Rekurzia v SQL

SQL a rekurzia

- Majme databázu: mesto(nazov), cesta(odkial, kam)
- WITH RECURSIVE dosiahnutelne_ba AS (

```
        SELECT kam FROM WHERE odkial=„bratislava“
        UNION
        SELECT c.kam
        FROM
                dosiahnutelne_ba as d
                        JOIN cesta as c on d.kam=c.odkial
)
SELECT * FROM dosiahnutelne_ba;
```

Recursive Query Evaluation

- Evaluate the non-recursive term. For UNION (but not UNION ALL), discard duplicate rows. Include all remaining rows in the result of the recursive query, and also place them in a temporary working table.
- So long as the working table is not empty, repeat these steps:
  - Evaluate the recursive term, substituting the current contents of the working table for the recursive self-reference. For UNION (but not UNION ALL), discard duplicate rows and rows that duplicate any previous result row. Include all remaining rows in the result of the recursive query, and also place them in a temporary intermediate table.
  - Replace the contents of the working table with the contents of the intermediate table, then empty the intermediate table.
- https://www.postgresql.org/docs/current/queries-with.html

Example

```
-- sum of 1..100
WITH RECURSIVE t(n) AS (
    VALUES (1)
UNION ALL
    SELECT n+1 FROM t WHERE n < 100
)
SELECT sum(n) FROM t;
```

Example

```
-- This is an infinite loop with UNION ALL, but not with UNION
WITH RECURSIVE t(n) AS (
    SELECT 1
UNION
    SELECT 10-n FROM t)
SELECT * FROM t ORDER BY n;
```

Example

```
-- This'd be an infinite loop, but outside query reads only as much as
        needed
WITH RECURSIVE t(n) AS (
    VALUES (1)
UNION ALL
    SELECT n+1 FROM t)
SELECT * FROM t LIMIT 10;
```