Name: Chitradevi Maruthavanan

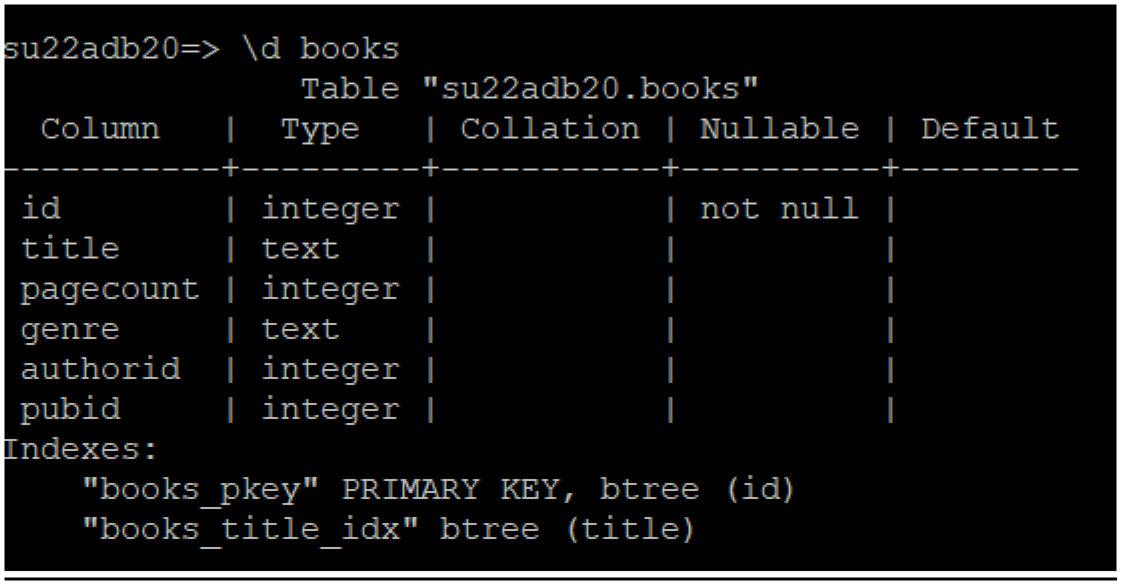
Week -6 Video Activity

Activity 11.1 –Storage and Indexes

1. **If you are in psql, run \d books to see that there is an index on books.id**

   Answer:

   Yes, I can see that there is an index on books.id



```
su22adb20=> \d books
                 Table "su22adb20.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_title_idx" btree (title)
```

2. **Create an index on books.pagecount**

   Answer:

   CREATE INDEX ON books (pagecount);

3. **Write a query that take advantage of that index**.

   Answer:

   SELECT * FROM books WHERE pagecount =100;

Activity 11.2 – Index Matching

For the below – indicate if the index matches the predicate or not

Assume index on books(id)

1. **id = 5132** yes
2. **title = 'It' AND id = 5132** **yes**
3. **title = 'It' OR id = 5132** **NO**

Assume index on books(genre,pagecount)

4.genre = 'Horror' AND pagecount > 2000 **Yes**

5.genre = 'Horror' **yes**

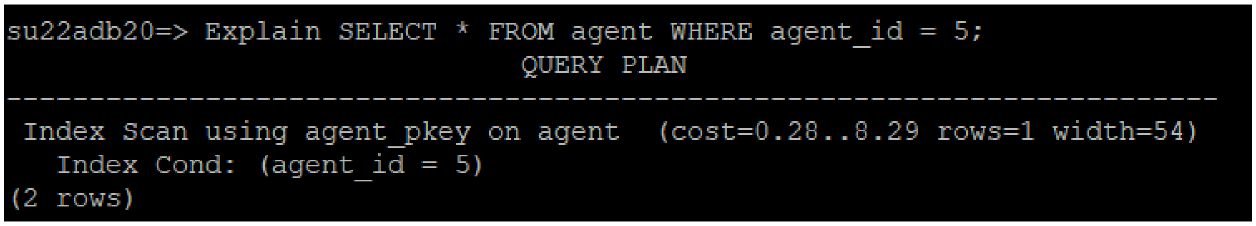
6.pagecount > 2000 **No**

Activity 12.1 –Index Structure

1. **Explain SELECT * FROM agent WHERE id = 5;**

   Answer:

   Yes, PostgreSQL uses an index

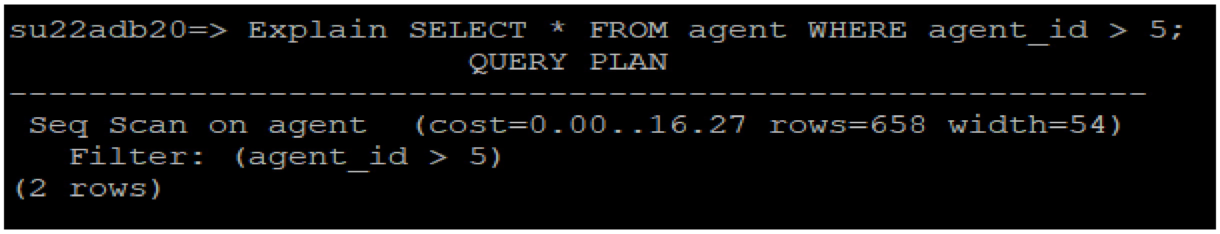


```
su22adb20=> Explain SELECT * FROM agent WHERE agent_id = 5;
                              QUERY PLAN
--------------------------------------------------------------------------
 Index Scan using agent_pkey on agent  (cost=0.28..8.29 rows=1 width=54)
   Index Cond: (agent_id = 5)
(2 rows)
```

2. **Explain SELECT * FROM agent WHERE id >5;**

   Answer:

   No, Postgre SQL does not an index

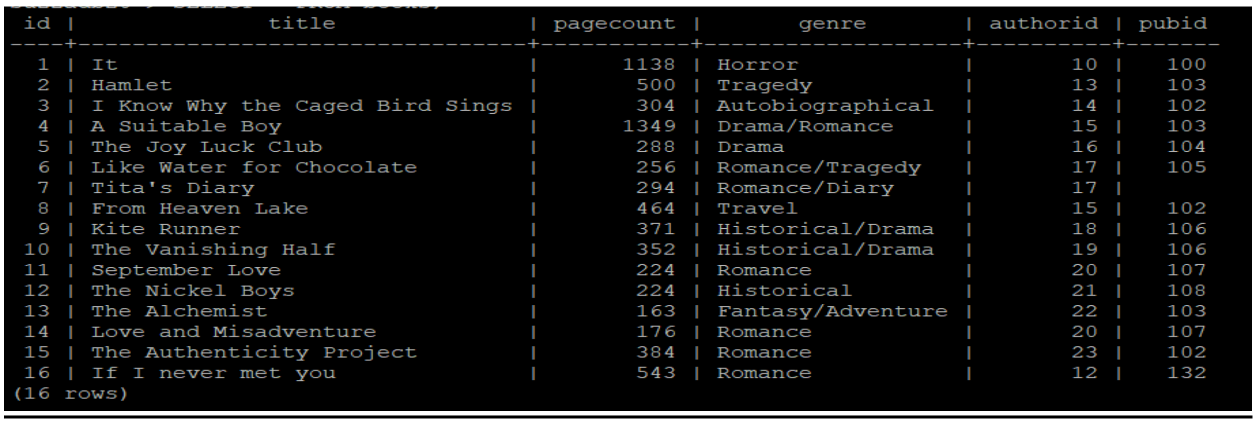


```
su22adb20=> Explain SELECT * FROM agent WHERE agent_id > 5;
                        QUERY PLAN
---------------------------------------------------------
 Seq Scan on agent  (cost=0.00..16.27 rows=658 width=54)
   Filter: (agent_id > 5)
(2 rows)
```

Activity 12.2 –Clustered and unclustered Indexes

1. **Run: SELECT * FROM books;**

   Answer:



```
 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 | If I never met you             |       543 | Romance            |       12 |   132
(16 rows)
```

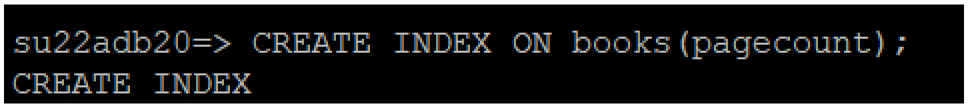
Note:

Before cluster the relation is not sorted on any attribute (the relation is almost sorted on id)

2. **CREATE an index on books on pagecount using the command**

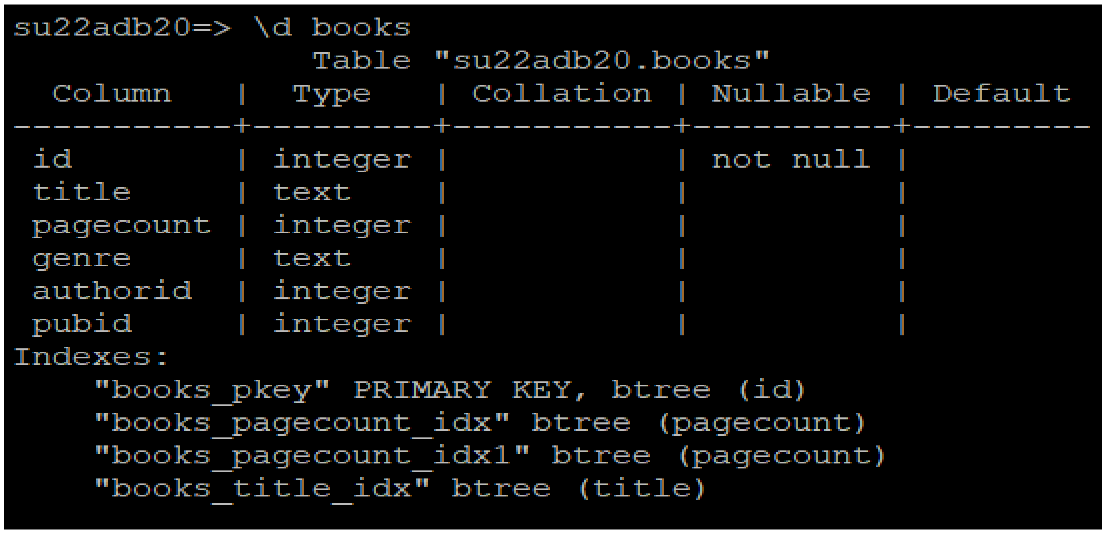
   CREATE INDEX ON books(pagecount);

   Answer:



```
su22adb20=> CREATE INDEX ON books(pagecount);
CREATE INDEX
```

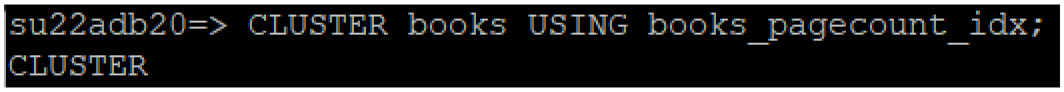
3. **If you are using psql, you can use \d books to see the name of the new index**



```
su22adb20=> \d books
               Table "su22adb20.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)
    "books_title_idx" btree (title)
```

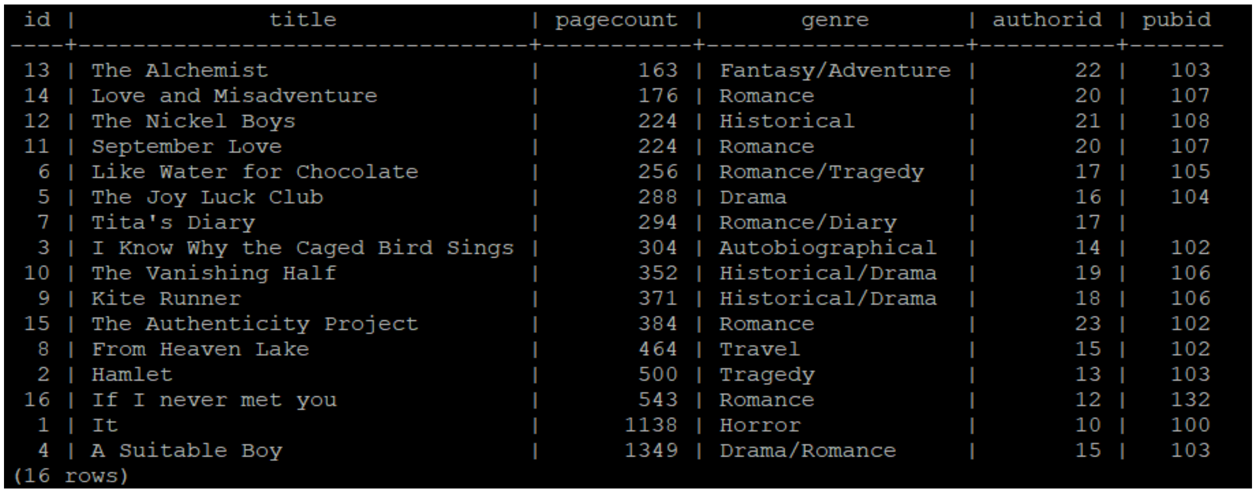
4. **Cluster books using the command:**

   CLUSTER books USING books_pagecount_index;



```
su22adb20=> CLUSTER books USING books_pagecount_idx;
CLUSTER
```

5. **Run:** SELECT * FROM books; again



```
 id |              title               | pagecount |       genre        | authorid | pubid
----+----------------------------------+-----------+--------------------+----------+-------
 13 | The Alchemist                    |       163 | Fantasy/Adventure  |       22 |   103
 14 | Love and Misadventure            |       176 | Romance            |       20 |   107
 12 | The Nickel Boys                  |       224 | Historical         |       21 |   108
 11 | September Love                   |       224 | Romance            |       20 |   107
  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
  8 | From Heaven Lake                 |       464 | Travel             |       15 |   102
  2 | Hamlet                           |       500 | Tragedy            |       13 |   103
 16 | If I never met you               |       543 | Romance            |       12 |   132
  1 | It                               |      1138 | Horror             |       10 |   100
  4 | A Suitable Boy                   |      1349 | Drama/Romance      |       15 |   103
(16 rows)
```

<u>Note:</u>

After cluster the relation is sorted on pagecount.

I cannot count on postgres giving me a sorted result like this. In the instance, postgres will return the tuples in the order they are stored; however, the database can return tuples in any order it wants to.