRABLE | NOT DEFERRABLE ] [ INITIALLY DEFERRED | INITIALLY IMMEDIATE ]
```


Kolom beperkingen

[CONSTRAINT constraint_name]
{ NOT NULL |
NULL |
CHECK (expression) [NO INHERIT] |
DEFAULT default_expr |
GENERATED ALWAYS AS (generation_expr) STORED |
GENERATED { ALWAYS | BY DEFAULT } AS IDENTITY [(sequence_options)] |
UNIQUE [NULLS [NOT] DISTINCT] index_parameters |
PRIMARY KEY index_parameters |
REFERENCES reftable [(refcolumn)] [MATCH FULL | MATCH PARTIAL | MATCH SIMPLE]
[ON DELETE referential_action] [ON UPDATE referential_action] }
[DEFERRABLE | NOT DEFERRABLE] [INITIALLY DEFERRED | INITIALLY IMMEDIATE]

Deferrable

r ..fk1 references t(pk1) deferrable..

```
start transaction;  
set constraints all deferred;  
insert into r(fk1) values (1);  
insert into t(pk1) values (1);  
commit;
```

<https://www.postgresql.org/docs/current/sql-set-constraints.html>

Referende actie

..fk1 references t(pk1) on delete set null..

Referentiële integriteit?:

- NO ACTION: default
- RESTRICT: =no action, niet uitstelbaar (deferrable)
- CASCADE: Waterval effect!
- SET NULL(fk1): optie om zinvolle selectie van verwijzende kolommen op te geven
- SET DEFAULT: als er een zinvolle default is (optie cf set null)

<https://www.postgresql.org/docs/current/sql-createtable.html>

Catalogus

Meta en organisatie data over de databank in de databank zelf

- information_schema
- pg_catalog

Eg. `select * from information_schema.tables;`

Schema en Locale

wim.bertels@ucll.be

Naamsvermelding-NietCommercieel-GelijkDelen 4.0 Unported
Licentie

Wat

- SCHEMA = namespace
- Binnen een namespace moeten alle objecten (b.v. tabellen, functies, views (later) ...) een unieke naam hebben

Welk schema

PostgreSQL:

de standaard namespace is "public"

je kan dit nakijken met:

```
> SHOW search_path;
```

Product specifiek

- Het standaard/default schema is opstelling specifiek
- Sommige andere producten:
de standaard namespace is de usernaam

Rechten

Privileges (standaard)

- Je moet rechten hebben op het parentobject om rechten uit te voeren op de kinderoBJECTen
- default: enkel eigenaar heeft toegangsrechten
- Dus Server>DB>Schema>Objecten(bv tabel)

Rechten toekennen

```
GRANT <privilege> ON  
<objecttype> <objectname>  
TO <role>;
```

Rechten wegnemen

```
REVOKE <privilege> ON  
<objecttype> <objectname>  
FROM <role>;
```

Privileges doorgeven?

- Recht toekennen om rechten toe te kennen:

```
GRANT <privilege> ON  
<objecttype> <objectname>  
TO <role>  
WITH GRANT OPTION;
```

Privilege voorbeelden

- SELECT
- INSERT
- UPDATE
- DELETE
- EXECUTE
- ALL PRIVILEGES

- GRANT DELETE ON TABLE boetes TO admin WITH GRANT OPTION;

Localisatie

Localisatie (1/2)

- Teken sets: speciale tekens ondersteunen (b.v. ç, Ъ ...)
- collating sequences: abc ...
- encoding: UTF-8, LATIN1, ...
- ..

Localisatie (2/2)

- LC_COLLATE : string volgorde
- LC_CTYPE : character classificatie
- LC_MESSAGES : taal van het bericht
- LC_MONETARY : formatteren valuta
- LC_NUMERIC : formatteren nummers
- LC_TIME : formatteren tijd

>SHOW LC_COLLATE;

Impact Locale?

- Wat zijn de gevolgen hiervan?
- Waarom moeten we hiervoor opletten?

Datatypes

wim.bertels@ucll.be

Naamsvermelding-NietCommercieel-GelijkDelen 4.0 Unported
Licentie

Kleinere topics

- Datatypes & Functies

Datatypes & Functies

<https://www.postgresql.org/docs/current/datatype.html>

<https://www.postgresql.org/docs/current/functions.html>

- Numerisch
- Geld
- Character
- Binair
- Tijd
- Boolean
- ..

Algemene verschillen voor teken-datatypes

- char(n): vaste lengte
 - vaste lengte, plaats wordt gereserveerd ook indien deze plaats niet gebruikt wordt
- varchar(n): variabele lengte
 - flexibele lengte met maximum, trager?
- text: onbeperkte variabele lengte
 - meest flexibel, traagst?

Strings vs Identifiers

'dit een string'

"dit is een identifier"

bv. 'hond' vs "54 mijngekketabelnaam"

sql : standard uppercase >

in de praktijk: sqlcode is case insensitive tenzij tussen quotes

IDS

wim.bertels@ucll.be

Naamsvermelding-NietCommercieel-GelijkDelen 4.0 Unported
Licentie

IDENTITY datatype

- Geen echte sleutel!
- Betekenis?
- Willekeur!
- Exploits?

IDENTITY kolom

```
CREATE TABLE id_demo(  
    id          integer PRIMARY KEY GENERATED BY DEFAULT AS IDENTITY,  
    content text  
);
```

OPMERKING:

- CREATE TABLE zal een impliciete sequentie
"id_demo_id_seq" aanmaken voor IDENTITY kolom "id" in tabel id_demo

OPMERKING:

- CREATE TABLE + PRIMARY KEY zal een impliciete unieke index
"id_demo_pkey" aanmaken voor table "id_demo"

IDENTITY kolom

Lijst van relaties			
Schema	Naam	Type	Eigenaar
public	id_demo	table	wim
public	id_demo_id_seq	sequence	wim
...			
(103 rijen)			

IDENTITY kolom

- Creëert extra object
 - Een sequence
- Wat als een gebruiker rechten heeft tot de tabel en niet tot de sequence?

Rechten op een sequence

```
GRANT <privilege> ON  
SEQUENCE sequence_name  
TO <role>;
```

of

```
GRANT <privilege> ON  
ALL SEQUENCES IN SCHEMA schema_name  
TO <role>;
```

ISO

- IDENTITY
- SERIAL, AUTOINCREMENT,.. --dialect