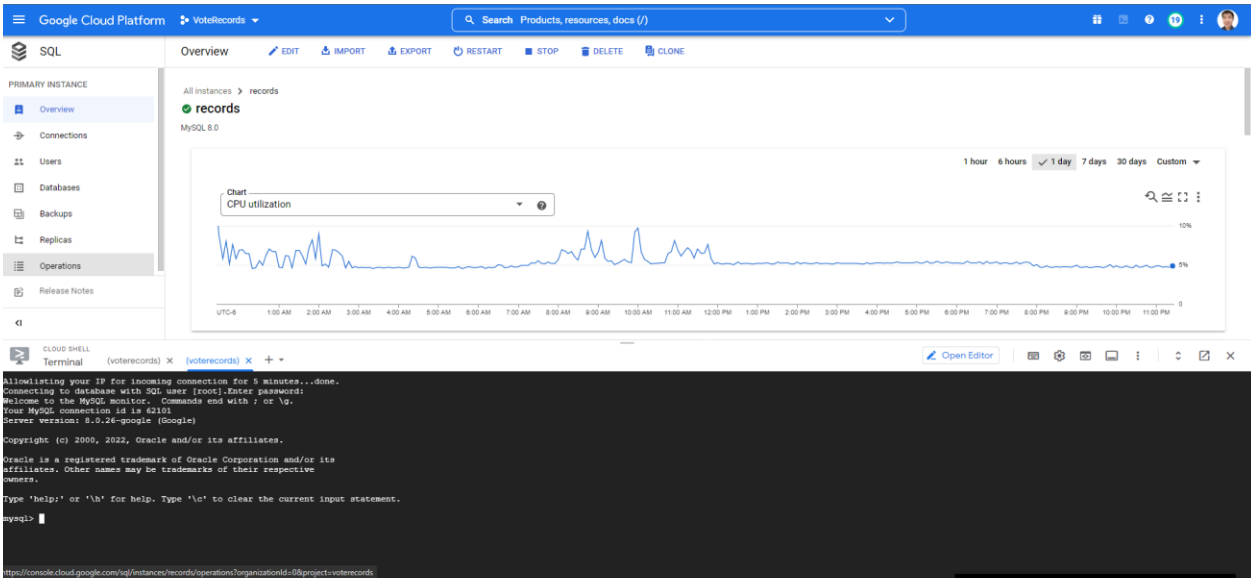
2.1 Screenshot of connection to GCP



```
Allowlisting your IP for incoming connection for 5 minutes...done.
Connecting to database with SQL user [root].Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 62101
Server version: 8.0.26-google (Google)

Copyright (c) 2000, 2022, Oracle and/or its affiliates.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql> 
```

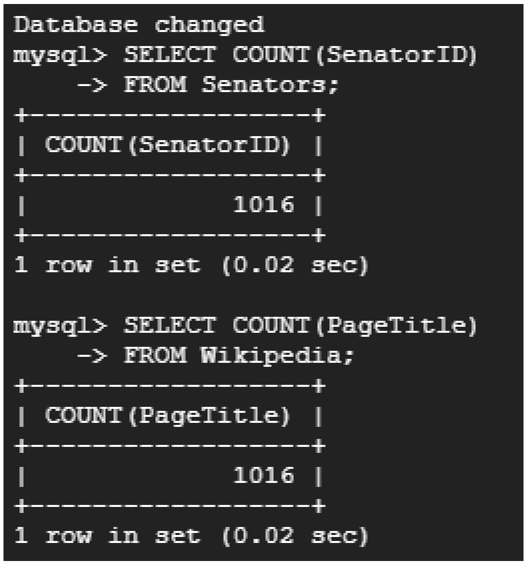
https://console.cloud.google.com/sql/instances/records/operations?organizationId=0&project=voterecords

2.2 DDL Commands

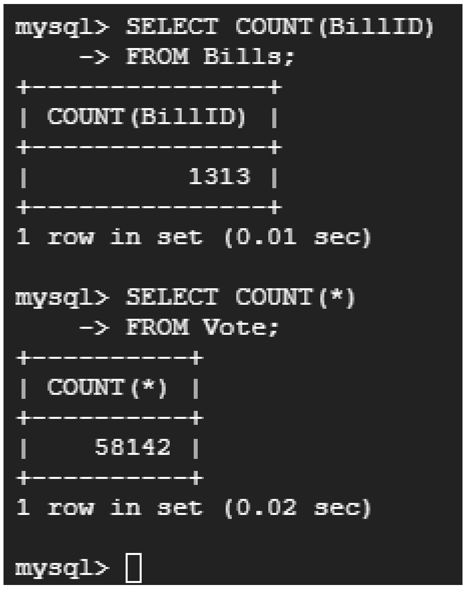
```
CREATE TABLE Senators(
  SenatorID INTEGER NOT NULL,
  Name VARCHAR(255) NOT NULL,
  BirthYear INTEGER,
  PRIMARY KEY (SenatorID)
 );
 CREATE TABLE Wikipedia(
   PageTitle VARCHAR(255) NOT NULL,
   PageURL VARCHAR(1024),
   PRIMARY KEY (PageTitle)
 );
 CREATE TABLE Bills(
   BillID INTEGER NOT NULL,
   Date,
   Results VARCHAR(255),
   Description VARCHAR(1024),
   PRIMARY KEY (BillID)
 );
 CREATE TABLE Parties(
   PartyName VARCHAR(255) NOT NULL,
   YearFounded INTEGER,
   PRIMARY KEY (PartyName)
 );
 CREATE TABLE States(
   StateID CHAR(2) NOT NULL,
   StateName VARCHAR(32),
   DominantParty VARCHAR(255),
   PRIMARY KEY (StateID)
 );
```

```
CREATE TABLE Vote(
  SenatorID INTEGER NOT NULL,
  BillID INTEGER NOT NULL,
  VoteType VARCHAR(32),
  PRIMARY KEY (SenatorID, BillID),
  FOREIGN KEY (SenatorID) REFERENCES Senators(SenatorID),
  FOREIGN KEY (BillID) REFERENCES Bills(BillID)
);
CREATE TABLE AffiliatedTo(
  SenatorID INTEGER NOT NULL,
  PartyName VARCHAR(255) NOT NULL,
  PRIMARY KEY (SenatorID, PartyName),
  FOREIGN KEY (SenatorID) REFERENCES Senators(SenatorID),
  FOREIGN KEY (PartyName) REFERENCES Parties(PartyName)
);
CREATE TABLE FromState(
  SenatorID INTEGER NOT NULL,
  StateID CHAR(2) NOT NULL,
  PRIMARY KEY (SenatorID, StateID),
  FOREIGN KEY (SenatorID) REFERENCES Senators(SenatorID),
  FOREIGN KEY (StateID) REFERENCES States(StateID)
);
CREATE TABLE LooksLike(
  SenatorID INTEGER,
  PageTitle VARCHAR(255),
  PRIMARY KEY (SenatorID, PageTitle),
  FOREIGN KEY (SenatorID) REFERENCES Senators(SenatorID),
  FOREIGN KEY (PageTitle) REFERENCES Wikipedia(PageTitle)
);
```

2.3 1000 rows on four tables

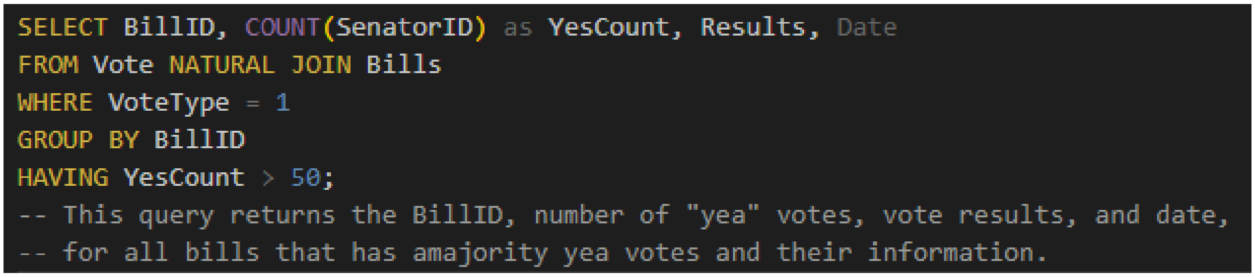


```
Database changed
mysql> SELECT COUNT(SenatorID)
    -> FROM Senators;
+------------------+
| COUNT(SenatorID) |
+------------------+
|             1016 |
+------------------+
1 row in set (0.02 sec)

mysql> SELECT COUNT(PageTitle)
    -> FROM Wikipedia;
+------------------+
| COUNT(PageTitle) |
+------------------+
|             1016 |
+------------------+
1 row in set (0.02 sec)
```



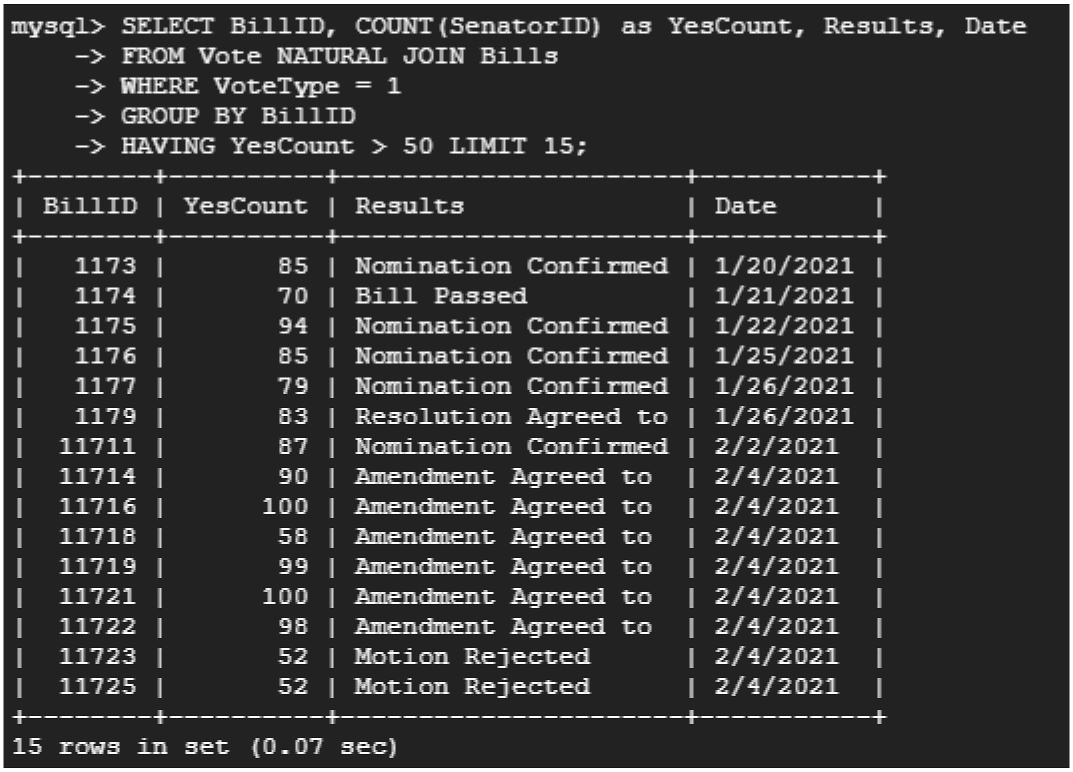
```
mysql> SELECT COUNT(BillID)
    -> FROM Bills;
+---------------+
| COUNT(BillID) |
+---------------+
|          1313 |
+---------------+
1 row in set (0.01 sec)

mysql> SELECT COUNT(*)
    -> FROM Vote;
+----------+
| COUNT(*) |
+----------+
|    58142 |
+----------+
1 row in set (0.02 sec)

mysql> 
```

3.1 SQL Query #1



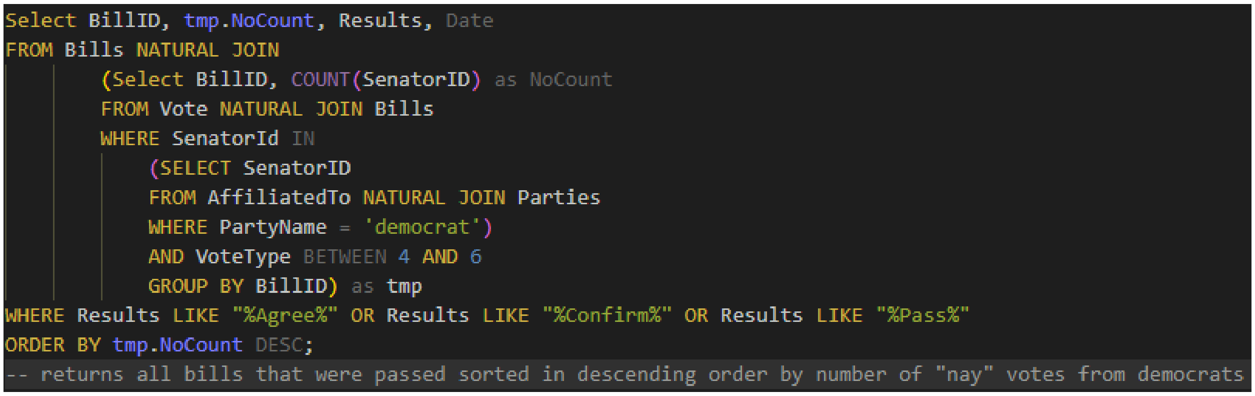
```
SELECT BillID, COUNT(SenatorID) as YesCount, Results, Date
FROM Vote NATURAL JOIN Bills
WHERE VoteType = 1
GROUP BY BillID
HAVING YesCount > 50;
-- This query returns the BillID, number of "yea" votes, vote results, and date,
-- for all bills that has amajority yea votes and their information.
```

Top 15 rows



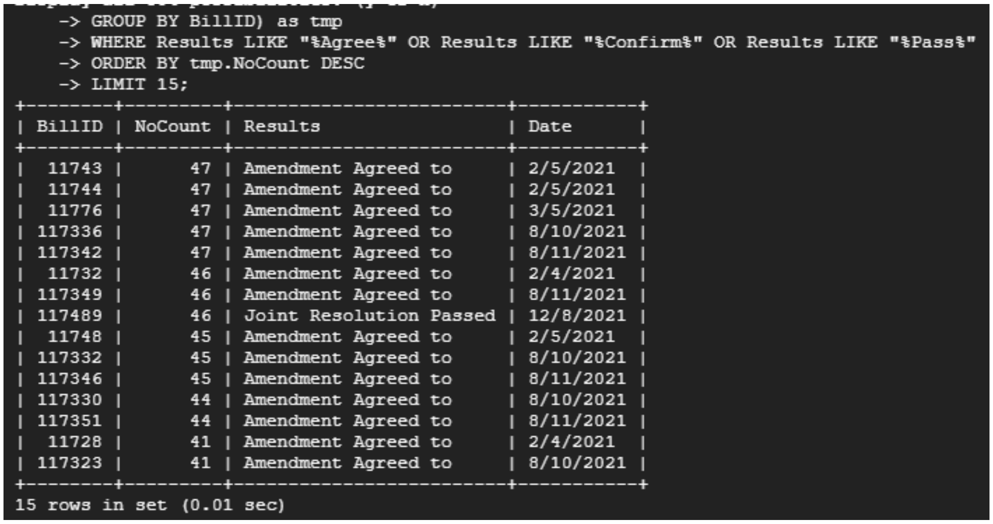
```
mysql> SELECT BillID, COUNT(SenatorID) as YesCount, Results, Date
    -> FROM Vote NATURAL JOIN Bills
    -> WHERE VoteType = 1
    -> GROUP BY BillID
    -> HAVING YesCount > 50 LIMIT 15;
+--------+----------+----------------------+-----------+
| BillID | YesCount | Results              | Date      |
+--------+----------+----------------------+-----------+
|   1173 |       85 | Nomination Confirmed | 1/20/2021 |
|   1174 |       70 | Bill Passed          | 1/21/2021 |
|   1175 |       94 | Nomination Confirmed | 1/22/2021 |
|   1176 |       85 | Nomination Confirmed | 1/25/2021 |
|   1177 |       79 | Nomination Confirmed | 1/26/2021 |
|   1179 |       83 | Resolution Agreed to | 1/26/2021 |
|  11711 |       87 | Nomination Confirmed | 2/2/2021  |
|  11714 |       90 | Amendment Agreed to  | 2/4/2021  |
|  11716 |      100 | Amendment Agreed to  | 2/4/2021  |
|  11718 |       58 | Amendment Agreed to  | 2/4/2021  |
|  11719 |       99 | Amendment Agreed to  | 2/4/2021  |
|  11721 |      100 | Amendment Agreed to  | 2/4/2021  |
|  11722 |       98 | Amendment Agreed to  | 2/4/2021  |
|  11723 |       52 | Motion Rejected      | 2/4/2021  |
|  11725 |       52 | Motion Rejected      | 2/4/2021  |
+--------+----------+----------------------+-----------+
15 rows in set (0.07 sec)
```

3.2 SQL Query #2



```
Select BillID, tmp.NoCount, Results, Date
FROM Bills NATURAL JOIN
        (Select BillID, COUNT(SenatorID) as NoCount
        FROM Vote NATURAL JOIN Bills
        WHERE SenatorId IN
            (SELECT SenatorID
            FROM AffiliatedTo NATURAL JOIN Parties
            WHERE PartyName = 'democrat')
            AND VoteType BETWEEN 4 AND 6
            GROUP BY BillID) as tmp
WHERE Results LIKE "%Agree%" OR Results LIKE "%Confirm%" OR Results LIKE "%Pass%"
ORDER BY tmp.NoCount DESC;
-- returns all bills that were passed sorted in descending order by number of "nay" votes from democrats
```

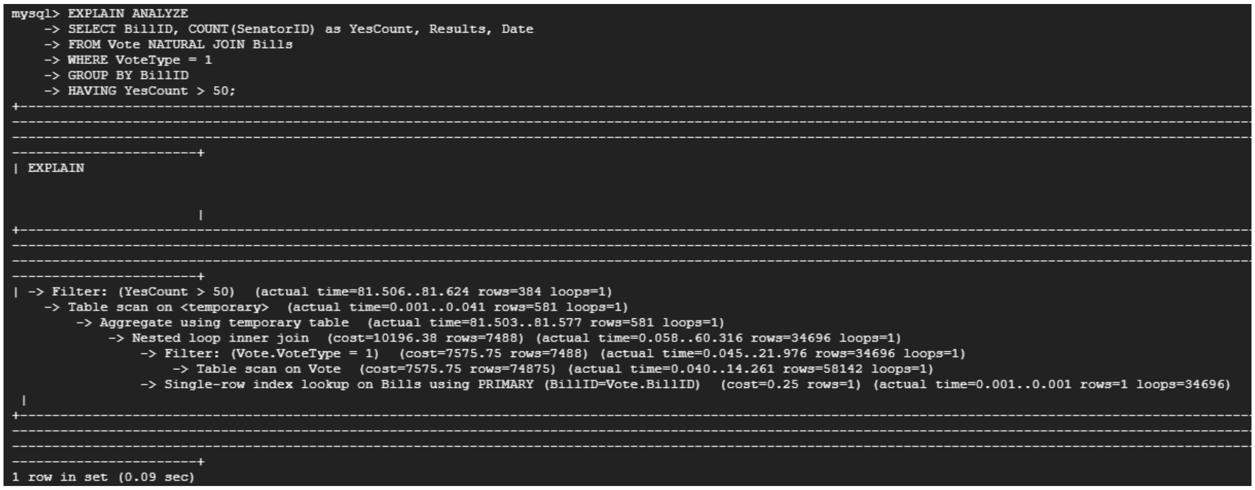
Top 15 rows



```
    -> GROUP BY BillID) as tmp
    -> WHERE Results LIKE "%Agree%" OR Results LIKE "%Confirm%" OR Results LIKE "%Pass%"
    -> ORDER BY tmp.NoCount DESC
    -> LIMIT 15;
+--------+---------+------------------------+-----------+
| BillID | NoCount | Results                | Date      |
+--------+---------+------------------------+-----------+
|  11743 |      47 | Amendment Agreed to    | 2/5/2021  |
|  11744 |      47 | Amendment Agreed to    | 2/5/2021  |
|  11776 |      47 | Amendment Agreed to    | 3/5/2021  |
| 117336 |      47 | Amendment Agreed to    | 8/10/2021 |
| 117342 |      47 | Amendment Agreed to    | 8/11/2021 |
|  11732 |      46 | Amendment Agreed to    | 2/4/2021  |
| 117349 |      46 | Amendment Agreed to    | 8/11/2021 |
| 117489 |      46 | Joint Resolution Passed | 12/8/2021 |
|  11748 |      45 | Amendment Agreed to    | 2/5/2021  |
| 117332 |      45 | Amendment Agreed to    | 8/10/2021 |
| 117346 |      45 | Amendment Agreed to    | 8/11/2021 |
| 117330 |      44 | Amendment Agreed to    | 8/10/2021 |
| 117351 |      44 | Amendment Agreed to    | 8/11/2021 |
|  11728 |      41 | Amendment Agreed to    | 2/4/2021  |
| 117323 |      41 | Amendment Agreed to    | 8/10/2021 |
+--------+---------+------------------------+-----------+
15 rows in set (0.01 sec)
```

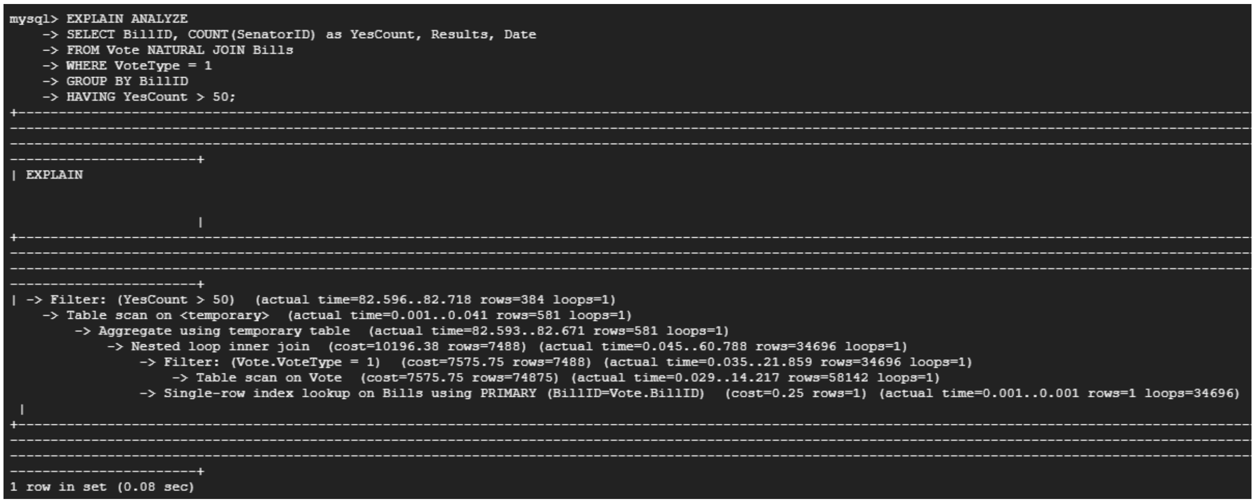
4.1.a Indexing Analysis for Query #1

Performance Before Indexing



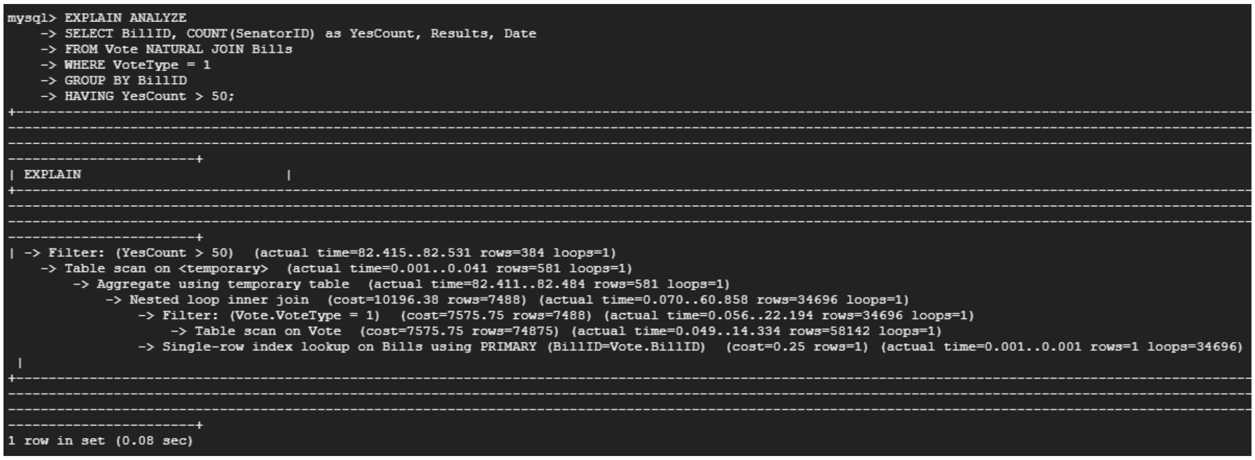
```
mysql> EXPLAIN ANALYZE
    -> SELECT BillID, COUNT(SenatorID) as YesCount, Results, Date
    -> FROM Vote NATURAL JOIN Bills
    -> WHERE VoteType = 1
    -> GROUP BY BillID
    -> HAVING YesCount > 50;
+----------------------------------------------------------------------------------------------------------------------------
------------------------------------------------------------------------------------------------------------------------------
------------------------------------------------------------------------------------------------------------------------------
-----------------------+
| EXPLAIN


                       |
+----------------------------------------------------------------------------------------------------------------------------
------------------------------------------------------------------------------------------------------------------------------
------------------------------------------------------------------------------------------------------------------------------
-----------------------+
| -> Filter: (YesCount > 50)  (actual time=81.506..81.624 rows=384 loops=1)
    -> Table scan on <temporary>  (actual time=0.001..0.041 rows=581 loops=1)
        -> Aggregate using temporary table  (actual time=81.503..81.577 rows=581 loops=1)
            -> Nested loop inner join  (cost=10196.38 rows=7488) (actual time=0.058..60.316 rows=34696 loops=1)
                -> Filter: (Vote.VoteType = 1)  (cost=7575.75 rows=7488) (actual time=0.045..21.976 rows=34696 loops=1)
                    -> Table scan on Vote  (cost=7575.75 rows=74875) (actual time=0.040..14.261 rows=58142 loops=1)
                -> Single-row index lookup on Bills using PRIMARY (BillID=Vote.BillID)  (cost=0.25 rows=1) (actual time=0.001..0.001 rows=1 loops=34696)
 |
+----------------------------------------------------------------------------------------------------------------------------
------------------------------------------------------------------------------------------------------------------------------
------------------------------------------------------------------------------------------------------------------------------
-----------------------+
1 row in set (0.09 sec)
```

1. CREATE INDEX idx1 ON Bills (BillID);



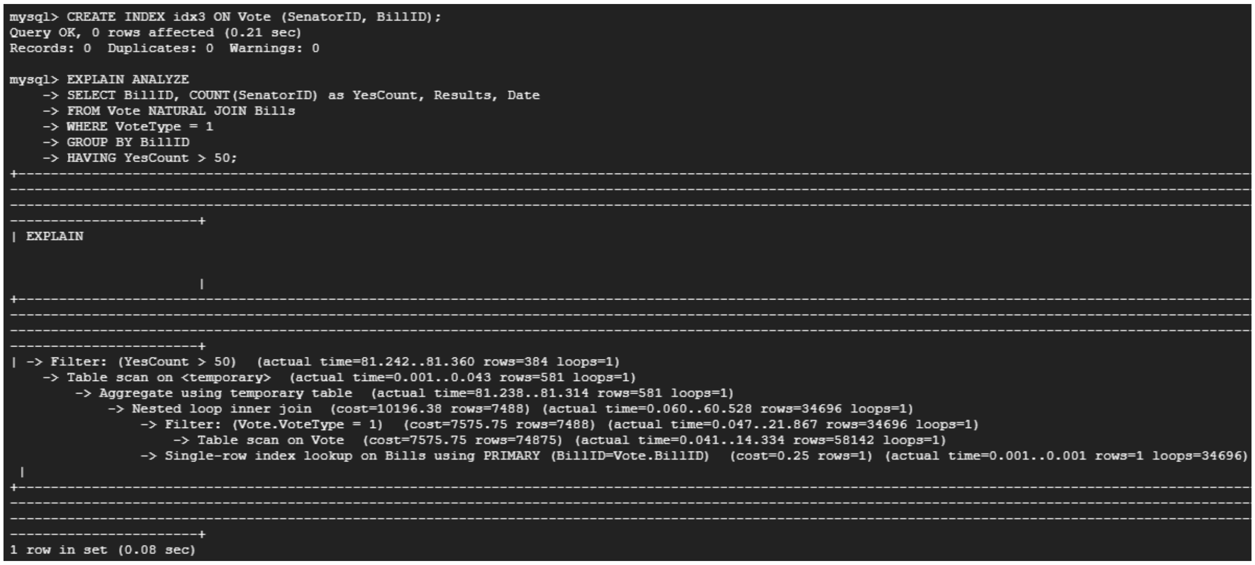
```
mysql> EXPLAIN ANALYZE
    -> SELECT BillID, COUNT(SenatorID) as YesCount, Results, Date
    -> FROM Vote NATURAL JOIN Bills
    -> WHERE VoteType = 1
    -> GROUP BY BillID
    -> HAVING YesCount > 50;
+----------------------------------------------------------------------------------------------------------------------------
------------------------------------------------------------------------------------------------------------------------------
------------------------------------------------------------------------------------------------------------------------------
-----------------------+
| EXPLAIN


                       |
+----------------------------------------------------------------------------------------------------------------------------
------------------------------------------------------------------------------------------------------------------------------
------------------------------------------------------------------------------------------------------------------------------
-----------------------+
| -> Filter: (YesCount > 50)  (actual time=82.596..82.718 rows=384 loops=1)
    -> Table scan on <temporary>  (actual time=0.001..0.041 rows=581 loops=1)
        -> Aggregate using temporary table  (actual time=82.593..82.671 rows=581 loops=1)
            -> Nested loop inner join  (cost=10196.38 rows=7488) (actual time=0.045..60.788 rows=34696 loops=1)
                -> Filter: (Vote.VoteType = 1)  (cost=7575.75 rows=7488) (actual time=0.035..21.859 rows=34696 loops=1)
                    -> Table scan on Vote  (cost=7575.75 rows=74875) (actual time=0.029..14.217 rows=58142 loops=1)
                -> Single-row index lookup on Bills using PRIMARY (BillID=Vote.BillID)  (cost=0.25 rows=1) (actual time=0.001..0.001 rows=1 loops=34696)
 |
+----------------------------------------------------------------------------------------------------------------------------
------------------------------------------------------------------------------------------------------------------------------
------------------------------------------------------------------------------------------------------------------------------
-----------------------+
1 row in set (0.08 sec)
```

2. CREATE INDEX idx2 ON Vote (SenatorID);



```
mysql> EXPLAIN ANALYZE
    -> SELECT BillID, COUNT(SenatorID) as YesCount, Results, Date
    -> FROM Vote NATURAL JOIN Bills
    -> WHERE VoteType = 1
    -> GROUP BY BillID
    -> HAVING YesCount > 50;
+-----------------------------------------------------------------------------------------------------------------------------------------------------
------------------------------------------------------------------------------------------------------------------------------------------------------
------------------------------------------------------------------------------------------------------------------------------------------------------
-----------------------+
| EXPLAIN                             |
+-----------------------------------------------------------------------------------------------------------------------------------------------------
------------------------------------------------------------------------------------------------------------------------------------------------------
------------------------------------------------------------------------------------------------------------------------------------------------------
-----------------------+
| -> Filter: (YesCount > 50)  (actual time=82.415..82.531 rows=384 loops=1)
    -> Table scan on <temporary>  (actual time=0.001..0.041 rows=581 loops=1)
        -> Aggregate using temporary table  (actual time=82.411..82.484 rows=581 loops=1)
            -> Nested loop inner join  (cost=10196.38 rows=7488) (actual time=0.070..60.858 rows=34696 loops=1)
                -> Filter: (Vote.VoteType = 1)  (cost=7575.75 rows=7488) (actual time=0.056..22.194 rows=34696 loops=1)
                    -> Table scan on Vote  (cost=7575.75 rows=74875) (actual time=0.049..14.334 rows=58142 loops=1)
                -> Single-row index lookup on Bills using PRIMARY (BillID=Vote.BillID)  (cost=0.25 rows=1) (actual time=0.001..0.001 rows=1 loops=34696)
 |
+-----------------------------------------------------------------------------------------------------------------------------------------------------
------------------------------------------------------------------------------------------------------------------------------------------------------
------------------------------------------------------------------------------------------------------------------------------------------------------
-----------------------+
1 row in set (0.08 sec)
```

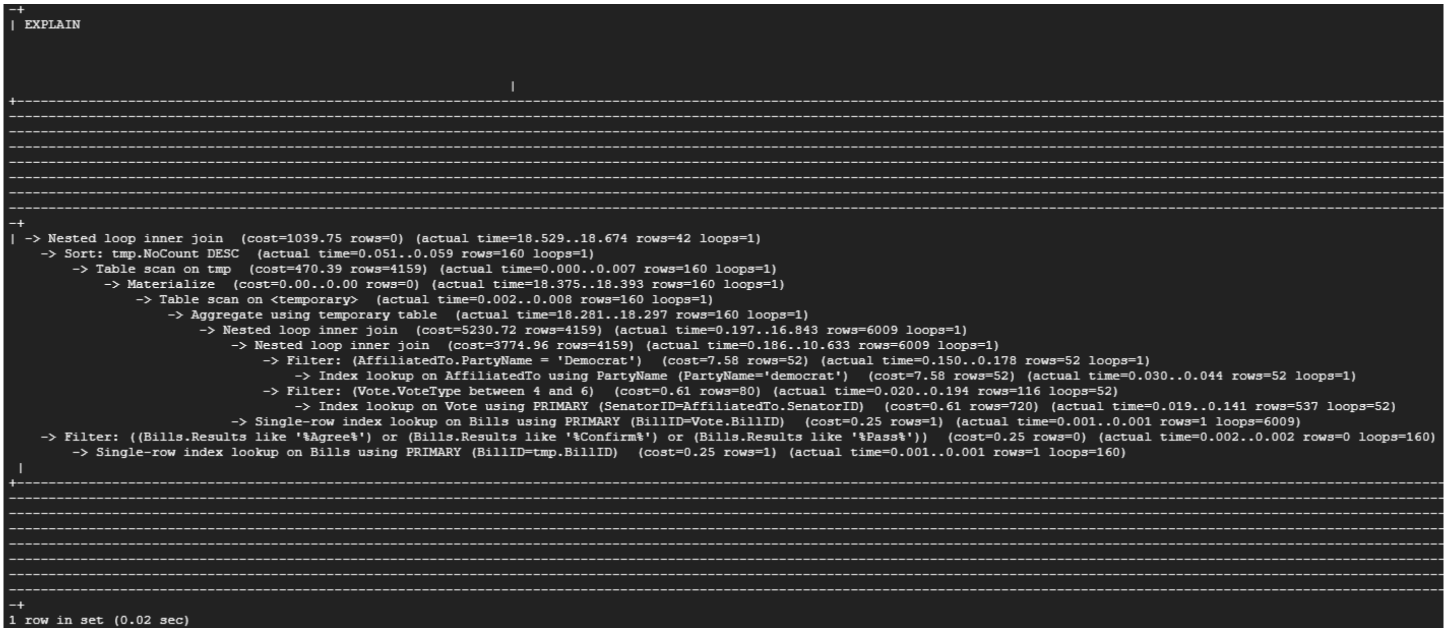
3. CREATE INDEX idx3 ON Vote (SenatorID, BillID);



```
mysql> CREATE INDEX idx3 ON Vote (SenatorID, BillID);
Query OK, 0 rows affected (0.21 sec)
Records: 0  Duplicates: 0  Warnings: 0

mysql> EXPLAIN ANALYZE
    -> SELECT BillID, COUNT(SenatorID) as YesCount, Results, Date
    -> FROM Vote NATURAL JOIN Bills
    -> WHERE VoteType = 1
    -> GROUP BY BillID
    -> HAVING YesCount > 50;
+-----------------------------------------------------------------------------------------------------------------------------------------------------
------------------------------------------------------------------------------------------------------------------------------------------------------
------------------------------------------------------------------------------------------------------------------------------------------------------
-----------------------+
| EXPLAIN


                        |
+-----------------------------------------------------------------------------------------------------------------------------------------------------
------------------------------------------------------------------------------------------------------------------------------------------------------
------------------------------------------------------------------------------------------------------------------------------------------------------
-----------------------+
| -> Filter: (YesCount > 50)  (actual time=81.242..81.360 rows=384 loops=1)
    -> Table scan on <temporary>  (actual time=0.001..0.043 rows=581 loops=1)
        -> Aggregate using temporary table  (actual time=81.238..81.314 rows=581 loops=1)
            -> Nested loop inner join  (cost=10196.38 rows=7488) (actual time=0.060..60.528 rows=34696 loops=1)
                -> Filter: (Vote.VoteType = 1)  (cost=7575.75 rows=7488) (actual time=0.047..21.867 rows=34696 loops=1)
                    -> Table scan on Vote  (cost=7575.75 rows=74875) (actual time=0.041..14.334 rows=58142 loops=1)
                -> Single-row index lookup on Bills using PRIMARY (BillID=Vote.BillID)  (cost=0.25 rows=1) (actual time=0.001..0.001 rows=1 loops=34696)
 |
+-----------------------------------------------------------------------------------------------------------------------------------------------------
------------------------------------------------------------------------------------------------------------------------------------------------------
------------------------------------------------------------------------------------------------------------------------------------------------------
-----------------------+
1 row in set (0.08 sec)
```

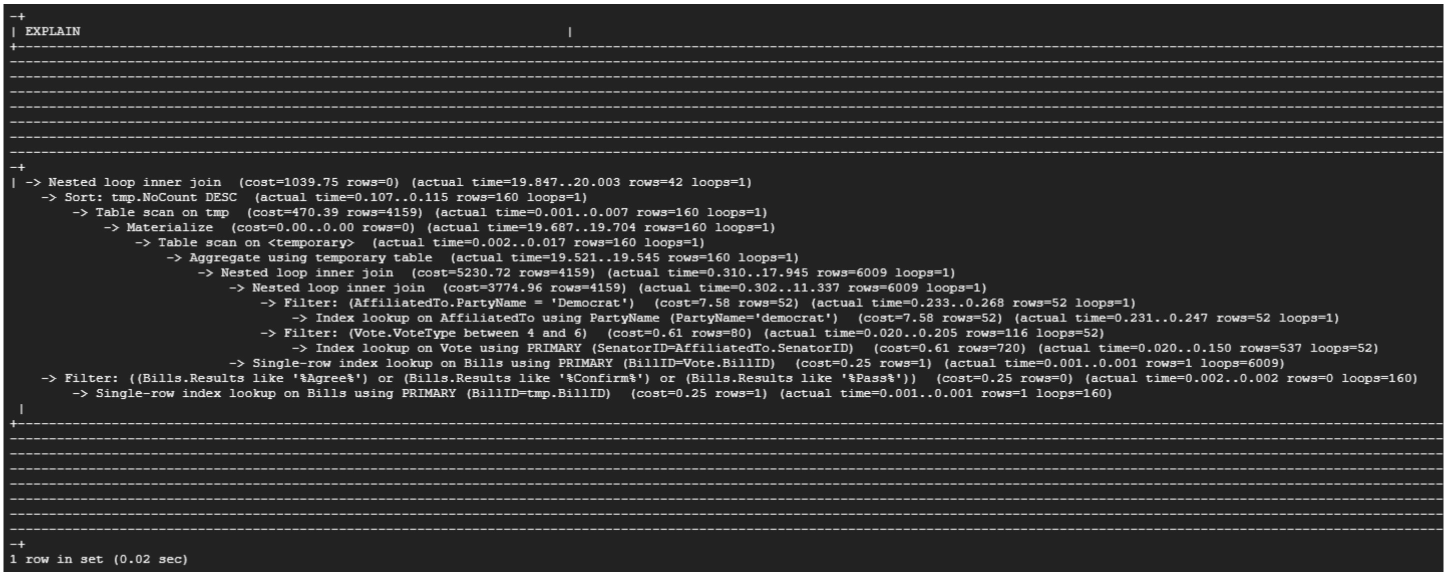
4.1.b Indexing Analysis for Query #2

Performance Before Indexing



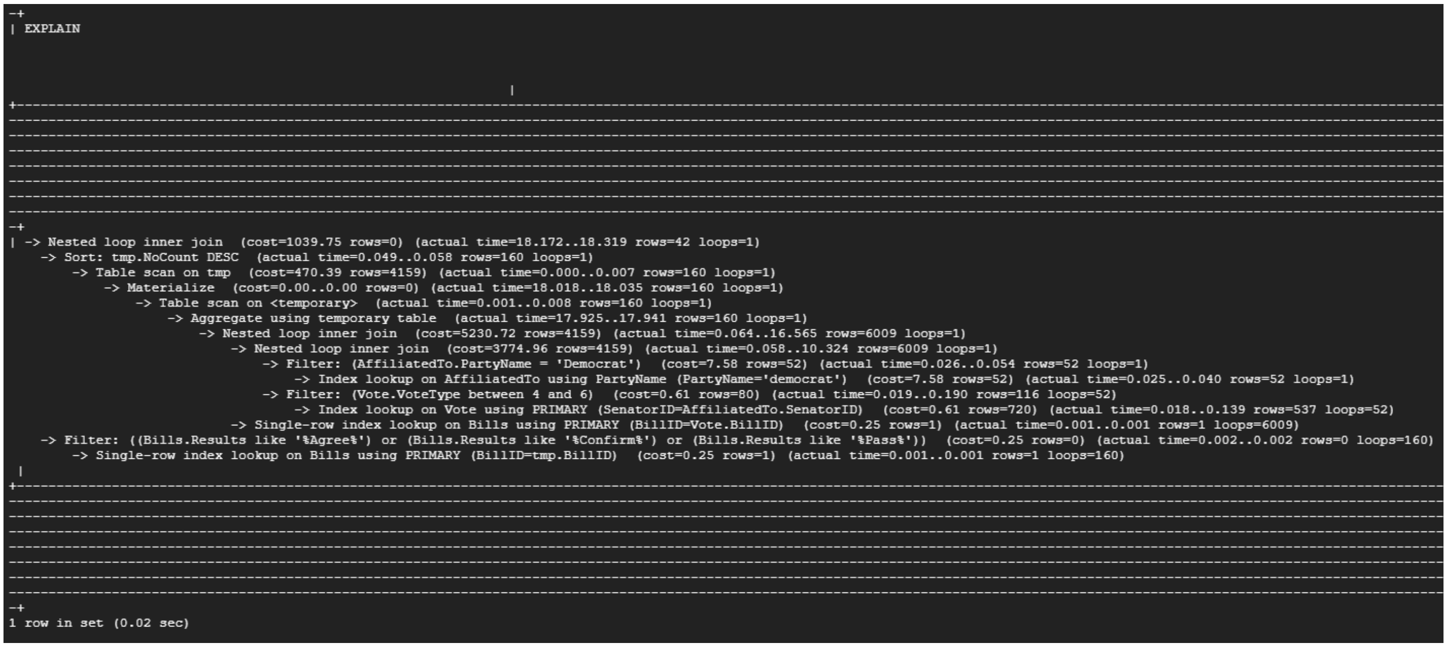
```
-+
| EXPLAIN



                                                              |
+-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------
--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------
--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------
--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------
--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------
--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------
--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------
--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------
-+
| -> Nested loop inner join  (cost=1039.75 rows=0) (actual time=18.529..18.674 rows=42 loops=1)
    -> Sort: tmp.NoCount DESC  (actual time=0.051..0.059 rows=160 loops=1)
        -> Table scan on tmp  (cost=470.39 rows=4159) (actual time=0.000..0.007 rows=160 loops=1)
            -> Materialize  (cost=0.00..0.00 rows=0) (actual time=18.375..18.393 rows=160 loops=1)
                -> Table scan on <temporary>  (actual time=0.002..0.008 rows=160 loops=1)
                    -> Aggregate using temporary table  (actual time=18.281..18.297 rows=160 loops=1)
                        -> Nested loop inner join  (cost=5230.72 rows=4159) (actual time=0.197..16.843 rows=6009 loops=1)
                            -> Nested loop inner join  (cost=3774.96 rows=4159) (actual time=0.186..10.633 rows=6009 loops=1)
                                -> Filter: (AffiliatedTo.PartyName = 'Democrat')  (cost=7.58 rows=52) (actual time=0.150..0.178 rows=52 loops=1)
                                    -> Index lookup on AffiliatedTo using PartyName (PartyName='democrat')  (cost=7.58 rows=52) (actual time=0.030..0.044 rows=52 loops=1)
                                -> Filter: (Vote.VoteType between 4 and 6)  (cost=0.61 rows=80) (actual time=0.020..0.194 rows=116 loops=52)
                                    -> Index lookup on Vote using PRIMARY (SenatorID=AffiliatedTo.SenatorID)  (cost=0.61 rows=720) (actual time=0.019..0.141 rows=537 loops=52)
                            -> Single-row index lookup on Bills using PRIMARY (BillID=Vote.BillID)  (cost=0.25 rows=1) (actual time=0.001..0.001 rows=1 loops=6009)
    -> Filter: ((Bills.Results like '%Agree%') or (Bills.Results like '%Confirm%') or (Bills.Results like '%Pass%'))  (cost=0.25 rows=0) (actual time=0.002..0.002 rows=0 loops=160)
        -> Single-row index lookup on Bills using PRIMARY (BillID=tmp.BillID)  (cost=0.25 rows=1) (actual time=0.001..0.001 rows=1 loops=160)
 |
+-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------
--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------
--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------
--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------
--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------
--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------
--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------
--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------
-+
1 row in set (0.02 sec)
```

1. CREATE INDEX idx1 ON Bills (BillID);



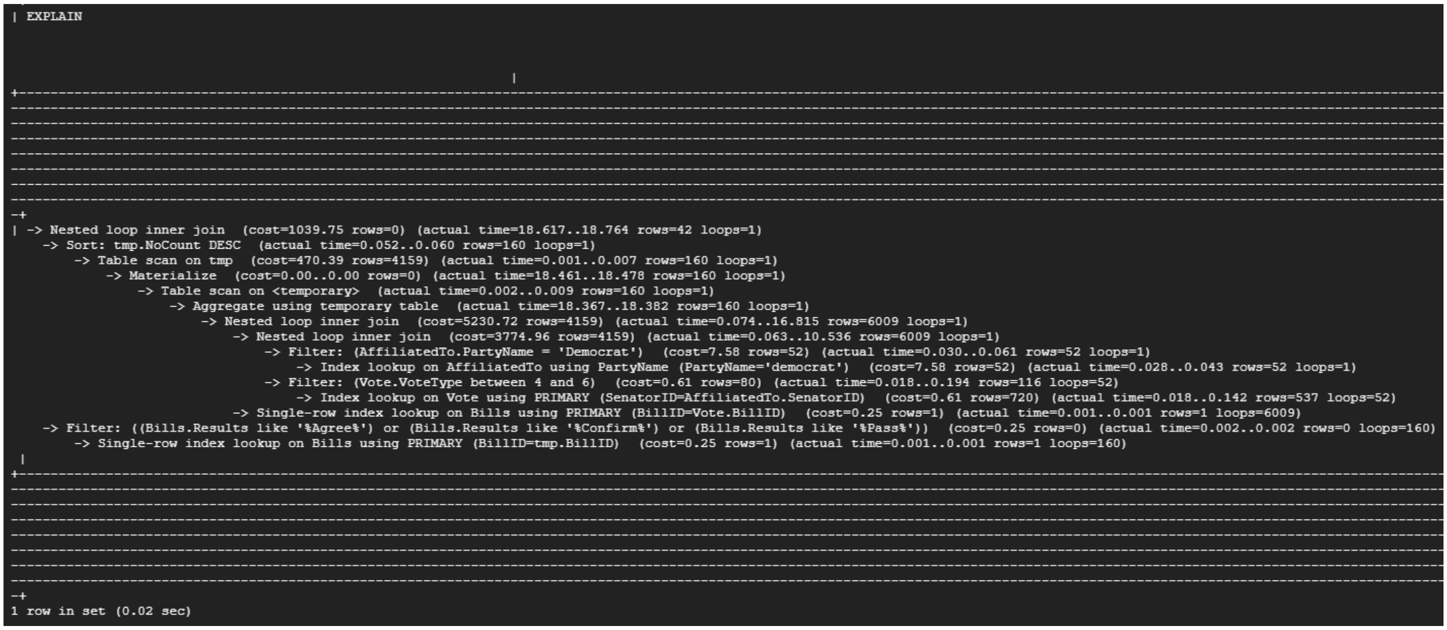
```
-+
| EXPLAIN                                                                  |
+----------------------------------------------------------------------------------------------------
----------------------------------------------------------------------------------------------------
----------------------------------------------------------------------------------------------------
----------------------------------------------------------------------------------------------------
----------------------------------------------------------------------------------------------------
----------------------------------------------------------------------------------------------------
----------------------------------------------------------------------------------------------------
----------------------------------------------------------------------------------------------------
-+
| -> Nested loop inner join  (cost=1039.75 rows=0) (actual time=19.847..20.003 rows=42 loops=1)
    -> Sort: tmp.NoCount DESC  (actual time=0.107..0.115 rows=160 loops=1)
        -> Table scan on tmp  (cost=470.39 rows=4159) (actual time=0.001..0.007 rows=160 loops=1)
            -> Materialize  (cost=0.00..0.00 rows=0) (actual time=19.687..19.704 rows=160 loops=1)
                -> Table scan on <temporary>  (actual time=0.002..0.017 rows=160 loops=1)
                    -> Aggregate using temporary table  (actual time=19.521..19.545 rows=160 loops=1)
                        -> Nested loop inner join  (cost=5230.72 rows=4159) (actual time=0.310..17.945 rows=6009 loops=1)
                            -> Nested loop inner join  (cost=3774.96 rows=4159) (actual time=0.302..11.337 rows=6009 loops=1)
                                -> Filter: (AffiliatedTo.PartyName = 'Democrat')  (cost=7.58 rows=52) (actual time=0.233..0.268 rows=52 loops=1)
                                    -> Index lookup on AffiliatedTo using PartyName (PartyName='democrat')  (cost=7.58 rows=52) (actual time=0.231..0.247 rows=52 loops=1)
                                -> Filter: (Vote.VoteType between 4 and 6)  (cost=0.61 rows=80) (actual time=0.020..0.205 rows=116 loops=52)
                                    -> Index lookup on Vote using PRIMARY (SenatorID=AffiliatedTo.SenatorID)  (cost=0.61 rows=720) (actual time=0.020..0.150 rows=537 loops=52)
                            -> Single-row index lookup on Bills using PRIMARY (BillID=Vote.BillID)  (cost=0.25 rows=1) (actual time=0.001..0.001 rows=1 loops=6009)
    -> Filter: ((Bills.Results like '%Agree%') or (Bills.Results like '%Confirm%') or (Bills.Results like '%Pass%'))  (cost=0.25 rows=0) (actual time=0.002..0.002 rows=0 loops=160)
        -> Single-row index lookup on Bills using PRIMARY (BillID=tmp.BillID)  (cost=0.25 rows=1) (actual time=0.001..0.001 rows=1 loops=160)
 |
+----------------------------------------------------------------------------------------------------
----------------------------------------------------------------------------------------------------
----------------------------------------------------------------------------------------------------
----------------------------------------------------------------------------------------------------
----------------------------------------------------------------------------------------------------
----------------------------------------------------------------------------------------------------
----------------------------------------------------------------------------------------------------
----------------------------------------------------------------------------------------------------
-+
1 row in set (0.02 sec)
```

2. CREATE INDEX idx2 ON Bills (BillID, Results);



```
-+
| EXPLAIN



                                                                     |
+----------------------------------------------------------------------------------------------------
----------------------------------------------------------------------------------------------------
----------------------------------------------------------------------------------------------------
----------------------------------------------------------------------------------------------------
----------------------------------------------------------------------------------------------------
----------------------------------------------------------------------------------------------------
----------------------------------------------------------------------------------------------------
----------------------------------------------------------------------------------------------------
-+
| -> Nested loop inner join  (cost=1039.75 rows=0) (actual time=18.172..18.319 rows=42 loops=1)
    -> Sort: tmp.NoCount DESC  (actual time=0.049..0.058 rows=160 loops=1)
        -> Table scan on tmp  (cost=470.39 rows=4159) (actual time=0.000..0.007 rows=160 loops=1)
            -> Materialize  (cost=0.00..0.00 rows=0) (actual time=18.018..18.035 rows=160 loops=1)
                -> Table scan on <temporary>  (actual time=0.001..0.008 rows=160 loops=1)
                    -> Aggregate using temporary table  (actual time=17.925..17.941 rows=160 loops=1)
                        -> Nested loop inner join  (cost=5230.72 rows=4159) (actual time=0.064..16.565 rows=6009 loops=1)
                            -> Nested loop inner join  (cost=3774.96 rows=4159) (actual time=0.058..10.324 rows=6009 loops=1)
                                -> Filter: (AffiliatedTo.PartyName = 'Democrat')  (cost=7.58 rows=52) (actual time=0.026..0.054 rows=52 loops=1)
                                    -> Index lookup on AffiliatedTo using PartyName (PartyName='democrat')  (cost=7.58 rows=52) (actual time=0.025..0.040 rows=52 loops=1)
                                -> Filter: (Vote.VoteType between 4 and 6)  (cost=0.61 rows=80) (actual time=0.019..0.190 rows=116 loops=52)
                                    -> Index lookup on Vote using PRIMARY (SenatorID=AffiliatedTo.SenatorID)  (cost=0.61 rows=720) (actual time=0.018..0.139 rows=537 loops=52)
                            -> Single-row index lookup on Bills using PRIMARY (BillID=Vote.BillID)  (cost=0.25 rows=1) (actual time=0.001..0.001 rows=1 loops=6009)
    -> Filter: ((Bills.Results like '%Agree%') or (Bills.Results like '%Confirm%') or (Bills.Results like '%Pass%'))  (cost=0.25 rows=0) (actual time=0.002..0.002 rows=0 loops=160)
        -> Single-row index lookup on Bills using PRIMARY (BillID=tmp.BillID)  (cost=0.25 rows=1) (actual time=0.001..0.001 rows=1 loops=160)
 |
+----------------------------------------------------------------------------------------------------
----------------------------------------------------------------------------------------------------
----------------------------------------------------------------------------------------------------
----------------------------------------------------------------------------------------------------
----------------------------------------------------------------------------------------------------
----------------------------------------------------------------------------------------------------
----------------------------------------------------------------------------------------------------
----------------------------------------------------------------------------------------------------
-+
1 row in set (0.02 sec)
```

3. CREATE INDEX idx3 ON Vote (SenatorID, BillID);



```
| EXPLAIN



                                                                |
+-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------
------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------
------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------
------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------
------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------
------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------
------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------
------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------
-+
| -> Nested loop inner join  (cost=1039.75 rows=0) (actual time=18.617..18.764 rows=42 loops=1)
    -> Sort: tmp.NoCount DESC  (actual time=0.052..0.060 rows=160 loops=1)
        -> Table scan on tmp  (cost=470.39 rows=4159) (actual time=0.001..0.007 rows=160 loops=1)
            -> Materialize  (cost=0.00..0.00 rows=0) (actual time=18.461..18.478 rows=160 loops=1)
                -> Table scan on <temporary>  (actual time=0.002..0.009 rows=160 loops=1)
                    -> Aggregate using temporary table  (actual time=18.367..18.382 rows=160 loops=1)
                        -> Nested loop inner join  (cost=5230.72 rows=4159) (actual time=0.074..16.815 rows=6009 loops=1)
                            -> Nested loop inner join  (cost=3774.96 rows=4159) (actual time=0.063..10.536 rows=6009 loops=1)
                                -> Filter: (AffiliatedTo.PartyName = 'Democrat')  (cost=7.58 rows=52) (actual time=0.030..0.061 rows=52 loops=1)
                                    -> Index lookup on AffiliatedTo using PartyName (PartyName='democrat')  (cost=7.58 rows=52) (actual time=0.028..0.043 rows=52 loops=1)
                                -> Filter: (Vote.VoteType between 4 and 6)  (cost=0.61 rows=80) (actual time=0.018..0.194 rows=116 loops=52)
                                    -> Index lookup on Vote using PRIMARY (SenatorID=AffiliatedTo.SenatorID)  (cost=0.61 rows=720) (actual time=0.018..0.142 rows=537 loops=52)
                            -> Single-row index lookup on Bills using PRIMARY (BillID=Vote.BillID)  (cost=0.25 rows=1) (actual time=0.001..0.001 rows=1 loops=6009)
    -> Filter: ((Bills.Results like '%Agree%') or (Bills.Results like '%Confirm%') or (Bills.Results like '%Pass%'))  (cost=0.25 rows=0) (actual time=0.002..0.002 rows=0 loops=160)
        -> Single-row index lookup on Bills using PRIMARY (BillID=tmp.BillID)  (cost=0.25 rows=1) (actual time=0.001..0.001 rows=1 loops=160)
 |
+-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------
------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------
------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------
------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------
------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------
------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------
------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------
------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------
-+
1 row in set (0.02 sec)
```