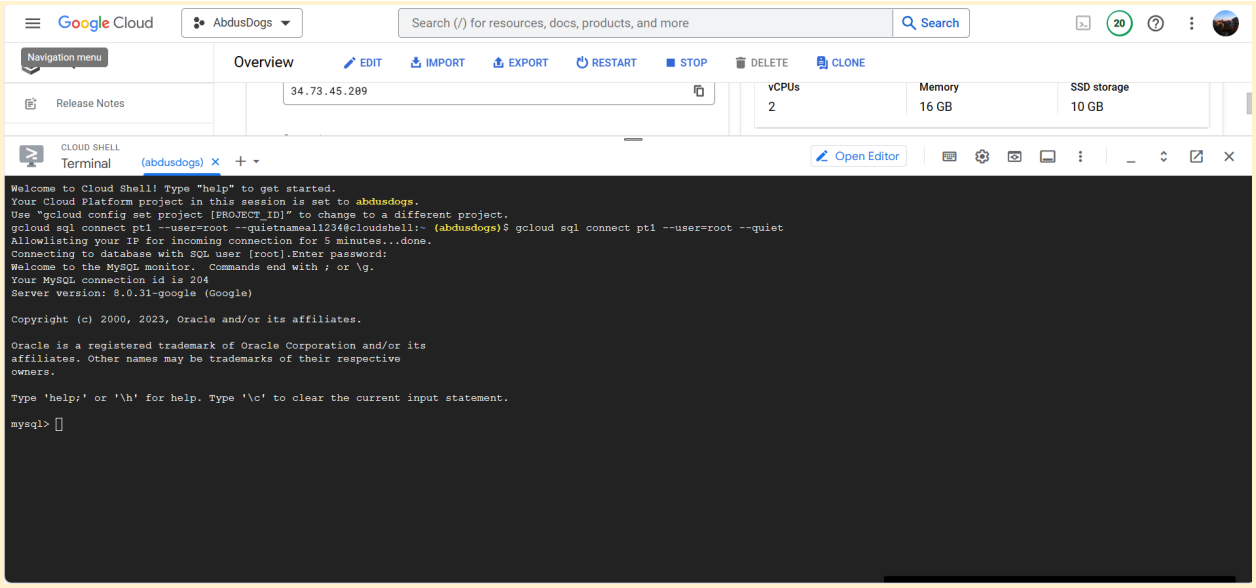
PROOF OF CONNECTION

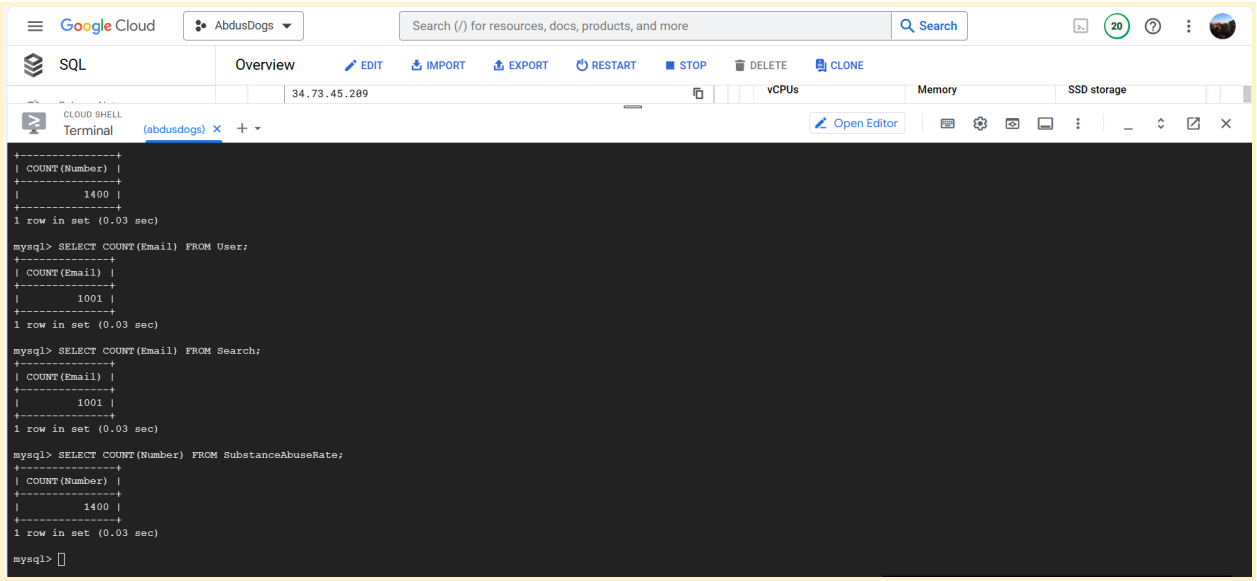


DDL

```
CREATE TABLE State
(
  StateName VARCHAR(255),
  Population DECIMAL(10),
  CorrelationCoefficient FLOAT,
  PRIMARY KEY (StateName)
);

CREATE TABLE Substance
(
  SubstanceName VARCHAR(255),
  SubstanceCategory INT,
  PRIMARY KEY (SubstanceName)
);

CREATE TABLE SubstanceAbuseRate
(
  AgeRange VARCHAR(255),
  Number INT,
  StateName VARCHAR(255),
  SubstanceName VARCHAR(255),
  PRIMARY KEY (AgeRange,StateName,SubstanceName),
  FOREIGN KEY (StateName) REFERENCES State(StateName),
  FOREIGN KEY (SubstanceName) REFERENCES Substance(SubstanceName)
);
```

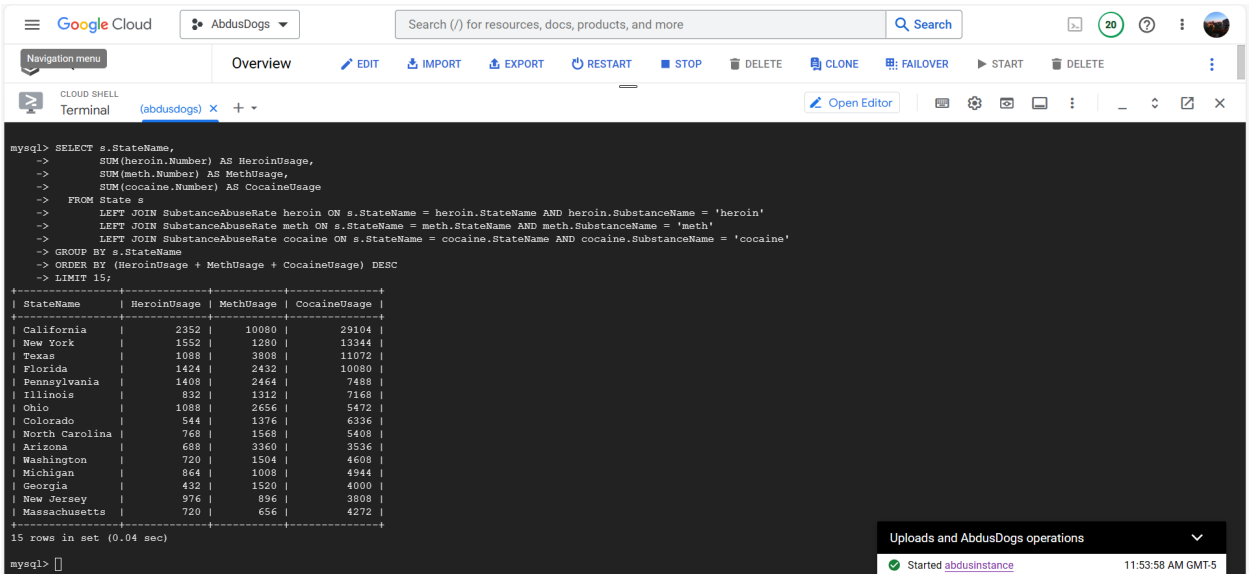
```
CREATE TABLE Pet
(
  Type VARCHAR(255),
  PercentHousehold DECIMAL(10,2),
  StateName VARCHAR(255),
  PRIMARY KEY (Type, StateName),
  FOREIGN KEY (StateName) REFERENCES State(StateName)
);

CREATE TABLE User
(
  Email VARCHAR(255),
  Password VARCHAR(255),
  PRIMARY KEY (Email)
);

CREATE TABLE Search
(
  SearchID INT,
  Email VARCHAR(255),
  Content VARCHAR(255),
  PRIMARY KEY (SearchID),
  FOREIGN KEY (Email) REFERENCES User(Email)
);
```

1000 ROWS



ADVANCED QUERIES/PROOF

TOP 15 STATES WITH HIGHEST HEROIN, COCAINE, METH USAGE




```
SELECT s.StateName, SUM(heroin.Number) AS HeroinUsage, SUM(meth.Number) AS
        MethUsage, SUM(cocaine.Number) AS CocaineUsage
FROM State s
        LEFT JOIN SubstanceAbuseRate as heroin ON s.StateName =
heroin.StateName
        LEFT JOIN SubstanceAbuseRate as meth on s.StateName = meth.StateName
        LEFT JOIN SubstanceAbuseRate as cocaine on s.StateName =
cocaine.StateName
WHERE heroin.SubstanceName = ‘heroin’ AND meth.SubstanceName = ‘meth’ AND
cocaine.SubstanceName = ‘cocaine’
GROUP BY s.StateName
ORDER BY (HeroinUsage + MethUsage + CocaineUsage) DESC LIMIT 15;
```

Correlation Coefficient Query

```
SELECT (SUM((P.PercentHousehold - PetAverageTable.petavg)*(S.Number -
        DrugAverageTable.drugavg))) / SQRT(SUM((P.PercentHousehold -
        PetAverageTable.petavg)*(P.PercentHousehold - PetAverageTable.petavg)) * SUM((S.Number -
        DrugAverageTable.drugavg)*(S.Number - DrugAverageTable.drugavg))) as cc
FROM Pet P JOIN SubstanceAbuseRate S ON P.StateName = S.StateName
        CROSS JOIN
                (SELECT AVG(SAR.Number) as drugavg
                FROM SubstanceAbuseRate SAR
                WHERE SAR.SubstanceName = 'total' AND SAR.AgeRange = 'All') as
                DrugAverageTable
        CROSS JOIN
                (SELECT AVG(Pet.PercentHousehold) as petavg
```

FROM Pet
WHERE Pet.Type = 'total') as PetAverageTable
WHERE S.SubstanceName = 'total' AND S.AgeRange = 'All' AND P.Type = 'total'

```
mysql> SELECT (SUM((P.PercentHousehold - PetAverageTable.petavg)*(S.Number -
    -> DrugAverageTable.drugavg))) / SQRT(SUM((P.PercentHousehold - PetAverageTable.petavg)*(P.PercentHousehold - PetAverageTable.petavg)) * SUM((S.Number - DrugAverageTable.drugav
g)*(S.Number - DrugAverageTable.drugavg))) as cc
    -> FROM Pet P JOIN SubstanceAbuseRate S ON P.StateName = S.StateName
    -> CROSS JOIN
    -> (SELECT AVG(SAR.Number) as drugavg
    -> FROM SubstanceAbuseRate SAR
    -> WHERE SAR.SubstanceName = 'total' AND SAR.AgeRange = 'All') as DrugAverageTable
    -> CROSS JOIN
    -> (SELECT AVG(Pet.PercentHousehold) as petavg
    ->
Display all 797 possibilities? (y or n)
    ->  FROM Pet
    -> WHERE Pet.Type = 'total') as PetAverageTable
    -> WHERE S.SubstanceName = 'total' AND S.AgeRange = 'All' AND P.Type = 'total'
    -> ;
+----------------------+
| cc                   |
+----------------------+
| -0.20776121555042595 |
+----------------------+
1 row in set (0.03 sec)
```

EXPLAIN ANALYZE

```
mysql> EXPLAIN ANALYZE SELECT (SUM((P.PercentHousehold - PetAverageTable.petavg)*(S.Number -  DrugAverageTable.drugavg))) / SQRT(SUM((P.PercentHousehold - PetAverageTable.petavg)*(
P.PercentHousehold - PetAverageTable.petavg)) * SUM((S.Number - DrugAverageTable.drugavg)*(S.Number - DrugAverageTable.drugavg))) as cc FROM Pet P JOIN SubstanceAbuseRate S ON P.St
ateName = S.StateName     CROSS JOIN   (SELECT AVG(SAR.Number) as drugavg   FROM SubstanceAbuseRate SAR    WHERE SAR.SubstanceName = 'total' AND SAR.AgeRange = 'All') as DrugAverag
eTable   CROSS JOIN   (SELECT AVG(Pet.PercentHousehold) as petavg  FROM Pet   WHERE Pet.Type = 'total') as PetAverageTable  WHERE S.SubstanceName = 'total' AND S.AgeRange = 'All' A
ND P.Type = 'total';
+------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------
------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------
------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------
------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------
------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------
------------------------------------------------------------------------------------------------------------------------------+
| EXPLAIN





                                                                                                                              |
+------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------
------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------
------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------
------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------
------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------
------------------------------------------------------------------------------------------------------------------------------+
| -> Aggregate: sum(((S.`Number` - '405.2000') * (S.`Number` - '405.2000'))), sum(((P.PercentHousehold - '59.260417') * (P.PercentHousehold - '59.260417'))), sum(((P.PercentHouseho
ld - '59.260417') * (S.`Number` - '405.2000')))  (cost=168.10 rows=1) (actual time=0.981..0.981 rows=1 loops=1)
    -> Filter: (P.StateName = S.StateName)  (cost=168.07 rows=0.3) (actual time=0.869..0.915 rows=48 loops=1)
        -> Inner hash join (<hash>(P.StateName)=<hash>(S.StateName))  (cost=168.07 rows=0.3) (actual time=0.869..0.902 rows=48 loops=1)
            -> Filter: (P.`Type` = 'total')  (cost=1.46 rows=2) (actual time=0.092..0.109 rows=48 loops=1)
                -> Table scan on P  (cost=1.46 rows=191) (actual time=0.021..0.074 rows=191 loops=1)
            -> Hash
                -> Filter: ((S.AgeRange = 'All') and (S.SubstanceName = 'total'))  (cost=145.10 rows=14) (actual time=0.638..0.746 rows=50 loops=1)
                    -> Table scan on S  (cost=145.10 rows=1400) (actual time=0.023..0.542 rows=1400 loops=1)
 |
+------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------
------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------
------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------
------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------
------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------
------------------------------------------------------------------------------------------------------------------------------+
1 row in set (0.03 sec)
```

Above is the original EXPLAIN ANALYZE for the correlation coefficient advanced query we created. We can see that in this case, the cost is 1.46

Next, we have our first indexes created on the substance name. As you can see, the index wasn't used. However, we still can see an order swap between Filter and Hash. We can also see that the cost actually went up for some reason. It jumped from 1.46 to 19.35. This result was totally unexpected, but this could be due to the fact that we are adding a useless index, which increases the storage cost. This could have

also increased the query optimization time, which causes the cost to increase.

```
mysql> CREATE INDEX SubName
    -> ON SubstanceAbuseRate(SubstanceName);
Query OK, 0 rows affected (0.06 sec)
Records: 0  Duplicates: 0  Warnings: 0

mysql> EXPLAIN ANALYZE SELECT (SUM((P.PercentHousehold - PetAverageTable.petavg)*(S.Number -  DrugAverageTable.drugavg))) / SQRT(SUM((P.PercentHousehold - PetAverageTable.petavg)*(P.PercentHousehold - PetAverageTa
ble.petavg)) * SUM((S.Number - DrugAverageTable.drugavg)*(S.Number - DrugAverageTable.drugavg))) as cc FROM Pet P JOIN SubstanceAbuseRate S ON P.StateName = S.StateName     CROSS JOIN   (SELECT AVG(SAR.Number) as
drugavg   FROM SubstanceAbuseRate SAR    WHERE SAR.SubstanceName = 'total' AND SAR.AgeRange = 'All') as DrugAverageTable   CROSS JOIN   (SELECT AVG(Pet.PercentHousehold) as petavg  FROM Pet   WHERE Pet.Type = 'tot
al') as PetAverageTable  WHERE S.SubstanceName = 'total' AND S.AgeRange = 'All' AND P.Type = 'total';
+-----------------------------------------------------------------------------------------------------------------------
-----------------------------------------------------------------------------------------------------------------------
-----------------------------------------------------------------------------------------------------------------------
-----------------------------------------------------------------------------------------------------------------------
-----------------------------------------------------------------------------------------------+
| EXPLAIN

                                                                                                 |
+-----------------------------------------------------------------------------------------------------------------------
-----------------------------------------------------------------------------------------------------------------------
-----------------------------------------------------------------------------------------------------------------------
-----------------------------------------------------------------------------------------------------------------------
-----------------------------------------------------------------------------------------------+
| -> Aggregate: sum(((S.`Number` - '405.2000') * (S.`Number` - '405.2000'))), sum(((P.PercentHousehold - '59.260417') * (P.PercentHousehold - '59.260417'))), sum(((P.PercentHousehold - '59.260417') * (S.`Number` -
 '405.2000')))  (cost=185.74 rows=1) (actual time=0.578..0.578 rows=1 loops=1)
    -> Filter: (S.StateName = P.StateName)  (cost=185.73 rows=0.01) (actual time=0.455..0.536 rows=48 loops=1)
        -> Inner hash join (<hash>(S.StateName)=<hash>(P.StateName))  (cost=185.73 rows=0.01) (actual time=0.454..0.528 rows=48 loops=1)
            -> Filter: ((S.AgeRange = 'All') and (S.SubstanceName = 'total'))  (cost=8.62 rows=1) (actual time=0.376..0.440 rows=50 loops=1)
                -> Table scan on S  (cost=8.62 rows=1400) (actual time=0.016..0.328 rows=1400 loops=1)
            -> Hash
                -> Filter: (P.`Type` = 'total')  (cost=19.35 rows=19) (actual time=0.051..0.061 rows=48 loops=1)
                    -> Table scan on P  (cost=19.35 rows=191) (actual time=0.016..0.046 rows=191 loops=1)
 |
+-----------------------------------------------------------------------------------------------------------------------
-----------------------------------------------------------------------------------------------------------------------
-----------------------------------------------------------------------------------------------------------------------
-----------------------------------------------------------------------------------------------------------------------
-----------------------------------------------------------------------------------------------+
1 row in set (0.03 sec)
```

Next, we have our second index on the attribute AgeRange, which we name Age Ran. We see that the price for Scan on P has stayed the same at 19.35. This shows that the index didn't create an improvement in cost at all. I'm also assuming that since there wasn't anything improvement, the cost was just kept at what the previous index was at. Therefore the price stayed at 19.35, and that is the better index out of the two, providing us with the lower cost comparing the two.

```
mysql> CREATE INDEX AgeRan
    -> ON SubstanceAbuseRate(AgeRange);
Query OK, 0 rows affected (0.06 sec)
Records: 0  Duplicates: 0  Warnings: 0

mysql> EXPLAIN ANALYZE SELECT (SUM((P.PercentHousehold - PetAverageTable.petavg)*(S.Number -  DrugAverageTable.drugavg))) / SQRT(SUM((P.PercentHousehold - PetAverageTable.petavg)*(P.PercentHousehold - PetAverageTa
ble.petavg)) * SUM((S.Number - DrugAverageTable.drugavg)*(S.Number - DrugAverageTable.drugavg))) as cc FROM Pet P JOIN SubstanceAbuseRate S ON P.StateName = S.StateName     CROSS JOIN   (SELECT AVG(SAR.Number) as
drugavg   FROM SubstanceAbuseRate SAR    WHERE SAR.SubstanceName = 'total' AND SAR.AgeRange = 'All') as DrugAverageTable   CROSS JOIN   (SELECT AVG(Pet.PercentHousehold) as petavg  FROM Pet   WHERE Pet.Type = 'tot
al') as PetAverageTable  WHERE S.SubstanceName = 'total' AND S.AgeRange = 'All' AND P.Type = 'total';
+-----------------------------------------------------------------------------------------------------------------------
-----------------------------------------------------------------------------------------------------------------------
-----------------------------------------------------------------------------------------------------------------------
-----------------------------------------------------------------------------------------------------------------------
-----------------------------------------------------------------------------------------------+
| EXPLAIN

                                                                                                 |
+-----------------------------------------------------------------------------------------------------------------------
-----------------------------------------------------------------------------------------------------------------------
-----------------------------------------------------------------------------------------------------------------------
-----------------------------------------------------------------------------------------------------------------------
-----------------------------------------------------------------------------------------------+
| -> Aggregate: sum(((S.`Number` - '405.2000') * (S.`Number` - '405.2000'))), sum(((P.PercentHousehold - '59.260417') * (P.PercentHousehold - '59.260417'))), sum(((P.PercentHousehold - '59.260417') * (S.`Number` -
 '405.2000')))  (cost=182.29 rows=1) (actual time=0.580..0.580 rows=1 loops=1)
    -> Filter: (S.StateName = P.StateName)  (cost=182.29 rows=0.01) (actual time=0.447..0.537 rows=48 loops=1)
        -> Inner hash join (<hash>(S.StateName)=<hash>(P.StateName))  (cost=182.29 rows=0.01) (actual time=0.447..0.530 rows=48 loops=1)
            -> Filter: ((S.AgeRange = 'All') and (S.SubstanceName = 'total'))  (cost=8.44 rows=1) (actual time=0.368..0.440 rows=50 loops=1)
                -> Table scan on S  (cost=8.44 rows=1400) (actual time=0.016..0.327 rows=1400 loops=1)
            -> Hash
                -> Filter: (P.`Type` = 'total')  (cost=19.35 rows=19) (actual time=0.050..0.060 rows=48 loops=1)
                    -> Table scan on P  (cost=19.35 rows=191) (actual time=0.015..0.046 rows=191 loops=1)
 |
+-----------------------------------------------------------------------------------------------------------------------
-----------------------------------------------------------------------------------------------------------------------
-----------------------------------------------------------------------------------------------------------------------
-----------------------------------------------------------------------------------------------------------------------
-----------------------------------------------------------------------------------------------+
1 row in set (0.04 sec)
```

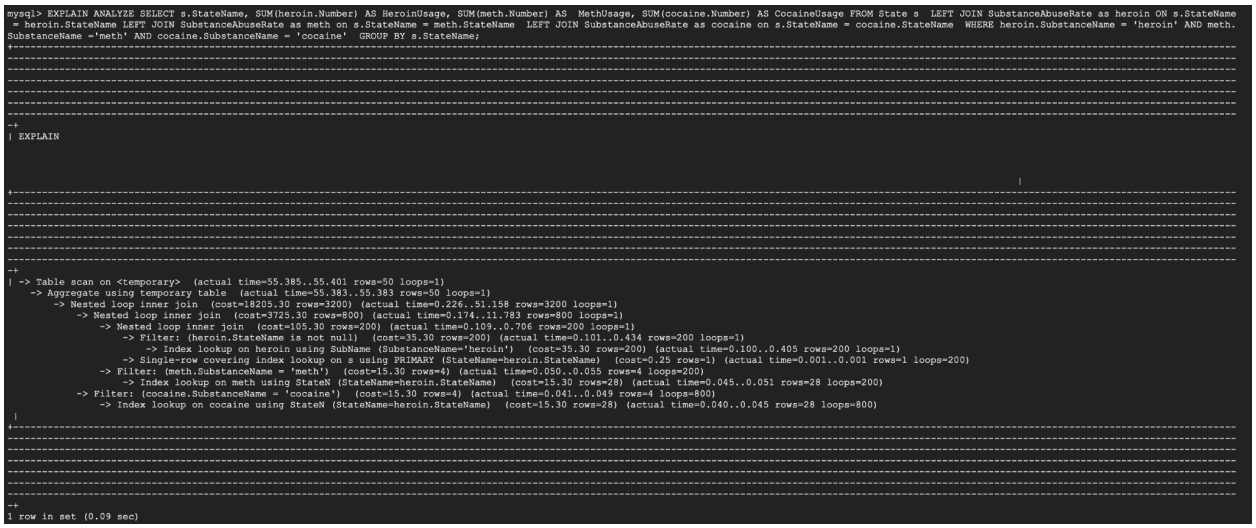
Finally, we have our third index on the combination of the two attributes. Unlikely, we didn't see a change in cost for this one either. Therefore, it was quite useless since that didn't guide us anywhere. It could also not have changed in price due to the fact that the query is not utilizing the index, which means the index wasn't providing any sort of benefit to the query itself. Another possibility is that the table is already small enough so that adding an index doesn't affect the amount of "pages" that have been scanned. Therefore the cost wasn't lowered.

```
mysql> CREATE INDEX Combination
    -> ON SubstanceAbuseRate(SubstanceName,AgeRange);
Query OK, 0 rows affected (0.05 sec)
Records: 0  Duplicates: 0  Warnings: 0

mysql> EXPLAIN ANALYZE SELECT (SUM((P.PercentHousehold - PetAverageTable.petavg)*(S.Number -  DrugAverageTable.drugavg))) / SQRT(SUM((P.PercentHousehold - PetAverageTable.petavg)*(P.PercentHousehold - PetAverageTa
ble.petavg)) * SUM((S.Number - DrugAverageTable.drugavg)*(S.Number - DrugAverageTable.drugavg))) as cc FROM Pet P JOIN SubstanceAbuseRate S ON P.StateName = S.StateName     CROSS JOIN   (SELECT AVG(SAR.Number) as
drugavg   FROM SubstanceAbuseRate SAR    WHERE SAR.SubstanceName = 'total' AND SAR.AgeRange = 'All') as DrugAverageTable   CROSS JOIN   (SELECT AVG(Pet.PercentHousehold) as petavg  FROM Pet   WHERE Pet.Type = 'tot
al') as PetAverageTable  WHERE S.SubstanceName = 'total' AND S.AgeRange = 'All' AND P.Type = 'total';
+-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------
-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------
-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------
-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------
----------+
| EXPLAIN


          |
+-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------
-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------
-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------
-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------
----------+
| -> Aggregate: sum(((S.`Number` - '405.2000') * (S.`Number` - '405.2000'))), sum(((P.PercentHousehold - '59.260417') * (P.PercentHousehold - '59.260417'))), sum(((P.PercentHousehold - '59.260417') * (S.`Number` -
 '405.2000')))  (cost=366.97 rows=1) (actual time=1.670..1.670 rows=1 loops=1)
    -> Nested loop inner join  (cost=365.06 rows=19) (actual time=0.120..1.624 rows=48 loops=1)
        -> Filter: (P.`Type` = 'total')  (cost=19.35 rows=19) (actual time=0.054..0.070 rows=48 loops=1)
            -> Table scan on P  (cost=19.35 rows=191) (actual time=0.015..0.053 rows=191 loops=1)
        -> Filter: ((S.AgeRange = 'All') and (S.SubstanceName = 'total'))  (cost=15.31 rows=1) (actual time=0.032..0.032 rows=1 loops=48)
            -> Index lookup on S using c2 (StateName=P.StateName)  (cost=15.31 rows=28) (actual time=0.027..0.030 rows=28 loops=48)
 |
+-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------
-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------
-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------
-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------
----------+
1 row in set (0.04 sec)
```

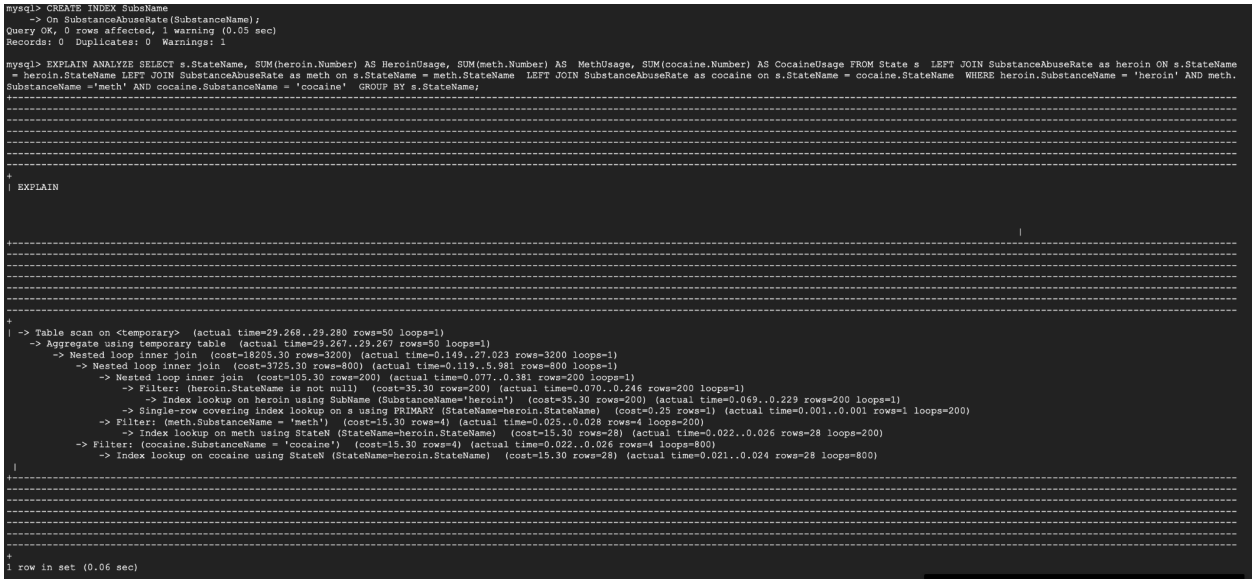
Below is the original EXPLAIN ANALYZE on our second query.

```
mysql> EXPLAIN ANALYZE SELECT s.StateName, SUM(heroin.Number) AS HeroinUsage, SUM(meth.Number) AS  MethUsage, SUM(cocaine.Number) AS CocaineUsage FROM State s  LEFT JOIN SubstanceAbuseRate as heroin ON s.StateName
 = heroin.StateName LEFT JOIN SubstanceAbuseRate as meth on s.StateName = meth.StateName  LEFT JOIN SubstanceAbuseRate as cocaine on s.StateName = cocaine.StateName  WHERE heroin.SubstanceName = 'heroin' AND meth.
SubstanceName ='meth' AND cocaine.SubstanceName = 'cocaine'  GROUP BY s.StateName;
+-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------
-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------
-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------
-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------
-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------
-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------
-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------+
| EXPLAIN


                                                                                                                                                                                                          |
+-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------
-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------
-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------
-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------
-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------
-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------
-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------+
| -> Table scan on <temporary>  (actual time=29.608..29.615 rows=50 loops=1)
    -> Aggregate using temporary table  (actual time=29.606..29.606 rows=50 loops=1)
        -> Nested loop inner join  (cost=18205.30 rows=3200) (actual time=0.157..27.364 rows=3200 loops=1)
            -> Nested loop inner join  (cost=3725.30 rows=800) (actual time=0.118..6.013 rows=800 loops=1)
                -> Nested loop inner join  (cost=105.30 rows=200) (actual time=0.075..0.403 rows=200 loops=1)
                    -> Filter: (heroin.StateName is not null)  (cost=35.30 rows=200) (actual time=0.068..0.265 rows=200 loops=1)
                        -> Index lookup on heroin using SubName (SubstanceName='heroin')  (cost=35.30 rows=200) (actual time=0.066..0.246 rows=200 loops=1)
                    -> Single-row covering index lookup on s using PRIMARY (StateName=heroin.StateName)  (cost=0.25 rows=1) (actual time=0.001..0.001 rows=1 loops=200)
                -> Filter: (meth.SubstanceName = 'meth')  (cost=15.30 rows=4) (actual time=0.025..0.028 rows=4 loops=200)
                    -> Index lookup on meth using c2 (StateName=heroin.StateName)  (cost=15.30 rows=28) (actual time=0.022..0.026 rows=28 loops=200)
            -> Filter: (cocaine.SubstanceName = 'cocaine')  (cost=15.30 rows=4) (actual time=0.022..0.026 rows=4 loops=800)
                -> Index lookup on cocaine using c2 (StateName=heroin.StateName)  (cost=15.30 rows=28) (actual time=0.021..0.024 rows=28 loops=800)
 |
+-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------
-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------
-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------
-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------
-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------
-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------
-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------+
1 row in set (0.07 sec)
```

After that is our first index on our attribute, StateName, which we named after StateN. We see that the price was the same as it was before we ran anything. I'm assuming that this by itself doesn't reduce the data size or the "pages" of the dataset that need to be scanned. This just means that our original SQL query was already efficient enough without adding this index.



Next, we have our second index on our attribute SubstanceName, and we call it SubsName. We can see that the run time was kept at 15.3, meaning no change to the price at all. At the same time, we can see that the index wasn't really being run. We're assuming that's because the previous execution with StateN had a lower cost than this one, therefore this index wasn't running, since we didn't drop the previous index. Though we tried to test this by dropping the StateN index, for some reason, it was popping up with an error relating to a foreign key, which didn't make sense to either our team or the TA during office hours.



Finally, we have our third index on the combination of the two before. We can clearly see that the cost has been lowered from 15.3 to 3.40. This was definitely a successful indexing that we have created. The cost has been lowered by almost 5 times. I think the reason as to why this happened could be because the dataset could have been more than 1 page to read, and using this combination of the two indexes, we were able to split it up into several pages to be scanned for. But only now we can lower the amount of time that is needed to search for that one specific thing. Instead of scanning the entire file and dataset, we only need

one portion, which would lower the cost.

```
mysql> CREATE INDEX Combinations
    -> ON SubstanceAbuseRate(StateName,SubstanceName)
    ->
    ->
    -> ;
Query OK, 0 rows affected (0.06 sec)
Records: 0  Duplicates: 0  Warnings: 0

mysql> EXPLAIN ANALYZE SELECT s.StateName, SUM(heroin.Number) AS HeroinUsage, SUM(meth.Number) AS  MethUsage, SUM(cocaine.Number) AS CocaineUsage FROM State s  LEFT JOIN SubstanceAbuseRate as heroin ON s.StateName
 = heroin.StateName LEFT JOIN SubstanceAbuseRate as meth on s.StateName = meth.StateName  LEFT JOIN SubstanceAbuseRate as cocaine on s.StateName = cocaine.StateName  WHERE heroin.SubstanceName = 'heroin' AND meth.
SubstanceName ='meth' AND cocaine.SubstanceName = 'cocaine'  GROUP BY s.StateName;
+----------------------------------------------------------------------------------------------------
----------------------------------------------------------------------------------------------------
----------------------------------------------------------------------------------------------------
----------------------------------------------------------------------------------------------------
----------------------------------------------------------------------------------------------------
----------------------------------------------------------------------------------------------------
-----+
| EXPLAIN



                                                                                                    |
+----------------------------------------------------------------------------------------------------
----------------------------------------------------------------------------------------------------
----------------------------------------------------------------------------------------------------
----------------------------------------------------------------------------------------------------
----------------------------------------------------------------------------------------------------
----------------------------------------------------------------------------------------------------
-----+
| -> Table scan on <temporary>  (actual time=8.483..8.489 rows=50 loops=1)
    -> Aggregate using temporary table  (actual time=8.481..8.481 rows=50 loops=1)
        -> Nested loop inner join  (cost=3905.30 rows=3200) (actual time=0.107..6.281 rows=3200 loops=1)
            -> Nested loop inner join  (cost=865.30 rows=800) (actual time=0.096..1.568 rows=800 loops=1)
                -> Nested loop inner join  (cost=105.30 rows=200) (actual time=0.084..0.375 rows=200 loops=1)
                    -> Filter: (heroin.StateName is not null)  (cost=35.30 rows=200) (actual time=0.078..0.250 rows=200 loops=1)
                        -> Index lookup on heroin using SubName (SubstanceName='heroin')  (cost=35.30 rows=200) (actual time=0.077..0.237 rows=200 loops=1)
                    -> Single-row covering index lookup on s using PRIMARY (StateName=heroin.StateName)  (cost=0.25 rows=1) (actual time=0.000..0.000 rows=1 loops=200)
                -> Index lookup on meth using Combinations (StateName=heroin.StateName, SubstanceName='meth')  (cost=3.40 rows=4) (actual time=0.005..0.006 rows=4 loops=200)
            -> Index lookup on cocaine using Combinations (StateName=heroin.StateName, SubstanceName='cocaine')  (cost=3.40 rows=4) (actual time=0.005..0.006 rows=4 loops=800)
 |
+----------------------------------------------------------------------------------------------------
----------------------------------------------------------------------------------------------------
----------------------------------------------------------------------------------------------------
----------------------------------------------------------------------------------------------------
----------------------------------------------------------------------------------------------------
----------------------------------------------------------------------------------------------------
-----+
1 row in set (0.04 sec)
```