

İsim

FETCH – bir gösterici kullanarak bir sorgudan satırları çeker

KULLANIM

FETCH [*yön* { **FROM** | **IN** }] *gösterici_ismi*

burada *yön* ya boş ya da şunlardan biri olabilir:

NEXT
PRIOR
FIRST
LAST
ABSOLUTE *miktar*
RELATIVE *miktar*
miktar
ALL
FORWARD
FORWARD *miktar*
FORWARD ALL
BACKWARD
BACKWARD *miktar*
BACKWARD ALL

Açıklama

FETCH evvelce oluşturulmuş bir göstericiyi kullanarak satırları çeker.

Bir gösterici **FETCH** tarafından da kullanılan ortak bir konuma sahiptir. Göstericinin konumu sorgu sonucunun ilk satırının öncesinde, herhangi bir satırında veya son satırının sonrasında olabilir. İlk oluşturulduğunda bir gösterici ilk satırın öncesine konumlanır. Satırlar çekildikten sonra gösterici en son alınan satıra konumlanır. **FETCH**'in satırları çekip çalışması durduğunda gösterici son satırın sonrasına ya da eğer geriye çekilmişse ilk satırın öncesine konumlanır. **FETCH ALL** göstericiyi daima son satırın sonrasına, **FETCH BACKWARD ALL** ise daima ilk satırın öncesine konumlandırır.

NEXT (sonraki), **PRIOR** (önceki), **FIRST** (ilk), **LAST** (son), **ABSOLUTE** (mutlak), **RELATIVE** (görelî) biçimleri göstericiyi ilgili konuma taşıdıktan sonra tek bir satırı çeker. O konumda bir satır yoksa, sonuç boş olarak döner ve gösterici duruma göre ya ilk satırın öncesine ya da son satırın sonrasına konumlanır.

FORWARD (ileri) ve **BACKWARD** (geri) biçimleri, ileri ya da geri yönde belirtilen miktarda satırı çekmek için kullanılabilir. Gösterici son çekilen satıra konumlanmış olarak bırakılır (eğer *miktar* alınabilecek satır sayısından fazla ise, gösterici duruma göre son satırın sonrasına ya da ilk satırın öncesine konumlanır).

RELATIVE 0, **FORWARD 0**, ve **BACKWARD 0** biçimlerinin hepsi gösterici hareket ettirilmeden o anki satırın çekilmesini sağlar, yani son çekilen satır yeniden çekilir. Bu, ancak, gösterici ilk satırın öncesinde ya da son satırın sonrasında değilse bir satırın çekilmesini sağlar; aksi takdirde, bir satır döndürülmez.

Parametreler

yön

Çekim yönünü ve çekilecek satır sayısını tanımlar. Aşağıdakilerden biri olabilir:

NEXT

Sonraki satırı çeker. *yön* belirtilmezse bu öntanımlıdır.

PRIOR

Önceki satırı çeker.

FIRST

Sorgunun ilk satırını çeker (**ABSOLUTE 1** ile aynıdır).

LAST

Sorgunun son satırını çeker (**ABSOLUTE -1** ile aynıdır).

ABSOLUTE miktar

miktar pozitifse, sorgunun *miktar*'ıncı satırını; negatifse, sondan **abs** (*miktar*) 'ıncı satırını çeker. *miktar* gereğinden büyükse, gösterici duruma göre ya ilk satırın öncesine ya da son satırın sonrasına konumlanır; özellikle, **ABSOLUTE 0** göstericinin ilk satırın öncesine konumlanmasını sağlar.

RELATIVE miktar

Göstericinin konumuna göre, *miktar* pozitifse, sonraki *miktar*'ıncı satırı; negatifse, önceki **abs** (*miktar*) 'ıncı satırı çeker. **RELATIVE 0** o anki satırı varsa yeniden çeker.

miktar

Sonraki *miktar* satırı çeker (**FORWARD miktar** ile aynıdır).

ALL

Kalan satırların hepsini çeker (**FORWARD ALL** ile aynıdır).

FORWARD

Sonraki satırı çeker (**NEXT** ile aynıdır).

FORWARD miktar

Sonraki *miktar* satırı çeker. **FORWARD 0** o anki satırı yeniden çeker.

FORWARD ALL

Kalan satırların hepsini çeker.

BACKWARD

Önceki satırı çeker (**PRIOR** ile aynı).

BACKWARD miktar

Önceki *miktar* satırı çeker. **BACKWARD 0** o anki satırı yeniden çeker.

BACKWARD ALL

Önceki tüm satırları çeker (geriye doğru tarama).

miktar

Konumu ya da çekilecek satır sayısını belirtmek üzere kullanılan bir işaretli tamsayıdır. **FORWARD** ve **BACKWARD** için belirtilen bir negatif *miktar* yönün değişmesine sebep olur.

gösterici_ismi

Açık bir göstericinin ismi.

Çıktılar

FETCH başarılı olduğunda şöyle bir çıktı döndürür:

FETCH miktar

Burada, *miktar* çekilen satırların sayısıdır (sıfır olabilir). Yalnız, **psql**'de komutun ismi çıktıda gösterilmez, sadece çekilen satır sayısı gösterilir.

Ek Bilgi

FETCH cümlesinin, pozitif bir miktar ile **FETCH NEXT** veya **FETCH FORWARD** biçimleri dışında bir biçimle kullanılması düşünülüyorsa, gösterici **SCROLL** seçeneği ile bildirilmelidir. Basit sorgular için PostgreSQL, **SCROLL** ile bildirilmemiş göstericilerden geriye doğru satır çekilmesine izin verir, fakat bu davranışın en iyisi olduğuna güvenmeyin. Eğer gösterici **NO SCROLL** ile bildirilmişse geri yönde satır çekmeye izin verilmez.

ABSOLUTE ile satır çekimleri istenen satırı bir görelî hareketle çekmekten daha hızlı değildir: temel gerçeklenim aradaki satırların tümünü bir şekilde geçmelidir. Negatif mutlak çekimlerin durumu daha bile kötüdür: Sorgunun, son satırı bulana kadar tüm satırları okuması ve buradan, çekilecek satıra kadar olan satırları tekrar geçmesi gerekir. Yine de, sorgunun başa dönmesi hızlı olur (**FETCH ABSOLUTE 0** ile olduğu gibi).

Gösterici üzerinden güncelleme PostgreSQL tarafından şimdilik desteklenmemektedir.

Bir göstericiyi tanımlamak için **DECLARE** [*declare (7)*] kullanılır. Veri döndürmeksizin göstericiyi hareket ettirmek için **MOVE** [*move (7)*] kullanabilirsiniz.

Örnekler

Aşağıdaki örnekte bir tablo gösterici kullanılarak incelenmektedir:

```
BEGIN WORK;

-- Göstericiyi bildirelim:
DECLARE liahona SCROLL CURSOR FOR SELECT * FROM films;

-- liahona göstericisinden ilk 5 satırı çekelim:
FETCH FORWARD 5 FROM liahona;
```

code	title	did	date_prod	kind	len
BL101	The Third Man	101	1949-12-23	Drama	01:44
BL102	The African Queen	101	1951-08-11	Romantic	01:43
JL201	Une Femme est une Femme	102	1961-03-12	Romantic	01:25
P_301	Vertigo	103	1958-11-14	Action	02:08
P_302	Becket	103	1964-02-03	Drama	02:28

```
-- Önceki satırı çekelim:
FETCH PRIOR FROM liahona;
```

code	title	did	date_prod	kind	len
P_301	Vertigo	103	1958-11-14	Action	02:08

```
-- Göstericiyi kapatıp hareketi sonlandıralım:
CLOSE liahona;
COMMIT WORK;
```

Uyumluluk

SQL standardındaki **FETCH** deyimini sadece gömülü SQL'de kullanmak içindir. Burada açıklanan **FETCH** ise, veriyi bir konak değişkenine yerleştirilmiş olarak değil, bir **SELECT** sonucu gibi döndürür. Bunun dışında **FETCH**, SQL standardı ile tamamen uyumludur.

FORWARD ve **BACKWARD** biçimleriyle **FORWARD** örtük olma üzere **FETCH miktar** ve **FETCH ALL** biçimleri birer **PostgreSQL** oluşumdur.

SQL standardı **FROM** deyimine sadece ismin önünde izin verir; **IN** kullanımı bir **PostgreSQL** oluşumdur.

İlgili Belgeler

CLOSE [`close (7)`], **DECLARE** [`declare (7)`], **MOVE** [`move (7)`].

Çeviren

Nilgün Belma Bugüner <nilgun (at) belgeler-gen-tr>, Nisan 2005

YASAL UYARI

Bu çevirinin telif hakkı yukarıda belirtilen çevirmen(ler)e aittir. Özgün belgenin telif hakkı ve lisans bilgileri varsa ve belge içinde belirtilmemişse belge sonunda belirtilmiş olacaktır. Bu çevirinin lisansı, özgün belge için belirtilmiş bir lisans varsa ve bu lisans çevirinin de aynı lisansa sahip olmasını gerektiriyorsa onunla aynıdır, yoksa GNU GPL lisansı ve her iki durumda da ek olarak aşağıdaki koşullar geçerlidir. GNU GPL lisansı <<http://www.gnu.org/licenses/gpl.html>> adresinden edinilebilir.

BU BELGE ÜCRETSİZ OLARAK RUHSATLANDIĞI İÇİN, BELGENİN İÇERDİĞİ BİLGİLERİN VEYA KODLARIN NİTELİKLERİ İÇİN İLGİLİ KANUNLARIN İZİN VERDİĞİ ÖLÇÜDE HERHANGİ BİR GARANTİ VERİLMEMEKTEDİR. AKSİ YAZILI OLARAK BELİRTİLMEDİĞİ MÜDDETÇE TELİF HAKKI SAHİPLERİ VE/VEYA BAŞKA ŞAHISLAR BELGELERİ "OLDUĞU GİBİ", AŞIKAR VEYA ZIMNEN, SATILABİLİRLİĞİ VEYA HERHANGİ BİR AMACA UYGUNLUĞU DA DAHİL OLMAK ÜZERE HİÇBİR GARANTİ VERMEKSİZİN DAĞITMAKTADIRLAR. BELGELERİN KALİTESİ VEYA PERFORMANSI İLE İLGİLİ TÜM SORUNLAR SİZE AİTTİR. HERHANGİ BİR HATA VEYA EKSİKLİKTEN DOLAYI DOĞABİLECEK OLAN BÜTÜN SERVİS, TAMİR VEYA DÜZELTME MASRAFLARI SİZE AİTTİR.

İLGİLİ KANUNUN İCBAR ETTİĞİ DURUMLAR VEYA YAZILI ANLAŞMA HARİCİNDE HERHANGİ BİR ŞEKİLDE TELİF HAKKI SAHİBİ VEYA YUKARIDA İZİN VERİLDİĞİ ŞEKİLDE BELGEYİ DEĞİŞTİREN VEYA YENİDEN DAĞITAN HERHANGİ BİR KİŞİ, BELGENİN İÇERDİĞİ BİLGİNİN KULLANIMI VEYA KULLANILAMAMASI (VEYA VERİ KAYBI OLUŞMASI, VERİNİN YANLIŞ HALE GELMESİ, SİZİN VEYA ÜÇÜNCÜ ŞAHISLARIN ZARARA UĞRAMASI VEYA BİLGİNİN BAŞKA BİLGİLERLE UYUMSUZ OLMASI) YÜZÜNDEN OLUŞAN GENEL, ÖZEL, DOĞRUDAN YA DA DOLAYLI HERHANGİ BİR ZARARDAN, BÖYLE BİR TAZMİNAT TALEBİ TELİF HAKKI SAHİBİ VEYA İLGİLİ KİŞİYE BİLDİRİLMİŞ OLSA DAHİ, SORUMLU DEĞİLDİR.