Lab 7: Indexing and Query Plans

Make sure your PostgreSQL setup is working before starting on this assignment. Also upload the large university dataset using the instructions available at https://moodle.iitb.ac.in/mod/page/view.php?id=58587 before lab time

In this assignment you should use the PostgreSQL functions *explain <query>* ,to find the query plan, and *explain analyse <query>* to find the execution time and other statistics in addition to the query plan. Your submission should be a document (created using google doc or libreoffice) which contains the answer for each part of each question along with query.

1. This part focuses on time for various queries.

   First create a relation player(A INT,B INT). (There is no primary key declaration.)

   Then execute the queries as below. Show the plan + time to run each of the queries as below. Also provide explanations for what you observe.

   a. Starting with an empty relation with no index, insert 10,000 records ino relation player, where the ith record is of the form (i, i+1). Use the following recursive query to generate the values to be inserted.

      ```
      WITH RECURSIVE GeneratePlayers AS (
         SELECT 1 AS A, 2 AS B
           UNION
         SELECT 1+A, 1+B FROM GeneratePlayers
         WHERE 1+A < 10001
      ```

      Report the time to execute the complete set of inserts.

   b. Query the database to retrieve a single record from the database with value of A=1000. Report the query plan and time.

   c. Create an index indexA on A.

   d. Query the database to retrieve a single record from the database with value of A=1000. Report the query plan and time

   e. Delete all tuples from the relation

   f. Insert tuples as in part a. Explain the difference in time between part a. and part f.

2. Load the large university dataset using the instructions provided on Moodle https://moodle.iitb.ac.in/mod/page/view.php?id=58587

   Next create queries as below on the university schema using the large university dataset

   Submit the query and the plan (copy-paste from Explain output) for each part below. Use Explain, instead of Explain analyze, except where time is asked for.

   a. Create a selection query whose chosen plan is a file scan.
   b. Create a selection query with and AND of two predicates, whose chosen plan uses an index scan on one of the predicates.
   c. Create a selection query where PostgreSQL uses the bitmap index scan operation. You can create indices on appropriate relation attributes to create such a case. Explain why PostgreSQL chose that plan.
   d. Create a query where PostgreSQL chooses an index nested loops join (NOTE: the nested loops operator has 2 children. The first child is the outer input, and it may have an index scan or anything else, that is irrelevant. The second child must have an index scan or bitmap index scan, using an attribute from the first child.)
   e. Create a table takes2 with the same schema as takes but no primary keys or foreign keys.   Find how long it takes to execute the query
      insert into takes2 select * from takes
      Report the explain analyze result for the above insert statement.
   f. Next drop the table takes2 (and its rows, as a result), and create it again, but this time with a primary key declaration which is the same as in takes, but no foreign key.

      Run the insert again and measure how long it takes to run. Report the explain analyze result, and explain why the time taken is different this time.
   g. Consider the following nested subquery:
      select count(*) from course c
      *where exists (select * from takes t where t.course_id < c.course_id)*
      What is the plan is chosen by PostgreSQL. Explain what is happening.
   h. As above, but with the query
      select count(*) from course c
      where c.course_id in (select course_id from takes t)

i. As above, but with the query

```
select count(*) from course c
where c.course_id  not in (select course_id from takes t)
```