

Chapter 14 – Lab Solution

Indexing 1

Exercise 1 Answer

■ Index creation

- CREATE INDEX idx_sorted on table1(sorted);
- CREATE INDEX idx_unsorted on table1(unsorted);

```
postgres=# \h CREATE INDEX
명령: CREATE INDEX
설명: 새 인덱스 만들기
구문:
CREATE [ UNIQUE ] INDEX [ CONCURRENTLY ] [ [ IF NOT EXISTS ] 이름 ] ON [ ONLY ] 테이블이름 [ USING 색인방법 ]
    ( { 칼럼이름 | ( 표현식 ) } [ COLLATE collation ] [ 연산자클래스 [ ( opclass_매개변수 = 값 [, ... ] ) ] ] [ ASC | DESC ] [ NULLS { FIRST | LAST } ] [, ... ] )
    [ INCLUDE ( 칼럼이름 [, ... ] ) ]
    [ WITH ( 스토리지 매개변수 [= 값] [, ... ] ) ]
    [ TABLESPACE 테이블스페이스이름 ]
    [ WHERE 범위한정구문 ]
URL: https://www.postgresql.org/docs/14/sql-createindex.html
```

```
postgres=# CREATE INDEX idx_sorted on table1(sorted);
CREATE INDEX
postgres=# CREATE INDEX idx_unsorted on table1(unsorted);
CREATE INDEX
```

Exercise 2.a Answer

- [Seq scan] SELECT unsorted FROM table1 WHERE unsorted>10;

```
postgres=# EXPLAIN ANALYZE SELECT unsorted FROM table1 WHERE unsorted>10;
               QUERY PLAN
-----
Seq Scan on table1 (cost=0.00..228092.36 rows=9999895 width=4) (actual time=0.009..913.586 rows=9999943 loops=1)
  Filter: (unsorted > 10)
  Rows Removed by Filter: 57
  Planning Time: 0.076 ms
  Execution Time: 1065.456 ms
(5개 행)
```

- [Index scan] SELECT sorted, rndm FROM table1 WHERE sorted<9 AND rndm=5;

```
postgres=# EXPLAIN ANALYZE SELECT sorted, rndm FROM table1 WHERE sorted<9 AND rndm=5;
               QUERY PLAN
-----
Index Scan using idx_sorted on table1 (cost=0.43..9.42 rows=1 width=8) (actual time=0.011..0.011 rows=0 loops=1)
  Index Cond: (sorted < 9)
  Filter: (rndm = 5)
  Rows Removed by Filter: 45
  Planning Time: 0.078 ms
  Execution Time: 0.021 ms
(6개 행)
```

- [Index only scan] SELECT sorted FROM table1 WHERE sorted>1234567;

```
postgres=# EXPLAIN ANALYZE SELECT sorted FROM table1 WHERE sorted>1234567;
               QUERY PLAN
-----
Index Only Scan using idx_sorted on table1 (cost=0.43..88053.74 rows=3796189 width=4) (actual time=0.012..188.678 rows=3827160 loops=1)
  Index Cond: (sorted > 1234567)
  Heap Fetches: 0
  Planning Time: 0.059 ms
  Execution Time: 247.588 ms
(5개 행)
```

Exercise 2.b Answer

- Using the index on “sorted” (Data records are clustered based on “sorted” values.)
 - SELECT sorted, rndm FROM table1 WHERE sorted>100 AND sorted<500 AND rndm=1005;

```
postgres=# EXPLAIN ANALYZE SELECT sorted, rndm FROM table1 WHERE sorted>100 and sorted<500 and rndm=1005;
               QUERY PLAN
-----
Index Scan using idx_sorted on table1 (cost=0.44..163.66 rows=1 width=8) (actual time=1.875..1.875 rows=0 loops=1)
  Index Cond: ((sorted > 100) AND (sorted < 500))
  Filter: (rndm = 1005)
  Rows Removed by Filter: 3990
  Planning Time: 9.170 ms
  Execution Time: 1.884 ms
(6개 행)
```

- Using the index on “unsorted” (Data records are not clustered w.r.t. “unsorted” values.)
 - SELECT unsorted, rndm FROM table1 WHERE unsorted>100 AND unsorted<500 AND rndm=1005;

```
postgres=# EXPLAIN ANALYZE SELECT unsorted, rndm FROM table1 WHERE unsorted>100 and unsorted<500 and rndm=1005;
               QUERY PLAN
-----
Index Scan using idx_unsorted on table1 (cost=0.44..24957.09 rows=1 width=8) (actual time=29.600..29.600 rows=0 loops=1)
  Index Cond: ((unsorted > 100) AND (unsorted < 500))
  Filter: (rndm = 1005)
  Rows Removed by Filter: 6063
  Planning Time: 1.465 ms
  Execution Time: 29.613 ms
(6개 행)
```

Exercise 2.c Answer

- CLUSTER table1 USING idx_sorted;
 - SELECT sorted, rndm FROM table1 WHERE sorted>100 AND sorted<500 AND rndm=1005;

```
postgres=# CLUSTER table1 USING idx_sorted;
CLUSTER
postgres=# EXPLAIN ANALYZE SELECT sorted, rndm FROM table1 WHERE sorted>100 and sorted<500 and rndm=1005;
               QUERY PLAN
-----
Index Scan using idx_sorted on table1 (cost=0.44..159.66 rows=1 width=8) (actual time=0.527..0.527 rows=0 loops=1)
  Index Cond: ((sorted > 100) AND (sorted < 500))
  Filter: (rndm = 1005)
  Rows Removed by Filter: 3990
  Planning Time: 2.472 ms
  Execution Time: 0.536 ms (Compare it with the execution time of the previous problem.)
(6개 행)
```

- CLUSTER table1 USING idx_unsorted;
 - SELECT unsorted, rndm FROM table1 WHERE unsorted>100 AND unsorted<500 AND rndm=1005;

```
postgres=# CLUSTER table1 USING idx_unsorted;
CLUSTER
postgres=# EXPLAIN ANALYZE SELECT unsorted, rndm FROM table1 WHERE unsorted>100 and unsorted<500 and rndm=1005;
               QUERY PLAN
-----
Index Scan using idx_unsorted on table1 (cost=0.44..24957.09 rows=1 width=8) (actual time=0.782..0.782 rows=0 loops=1)
  Index Cond: ((unsorted > 100) AND (unsorted < 500))
  Filter: (rndm = 1005)
  Rows Removed by Filter: 6063
  Planning Time: 1.923 ms
  Execution Time: 0.794 ms (Compare it with the execution time of the previous problem.)
(6개 행)
```

Exercise 2.d Answer

- Query plans are different according to the "WHERE condition"

```
postgres=# EXPLAIN ANALYZE SELECT sorted, rndm FROM table1 WHERE sorted>1999231 AND rndm=1005;
               QUERY PLAN
-----
Index Scan using idx_sorted on table1  (cost=0.43..161.88 rows=1 width=8) (actual time=0.454..0.455 rows=0 loops=1)
  Index Cond: (sorted > 1999231)
  Filter: (rndm = 1005)
  Rows Removed by Filter: 3840
  Planning Time: 0.081 ms
  Execution Time: 0.466 ms
(6개 행)
```

```
postgres=# EXPLAIN ANALYZE SELECT sorted, rndm FROM table1 WHERE sorted<1999231 AND rndm=1005;
               QUERY PLAN
-----
Seq Scan on table1  (cost=0.00..253092.23 rows=101 width=8) (actual time=12.399..806.499 rows=114 loops=1)
  Filter: ((sorted < 1999231) AND (rndm = 1005))
  Rows Removed by Filter: 9999886
  Planning Time: 0.079 ms
  Execution Time: 806.521 ms
(5개 행)
```

Exercise 3 Answer

- Measure the t_1 value
 - $t_{1.insert} > \text{INSERT INTO table10 (SELECT * FROM pool);}$
 - $t_{1.create_index} > \text{CREATE INDEX t10_idx on table10 (val asc);}$

- Measure the t_2 value
 - $t_{2.create_index} > \text{CREATE INDEX t20_idx on table20 (val asc);}$
 - $t_{2.insert} > \text{INSERT INTO table20 (SELECT * FROM pool);}$

Exercise 3 Answer

- t_1 is smaller than t_2
 - $t_1 = 6173.896 + 3575.701 = 9749.597(\text{ms})$
 - $t_2 = 9.639 + 15894.618 = 15904.257(\text{ms})$

```
postgres=# INSERT INTO table10 (SELECT * FROM pool);  
INSERT 0 5000000  
작업시간: 6173.896 ms (00:06.174)  
postgres=# CREATE INDEX t10_idx on table10 (val asc);  
CREATE INDEX  
작업시간: 3575.701 ms (00:03.576)
```

$t_{1.insert} = 6173.896(\text{ms})$

$t_{1.create_index} = 3575.701(\text{ms})$

```
postgres=# CREATE INDEX t20_idx on table20 (val asc);  
CREATE INDEX  
작업시간: 9.639 ms  
postgres=# INSERT INTO table20 (SELECT * FROM pool);  
INSERT 0 5000000  
작업시간: 15894.618 ms (00:15.895)
```

$t_{2.create_index} = 9.639(\text{ms})$

$t_{2.insert} = 15894.618(\text{ms})$