

Veritabanı Yönetim Sistemleri (335)

Dr. Öğr. Üyesi Ahmet Arif AYDIN

L18- PostgreSQL (Alter, Cascade, Function)

GÜZ -2022

- İlişkisel Cebirde kullanılan işlemlerin PostgreSQL ile gerçekleştirilmesi
- ALTER
- CASCADE
- FUNCTION

PostgreSQL & İlişkisel Cebir

İlişkisel Cebir	PostgreSQL
$\Pi a, b (T1)$	SELECT a, b FROM TABLE
$\sigma (d > e) \wedge (f = g) (T1)$	SELECT * FROM T1 WHERE d > e AND f = g
$p \times q$	SELECT * FROM p, q
$\pi a, b \sigma (d > e) \wedge (f = g) (p \times q)$	SELECT a, b FROM p, q WHERE d > e AND f = g;

İlişkisel Cebir	PostgreSQL
Renaming	AS
$p \cup q$	SELECT * FROM p UNION SELECT * FROM q
$p - q$	SELECT * FROM p EXCEPT SELECT * FROM q
$p \cap q$	SELECT * FROM p INTERSECT SELECT * FROM q

ALTER

PostgreSQL'de oluşturulan bir tablo üzerinde değişiklik gerçekleştirmek için

ALTER TABLE komutu kullanılır.

- Bir tablonun ismi değiştirilebilir

```
ALTER TABLE test3 RENAME TO personel;
```

- Bir kolonun ismi değiştirilebilir

```
ALTER TABLE personel RENAME COLUMN öğrencino TO personalid;
```

- Bir tabloya yeni bir kolon eklenebilir

```
ALTER TABLE personel ADD COLUMN adres varchar(20);
```

ALTER

PostgreSQL'de oluşturulan bir tablo üzerinde değişiklik gerçekleştirmek için

ALTER TABLE komutu kullanılır.

- Bir kolon silinebilir

```
ALTER TABLE personel DROP COLUMN adres;
```

- Bir kolona yeni bir kısıtlama eklenebilir.

```
ALTER TABLE personel ADD CONSTRAINT yaşkontrol CHECK (yaş>20);
```

PostgreSQL: Cascade

PostgreSQL'de **CASCADE** komutu

Primary key ve Foreign Key ilişkisi bulunan tabloların ana tabloda bulunan kayıtların silinmesi veya güncellenmesi durumunda ilişkili tabloda otomatik olarak silme ve güncelleme işleminin gerçekleştirmeyi sağlar

In PostgreSQL, a cascade means that

a delete or update of records in a parent table

will automatically delete or update matching records in a child table

where a foreign key relationship is in place

<https://www.postgresql.org/docs/9.5/ddl-constraints.html>

PostgreSQL: Cascade

PostgreSQL'de **CASCADE** komutu

Primary key ve Foreign Key ilişkisi bulunan tabloların ana tabloda bulunan kayıtların silinmesi veya güncellenmesi durumunda ilişkili tabloda otomatik olarak silme ve güncelleme işleminin gerçekleştirmeyi sağlar

```
CREATE TABLE öğrenci (  
    öğrencino int primary key,  
    isim varchar(20) ,  
    kullanıcıadı varchar(20) ,  
    yaş int,  
    ortalama real  
);
```

```
CREATE TABLE dersler (  
    öğrencino integer references  
    öğrenci(öğrencino),  
    derskodu varchar ,  
    derskredi integer,  
    dersnotu integer  
);
```

<https://www.postgresql.org/docs/9.5/ddl-constraints.html>

PostgreSQL: Cascade

PostgreSQL'de **CASCADE** komutu

Primary key ve Foreign Key ilişkisi bulunan tabloların ana tabloda bulunan kayıtların silinmesi veya güncellenmesi durumunda ilişkili tabloda otomatik olarak silme ve güncelleme işleminin gerçekleştirmeyi sağlar

FOREIGN KEY(fk_columns)

REFERENCES anatablo (ilişkili kolon)

[ON DELETE delete_action]

[ON UPDATE update_action]

- SET NULL
- SET DEFAULT
- RESTRICT
- NO ACTION
- CASCADE

PostgreSQL: Cascade

```
CREATE TABLE products (  
    product_no integer PRIMARY KEY,  
    name text,  
    price numeric );
```

```
CREATE TABLE orders (  
    order_id integer PRIMARY KEY,  
    shipping_address text,  
    ... );
```

the child data is either deleted or updated when the parent data is deleted or updated

PostgreSQL: Cascade

```
CREATE TABLE products (  
    product_no integer PRIMARY KEY,  
    name text,  
    price numeric );
```

```
CREATE TABLE orders (  
    order_id integer PRIMARY KEY,  
    shipping_address text,  
    ... );
```

```
CREATE TABLE order_items (  
    product_no integer REFERENCES products(product_no) ON DELETE RESTRICT,  
    order_id integer REFERENCES orders(order_id ) ON DELETE CASCADE,  
    quantity integer,  
    PRIMARY KEY (product_no, order_id)  
);
```

Ana tablodaki veri silindiğinde veya güncellendiğinde ilişkili olunan tablodaki satırlar üzerinde tanımlanan action gerçekleştirilir.

PostgreSQL: Cascade

```
CREATE TABLE products (  
    product_no integer PRIMARY KEY,  
    name text,  
    price numeric );
```

```
CREATE TABLE orders (  
    order_id integer PRIMARY KEY,  
    shipping_address text,  
    ... );
```

```
CREATE TABLE order_items (  
    product_no integer REFERENCES products(product_no) ON DELETE RESTRICT,  
    order_id integer REFERENCES orders(order_id) ON DELETE CASCADE,  
    quantity integer,  
    PRIMARY KEY (product_no, order_id)  
);
```

Restrict default olarak primary key alanına bağlı foreign key'ler mevcut olduğunda silme işlemini engeller.

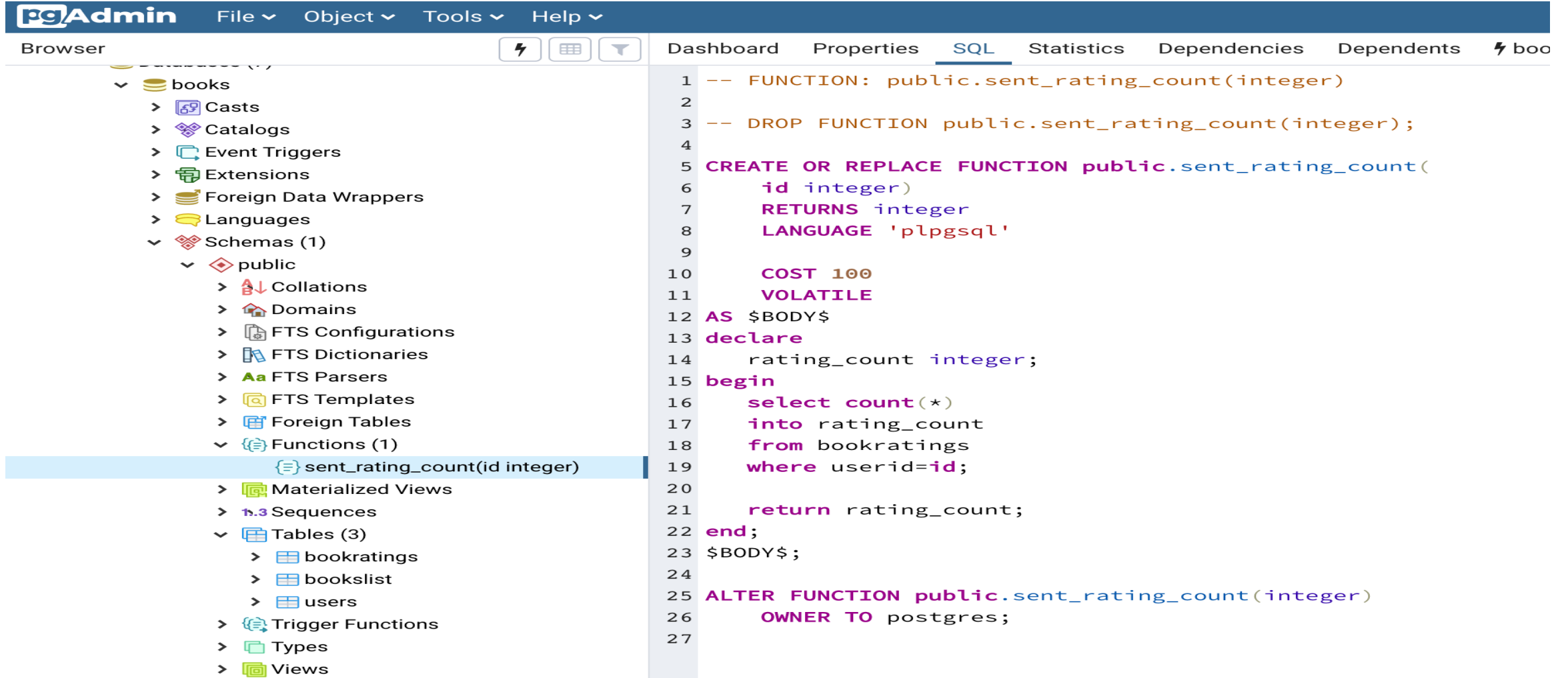
Cascade products tablosundan bir ürün silindiğinde ilişkili tablolardan da otomatik olarak silinmesini sağlar

NO ACTION
ON UPDATE CASCADE
ON UPDATE RESTRICT

<https://www.postgresqltutorial.com/postgresql-foreign-key/>

PostgreSQL: Functions

PostgreSQL'de ile kullanıcı tanımlı fonksiyonlar (user-defined functions) yaradımıyla istenilen sorgulama işleminin fonksiyonunu oluşturma imkanı sağlar.



The screenshot displays the PgAdmin 4 interface. On the left, the 'Browser' pane shows the database structure. The 'public' schema is expanded, and 'Functions (1)' is selected, showing the function 'sent_rating_count(id integer)'. The right pane shows the SQL editor with the following code:

```
1 -- FUNCTION: public.sent_rating_count(integer)
2
3 -- DROP FUNCTION public.sent_rating_count(integer);
4
5 CREATE OR REPLACE FUNCTION public.sent_rating_count(
6     id integer)
7     RETURNS integer
8     LANGUAGE 'plpgsql'
9
10    COST 100
11    VOLATILE
12 AS $BODY$
13 declare
14     rating_count integer;
15 begin
16     select count(*)
17     into rating_count
18     from bookratings
19     where userid=id;
20
21     return rating_count;
22 end;
23 $BODY$;
24
25 ALTER FUNCTION public.sent_rating_count(integer)
26     OWNER TO postgres;
27
```

PostgreSQL: Functions

```
create [or replace ]function sent_rating_count (id int)
returns int
language plpgsql
as
$$
declare
    rating_count integer;
begin
    select count(*)
    into rating_count
    from bookratings
    where userid=id;

    return rating_count ;
end;
$$;
```

<https://www.postgresqltutorial.com/postgresql-create-function/>

PostgreSQL: Functions

Fonksiyondan
gönderilen
verinin tipi

```
create [or replace ]function sent_rating_count (id int)
```

```
returns int
```

Fonksiyon adı

Parametre ve tipi

```
language plpgsql
```

Fonksiyon
oluşturmada
kullanılan dil

<https://www.postgresqltutorial.com/introduction-to-postgresql-stored-procedures/>

```
as
```

```
$$
```

```
declare
```

```
rating_count integer;
```

Değişken tanımlanıyor

```
begin
```

```
select count(*)
```

```
into rating_count
```

İşlem sonucu değişkene atanıyor

```
from bookratings
```

```
where userid=id;
```

```
return rating_count ;
```

Fonksiyonun geri gönderdiği değer

```
end;
```

```
$$;
```

<https://www.postgresqltutorial.com/postgresql-create-function/>

PostgreSQL: Functions

The screenshot shows the PostgreSQL GUI with the 'books' database selected. The 'Functions (1)' folder is expanded, showing the 'sent_rating_count(id integer)' function. The 'Definition' tab is active, displaying the following SQL code:

```
1 declare
2   rating_count integer;
3 begin
4   select count(*)
5   into rating_count
6   from bookratings
7   where userid=id;
8
9   return rating_count;
10 end;
11
```

The function's arguments are 'id integer', the return type is 'integer', and the language is 'plpgsql'.

Select sent_rating_count(8);

	sent_rating_count
	integer
1	18

PostgreSQL: PL/pgSQL

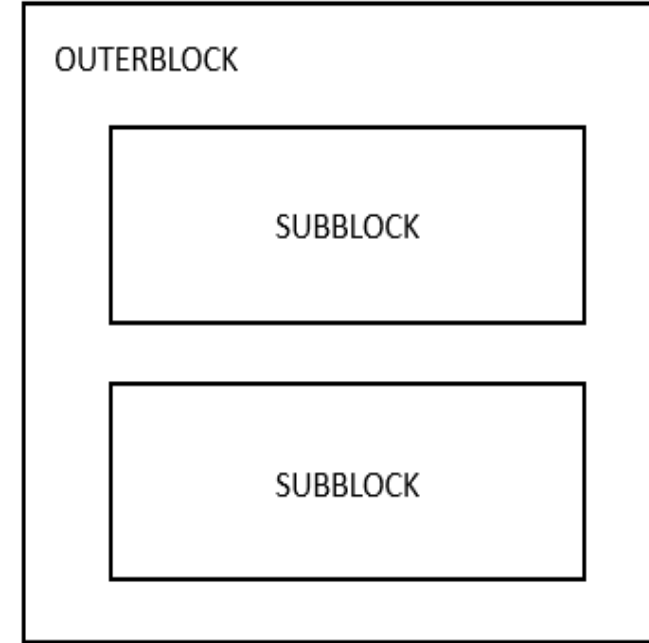
PL/pgSQL

- Prosedürel bir programlama dilidir.
- Kullanıcı tanımlı fonksiyonların (user-defined functions)
- Stored procedure ve Trigger tanımlanmasına imkan sağlar

PostgreSQL: PL/pgSQL

```
create [or replace ]function sent_rating_count (id int)
returns int
language plpgsql
as
$$
declare
    rating_count integer;
begin
    select count(*)
    into rating_count
    from bookratings
    where userid=id;

    return rating_count ;
end;
$$;
```



PL/pgSQL

- Blok yapısını kullanır
- Tanımlama ve body'den oluşur
- İç içe blok kullanılır.

PL/pgSQL

- Değişken ve sabit tanımlama
- Hata mesajı oluşturma
- Kontrol yapıları (if-while-for)

<https://www.postgresqltutorial.com/plpgsql-variables/>

<https://www.postgresqltutorial.com/plpgsql-errors-messages/>

<https://www.postgresqltutorial.com/plpgsql-if-else-statements/>

Dinlediğiniz için
Teşekkürler...
İyi çalışmalar...