DBMS PROJECT

DATABASE CREATORS TEAM

TEAM MEMBERS:

1. Banoth Chethan Naik - 2020CS10333
2. Varanasi Yaswanth Krishna - 2020CS10406
3. Ankit Kumar - 2020CS10323

Milestone -2

Queries that will drive application.

Query -1:

```
SELECT  DISTINCT ON (Diseases.name) Diseases.name ,Diseases.born_country,Diseases.born_year,Diseases.mortality_rate
FROM Diseases
WHERE Diseases.name='disease_name';
```

Query -2:

```
--2--
SELECT DISTINCT Diseases.name
FROM Diseases
WHERE name LIKE 'starting_characters%';
```

Query -3:

```
--3--
SELECT Treatment.name,Treatment.type ,Treatment.description
FROM Treatment
WHERE Treatment.id IN (
    SELECT Has_Treatment.Treatment_id
    FROM Has_Treatment
    WHERE Has_Treatment.Disease_id IN (
        SELECT Diseases.id
        FROM Diseases
        WHERE Diseases.name='disease_name'
));
```

Query -4:

```
--4--
SELECT Prevention.description
FROM Prevention
WHERE Prevention.Disease_id IN (
    SELECT Diseases.id
    FROM Diseases
    WHERE Diseases.name='disease_name'

);
```

Query -5:

```
--5--
SELECT Medicine.name,Medicine.prescription ,Medicine.side_effects
FROM Medicine
WHERE Medicine.id IN (
    SELECT Has_Medicine.Medicine_id
    FROM Has_Medicine
    WHERE Has_Medicine.Disease_id IN (
        SELECT Diseases.id
        FROM Diseases
        WHERE Diseases.name='disease_name'
));
```

Query -6:

```
--6--
SELECT Types_of_tests.name,Types_of_tests.description
FROM Types_of_tests
WHERE Types_of_tests.id IN (
    SELECT Identification_test.test_id
    FROM Identification_test
    WHERE Identification_test.Disease_id IN (
        SELECT Diseases.id
        FROM Diseases
        WHERE Diseases.name='disease_name'
));
```

Query -7:

```
--7--
SELECT Types_of_symptoms.name
FROM Types_of_symptoms
WHERE Types_of_symptoms.id IN (
    SELECT Has_symptoms.symptom_id
    FROM Has_symptoms
    WHERE Has_symptoms.Disease_id IN (
        SELECT Diseases.id
        FROM Diseases
        WHERE Diseases.name='disease_name'
));
```

Query -8:

```
 --8--
 SELECT Vaccines.name
 FROM Vaccines
 WHERE Vaccines.id IN (
     SELECT Has_Vaccine.Vaccine_id
     FROM Has_Vaccine
     WHERE Has_Vaccine.Disease_id IN (
         SELECT Diseases.id
         FROM Diseases
         WHERE Diseases.name='disease_name'
 ));
```

Query -9:

```
--9--
SELECT Transmission_modes.name ,Transmission_modes.description
FROM Transmission_modes
WHERE Transmission_modes.id IN (
    SELECT Modes_of_Transmission.Transmission_id
    FROM Modes_of_Transmission
    WHERE Modes_of_Transmission.Disease_id IN (
        SELECT Diseases.id
        FROM Diseases
        WHERE Diseases.name='disease_name'
));
```

Query -10:

```
--10--
SELECT DISTINCT Treatment.name
FROM Treatment
WHERE name LIKE 'starting_characters%';
```

Query -11:

```
--11--
SELECT DISTINCT Transmission_modes.name
FROM Transmission_modes
WHERE name LIKE 'starting_characters%';
```

Query -12:

```
--12--
SELECT DISTINCT Medicine.name
FROM Medicine
WHERE name LIKE 'starting_characters%';
```

Query -13:

```
--13--
SELECT DISTINCT Vaccines.name
FROM Vaccines
WHERE name LIKE 'starting_characters%';
```

Query -14:

```
--14--
SELECT DISTINCT Types_of_tests.name
FROM Types_of_tests
WHERE name LIKE 'starting_characters%';
```

Query -15:

```sql
--15--
SELECT DISTINCT Types_of_symptoms.name
FROM Types_of_symptoms
WHERE name LIKE 'starting_characters%';
```

Query -16:

```sql
--16--
SELECT Treatment.name,Treatment.type ,Treatment.description
FROM Treatment
WHERE Treatment.name='treatment_name';
```

Query -17:

```sql
--17--
SELECT Types_of_tests.name,Types_of_tests.description
FROM Types_of_tests
WHERE Types_of_tests.name='test_name';
```

Query -18:

```sql
--18--
SELECT Has_Medicine.Dosage
FROM Has_Medicine
WHERE Has_Medicine.Medicine_id=(
    SELECT Medicine.id
    FROM Medicine
    WHERE Medicine.name='medicine_name'
)
AND Has_Medicine.Disease_id=(
    SELECT Diseases.id
    FROM Diseases
    WHERE Diseases.name='disease_name'
);
```

Query -19:

```
--19--
SELECT DISTINCT Diseases.name
FROM Diseases
WHERE Diseases.id IN(
    SELECT Has_Medicine.Disease_id
    FROM Has_Medicine
    WHERE Has_Medicine.Medicine_id IN(
        SELECT Medicine.id
        FROM Medicine
        WHERE Medicine.name='medicine_name'
    )
);
```

Query -20:

```
--20--
SELECT DISTINCT Diseases.name
FROM Diseases
WHERE Diseases.id IN(
    SELECT Has_Treatment.Disease_id
    FROM Has_Treatment
    WHERE Has_Treatment.Treatment_id IN(
        SELECT Treatment.id
        FROM Treatment
        WHERE Treatment.name='Treatment_name'
    )
);
```

Query -21:

```
--21--
SELECT DISTINCT Diseases.name
FROM Diseases
WHERE Diseases.id IN(
    SELECT Modes_of_Transmission.Disease_id
    FROM Modes_of_Transmission
    WHERE Modes_of_Transmission.Transmission_id IN(
        SELECT Transmission_modes.id
        FROM Transmission_modes
        WHERE Transmission_modes.name='transmission_name'
    )
);
```

Query -22:

```
--22--
SELECT DISTINCT Diseases.name
FROM Diseases
WHERE Diseases.id IN(
    SELECT Has_Vaccine.Disease_id
    FROM Has_Vaccine
    WHERE Has_Vaccine.Vaccine_id IN(
        SELECT Vaccines.id
        FROM Vaccines
        WHERE Vaccines.name='Vaccine_name'
    )
);
```

Query -23:

```
--23--
SELECT DISTINCT Diseases.name
FROM Diseases
WHERE Diseases.id IN(
    SELECT Identification_test.Disease_id
    FROM Identification_test
    WHERE Identification_test.test_id IN(
        SELECT Types_of_tests.id
        FROM Types_of_tests
        WHERE Types_of_tests.name='test_name'
    )
);
```

Query -24:

```
--24--
SELECT DISTINCT Diseases.name
FROM Diseases
WHERE Diseases.id IN(
    SELECT Has_symptoms.Disease_id
    FROM Has_symptoms
    WHERE Has_symptoms.symptom_id IN (
        SELECT Types_of_symptoms.id
        FROM Types_of_symptoms
        WHERE Types_of_symptoms.name IN :symptom_table
    )
);
```

Query -25:

```
--25--
SELECT DISTINCT Diseases.name
FROM Diseases
JOIN Has_symptoms ON Diseases.id = Has_symptoms.Disease_id
JOIN Types_of_symptoms ON Has_symptoms.symptom_id = Types_of_symptoms.id
WHERE Types_of_symptoms.name IN :symptom_table
GROUP BY Diseases.id, Diseases.name
HAVING COUNT(DISTINCT Types_of_symptoms.name) = :num_of_symptoms;
```

Query -26:

```
--26--
SELECT Prevention.description
FROM Prevention
WHERE Prevention.Disease_id IN (
    SELECT Diseases.id
    FROM Diseases
    WHERE Diseases.name IN (
        SELECT DISTINCT Diseases.name
        FROM Diseases
        WHERE Diseases.id IN(
            SELECT Has_symptoms.Disease_id
            FROM Has_symptoms
            WHERE Has_symptoms.symptom_id IN (
                SELECT Types_of_symptoms.id
                FROM Types_of_symptoms
                WHERE Types_of_symptoms.name IN :symptom_table
    )
)

    )

);
```

Query -27:

```
--27--
SELECT Prevention.description
FROM Prevention
WHERE Prevention.Disease_id IN (
    SELECT Diseases.id
    FROM Diseases
    WHERE Diseases.name IN (
        SELECT DISTINCT Diseases.name
        FROM Diseases
        JOIN Has_symptoms ON Diseases.id = Has_symptoms.Disease_id
        JOIN Types_of_symptoms ON Has_symptoms.symptom_id = Types_of_symptoms.id
        WHERE Types_of_symptoms.name IN :symptom_table
        GROUP BY Diseases.id, Diseases.name
        HAVING COUNT(DISTINCT Types_of_symptoms.name) = :num_of_symptoms
    )

);
```

Query -28:

```
--28--
SELECT DISTINCT ON (Diseases.name) Diseases.name,Diseases.born_country,Diseases.born_year,
Diseases.mortality_rate,Types_of_symptoms.name as symptoms_name,Medicine.name as medicine_name,Medicine.prescription as medicine_prescription,
Medicine.side_effects as medicine_side_effects,Has_Medicine.Dosage as medicine_dosage,Treatment.name as treatment_name,Treatment.type as treatment_type,
Treatment.description as treatment_description,Vaccines.name as Vaccine_name,Types_of_tests.name as test_name,Types_of_tests.description as test_description,
Transmission_modes.name as modes_name,Transmission_modes.description as modes_description ,Prevention.description as Prevention
FROM Diseases
JOIN Has_symptoms ON Diseases.id = Has_symptoms.Disease_id
JOIN Types_of_symptoms ON Has_symptoms.symptom_id = Types_of_symptoms.id
JOIN Has_Medicine ON Diseases.id=Has_Medicine.Disease_id
JOIN Medicine ON Has_Medicine.Medicine_id=Medicine.id
JOIN Has_Treatment ON Diseases.id=Has_Treatment.Disease_id
JOIN Treatment ON Has_Treatment.Treatment_id=Treatment.id
JOIN Has_Vaccine ON Diseases.id=Has_Vaccine.Disease_id
JOIN Vaccines ON Has_Vaccine.Vaccine_id=Vaccines.id
JOIN Identification_test ON Diseases.id=Identification_test.Disease_id
JOIN Types_of_tests ON Identification_test.test_id=Types_of_tests.id
JOIN Modes_of_Transmission ON Diseases.id=Modes_of_Transmission.Disease_id
JOIN Transmission_modes ON Modes_of_Transmission.Transmission_id=Transmission_modes.id
JOIN Prevention ON Diseases.id=Prevention.Disease_id
WHERE Diseases.name='disease_name';
```

Index Choices for optimizing the queries:

- Basically, all queries depend on the diseases table which is our main application as all users search basically on diseases.

  -**->Create index A1 on diseases(name)**
  ->Create index A2 on Treatment(name)
  ->Create index A3 on Medicine(name)
  ->Create index A4 on Types_of_tests(name)

  The above indexing is used for optimizing the queries.
  We created indexes A1, A2, A3, and A4 on the name columns of the diseases, treatment, medicine, and types_of_tests tables, respectively. By creating indexes on these columns, the database engine can quickly find the data that matches the query conditions for queries that involve these columns.

  For example, if a query is executed to find all diseases with a particular name, the database engine can use the A1 index to quickly locate the relevant rows in the diseases table. Similarly, if a query is executed to find all treatments with a particular name, the database engine can use the A2 index to quickly locate the relevant rows in the treatment table.

  The execution cost of the queries after using the indexing decreases very highly its around 10 times smaller than that without using indexing.

Database size and performance

Our database size:

```
 pg_size_pretty
----------------
 15 MB
(1 row)
```

Performance:

Query 1:

Query 1

```
WHERE Diseases.name='disease_name';
                        QUERY PLAN
-----------------------------------------------------------
 Unique  (cost=0.00..121.94 rows=1 width=86)
   ->  Seq Scan on diseases  (cost=0.00..121.94 rows=1 width=86)
         Filter: ((name)::text = 'disease_name'::text)
(3 rows)

Time: 0.787 ms
```

Query 2

```
final=# EXPLAIN
SELECT DISTINCT Diseases.name
FROM Diseases
WHERE name LIKE 'starting_characters%';
                              QUERY PLAN
-----------------------------------------------------------------------
 Unique  (cost=121.95..121.95 rows=1 width=22)
   ->  Sort  (cost=121.95..121.95 rows=1 width=22)
         Sort Key: name
         ->  Seq Scan on diseases  (cost=0.00..121.94 rows=1 width=22)
               Filter: ((name)::text ~~ 'starting_characters%'::text)
(5 rows)

Time: 0.856 ms
```

Query 3

```
final=# EXPLAIN
SELECT DISTINCT Diseases.name
FROM Diseases
WHERE name LIKE 'starting_characters%';
                            QUERY PLAN
-------------------------------------------------------------------------
 Unique  (cost=121.95..121.95 rows=1 width=22)
   ->  Sort  (cost=121.95..121.95 rows=1 width=22)
         Sort Key: name
         ->  Seq Scan on diseases  (cost=0.00..121.94 rows=1 width=22)
               Filter: ((name)::text ~~ 'starting_characters%'::text)
(5 rows)

Time: 0.856 ms
final=# EXPLAIN
SELECT Treatment.name,Treatment.type ,Treatment.description
FROM Treatment
WHERE Treatment.id IN (
    SELECT Has_Treatment.Treatment_id
    FROM Has_Treatment
    WHERE Has_Treatment.Disease_id IN (
        SELECT Diseases.id
        FROM Diseases
        WHERE Diseases.name='disease_name'
));
                                                        QUERY PLAN
---------------------------------------------------------------------------------------------------------------------------
 Nested Loop  (cost=132.31..132.75 rows=2 width=48)
   ->  Unique  (cost=132.04..132.05 rows=2 width=4)
         ->  Sort  (cost=132.04..132.04 rows=2 width=4)
               Sort Key: has_treatment.treatment_id
               ->  Nested Loop  (cost=0.29..132.03 rows=2 width=4)
                     ->  Seq Scan on diseases  (cost=0.00..121.94 rows=1 width=4)
                           Filter: ((name)::text = 'disease_name'::text)
                     ->  Index Only Scan using has_treatment_pkey on has_treatment  (cost=0.29..10.07 rows=2 width=8)
                           Index Cond: (disease_id = diseases.id)
   ->  Index Scan using treatment_pkey on treatment  (cost=0.28..0.35 rows=1 width=52)
         Index Cond: (id = has_treatment.treatment_id)
(11 rows)

Time: 1.289 ms
final=#
```

Query 4

```
);
                                      QUERY PLAN
---------------------------------------------------------------------------------------
 Nested Loop  (cost=0.28..130.25 rows=1 width=107)
   ->  Seq Scan on diseases  (cost=0.00..121.94 rows=1 width=4)
         Filter: ((name)::text = 'disease_name'::text)
   ->  Index Scan using prevention_pkey on prevention  (cost=0.28..8.30 rows=1 width=111)
         Index Cond: (disease_id = diseases.id)
(5 rows)

Time: 0.853 ms
final=#
```

Query 5

```
                                      QUERY PLAN
---------------------------------------------------------------------------------------
 Nested Loop  (cost=0.28..130.25 rows=1 width=107)
   ->  Seq Scan on diseases  (cost=0.00..121.94 rows=1 width=4)
         Filter: ((name)::text = 'disease_name'::text)
   ->  Index Scan using prevention_pkey on prevention  (cost=0.28..8.30 rows=1 width=111)
         Index Cond: (disease_id = diseases.id)
(5 rows)

Time: 0.853 ms
final=# EXPLAIN
SELECT Medicine.name,Medicine.prescription ,Medicine.side_effects
FROM Medicine
WHERE Medicine.id IN (
    SELECT Has_Medicine.Medicine_id
    FROM Has_Medicine
    WHERE Has_Medicine.Disease_id IN (
        SELECT Diseases.id
        FROM Diseases
        WHERE Diseases.name='disease_name'
));
                                                  QUERY PLAN
---------------------------------------------------------------------------------------------------------------
 Nested Loop  (cost=130.53..130.64 rows=1 width=99)
   ->  HashAggregate  (cost=130.25..130.26 rows=1 width=4)
         Group Key: has_medicine.medicine_id
         ->  Nested Loop  (cost=0.28..130.25 rows=1 width=4)
               ->  Seq Scan on diseases  (cost=0.00..121.94 rows=1 width=4)
                     Filter: ((name)::text = 'disease_name'::text)
               ->  Index Only Scan using has_medicine_pkey on has_medicine  (cost=0.28..8.30 rows=1 width=8)
                     Index Cond: (disease_id = diseases.id)
   ->  Index Scan using medicine_pkey on medicine  (cost=0.28..0.38 rows=1 width=103)
         Index Cond: (id = has_medicine.medicine_id)
(10 rows)

Time: 1.283 ms
final=#
```

Query 6

```
));
                                        QUERY PLAN
-----------------------------------------------------------------------------------------------------------
 Nested Loop  (cost=130.53..130.64 rows=1 width=99)
   ->  HashAggregate  (cost=130.25..130.26 rows=1 width=4)
         Group Key: has_medicine.medicine_id
         ->  Nested Loop  (cost=0.28..130.25 rows=1 width=4)
               ->  Seq Scan on diseases  (cost=0.00..121.94 rows=1 width=4)
                     Filter: ((name)::text = 'disease_name'::text)
               ->  Index Only Scan using has_medicine_pkey on has_medicine  (cost=0.28..8.30 rows=1 width=8)
                     Index Cond: (disease_id = diseases.id)
   ->  Index Scan using medicine_pkey on medicine  (cost=0.28..0.38 rows=1 width=103)
         Index Cond: (id = has_medicine.medicine_id)
(10 rows)

Time: 1.283 ms
final=# EXPLAIN
SELECT Types_of_tests.name,Types_of_tests.description
FROM Types_of_tests
WHERE Types_of_tests.id IN (
    SELECT Identification_test.test_id
    FROM Identification_test
    WHERE Identification_test.Disease_id IN (
        SELECT Diseases.id
        FROM Diseases
        WHERE Diseases.name='disease_name'
));
                                                QUERY PLAN
-----------------------------------------------------------------------------------------------------------------------
 Nested Loop  (cost=130.53..130.69 rows=1 width=192)
   ->  HashAggregate  (cost=130.25..130.26 rows=1 width=4)
         Group Key: identification_test.test_id
         ->  Nested Loop  (cost=0.28..130.25 rows=1 width=4)
               ->  Seq Scan on diseases  (cost=0.00..121.94 rows=1 width=4)
                     Filter: ((name)::text = 'disease_name'::text)
               ->  Index Only Scan using identification_test_pkey on identification_test  (cost=0.28..8.30 rows=1 width=8)
                     Index Cond: (disease_id = diseases.id)
   ->  Index Scan using types_of_tests_pkey on types_of_tests  (cost=0.28..0.43 rows=1 width=196)
         Index Cond: (id = identification_test.test_id)
(10 rows)

Time: 1.172 ms
```

Query 7

```
));
                                        QUERY PLAN
-----------------------------------------------------------------------------------------------------------
 Nested Loop  (cost=134.09..134.88 rows=3 width=33)
   ->  HashAggregate  (cost=133.81..133.84 rows=3 width=4)
         Group Key: has_symptoms.symptom_id
         ->  Nested Loop  (cost=0.29..133.81 rows=3 width=4)
               ->  Seq Scan on diseases  (cost=0.00..121.94 rows=1 width=4)
                     Filter: ((name)::text = 'disease_name'::text)
               ->  Index Only Scan using has_symptoms_pkey on has_symptoms  (cost=0.29..11.84 rows=3 width=8)
                     Index Cond: (disease_id = diseases.id)
   ->  Index Scan using types_of_symptoms_pkey on types_of_symptoms  (cost=0.28..0.35 rows=1 width=37)
         Index Cond: (id = has_symptoms.symptom_id)
(10 rows)

Time: 1.216 ms
```

Query 8

```
));
                                        QUERY PLAN
-----------------------------------------------------------------------------------------------------------
 Nested Loop  (cost=130.53..130.60 rows=1 width=17)
   ->  HashAggregate  (cost=130.25..130.26 rows=1 width=4)
         Group Key: has_vaccine.vaccine_id
         ->  Nested Loop  (cost=0.28..130.25 rows=1 width=4)
               ->  Seq Scan on diseases  (cost=0.00..121.94 rows=1 width=4)
                     Filter: ((name)::text = 'disease_name'::text)
               ->  Index Only Scan using has_vaccine_pkey on has_vaccine  (cost=0.28..8.30 rows=1 width=8)
                     Index Cond: (disease_id = diseases.id)
   ->  Index Scan using vaccines_pkey on vaccines  (cost=0.28..0.34 rows=1 width=21)
         Index Cond: (id = has_vaccine.vaccine_id)
(10 rows)

Time: 1.212 ms
```

Query 9

```
                                                   QUERY PLAN
-------------------------------------------------------------------------------------------------------------------------
 Nested Loop  (cost=130.53..130.61 rows=1 width=40)
   ->  HashAggregate  (cost=130.25..130.26 rows=1 width=4)
         Group Key: modes_of_transmission.transmission_id
         ->  Nested Loop  (cost=0.28..130.25 rows=1 width=4)
               ->  Seq Scan on diseases  (cost=0.00..121.94 rows=1 width=4)
                     Filter: ((name)::text = 'disease_name'::text)
               ->  Index Only Scan using modes_of_transmission_pkey on modes_of_transmission  (cost=0.28..8.30 rows=1 width=8)
                     Index Cond: (disease_id = diseases.id)
   ->  Index Scan using transmission_modes_pkey on transmission_modes  (cost=0.28..0.35 rows=1 width=44)
         Index Cond: (id = modes_of_transmission.transmission_id)
(10 rows)

Time: 1.457 ms
```

Query 10

```
Time: 1.457 ms
final=# EXPLAIN
SELECT DISTINCT Treatment.name
FROM Treatment
WHERE name LIKE 'starting_characters%';
                              QUERY PLAN
-----------------------------------------------------------------------
 Unique  (cost=46.25..46.25 rows=1 width=28)
   ->  Sort  (cost=46.25..46.25 rows=1 width=28)
         Sort Key: name
         ->  Seq Scan on treatment  (cost=0.00..46.24 rows=1 width=28)
               Filter: ((name)::text ~~ 'starting_characters%'::text)
(5 rows)

Time: 0.690 ms
```

Query 11

```
Time: 0.690 ms
final=# EXPLAIN
SELECT DISTINCT Transmission_modes.name
FROM Transmission_modes
WHERE name LIKE 'starting_characters%';
                                   QUERY PLAN
------------------------------------------------------------------------------
 Unique  (cost=36.31..36.32 rows=1 width=31)
   ->  Sort  (cost=36.31..36.31 rows=1 width=31)
         Sort Key: name
         ->  Seq Scan on transmission_modes  (cost=0.00..36.30 rows=1 width=31)
               Filter: ((name)::text ~~ 'starting_characters%'::text)
(5 rows)

Time: 0.724 ms
```

Query 12

```
Time: 0.724 ms
final=# EXPLAIN
SELECT DISTINCT Medicine.name
FROM Medicine
WHERE name LIKE 'starting_characters%';
                              QUERY PLAN
-----------------------------------------------------------------------
 Unique  (cost=128.95..128.95 rows=1 width=34)
   ->  Sort  (cost=128.95..128.95 rows=1 width=34)
         Sort Key: name
         ->  Seq Scan on medicine  (cost=0.00..128.94 rows=1 width=34)
               Filter: ((name)::text ~~ 'starting_characters%'::text)
(5 rows)

Time: 0.868 ms
```

Query 13

```
Time: 0.868 ms
final=# EXPLAIN
SELECT DISTINCT Vaccines.name
FROM Vaccines
WHERE name LIKE 'starting_characters%';
                            QUERY PLAN
------------------------------------------------------------------------
 Unique  (cost=84.95..84.95 rows=1 width=17)
   ->  Sort  (cost=84.95..84.95 rows=1 width=17)
         Sort Key: name
         ->  Seq Scan on vaccines  (cost=0.00..84.94 rows=1 width=17)
               Filter: ((name)::text ~~ 'starting_characters%'::text)
(5 rows)

Time: 0.586 ms
final=#
```

Query 14

```
Time: 0.586 ms
final=# EXPLAIN
SELECT DISTINCT Types_of_tests.name
FROM Types_of_tests
WHERE name LIKE 'starting_characters%';
                                QUERY PLAN
---------------------------------------------------------------------------
 Unique  (cost=178.95..178.95 rows=1 width=31)
   ->  Sort  (cost=178.95..178.95 rows=1 width=31)
         Sort Key: name
         ->  Seq Scan on types_of_tests  (cost=0.00..178.94 rows=1 width=31)
               Filter: ((name)::text ~~ 'starting_characters%'::text)
(5 rows)

Time: 0.830 ms
final=#
```

Query 15

```
Time: 0.830 ms
final=# EXPLAIN
SELECT DISTINCT Types_of_symptoms.name
FROM Types_of_symptoms
WHERE name LIKE 'starting_characters%';
                                  QUERY PLAN
-------------------------------------------------------------------------------
 Unique  (cost=110.64..110.64 rows=1 width=33)
   ->  Sort  (cost=110.64..110.64 rows=1 width=33)
         Sort Key: name
         ->  Seq Scan on types_of_symptoms  (cost=0.00..110.62 rows=1 width=33)
               Filter: ((name)::text ~~ 'starting_characters%'::text)
(5 rows)

Time: 0.761 ms
final=#
```

Query 16

```
               Filter: ((name)::text ~~ 'starting_characters%'::text)
(5 rows)

Time: 0.761 ms
final=# EXPLAIN
SELECT Treatment.name,Treatment.type ,Treatment.description
FROM Treatment
WHERE Treatment.name='treatment_name';
                        QUERY PLAN
-----------------------------------------------------------
 Seq Scan on treatment  (cost=0.00..46.24 rows=1 width=48)
   Filter: ((name)::text = 'treatment_name'::text)
(2 rows)

Time: 0.646 ms
final=#
```

Query 17

```
   Filter: ((name)::text = 'treatment_name'::text)
(2 rows)

Time: 0.646 ms
final=# EXPLAIN
SELECT Types_of_tests.name,Types_of_tests.description
FROM Types_of_tests
WHERE Types_of_tests.name='test_name';
                          QUERY PLAN
---------------------------------------------------------------
 Seq Scan on types_of_tests  (cost=0.00..178.94 rows=10 width=192)
   Filter: ((name)::text = 'test_name'::text)
(2 rows)

Time: 0.603 ms
final=#
```

Query 18

```
    WHERE Diseases.name='disease_name'
);
                                    QUERY PLAN
-----------------------------------------------------------------------------------------
 Index Scan using has_medicine_pkey on has_medicine  (cost=251.16..259.18 rows=1 width=36)
   Index Cond: ((disease_id = $1) AND (medicine_id = $0))
   InitPlan 1 (returns $0)
     ->  Seq Scan on medicine  (cost=0.00..128.94 rows=12 width=4)
           Filter: ((name)::text = 'medicine_name'::text)
   InitPlan 2 (returns $1)
     ->  Seq Scan on diseases  (cost=0.00..121.94 rows=1 width=4)
           Filter: ((name)::text = 'disease_name'::text)
(8 rows)

Time: 0.934 ms
final=#
```

Query 19

```
                                    QUERY PLAN
-----------------------------------------------------------------------------------------------
 Unique  (cost=231.44..231.50 rows=12 width=22)
   ->  Sort  (cost=231.44..231.47 rows=12 width=22)
         Sort Key: diseases.name
         ->  Nested Loop  (cost=226.90..231.22 rows=12 width=22)
               ->  HashAggregate  (cost=226.62..226.74 rows=12 width=4)
                     Group Key: has_medicine.disease_id
                     ->  Hash Join  (cost=129.09..226.59 rows=12 width=4)
                           Hash Cond: (has_medicine.medicine_id = medicine.id)
                           ->  Seq Scan on has_medicine  (cost=0.00..85.95 rows=4395 width=8)
                           ->  Hash  (cost=128.94..128.94 rows=12 width=4)
                                 ->  Seq Scan on medicine  (cost=0.00..128.94 rows=12 width=4)
                                       Filter: ((name)::text = 'medicine_name'::text)
               ->  Index Scan using diseases_pkey on diseases  (cost=0.28..0.37 rows=1 width=26)
                     Index Cond: (id = has_medicine.disease_id)
(14 rows)

Time: 1.215 ms
final=#
```

Query 20

```
                                    QUERY PLAN
-----------------------------------------------------------------------------------------------
 Unique  (cost=217.55..217.57 rows=5 width=22)
   ->  Sort  (cost=217.55..217.56 rows=5 width=22)
         Sort Key: diseases.name
         ->  Nested Loop  (cost=215.85..217.49 rows=5 width=22)
               ->  HashAggregate  (cost=215.57..215.62 rows=5 width=4)
                     Group Key: has_treatment.disease_id
                     ->  Hash Join  (cost=46.25..215.56 rows=5 width=4)
                           Hash Cond: (has_treatment.treatment_id = treatment.id)
                           ->  Seq Scan on has_treatment  (cost=0.00..143.21 rows=9921 width=8)
                           ->  Hash  (cost=46.24..46.24 rows=1 width=4)
                                 ->  Seq Scan on treatment  (cost=0.00..46.24 rows=1 width=4)
                                       Filter: ((name)::text = 'Treatment_name'::text)
               ->  Index Scan using diseases_pkey on diseases  (cost=0.28..0.37 rows=1 width=26)
                     Index Cond: (id = has_treatment.disease_id)
(14 rows)

Time: 1.170 ms
final=#
```

Query 21

```
                                                QUERY PLAN
----------------------------------------------------------------------------------------------------
 Unique  (cost=145.34..145.36 rows=4 width=22)
   ->  Sort  (cost=145.34..145.35 rows=4 width=22)
         Sort Key: diseases.name
         ->  Nested Loop  (cost=144.05..145.30 rows=4 width=22)
               ->  HashAggregate  (cost=143.77..143.81 rows=4 width=4)
                     Group Key: modes_of_transmission.disease_id
                     ->  Hash Join  (cost=36.31..143.76 rows=4 width=4)
                           Hash Cond: (modes_of_transmission.transmission_id = transmission_modes.id)
                           ->  Seq Scan on modes_of_transmission  (cost=0.00..90.89 rows=6289 width=8)
                           ->  Hash  (cost=36.30..36.30 rows=1 width=4)
                                 ->  Seq Scan on transmission_modes  (cost=0.00..36.30 rows=1 width=4)
                                       Filter: ((name)::text = 'transmission_name'::text)
               ->  Index Scan using diseases_pkey on diseases  (cost=0.28..0.37 rows=1 width=26)
                     Index Cond: (id = modes_of_transmission.disease_id)
(14 rows)

Time: 1.217 ms
```

Query 22

```
                                             QUERY PLAN
----------------------------------------------------------------------------------------------
 Unique  (cost=160.84..160.85 rows=1 width=22)
   ->  Sort  (cost=160.84..160.85 rows=1 width=22)
         Sort Key: diseases.name
         ->  Nested Loop  (cost=160.73..160.83 rows=1 width=22)
               ->  HashAggregate  (cost=160.45..160.46 rows=1 width=4)
                     Group Key: has_vaccine.disease_id
                     ->  Hash Join  (cost=84.95..160.45 rows=1 width=4)
                           Hash Cond: (has_vaccine.vaccine_id = vaccines.id)
                           ->  Seq Scan on has_vaccine  (cost=0.00..63.95 rows=4395 width=8)
                           ->  Hash  (cost=84.94..84.94 rows=1 width=4)
                                 ->  Seq Scan on vaccines  (cost=0.00..84.94 rows=1 width=4)
                                       Filter: ((name)::text = 'Vaccine_name'::text)
               ->  Index Scan using diseases_pkey on diseases  (cost=0.28..0.37 rows=1 width=26)
                     Index Cond: (id = has_vaccine.disease_id)
(14 rows)

Time: 1.224 ms
```

Query 23

```
                                               QUERY PLAN
--------------------------------------------------------------------------------------------------
 Unique  (cost=258.59..258.64 rows=10 width=22)
   ->  Sort  (cost=258.59..258.61 rows=10 width=22)
         Sort Key: diseases.name
         ->  Nested Loop  (cost=254.87..258.42 rows=10 width=22)
               ->  HashAggregate  (cost=254.59..254.69 rows=10 width=4)
                     Group Key: identification_test.disease_id
                     ->  Hash Join  (cost=179.06..254.56 rows=10 width=4)
                           Hash Cond: (identification_test.test_id = types_of_tests.id)
                           ->  Seq Scan on identification_test  (cost=0.00..63.95 rows=4395 width=8)
                           ->  Hash  (cost=178.94..178.94 rows=10 width=4)
                                 ->  Seq Scan on types_of_tests  (cost=0.00..178.94 rows=10 width=4)
                                       Filter: ((name)::text = 'test_name'::text)
               ->  Index Scan using diseases_pkey on diseases  (cost=0.28..0.37 rows=1 width=26)
                     Index Cond: (id = identification_test.disease_id)
(14 rows)

Time: 1.302 ms
```

Query 24

```
);
                                                          QUERY PLAN
------------------------------------------------------------------------------------------------------------------------------------------
 Unique  (cost=361.45..361.51 rows=13 width=22)
   ->  Sort  (cost=361.45..361.48 rows=13 width=22)
         Sort Key: diseases.name
         ->  Nested Loop  (cost=356.50..361.21 rows=13 width=22)
               ->  HashAggregate  (cost=356.22..356.35 rows=13 width=4)
                     Group Key: has_symptoms.disease_id
                     ->  Hash Join  (cost=130.38..356.18 rows=13 width=4)
                           Hash Cond: (has_symptoms.symptom_id = types_of_symptoms.id)
                           ->  Seq Scan on has_symptoms  (cost=0.00..191.10 rows=13210 width=8)
                           ->  Hash  (cost=130.31..130.31 rows=5 width=4)
                                 ->  Seq Scan on types_of_symptoms  (cost=0.00..130.31 rows=5 width=4)
                                       Filter: ((name)::text = ANY ('{Frank_Sinatra,William_Shakespeare,Elizabeth_II,Adolf_Hitler,George_W._Bush}'::text[]))
               ->  Index Scan using diseases_pkey on diseases  (cost=0.28..0.37 rows=1 width=26)
                     Index Cond: (id = has_symptoms.disease_id)
(14 rows)

Time: 1.471 ms
```

Query 25

```
HAVING COUNT(DISTINCT types_of_symptoms.name) = :num_of_symptoms;
                                                      QUERY PLAN
----------------------------------------------------------------------------------------------------------------------------------------
 Unique  (cost=360.91..360.92 rows=1 width=26)
   ->  Sort  (cost=360.91..360.92 rows=1 width=26)
         Sort Key: diseases.name
         ->  GroupAggregate  (cost=360.64..360.90 rows=1 width=26)
               Group Key: diseases.id
               Filter: (count(DISTINCT types_of_symptoms.name) = 5)
               ->  Sort  (cost=360.64..360.68 rows=13 width=59)
                     Sort Key: diseases.id
                     ->  Nested Loop  (cost=130.66..360.40 rows=13 width=59)
                           ->  Hash Join  (cost=130.38..356.18 rows=13 width=37)
                                 Hash Cond: (has_symptoms.symptom_id = types_of_symptoms.id)
                                 ->  Seq Scan on has_symptoms  (cost=0.00..191.10 rows=13210 width=8)
                                 ->  Hash  (cost=130.31..130.31 rows=5 width=37)
                                       ->  Seq Scan on types_of_symptoms  (cost=0.00..130.31 rows=5 width=37)
                                             Filter: ((name)::text = ANY ('{Frank_Sinatra,William_Shakespeare,Elizabeth_II,Adolf_Hitler,George_W._Bush}'::text[])
                           ->  Index Scan using diseases_pkey on diseases  (cost=0.28..0.32 rows=1 width=26)
                                 Index Cond: (id = has_symptoms.disease_id)
(17 rows)

Time: 1.859 ms
```

Query 26

```
                                                      QUERY PLAN
----------------------------------------------------------------------------------------------------------------------------------------
Nested Loop  (cost=484.62..490.44 rows=15 width=107)
  ->  HashAggregate  (cost=484.34..484.49 rows=15 width=4)
        Group Key: diseases.id
        ->  Hash Join  (cost=361.80..484.30 rows=15 width=4)
              Hash Cond: ((diseases.name)::text = (diseases_1.name)::text)
              ->  Seq Scan on diseases  (cost=0.00..110.95 rows=4395 width=26)
              ->  Hash  (cost=361.64..361.64 rows=13 width=22)
                    ->  Unique  (cost=361.45..361.51 rows=13 width=22)
                          ->  Sort  (cost=361.45..361.48 rows=13 width=22)
                                Sort Key: diseases_1.name
                                ->  Nested Loop  (cost=356.50..361.21 rows=13 width=22)
                                      ->  HashAggregate  (cost=356.22..356.35 rows=13 width=4)
                                            Group Key: has_symptoms.disease_id
                                            ->  Hash Join  (cost=130.38..356.18 rows=13 width=4)
                                                  Hash Cond: (has_symptoms.symptom_id = types_of_symptoms.id)
                                                  ->  Seq Scan on has_symptoms  (cost=0.00..191.10 rows=13210 width=8)
                                                  ->  Hash  (cost=130.31..130.31 rows=5 width=4)
                                                        ->  Seq Scan on types_of_symptoms  (cost=0.00..130.31 rows=5 width=4)
                                                              Filter: ((name)::text = ANY ('{Frank_Sinatra,William_Shakespeare,Elizabeth_II,Adolf_Hitler,George_W._Bush}'::text[]))
                                      ->  Index Scan using diseases_pkey on diseases diseases_1  (cost=0.28..0.37 rows=1 width=26)
                                            Index Cond: (id = has_symptoms.disease_id)
  ->  Index Scan using prevention_pkey on prevention  (cost=0.28..0.39 rows=1 width=111)
        Index Cond: (disease_id = diseases.id)
(23 rows)

Time: 1.663 ms
```

Query 27

```
                                                      QUERY PLAN
----------------------------------------------------------------------------------------------------------------------------------------
Nested Loop  (cost=483.73..483.85 rows=1 width=107)
  ->  HashAggregate  (cost=483.44..483.45 rows=1 width=4)
        Group Key: diseases.id
        ->  Hash Join  (cost=360.94..483.44 rows=1 width=4)
              Hash Cond: ((diseases.name)::text = ("ANY_subquery".name)::text)
              ->  Seq Scan on diseases  (cost=0.00..110.95 rows=4395 width=26)
              ->  Hash  (cost=360.93..360.93 rows=1 width=22)
                    ->  Subquery Scan on "ANY_subquery"  (cost=360.91..360.93 rows=1 width=22)
                          ->  Unique  (cost=360.91..360.92 rows=1 width=26)
                                ->  Sort  (cost=360.91..360.92 rows=1 width=26)
                                      Sort Key: diseases_1.name
                                      ->  GroupAggregate  (cost=360.64..360.90 rows=1 width=26)
                                            Group Key: diseases_1.id
                                            Filter: (count(DISTINCT types_of_symptoms.name) = 5)
                                            ->  Sort  (cost=360.64..360.68 rows=13 width=59)
                                                  Sort Key: diseases_1.id
                                                  ->  Nested Loop  (cost=130.66..360.40 rows=13 width=59)
                                                        ->  Hash Join  (cost=130.38..356.18 rows=13 width=37)
                                                              Hash Cond: (has_symptoms.symptom_id = types_of_symptoms.id)
                                                              ->  Seq Scan on has_symptoms  (cost=0.00..191.10 rows=13210 width=8)
                                                              ->  Hash  (cost=130.31..130.31 rows=5 width=37)
                                                                    ->  Seq Scan on types_of_symptoms  (cost=0.00..130.31 rows=5 width=37)
                                                                          Filter: ((name)::text = ANY ('{Frank_Sinatra,William_Shakespeare,Elizabeth_II,Adolf_Hitler,George_W._Bush}'::text[]))
                                                        ->  Index Scan using diseases_pkey on diseases diseases_1  (cost=0.28..0.32 rows=1 width=26)
                                                              Index Cond: (id = has_symptoms.disease_id)
  ->  Index Scan using prevention_pkey on prevention  (cost=0.28..0.39 rows=1 width=111)
        Index Cond: (disease_id = diseases.id)
(27 rows)

Time: 1.838 ms
```

Query 28

```
                                                                 QUERY PLAN
---------------------------------------------------------------------------------------------------------------------------------------------
 Unique  (cost=3.67..150.45 rows=1 width=658)
   ->  Nested Loop  (cost=3.67..150.45 rows=10 width=658)
         ->  Nested Loop  (cost=3.39..147.35 rows=10 width=622)
               Join Filter: (diseases.id = modes_of_transmission.disease_id)
               ->  Nested Loop  (cost=3.11..145.06 rows=7 width=646)
                     Join Filter: (diseases.id = prevention.disease_id)
                     ->  Nested Loop  (cost=2.82..142.68 rows=7 width=535)
                           ->  Nested Loop  (cost=2.54..139.70 rows=7 width=347)
                                 Join Filter: (diseases.id = identification_test.disease_id)
                                 ->  Nested Loop  (cost=2.26..137.44 rows=7 width=339)
                                       ->  Nested Loop  (cost=1.98..135.06 rows=7 width=326)
                                             ->  Nested Loop  (cost=1.69..132.83 rows=7 width=297)
                                                   Join Filter: (diseases.id = has_symptoms.disease_id)
                                                   ->  Nested Loop  (cost=1.41..132.00 rows=2 width=289)
                                                         ->  Nested Loop  (cost=1.13..131.38 rows=2 width=245)
                                                               Join Filter: (diseases.id = has_treatment.disease_id)
                                                               ->  Nested Loop  (cost=0.85..130.97 rows=1 width=237)
                                                                     Join Filter: (diseases.id = has_vaccine.disease_id)
                                                                     ->  Nested Loop  (cost=0.56..130.63 rows=1 width=229)
                                                                           ->  Nested Loop  (cost=0.28..130.25 rows=1 width=134)
                                                                                 ->  Seq Scan on diseases  (cost=0.00..121.94 rows=1 width=90)
                                                                                       Filter: ((name)::text = 'disease_name'::text)
                                                                                 ->  Index Scan using has_medicine_pkey on has_medicine  (cost=0.28..8.30 rows=1 width=44)
                                                                                       Index Cond: (disease_id = diseases.id)
                                                                           ->  Index Scan using medicine_pkey on medicine  (cost=0.28..0.38 rows=1 width=103)
                                                                                 Index Cond: (id = has_medicine.medicine_id)
                                                                     ->  Index Only Scan using has_vaccine_pkey on has_vaccine  (cost=0.28..0.33 rows=1 width=8)
                                                                           Index Cond: (disease_id = has_medicine.disease_id)
                                                               ->  Index Only Scan using has_treatment_pkey on has_treatment  (cost=0.29..0.39 rows=2 width=8)
                                                                     Index Cond: (disease_id = has_medicine.disease_id)
                                                         ->  Index Scan using treatment_pkey on treatment  (cost=0.28..0.31 rows=1 width=52)
                                                               Index Cond: (id = has_treatment.treatment_id)
                                                   ->  Index Only Scan using has_symptoms_pkey on has_symptoms  (cost=0.29..0.38 rows=3 width=8)
                                                         Index Cond: (disease_id = has_treatment.disease_id)
                                             ->  Index Scan using types_of_symptoms_pkey on types_of_symptoms  (cost=0.28..0.32 rows=1 width=37)
                                                   Index Cond: (id = has_symptoms.symptom_id)
                                       ->  Index Scan using vaccines_pkey on vaccines  (cost=0.28..0.34 rows=1 width=21)
                                             Index Cond: (id = has_vaccine.vaccine_id)
                                 ->  Index Only Scan using identification_test_pkey on identification_test  (cost=0.28..0.31 rows=1 width=8)
                                       Index Cond: (disease_id = has_symptoms.disease_id)
                           ->  Index Scan using types_of_tests_pkey on types_of_tests  (cost=0.28..0.43 rows=1 width=196)
                                 Index Cond: (id = identification_test.test_id)
                     ->  Index Scan using prevention_pkey on prevention  (cost=0.28..0.33 rows=1 width=111)
                           Index Cond: (disease_id = has_symptoms.disease_id)
               ->  Index Only Scan using modes_of_transmission_pkey on modes_of_transmission  (cost=0.28..0.31 rows=1 width=8)
                     Index Cond: (disease_id = has_symptoms.disease_id)
         ->  Index Scan using transmission_modes_pkey on transmission_modes  (cost=0.28..0.31 rows=1 width=44)
               Index Cond: (id = modes_of_transmission.transmission_id)
(48 rows)

Time: 13.024 ms
```

After using the indexing, we optimized the queries very well It reduced the cost approximately 10 times.

Performance is increased by 10 times by using the indexing.