Rekursive Abfragen

Prof. Dr. Ingo Claßen

Mittwoch, 9. März 2022 08:09

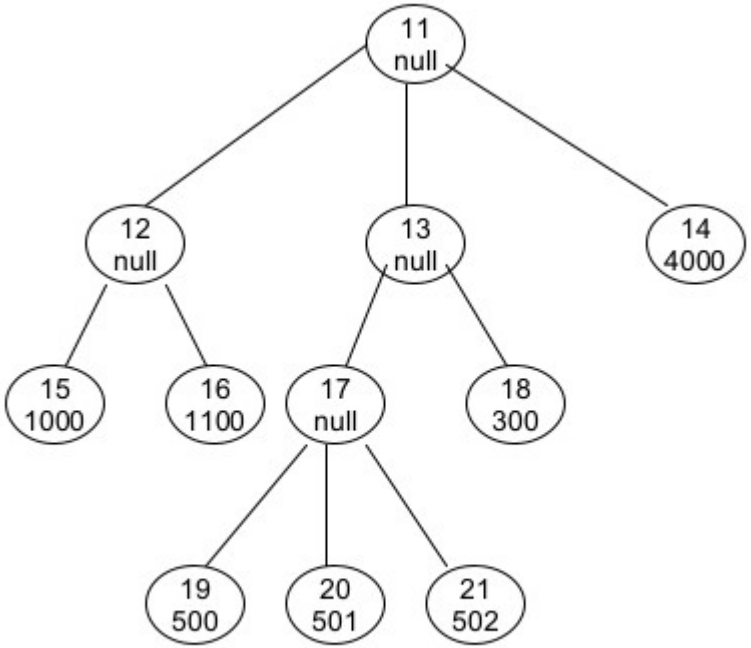
- Allgemeine Struktur
- Daten
- Abfragen

```
with recursive t(n) as (
    values (1)
  union all
    select n+1 from t where n < 100
)
select sum(n) from t;
```

123 sum
5.050

Tabelle tree0
p: Parent
v: Value

123 id	123 p	123 v
11	[NULL]	[NULL]
12	11	[NULL]
13	11	[NULL]
14	11	4.000
15	12	1.000
16	12	1.100
17	13	[NULL]
18	13	300
19	17	500
20	17	501
21	17	502



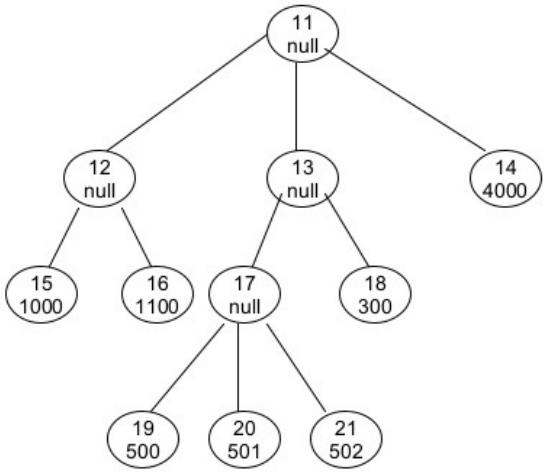
Werte gibt es nur bei den Blättern

Rekursive Abfrage

Prof. Dr. Ingo Claßen

Mittwoch, 9. März 2022 09:22

```
with recursive
  tree1(id, p, v, lvl, pth) as (
      select id, p, v, 1, cast(id as varchar(200))
      from tree0
      where p is null
    union all
      select t0.id, t0.p, t0.v, lvl+1, cast(pth || '/' || t0.id as varchar(200))
      from tree0 t0 join tree1 t1 on t0.p=t1.id
  ),
  tree2(id, p, v, lvl, pth, kind) as (
    select
      id, p, v, lvl, pth,
      case
        when lvl=(select min(lvl) from tree1) then 'root'
        when (not exists (select * from tree1 below where below.p =tree1.id)) then 'leaf'
        else 'inner'
      end
    from tree1
  )
select id, p, v, lvl, pth, kind
from tree2
order by lvl
;
```



123 id	123 p	123 v	123 lvl	ABC pth	ABC kind
11	[NULL]	[NULL]	1	11	root
12	11	[NULL]	2	11/12	inner
13	11	[NULL]	2	11/13	inner
14	11	4.000	2	11/14	leaf
15	12	1.000	3	11/12/15	leaf
16	12	1.100	3	11/12/16	leaf
17	13	[NULL]	3	11/13/17	inner
18	13	300	3	11/13/18	leaf
19	17	500	4	11/13/17/19	leaf
20	17	501	4	11/13/17/20	leaf
21	17	502	4	11/13/17/21	leaf