

# Rekurzivni SQL

Stefan Kapunac

Univerzitet u Beogradu, Matematički fakultet

26.3.2020.

# Pregled

1 Uvod

2 Zadaci

# Rekurzija

- Neki podaci su prirodno rekurzivni
- Kao u primeru 1



# Primer 1

- deo - šifra dela
- poddeo - šifra poddela potrebnog za deo
- količina - potrebna količina poddela

```
create table delovi(  
    deo varchar(10),  
    poddeo varchar(10),  
    kolicina integer  
);
```

# Primer 1

- Dodavanje podataka u tabelu

```
insert into delovi values ('00', '01', 5);
insert into delovi values ('00', '05', 3);
insert into delovi values ('01', '02', 2);
insert into delovi values ('01', '03', 3);
insert into delovi values ('01', '04', 4);
insert into delovi values ('01', '06', 3);
insert into delovi values ('02', '05', 7);
insert into delovi values ('02', '06', 6);
insert into delovi values ('03', '07', 6);
insert into delovi values ('04', '08', 10);
insert into delovi values ('04', '09', 11);
insert into delovi values ('05', '10', 10);
insert into delovi values ('05', '11', 10);
insert into delovi values ('06', '12', 10);
insert into delovi values ('06', '13', 10);
insert into delovi values ('07', '14', 8);
insert into delovi values ('07', '12', 8);
```

## Primer 1a

- Koji delovi su potrebni da se napravi deo 01?

```
with rek (deo, poddeo, kolicina) as
  (select deo, poddeo, kolicina
   from delovi
   where deo = '01'
   union all
   select b.deo, b.poddeo, b.kolicina
   from rek a, delovi b
   where a.poddeo = b.deo
  )
select distinct *
from rek
order by deo, poddeo, kolicina;
```

# Rezultat izvršavanja

deo	poddeo	količina
01	02	2
01	03	3
01	04	4
01	06	3
02	05	7
02	06	6
03	07	6
04	08	10
04	09	11
05	10	10
05	11	10
06	12	10
06	13	10
07	12	8
07	14	8



## Primer 1b

- Koja je ukupna količina svakog dela potrebnog za deo 01?

```
with rek (deo, poddeo, kolicina) as
  (select deo, poddeo, kolicina
   from delovi
   where deo = '01'
   union all
   select a.deo, b.poddeo, a.kolicina * b.kolicina
   from rek a, delovi b
   where a.poddeo = b.deo
  )
select deo, poddeo, sum(kolicina) as 'ukupna kolicina'
from rek
group by deo, poddeo
order by deo, poddeo;
```

## Rezultat izvršavanja

deo	poddeo	ukupna količina
01	02	2
01	03	3
01	04	4
01	05	14
01	06	15
01	07	18
01	08	40
01	09	44
01	10	140
01	11	140
01	12	294
01	13	150
01	14	144

## Primer 1c

- Koja su prva dva nivoa delova potrebnih za deo 01?

```
with rek (deo, poddeo, kolicina, nivo) as
  (select deo, poddeo, kolicina, 1
   from delovi
   where deo = '01'
   union all
   select b.deo, b.poddeo, b.kolicina, a.nivo + 1
   from rek a, delovi b
   where a.poddeo = b.deo
         and a.nivo < 2
  )
select deo, poddeo, kolicina, nivo
from rek;
```

# Rezultat izvršavanja

deo	poddeo	količina	nivo
01	02	2	1
01	03	3	1
01	04	4	1
01	06	3	1
02	05	7	2
02	06	6	2
03	07	6	2
04	08	10	2
04	09	11	2
06	12	10	2
06	13	10	2

## Primer 2

- id - identifikator
- pocetak - polazni aerodrom
- kraj - dolazni aerodrom
- cena - cena karte

```
CREATE TABLE letovi(  
  id integer PRIMARY KEY,  
  pocetak varchar(255),  
  kraj varchar(255),  
  cena integer);
```

## Primer 2

- Dodavanje podataka u tabelu

```
INSERT INTO letovi VALUES(1, 'Beograd', 'Bec', 300);
INSERT INTO letovi VALUES(2, 'Beograd', 'Berlin', 150);
INSERT INTO letovi VALUES(3, 'Bec', 'Dubai', 500);
INSERT INTO letovi VALUES(4, 'Berlin', 'Doha', 600);
INSERT INTO letovi VALUES(5, 'Berlin', 'Peking', 700);
INSERT INTO letovi VALUES(6, 'Dubai', 'Hanoj', 1000);
INSERT INTO letovi VALUES(7, 'Doha', 'Tokio', 100);
INSERT INTO letovi VALUES(8, 'Tokio', 'Peking', 300);
INSERT INTO letovi VALUES(9, 'Peking', 'Hanoj', 10);
INSERT INTO letovi VALUES(10, 'Hanoj', 'Sidnej', 200);
```

## Primer 2a

- Pronaći sva mesta u koja se može stići iz Beograda, uz moguća presedanja

```
with destinacije (polaziste , pocetak , kraj) as
(select pocetak , pocetak , kraj
 from letovi
 where pocetak = 'Beograd'
 union all
 select a.polaziste , b.pocetak , b.kraj
 from destinacije a, letovi b
 where a.kraj = b.pocetak
 )
select *
from destinacije;
```

# Rezultat izvršavanja

polazište	početak	kraj
Beograd	Beograd	Bec
Beograd	Beograd	Berlin
Beograd	Bec	Dubai
Beograd	Berlin	Doha
Beograd	Berlin	Peking
Beograd	Dubai	Hanoj
Beograd	Doha	Tokio
Beograd	Peking	Hanoj
Beograd	Hanoj	Sidnej
Beograd	Tokio	Peking
Beograd	Hanoj	Sidnej
Beograd	Peking	Hanoj
Beograd	Hanoj	Sidnej



## Primer 2b

- U prethodnom rezultatu par Hanoj, Sidnej se pojavljuje 3 puta
- Da bismo takva ponavljanja izbegli, dodajemo `distinct`

```
with destinacije (polaziste , pocetak , kraj) as
(select pocetak , pocetak , kraj
 from letovi
 where pocetak = 'Beograd'
 union all
 select a.polaziste , b.pocetak , b.kraj
 from destinacije a, letovi b
 where a.kraj = b.pocetak
 )
select distinct *
from destinacije;
```

# Rezultat izvršavanja

polazište	početak	kraj
Beograd	Beograd	Bec
Beograd	Beograd	Berlin
Beograd	Bec	Dubai
Beograd	Berlin	Doha
Beograd	Berlin	Peking
Beograd	Dubai	Hanoj
Beograd	Doha	Tokio
Beograd	Peking	Hanoj
Beograd	Hanoj	Sidnej
Beograd	Tokio	Peking

## Primer 2c

- Pronaći sva mesta u koja se može stići iz Beograda, uz moguća presedanja
- Prebrojati presedanja

```
with destinacije (polaziste , pocetak , kraj ,
                  presedanja) as
(select pocetak , pocetak , kraj , 0
 from letovi
 where pocetak = 'Beograd'
 union all
 select a.polaziste , b.pocetak , b.kraj ,
        a.presedanja + 1
 from destinacije a, letovi b
 where a.kraj = b.pocetak
 )
select *
from destinacije;
```

## Rezultat izvršavanja

polazište	početak	kraj	presedanja
Beograd	Beograd	Bec	0
Beograd	Beograd	Berlin	0
Beograd	Bec	Dubai	1
Beograd	Berlin	Doha	1
Beograd	Berlin	Peking	1
Beograd	Dubai	Hanoj	2
Beograd	Doha	Tokio	2
Beograd	Peking	Hanoj	2
Beograd	Hanoj	Sidnej	3
Beograd	Tokio	Peking	3
Beograd	Hanoj	Sidnej	3
Beograd	Peking	Hanoj	4
Beograd	Hanoj	Sidnej	5

## Primer 2d

- Pronaći sva mesta u koja se može stići iz Beograda, uz moguća presedanja
- Prebrojati presedanja i izračunati ukupnu cenu

```
with destinacije ( polaziste , pocetak , kraj ,
                    presedanja , cena ) as
( select pocetak , pocetak , kraj , 0 , cena
  from letovi
  where pocetak = 'Beograd'
  union all
  select a.polaziste , b.pocetak , b.kraj ,
         a.presedanja + 1 , a.cena + b.cena
  from destinacije a , letovi b
  where a.kraj = b.pocetak
)
select *
from destinacije ;
```

## Rezultat izvršavanja

polazište	početak	kraj	presedanja	cena
Beograd	Beograd	Bec	0	300
Beograd	Beograd	Berlin	0	150
Beograd	Bec	Dubai	1	800
Beograd	Berlin	Doha	1	750
Beograd	Berlin	Peking	1	850
Beograd	Dubai	Hanoj	2	1800
Beograd	Doha	Tokio	2	850
Beograd	Peking	Hanoj	2	860
Beograd	Hanoj	Sidnej	3	2000
Beograd	Tokio	Peking	3	1150
Beograd	Hanoj	Sidnej	3	1060
Beograd	Peking	Hanoj	4	1160
Beograd	Hanoj	Sidnej	5	1360

## Primer 2e

- Pronaći sve načine na koje se može stići iz Beograda do Hanoja, tako da broj presedanja bude manji od 3

```
with rez (ruta , pocetak , kraj , presedanja , cena) as
(select pocetak || '-' || kraj , pocetak , kraj , 0, cena
 from letovi
 where pocetak = 'Beograd'
 union all
 select a.ruta || '-' || b.kraj , b.pocetak ,
        b.kraj , a.presedanja + 1, a.cena + b.cena
 from rez a, letovi b
 where a.kraj = b.pocetak
        and a.presedanja < 2
        and b.pocetak <> 'Hanoj'
 )
select ruta , presedanja , cena
from rez
where ruta like '%Hanoj%';
```

# Rezultat izvršavanja

ruta	presedanja	cena
Beograd-Bec-Dubai-Hanoj	2	1800
Beograd-Berlin-Peking-Hanoj	2	860



## Primer 2f

- Pronaći sve načine na koje se može stići iz Beograda do Hanoja, tako da broj presedanja bude manji od 5

```
with rez (ruta , pocetak , kraj , presedanja , cena) as
(select pocetak || '-' || kraj , pocetak , kraj , 0, cena
 from letovi
 where pocetak = 'Beograd'
 union all
 select a.ruta || '-' || b.kraj , b.pocetak ,
        b.kraj , a.presedanja + 1, a.cena + b.cena
 from rez a, letovi b
 where a.kraj = b.pocetak
        and a.presedanja < 4
        and b.pocetak <> 'Hanoj'
 )
select ruta , presedanja , cena
from rez
where ruta like '%Hanoj%';
```

# Rezultat izvršavanja

ruta	presedanja	cena
Beograd-Bec-Dubai-Hanoj	2	1800
Beograd-Berlin-Peking-Hanoj	2	860
Beograd-Berlin-Doha-Tokio-Peking-Hanoj	4	1160