Video Activity – Week-7

Submitted by: - Parth Parashar

Activity 11.1

Question-1: - When should we use index in database?

Answer: - Indexes are a common way **to enhance database performance**. An index allows the database server to find and retrieve specific rows much faster than it could do without an index. But indexes also add overhead to the database system as a whole, so they should be used sensibly.

Question-2: - What does the data structure for an index look like?

Answer: - An Index is **the structure or object by which we can retrieve specific rows or data faster**. Indexes can be created using one or multiple columns or by using the partial data depending on your query requirement conditions. Index will create a pointer to the actual rows in the specified table.

Question-3: - If you are in psql, run \d books to see that there is an index on books.id?

Answer: -

```
spr2022adb35=> \d books;
                Table "spr2022adb35.books"
  Column   |  Type   | Collation | Nullable | Default
-----------+---------+-----------+----------+---------
 id        | integer |           | not null |
 title     | text    |           |          |
 pagecount | integer |           |          |
 genre     | text    |           |          |
 authorid  | integer |           |          |
 pubid     | integer |           |          |
Indexes:
    "books_pkey" PRIMARY KEY, btree (id)

spr2022adb35=>
```

Question-4: - Create an index on books.pagecount?

Answer: -

```
spr2022adb35=> create index on books(pagecount);
CREATE INDEX
spr2022adb35=> \d books;
              Table "spr2022adb35.books"
  Column   |  Type   | Collation | Nullable | Default
-----------+---------+-----------+----------+---------
 id        | integer |           | not null |
 title     | text    |           |          |
 pagecount | integer |           |          |
 genre     | text    |           |          |
 authorid  | integer |           |          |
 pubid     | integer |           |          |
Indexes:
    "books_pkey" PRIMARY KEY, btree (id)
    "books_pagecount_idx" btree (pagecount)

spr2022adb35=>
```

Question-5: - Write a query that could take advantage of that index?

Answer: -

```
spr2022adb35=> select * from books where pagecount = 500;
 id | title  | pagecount |  genre  | authorid | pubid
----+--------+-----------+---------+----------+-------
  2 | Hamlet |       500 | Tragedy |       13 |   103
(1 row)
```

Activity 11.2

Question-1: - indicate whether index matches predicate or not?

Answer: -

1) Id = 5132 → Yes
2) Title = 'It' and id = 5132 → yes
3) Title = 'It' or id = 5132 → no
4) Genre = ' Horror' and pagecount > 2000 → yes
5) Genre = 'Horror' → yes
6) pagecount > 2000 → no

Activity 12.1

Question -1: - Try running the following queries?

Answer: - 1)

OpenSSH SSH client

```
spr2022adb35=> explain select * from agent where agent_id = 5;
                              QUERY PLAN
-----------------------------------------------------------------------
 Index Scan using agent_pkey on agent  (cost=0.28..8.29 rows=1 width=54)
   Index Cond: (agent_id = 5)
(2 rows)

spr2022adb35=>
```

2)

```
spr2022adb35=> explain select * from agent where agent_id > 5;
                         QUERY PLAN
-------------------------------------------------------------
 Seq Scan on agent  (cost=0.00..16.27 rows=658 width=54)
   Filter: (agent_id > 5)
(2 rows)

spr2022adb35=>
```

Activity 12.2

Question-1: - Run the following queries

Answer: -

1) run select * from books;

OpenSSH SSH client

```
spr2022adb35=> select * from books;
 id |              title               | pagecount |       genre        | authorid | pubid
----+----------------------------------+-----------+--------------------+----------+-------
  1 | It                               |      1138 | Horror             |       10 |   100
  2 | Hamlet                           |       500 | Tragedy            |       13 |   103
  3 | I Know Why the Caged Bird Sings  |       304 | Autobiographical   |       14 |   102
  4 | A Suitable Boy                   |      1349 | Drama/Romance      |       15 |   103
  5 | The Joy Luck Club                |       288 | Drama              |       16 |   104
  6 | Like Water for Chocolate         |       256 | Romance/Tragedy    |       17 |   105
  7 | Tita's Diary                     |       294 | Romance/Diary      |       17 |
  8 | From Heaven Lake                 |       464 | Travel             |       15 |   102
  9 | Kite Runner                      |       371 | Historical/Drama   |       18 |   106
 10 | The Vanishing Half               |       352 | Historical/Drama   |       19 |   106
 11 | September Love                   |       224 | Romance            |       20 |   107
 12 | The Nickel Boys                  |       224 | Historical         |       21 |   108
 13 | The Alchemist                    |       163 | Fantasy/Adventure  |       22 |   103
 14 | Love and Misadventure            |       176 | Romance            |       20 |   107
 15 | The Authenticity Project         |       384 | Romance            |       23 |   102
 16 | Paper Towns                      |       420 | Young adult        |       24 |   100
 17 | Looking for Alaska               |       620 | Young Adult        |       24 |   100
(17 rows)

spr2022adb35=>
```

2) create index on books (pagecount);
3) \d books

Both of these queries are in the screenshot below.

OpenSSH SSH client

```
spr2022adb35=> create index on books(pagecount);
CREATE INDEX
spr2022adb35=> \d books;
                Table "spr2022adb35.books"
  Column   |  Type   | Collation | Nullable | Default
-----------+---------+-----------+----------+---------
 id        | integer |           | not null |
 title     | text    |           |          |
 pagecount | integer |           |          |
 genre     | text    |           |          |
 authorid  | integer |           |          |
 pubid     | integer |           |          |
Indexes:
    "books_pkey" PRIMARY KEY, btree (id)
    "books_pagecount_idx" btree (pagecount)
    "books_pagecount_idx1" btree (pagecount)

spr2022adb35=>
```

4) Cluster books using books_pagecount_index;

OpenSSH SSH client

```
spr2022adb35=> cluster books using books_pagecount_idx;
CLUSTER
spr2022adb35=> \d books;
                Table "spr2022adb35.books"
  Column   |  Type   | Collation | Nullable | Default
-----------+---------+-----------+----------+---------
 id        | integer |           | not null |
 title     | text    |           |          |
 pagecount | integer |           |          |
 genre     | text    |           |          |
 authorid  | integer |           |          |
 pubid     | integer |           |          |
Indexes:
    "books_pkey" PRIMARY KEY, btree (id)
    "books_pagecount_idx" btree (pagecount) CLUSTER
    "books_pagecount_idx1" btree (pagecount)

spr2022adb35=>
```

5) Select * from books;

OpenSSH SSH client

```
spr2022adb35=> select * from books;
 id |              title               | pagecount |       genre        | authorid | pubid
----+----------------------------------+-----------+--------------------+----------+-------
 13 | The Alchemist                    |       163 | Fantasy/Adventure  |       22 |   103
 14 | Love and Misadventure            |       176 | Romance            |       20 |   107
 11 | September Love                   |       224 | Romance            |       20 |   107
 12 | The Nickel Boys                  |       224 | Historical         |       21 |   108
  6 | Like Water for Chocolate         |       256 | Romance/Tragedy    |       17 |   105
  5 | The Joy Luck Club                |       288 | Drama              |       16 |   104
  7 | Tita's Diary                     |       294 | Romance/Diary      |       17 |
  3 | I Know Why the Caged Bird Sings  |       304 | Autobiographical   |       14 |   102
 10 | The Vanishing Half               |       352 | Historical/Drama   |       19 |   106
  9 | Kite Runner                      |       371 | Historical/Drama   |       18 |   106
 15 | The Authenticity Project         |       384 | Romance            |       23 |   102
 16 | Paper Towns                      |       420 | Young adult        |       24 |   100
  8 | From Heaven Lake                 |       464 | Travel             |       15 |   102
  2 | Hamlet                           |       500 | Tragedy            |       13 |   103
 17 | Looking for Alaska               |       620 | Young Adult        |       24 |   100
  1 | It                               |      1138 | Horror             |       10 |   100
  4 | A Suitable Boy                   |      1349 | Drama/Romance      |       15 |   103
(17 rows)

spr2022adb35=>
```

Observations: -

1) After cluster, the relation is sorted on the basis of pagecount
2) When select * is run, then the second time, it is based on the cluster and not according to the Asc order as is the default case for select *.
3) Cluster is effective to get results in the form of groups.