```
postgres=# EXPLAIN ANALYZE
SELECT
    v.vasarlo_nev,
    v.varos,
    r.rendeles_datum,
    k.konyv_cim,
    k.ar,
    r.rendeles_allapot,
    s.osszeg
FROM
    Vasarlok v
JOIN
    Rendelesek r ON v.vasarlo_id = r.vasarlo_id AND v.varos = r.varos
JOIN
    Tetelek t ON r.rendeles_id = t.rendeles_id
JOIN
    Konyvek k ON t.konyv_id = k.konyv_id
JOIN
    Szamlak s ON r.rendeles_id = s.rendeles_id
WHERE
    r.rendeles_allapot = 'Teljesitett'
ORDER BY
    r.rendeles_datum DESC;
                                                                  QUERY PLAN
-----------------------------------------------------------------------------------------------------------------------------------------------------
 Sort  (cost=639.75..639.76 rows=3 width=54) (actual time=16.137..16.251 rows=2502 loops=1)
   Sort Key: r.rendeles_datum DESC
   Sort Method: quicksort  Memory: 273kB
   ->  Nested Loop  (cost=285.85..639.72 rows=3 width=54) (actual time=4.402..13.593 rows=2502 loops=1)
         ->  Nested Loop  (cost=285.56..638.56 rows=3 width=42) (actual time=4.388..10.474 rows=2502 loops=1)
               ->  Hash Join  (cost=285.28..637.35 rows=3 width=46) (actual time=4.353..7.207 rows=2502 loops=1)
                     Hash Cond: ((v.vasarlo_id = r.vasarlo_id) AND ((v.varos)::text = (r.varos)::text))
                     ->  Append  (cost=0.00..277.03 rows=10002 width=29) (actual time=0.013..2.408 rows=10002 loops=1)
                           ->  Seq Scan on vasarlok_balassagyarmat v_1  (cost=0.00..59.00 rows=2500 width=34) (actual time=0.013..0.910 rows=2500 loops=1)
                           ->  Seq Scan on vasarlok_budapest v_2  (cost=0.00..56.01 rows=2501 width=28) (actual time=0.013..0.234 rows=2501 loops=1)
                           ->  Seq Scan on vasarlok_debrecen v_3  (cost=0.00..56.01 rows=2501 width=28) (actual time=0.005..0.202 rows=2501 loops=1)
                           ->  Seq Scan on vasarlok_default v_4  (cost=0.00..56.00 rows=2500 width=25) (actual time=0.003..0.592 rows=2500 loops=1)
                     ->  Hash  (cost=275.89..275.89 rows=626 width=34) (actual time=2.945..2.947 rows=2502 loops=1)
                           Buckets: 4096 (originally 1024)  Batches: 1 (originally 1)  Memory Usage: 199kB
                           ->  Hash Join  (cost=228.30..275.89 rows=626 width=34) (actual time=1.326..2.274 rows=2502 loops=1)
                                 Hash Cond: (s.rendeles_id = r.rendeles_id)
                                 ->  Seq Scan on szamlak s  (cost=0.00..41.02 rows=2502 width=9) (actual time=0.004..0.136 rows=2502 loops=1)
                                 ->  Hash  (cost=197.03..197.03 rows=2502 width=25) (actual time=1.308..1.309 rows=2502 loops=1)
                                       Buckets: 4096  Batches: 1  Memory Usage: 169kB
                                       ->  Seq Scan on rendelesek r  (cost=0.00..197.03 rows=2502 width=25) (actual time=0.007..0.894 rows=2502 loops=1)
                                             Filter: (rendeles_allapot = 'Teljesitett'::rendeles_allapot_enum)
                                             Rows Removed by Filter: 7500
               ->  Index Only Scan using tetelek_pkey on tetelek t  (cost=0.29..0.39 rows=1 width=8) (actual time=0.001..0.001 rows=1 loops=2502)
                     Index Cond: (rendeles_id = r.rendeles_id)
                     Heap Fetches: 0
         ->  Index Scan using konyvek_pkey on konyvek k  (cost=0.29..0.39 rows=1 width=20) (actual time=0.001..0.001 rows=1 loops=2502)
               Index Cond: (konyv_id = t.konyv_id)
 Planning Time: 1.050 ms
 Execution Time: 16.409 ms
(29 rows)
```

A fenti képen látható az indexelés előtti EXPLAIN ANALYZE. Lekérdezésemben több táblát is joinoltam.

```
postgres=# CREATE INDEX idx_rendelesek_teljesitett ON Rendelesek(rendeles_id, vasarlo_id, varos, rendeles_allapot)
WHERE rendeles_allapot = 'Teljesitett';
CREATE INDEX
postgres=# EXPLAIN ANALYZE
SELECT
    v.vasarlo_nev,
    v.varos,
    r.rendeles_datum,
    k.konyv_cim,
    k.ar,
    r.rendeles_allapot,
    s.osszeg
FROM
    Vasarlok v
JOIN
    Rendelesek r ON v.vasarlo_id = r.vasarlo_id AND v.varos = r.varos
JOIN
    Tetelek t ON r.rendeles_id = t.rendeles_id
JOIN
    Konyvek k ON t.konyv_id = k.konyv_id
JOIN
    Szamlak s ON r.rendeles_id = s.rendeles_id
WHERE
    r.rendeles_allapot = 'Teljesitett'
ORDER BY
    r.rendeles_datum DESC;
                                                                  QUERY PLAN
-----------------------------------------------------------------------------------------------------------------------------------------------------
 Sort  (cost=619.41..619.42 rows=3 width=54) (actual time=11.401..11.497 rows=2502 loops=1)
   Sort Key: r.rendeles_datum DESC
   Sort Method: quicksort  Memory: 273kB
   ->  Nested Loop  (cost=265.51..619.39 rows=3 width=54) (actual time=3.651..10.705 rows=2502 loops=1)
         ->  Nested Loop  (cost=265.23..618.23 rows=3 width=42) (actual time=3.638..8.220 rows=2502 loops=1)
               ->  Hash Join  (cost=264.94..617.02 rows=3 width=46) (actual time=3.617..5.769 rows=2502 loops=1)
                     Hash Cond: ((v.vasarlo_id = r.vasarlo_id) AND ((v.varos)::text = (r.varos)::text))
                     ->  Append  (cost=0.00..277.03 rows=10002 width=29) (actual time=0.031..1.397 rows=10002 loops=1)
                           ->  Seq Scan on vasarlok_balassagyarmat v_1  (cost=0.00..59.00 rows=2500 width=34) (actual time=0.029..0.247 rows=2500 loops=1)
                           ->  Seq Scan on vasarlok_budapest v_2  (cost=0.00..56.01 rows=2501 width=28) (actual time=0.011..0.207 rows=2501 loops=1)
                           ->  Seq Scan on vasarlok_debrecen v_3  (cost=0.00..56.01 rows=2501 width=28) (actual time=0.003..0.214 rows=2501 loops=1)
                           ->  Seq Scan on vasarlok_default v_4  (cost=0.00..56.00 rows=2500 width=25) (actual time=0.004..0.258 rows=2500 loops=1)
                     ->  Hash  (cost=255.55..255.55 rows=626 width=34) (actual time=3.020..3.023 rows=2502 loops=1)
                           Buckets: 4096 (originally 1024)  Batches: 1 (originally 1)  Memory Usage: 199kB
                           ->  Hash Join  (cost=207.97..255.55 rows=626 width=34) (actual time=1.758..2.472 rows=2502 loops=1)
                                 Hash Cond: (s.rendeles_id = r.rendeles_id)
                                 ->  Seq Scan on szamlak s  (cost=0.00..41.02 rows=2502 width=9) (actual time=0.012..0.141 rows=2502 loops=1)
                                 ->  Hash  (cost=176.69..176.69 rows=2502 width=25) (actual time=1.735..1.736 rows=2502 loops=1)
                                       Buckets: 4096  Batches: 1  Memory Usage: 169kB
                                       ->  Bitmap Heap Scan on rendelesek r  (cost=73.42..176.69 rows=2502 width=25) (actual time=0.419..1.117 rows=2502 loops=1)
                                             Recheck Cond: (rendeles_allapot = 'Teljesitett'::rendeles_allapot_enum)
                                             Heap Blocks: exact=72
                                             ->  Bitmap Index Scan on idx_rendelesek_teljesitett  (cost=0.00..72.79 rows=2502 width=0) (actual time=0.393..0.393 rows=2502 loops=1)
               ->  Index Only Scan using tetelek_pkey on tetelek t  (cost=0.29..0.39 rows=1 width=8) (actual time=0.001..0.001 rows=1 loops=2502)
                     Index Cond: (rendeles_id = r.rendeles_id)
                     Heap Fetches: 0
         ->  Index Scan using konyvek_pkey on konyvek k  (cost=0.29..0.39 rows=1 width=20) (actual time=0.001..0.001 rows=1 loops=2502)
               Index Cond: (konyv_id = t.konyv_id)
 Planning Time: 1.214 ms
 Execution Time: 11.816 ms
(30 rows)
```

Az indexelés utáni QUERY PLAN végrehajtási idejében látszódik, hogy már gyorsabb a lekérdezés, az execution time 10-13ms között mozog.

Optimálisabb és pontosabban meghatározott indexelésekkel növelhető a lekérdezés teljesítménye. A parciális indexelés mellett más indexelés típusokat is lehet használni.