



Zadatci za vježbu #12

[Previous step](#)[Submit answer](#)[Next step](#)[Layout ▼](#)[Tutorial content ▼](#)

4/4: Analiza plana izvršavanja upita u Postgresu pomoću pgAdmina - eksperimentalni dio

Obavljanje eksperimentalnog dijela namijenjeno je studentima koji žele znati više.

Zadatak je obaviti SELECT iz primjera 1 prethodnog koraka te proučiti plan izvršavanja upita kojeg je napravio optimizator.

Budući da smo nakon primjera 1 kreirali neke primarne i strane ključeve, da bismo imali situaciju kao u primjeru 1, uništit ćemo ta integritetska ograničenja naredbama:

```
ALTER TABLE ispitOp DROP CONSTRAINT JMBAGStudFk;  
ALTER TABLE studentOp DROP CONSTRAINT pbrRodFk;  
ALTER TABLE studentOp DROP CONSTRAINT studOpPk;  
ALTER TABLE mjestoOp DROP CONSTRAINT mjestoOpPk;
```

Ispis plana upita dobije se uključivanjem naredbe EXPLAIN ANALYZE VERBOSE ispred upita za kojeg se traži plan izvršavanja.

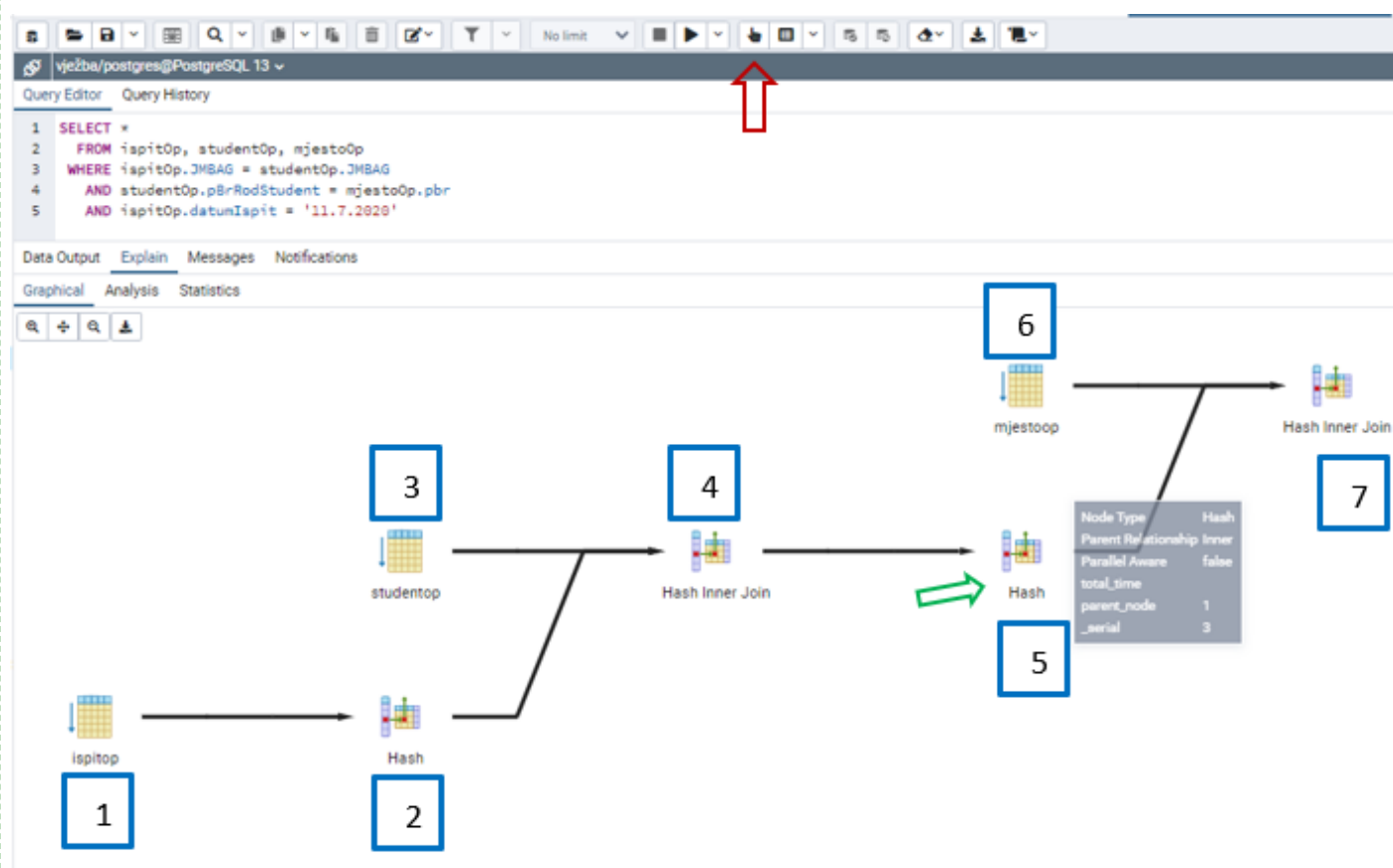
Primjer 1.

Obavite SELECT naredbu iz primjera 1 prethodnog koraka i proučite plan izvršavanja:

```
SELECT *  
FROM ispitOp, studentOp, mjestoOp  
WHERE ispitOp.JMBAG = studentOp.JMBAG  
AND studentOp.pBrRodStudent = mjestoOp.pbr  
AND ispitOp.datumIspit = '11.7.2020';
```

rješenje

Gornji upit prenesite (copy+paste) u pgAdmin i pokrenite opciju Explain (na slici označena crvenom strelicom). Nakon što je izvršavanje upita završeno, u kartici Explain-Graphical može se vidjeti stablo plana upita.



Kao što je naglašeno na predavanjima, redoslijed plana treba interpretirati od listova prema korijenu stabla. Svaki čvor predstavlja operaciju, to jest korak izvršavanja upita.

Čvorovi su povezani usmjerenim bridovima (strelicama), koji označavaju tok plana. Na dobivenom stablu prikazani su čvorovi, a naknadno je ucrtan redoslijed izvršavanja koraka plana upita. Ako se u pgAdmin 4 administracijskom alatu mišem pozicionirate iznad jednog od čvorova u planu upita, dobivate detalje o tom koraku plana. Na gornjoj slici to je napravljeno za korak označen brojem 5 i zelenom strelicom.

Detaljni tekstualni ispis plana upita dobiva se na sljedeći način:

[Feedback](#)

1. Otvorite alat za upite (query tool) nad bazom podataka u kojoj planirate izvršiti traženi upit. U našem slučaju to je baza podataka studAdmin. Kopirajte upit u alat za rad s upitima i dodajte EXPLAIN ANALYZE VERBOSE ispred upita.
2. Odaberite opciju Execute/Refresh.

Kao rezultat izvršavanja takvog upita dobije se detaljni tekstualni ispis plana upita:

	QUERY PLAN text
1	Hash Join (cost=21.95..45.71 rows=1 width=145) (actual time=0.200..0.328 rows=1 loops=1)
2	Output: ispitop.jmbag, ispitop.sifpred, ispitop.sifnastavnik, ispitop.datumispit, ispitop.ocjena, studentop.jmbag, studentop.imestudent, studentop.prezimestudent, studentop.pbrrodstudent, studentop.pbrstanstudent, studentop.datumrod, mjestoop.pbr, mjestoop.nazmjesto
3	Hash Cond: (mjestoop.pbr = studentop.pbrrodstudent)
4	-> Seq Scan on public.mjestoop (cost=0.00..20.00 rows=1000 width=45) (actual time=0.024..0.088 rows=1000 loops=1)
5	Output: mjestoop.pbr, mjestoop.nazmjesto
6	-> Hash (cost=21.93..21.93 rows=1 width=100) (actual time=0.151..0.152 rows=1 loops=1)
7	Output: ispitop.jmbag, ispitop.sifpred, ispitop.sifnastavnik, ispitop.datumispit, ispitop.ocjena, studentop.jmbag, studentop.imestudent, studentop.prezimestudent, studentop.pbrrodstudent, studentop.pbrstanstudent, studentop.datumrod
8	Buckets: 1024 Batches: 1 Memory Usage: 9kB
9	-> Hash Join (cost=20.51..21.93 rows=1 width=100) (actual time=0.144..0.146 rows=1 loops=1)
10	Output: ispitop.jmbag, ispitop.sifpred, ispitop.sifnastavnik, ispitop.datumispit, ispitop.ocjena, studentop.jmbag, studentop.imestudent, studentop.prezimestudent, studentop.pbrrodstudent, studentop.pbrstanstudent, studentop.datumrod
11	Hash Cond: (studentop.jmbag = ispitop.jmbag)
12	-> Seq Scan on public.studentop (cost=0.00..1.30 rows=30 width=75) (actual time=0.014..0.016 rows=30 loops=1)
13	Output: studentop.jmbag, studentop.imestudent, studentop.prezimestudent, studentop.pbrrodstudent, studentop.pbrstanstudent, studentop.datumrod
14	-> Hash (cost=20.50..20.50 rows=1 width=25) (actual time=0.114..0.114 rows=1 loops=1)
15	Output: ispitop.jmbag, ispitop.sifpred, ispitop.sifnastavnik, ispitop.datumispit, ispitop.ocjena
16	Buckets: 1024 Batches: 1 Memory Usage: 9kB
17	-> Seq Scan on public.ispitop (cost=0.00..20.50 rows=1 width=25) (actual time=0.103..0.103 rows=1 loops=1)
18	Output: ispitop.jmbag, ispitop.sifpred, ispitop.sifnastavnik, ispitop.datumispit, ispitop.ocjena
19	Filter: (ispitop.datumispit = '11.07.2020'::date)
20	Rows Removed by Filter: 999
21	Planning Time: 0.318 ms
22	Execution Time: 0.374 ms

U donjem prikazu detaljni plan je označen koracima kao i kod stabla plana izvršavanja. Čitajte ga od dolje prema gore tj. od koraka 1 prema koraku 7.

Korak 7:

- Hash Join – Raspršeno spajanje
- (cost=21.95..45.71 rows=1 width=145) – Predviđeni trošak spajanja je 45.71 apstraktnih jedinica (cost), 1 n-torka (rows), 145 byte-a po n-torki (rows) – Ovako visok trošak predviđen je zbog pročitanoog broja n-torki u koraku 6
- (actual time=0.200..0.328 rows=1 loops=1) – Stvarno trajanje je 0.328 ms, rezultat spajanja je 1 n-torki (rows) u jednom koraku (loops)
- Output: ispitop.jmbag, ispitop.sifpred, ... – Izlazna lista atributa
- Hash Cond: (mjestoop.pbr = studentop.pbrrodstudent) – Uvjet raspršenog spajanja

Hash Join (cost=21.95..45.71 rows=1 width=145) (actual time=0.200..0.328 rows=1 loops=1)
Output: ispitop.jmbag, ispitop.sifpred, ispitop.sifnastavnik, ispitop.datumispit, ispitop.ocjena, studentop.jmbag, studentop.imestudent, studentop.prezimestudent, studentop.pbrrodstudent, studentop.pbrstanstudent, studentop.datumrod, mjestoop.pbr, mjestoop.nazmjesto
Hash Cond: (mjestoop.pbr = studentop.pbrrodstudent)

Korak 6:

- Seq Scan on public.mjestoop – Relacija mjestoop čita se sekvencijski
- (cost=0.00..20.00 rows=1000 width=45) – Predviđeni trošak je 20.00 apstraktnih jedinica (cost), 1000 n-torki (rows), 45 byte-a po n-torki (rows)
- (actual time=0.024..0.088 rows=1000 loops=1) – Stvarno trajanje je 0.088 ms (actual time), pročitano je 1000 n-torki (rows) u jednom koraku (loops)
- Output: mjestoop.pbr, mjestoop.nazmjesto – Izlazna lista atributa

-> Seq Scan on public.mjestoop (cost=0.00..20.00 rows=1000 width=45) (actual time=0.024..0.088 rows=1000 loops=1)
Output: mjestoop.pbr, mjestoop.nazmjesto

Korak 5:

- Hash – Stvaranje raspršene tablice u primarnoj memoriji DBMS-a iz rezultata raspršenog spajanja u koraku 4
- (cost=21.93..21.93 rows=1 width=100) – Predviđeni trošak je 21.93 apstraktnih jedinica (cost), 1 n-torka (rows), 100 byte-a po n-torki (rows)
- (actual time=0.151..0.152 rows=1 loops=1) – Stvarno trajanje je 0.152 ms (actual time), spremljena je jedna n-torka (rows) u jednom koraku (loops) – računajte da stvaranje raspršene tablice uključuje i spajanje iz koraka 4
- Output: ispitop.jmbag, ispitop.sifpred, ... – Izlazna lista atributa

- Buckets: 1024 Batches: 1 Memory Usage: 9kB – 1024 inicijalnih džepova (buckets), koristi se samo primarna memorija (batches=1), koristi se 9kB memorije (memory usage)

```
-> Hash (cost=21.93..21.93 rows=1 width=100) (actual time=0.151..0.152 rows=1 loops=1)
      Output: ispitop.jmbag, ispitop.sifpred, ispitop.sifnastavnik, ispitop.datumispit, ispitop.ocjena, studentop.jmbag,
studentop.imestudent, studentop.prezimestudent, studentop.pbrrodstudent, studentop.pbrstanstudent, studentop.datumrod
      Buckets: 1024 Batches: 1 Memory Usage: 9kB
```

Korak 4:

- Hash Join – Raspršeno spajanje
- (cost=20.51..21.93 rows=1 width=100) – Predviđeni trošak spajanja je 21.93 apstraktnih jedinica (cost), 1 n-torka (rows), 100 byte-a po n-torki (rows)
- (actual time=0.144..0.146 rows=1 loops=1) – Stvarno trajanje je 0.146 ms (actual time), rezultat spajanja je jedna n-torka (rows) u jednom koraku (loops)
- Output: ispitop.jmbag, ispitop.sifpred, ... – Izlazna lista atributa
- Hash Cond: (studentop.jmbag = ispitop.jmbag) – Uvjet raspršenog spajanja

```
-> Hash Join (cost=20.51..21.93 rows=1 width=100) (actual time=0.144..0.146 rows=1 loops=1)
      Output: ispitop.jmbag, ispitop.sifpred, ispitop.sifnastavnik, ispitop.datumispit, ispitop.ocjena, studentop.jmbag,
studentop.imestudent, studentop.prezimestudent, studentop.pbrrodstudent, studentop.pbrstanstudent, studentop.datumrod
      Hash Cond: (studentop.jmbag = ispitop.jmbag)
```

Korak 3:

- Seq Scan on public.studentop – Relacija studentop čita se sekvencijski
- (cost=0.00..1.30 rows=30 width=75) – Predviđeni trošak je 1.30 apstraktnih jedinica (cost), 30 n-torki (rows), 75 byte-a po n-torki (rows)
- (actual time=0.014..0.016 rows=30 loops=1) – Stvarno trajanje je 0.016 ms (actual time), pročitano je 30 n-torki (rows) u jednom koraku (loops)
- Output: studentop.jmbag, studentop.imestudent, ... – Izlazna lista atributa

```
-> Seq Scan on public.studentop (cost=0.00..1.30 rows=30 width=75) (actual time=0.014..0.016 rows=30 loops=1)
      Output: studentop.jmbag, studentop.imestudent, studentop.prezimestudent, studentop.pbrrodstudent,
studentop.pbrstanstudent, studentop.datumrod
```

Korak 2:

- Hash – Stvaranje raspršene tablice u primarnoj memoriji DBMS-a iz rezultata slijednog čitanja u koraku 1
- (cost=20.50..20.50 rows=1 width=25) – Predviđeni trošak je 20.50 apstraktnih jedinica (cost), 1 n-torka (rows), 25 byte-a po n-torki (rows)
- (actual time=0.114..0.114 rows=1 loops=1) – Stvarno trajanje je 0.114 ms, spremljena je 1 n-torka (rows) u jednom koraku (loops) – računajte da stvaranje raspršene tablice uključuje i jedno čitanje iz koraka 1
- Output: ispitop.jmbag, ispitop.sifpred, ... – Izlazna lista atributa
- Buckets: 1024 Batches: 1 Memory Usage: 9kB – 1024 inicijalnih džepova (buckets), koristi se samo primarna memorija (batches=1), koristi se 9kB memorije (memory usage)

```
-> Hash (cost=20.50..20.50 rows=1 width=25) (actual time=0.114..0.114 rows=1 loops=1)
      Output: ispitop.jmbag, ispitop.sifpred, ispitop.sifnastavnik, ispitop.datumispit, ispitop.ocjena
      Buckets: 1024 Batches: 1 Memory Usage: 9kB
```

Korak 1:

- Budući da nismo stvorili primarne ključeve niti nad jednom relacijom, a time niti jedan indeks, za očekivati je da se sve relacije čitati slijedno (sequential scan)
- Seq Scan on public.ispitop – Relacija ispitop čita se slijedno
- (cost=0.00..20.50 rows=1 width=25) – Predviđeni trošak je 20.50 apstraktnih jedinica (cost), 1 n-torka (rows), 25 byte-a po n-torki (rows)
- (actual time=0.103..0.103 rows=1 loops=1) – Stvarno trajanje je 0.103 ms, pročitana je jedna n-torka (rows) u jednom koraku (loops)
- Output: ispitop.jmbag, ispitop.sifpred, ... – Izlazna lista atributa
- Filter: (ispitop.datumispit = '11.07.2011'::date) – Uvjet selekcije
- Rows Removed by Filter: 999 – Broj eliminiranih n-torki primjenom uvjeta selekcije

-> Seq Scan on public.ispitop (cost=0.00..20.50 rows=1 width=25) (actual time=0.103..0.103 rows=1 loops=1)
Output: ispitop.jmbag, ispitop.sifpred, ispitop.sifnastavnik, ispitop.datumispit, ispitop.ocjena
Filter: (ispitop.datumispit = '11.07.2020'::date)
Rows Removed by Filter: 999

Planirano vrijeme izvršavanja upita je 0.318 ms, a stvarno vrijeme je 0.374 ms.

Planning Time: 0.318 ms
Execution Time: 0.374 ms

Question

Playground

Napišite SQL upit kojim biste procijenili broj n-torki (u rezultatu sljedećeg upita u skladu s onim što smo učili na predavanjima):

```
SELECT *  
FROM ispit  
WHERE datumIspit = '16.02.2021'
```

Koliko n-torki vraća gornji upit? Je li procjena točna?

rješenje

```
SELECT (SELECT COUNT(*) FROM ispit)*1.0/  
      (SELECT COUNT(DISTINCT datumIspit) FROM ispit);
```

Upit vraća 0 n-torki. Procjena nije točna.

1

2

3

4

5

6

Run

Save