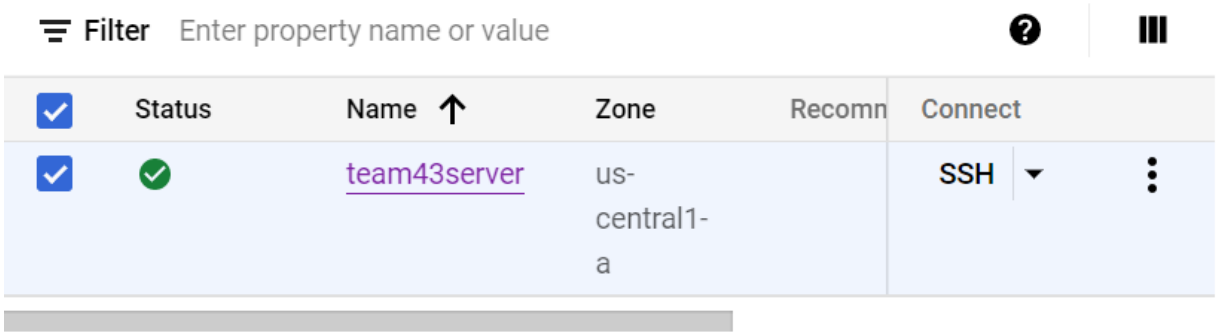
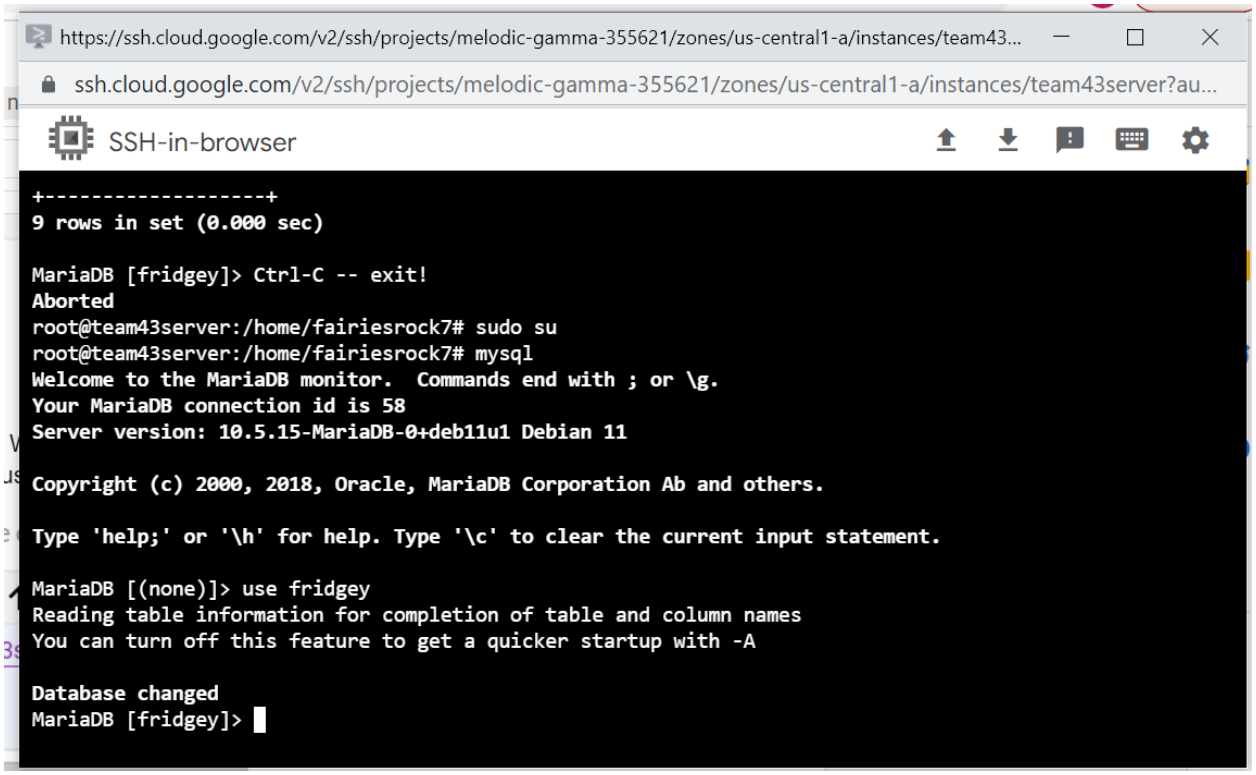
Stage 3: Database Design.

WHAT WE USED:

GCP VM was used for this stage. We used MariaDB. This choice was made as the demo for the next stages of web development used this.





I. DATABASE IMPLEMENTATION

Create Table Commands:

- create table Dish(DishId INT PRIMARY KEY, DishName VARCHAR(255) ,Calories INT);
- create table Ingredient(IngredientId INT PRIMARY KEY, IngredientName VARCHAR(255), Measurement REAL, Price REAL);
- create table User(UserId INT PRIMARY KEY, UserFN VARCHAR(255) , UserLN VARCHAR(255));
- create table Cuisine(CuisineId INT PRIMARY KEY, CuisineName VARCHAR(255) ,Taste VARCHAR(255));
- Create table Madeof(CuisineId INT, DishId INT,
  PRIMARY KEY(CuisineId , DishId),
  FOREIGN KEY (CuisineId) references Cuisine(CuisineId) ON DELETE CASCADE,
  FOREIGN KEY (DishId) references Dish(DishId) ON DELETE CASCADE
  );
- Create table Contains(IngredientId INT, DishId INT,
  PRIMARY KEY(IngredientId , DishId),
  FOREIGN KEY (IngredientId) references Ingredient(IngredientId) ON DELETE CASCADE,
  FOREIGN KEY (DishId) references Dish(DishId) ON DELETE CASCADE
  );
- Create table Creates(UserId INT, DishId INT,
  PRIMARY KEY(UserId , DishId),
  FOREIGN KEY (UserId) references User(UserId) ON DELETE CASCADE,
  FOREIGN KEY (DishId) references Dish(DishId) ON DELETE CASCADE
  );
- Create table Follows(UserId1 INT, UserId2 INT,
  PRIMARY KEY(UserId1 , UserId2),
  FOREIGN KEY (UserId1) references User(UserId) ON DELETE CASCADE,
  FOREIGN KEY (UserId2) references User(UserId) ON DELETE CASCADE
  );
- Create table Rating(UserId INT, DishId INT,Score REAL,
  PRIMARY KEY(UserId , DishId),
  FOREIGN KEY (UserId) references User(UserId) ON DELETE CASCADE,
  FOREIGN KEY (DishId) references Dish(DishId) ON DELETE CASCADE
  );

Main Tables Used: Cuisine, Dish, Ingredient, Ratings

```
MariaDB [fridgey]> show tables;
+-------------------+
| Tables_in_fridgey |
+-------------------+
| Contains          |
| Creates           |
| Cuisine           |
| Dish              |
| Follows           |
| Ingredient        |
| Madeof            |
| Rating            |
| User              |
+-------------------+
9 rows in set (0.000 sec)
```

Count commands:

```
MariaDB [fridgey]> SELECT COUNT(*)
    -> FROM Ingredient;
+----------+
| COUNT(*) |
+----------+
|     1024 |
+----------+
1 row in set (0.001 sec)
```

```
MariaDB [fridgey]> SELECT COUNT(*)  FROM Dish;
+----------+
| COUNT(*) |
+----------+
|     1024 |
+----------+
1 row in set (0.001 sec)
```

```
MariaDB [fridgey]> select count(*) from Cuisine
    -> ;
+----------+
| count(*) |
+----------+
|     1024 |
+----------+
1 row in set (0.001 sec)
```

II. ADVANCED QUERIES

Queries:

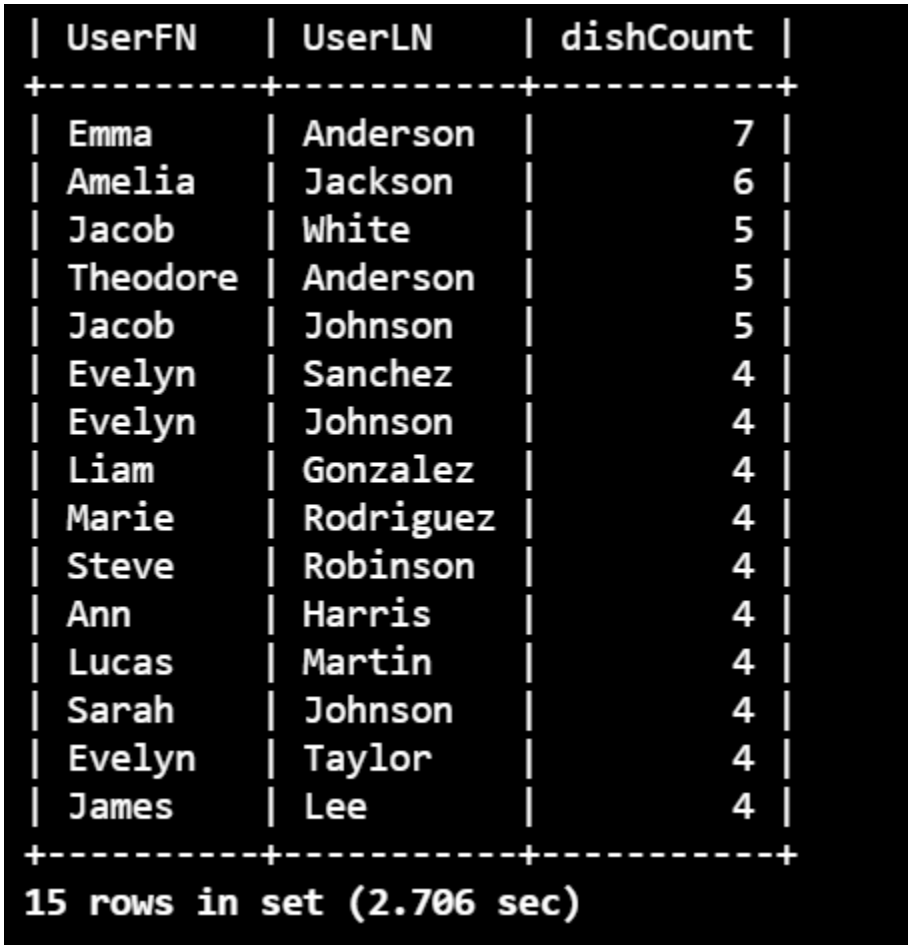
1) SELECT UserFN,UserLN, Calories,dishName, (SELECT AVG(Score) FROM Rating WHERE User.UserId=Rating.UserId GROUP BY UserId) AS averageRating
FROM User
NATURAL JOIN Rating
JOIN Dish ON Dish.DishId= Rating.DishId
WHERE Calories>200
GROUP BY User.UserId
HAVING averageRating>6
ORDER BY averageRating
LIMIT 15

UserFN	UserLN	Calories	dishName	averageRating
Theodore	Anderson	855	Chocolate Truffle	6.2
John	Ramirez	1036	Chicken Bread	6.333333333333333
Benjamin	Jackson	256	Green Bean Pasta	6.333333333333333
Eva	Garcia	808	Grapefruit Hotdog	6.333333333333333
Emma	Gonzalez	1908	Apple Waffles	6.333333333333333
Bill	White	730	Mango Cupcake	6.333333333333333
Lucas	Gonzalez	1369	Peach Tart	6.5
Benjamin	Robinson	1023	Spinach Smoothie	6.5
Mike	Davis	338	Lime Ramen	6.5
Elijah	Lopez	262	Potato Rice	6.5
Eva	Clark	1683	Cheese Cupcake	6.5
Isabella	Walker	470	Lime Pancakes	6.5
Sarah	Garcia	1825	Peach Pasta	6.5
Noah	Thomas	1609	Ginger Bread	6.5
Noah	Ramirez	1695	Tomato Sushi	6.5

15 rows in set (0.007 sec)

This query checks for users who have an average rating over 6 on their dishes. Further this query checks if the dish has more than 200 calories. On passing these conditions we display the user's first name, last name, the dish calories and its name along with the user's average rating.

2) SELECT u.UserFN, u.UserLN, (Select Count(*) from Creates c1 where u.UserId = c1.UserId) as dishCount
FROM User u,Creates c
GROUP BY u.UserId
Having dishCount > 1
ORDER BY dishCount desc
LIMIT 15



UserFN	UserLN	dishCount
Emma	Anderson	7
Amelia	Jackson	6
Jacob	White	5
Theodore	Anderson	5
Jacob	Johnson	5
Evelyn	Sanchez	4
Evelyn	Johnson	4
Liam	Gonzalez	4
Marie	Rodriguez	4
Steve	Robinson	4
Ann	Harris	4
Lucas	Martin	4
Sarah	Johnson	4
Evelyn	Taylor	4
James	Lee	4

15 rows in set (2.706 sec)

This query counts the number of times a user has created a dish and if they have created more than one dish, it prints their first and last names. This query could be used to display users who have more content to other users. We want our website to display people with multiple recipes first as they're more likely to be active posters.

III. INDEXING ANALYSIS

We used ANALYZE FORMAT=JSON .

Query 1:

```
SELECT UserFN,UserLN, Calories,dishName, (SELECT AVG(Score) FROM Rating WHERE
User.UserId=Rating.UserId GROUP BY UserId) AS averageRating
   FROM User
   NATURAL JOIN Rating
    JOIN Dish ON Dish.DishId= Rating.DishId
   WHERE Calories>200
   GROUP BY User.UserId
   HAVING averageRating>6
   ORDER BY averageRating
```

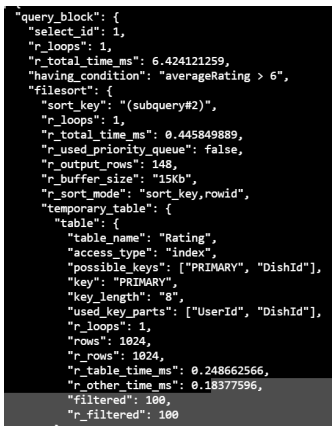
First, we tried to index on the userId of the user table. However, this gave us a slower run time on the query and the subquery than default by approximately 0.2 ms. This index generated more loops on individual tables and made the performance of the query worse.

Second, we decided to index on the dishId of the dish table. This improved run times of the main query by 0.3 ms and also made the subquery run marginally faster. This however, invoked the same amount of loops.

Third, we indexed on the score of the Ratings table. This made the run time even more efficient in comparison to the query indexed on dishId. The subquery was marginally slower in comparison to DishId however.

The read rows and filtered rows remained the same according to the tool for all 3 indices and thus we decided that the index on the dishId would be most appropriate since it was marginally slower and the subquery is run with a loop multiple times and it would be more efficient to have this as the index.

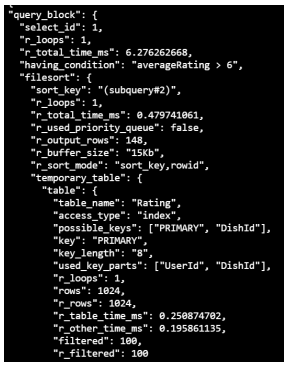
Screenshots of Runtimes:
No Filter



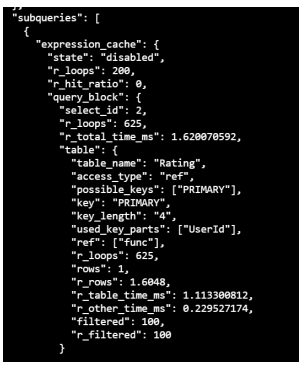
```
{
  "query_block": {
    "select_id": 1,
    "r_loops": 1,
    "r_total_time_ms": 6.424121259,
    "having_condition": "averageRating > 6",
    "filesort": {
      "sort_key": "(subquery#2)",
      "r_loops": 1,
      "r_total_time_ms": 0.445849889,
      "r_used_priority_queue": false,
      "r_output_rows": 148,
      "r_buffer_size": "15Kb",
      "r_sort_mode": "sort_key,rowid",
      "temporary_table": {
        "table": {
          "table_name": "Rating",
          "access_type": "index",
          "possible_keys": ["PRIMARY", "DishId"],
          "key": "PRIMARY",
          "key_length": "8",
          "used_key_parts": ["UserId", "DishId"],
          "r_loops": 1,
          "rows": 1024,
          "r_rows": 1024,
          "r_table_time_ms": 0.248662566,
          "r_other_time_ms": 0.18377596,
          "filtered": 100,
          "r_filtered": 100
```

```
    },
    "subqueries": [
      {
        "expression_cache": {
          "state": "disabled",
          "r_loops": 200,
          "r_hit_ratio": 0,
          "query_block": {
            "select_id": 2,
            "r_loops": 625,
            "r_total_time_ms": 1.605489813,
            "table": {
              "table_name": "Rating",
              "access_type": "ref",
              "possible_keys": ["PRIMARY"],
              "key": "PRIMARY",
              "key_length": "4",
              "used_key_parts": ["UserId"],
              "ref": ["func"],
              "r_loops": 625,
              "rows": 1,
              "r_rows": 1.6048,
              "r_table_time_ms": 1.091492992,
              "r_other_time_ms": 0.23318399,
              "filtered": 100,
              "r_filtered": 100
            }
```

User UserId

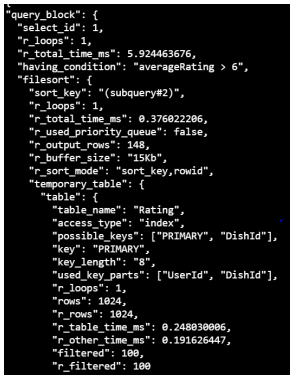


```
{
  "query_block": {
    "select_id": 1,
    "r_loops": 1,
    "r_total_time_ms": 6.276262668,
    "having_condition": "averageRating > 6",
    "filesort": {
      "sort_key": "(subquery#2)",
      "r_loops": 1,
      "r_total_time_ms": 0.479741061,
      "r_used_priority_queue": false,
      "r_output_rows": 148,
      "r_buffer_size": "15Kb",
      "r_sort_mode": "sort_key,rowid",
      "temporary_table": {
        "table": {
          "table_name": "Rating",
          "access_type": "index",
          "possible_keys": ["PRIMARY", "DishId"],
          "key": "PRIMARY",
          "key_length": "8",
          "used_key_parts": ["UserId", "DishId"],
          "r_loops": 1,
          "rows": 1024,
          "r_rows": 1024,
          "r_table_time_ms": 0.250874702,
          "r_other_time_ms": 0.195861135,
          "filtered": 100,
          "r_filtered": 100
```

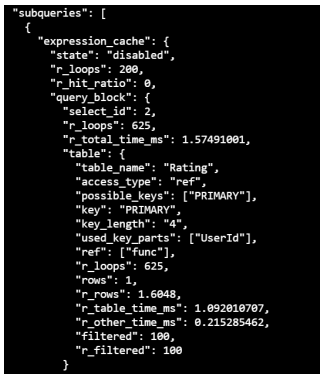


```
  "subqueries": [
    {
      "expression_cache": {
        "state": "disabled",
        "r_loops": 200,
        "r_hit_ratio": 0,
        "query_block": {
          "select_id": 2,
          "r_loops": 625,
          "r_total_time_ms": 1.620070592,
          "table": {
            "table_name": "Rating",
            "access_type": "ref",
            "possible_keys": ["PRIMARY"],
            "key": "PRIMARY",
            "key_length": "4",
            "used_key_parts": ["UserId"],
            "ref": ["func"],
            "r_loops": 625,
            "rows": 1,
            "r_rows": 1.6048,
            "r_table_time_ms": 1.113300812,
            "r_other_time_ms": 0.229527174,
            "filtered": 100,
            "r_filtered": 100
          }
```

Dish DishId

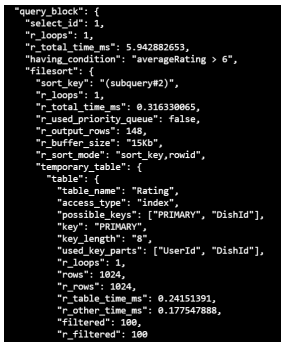


```
{
  "query_block": {
    "select_id": 1,
    "r_loops": 1,
    "r_total_time_ms": 5.924463676,
    "having_condition": "averageRating > 6",
    "filesort": {
      "sort_key": "(subquery#2)",
      "r_loops": 1,
      "r_total_time_ms": 0.376022206,
      "r_used_priority_queue": false,
      "r_output_rows": 148,
      "r_buffer_size": "15Kb",
      "r_sort_mode": "sort_key,rowid",
      "temporary_table": {
        "table": {
          "table_name": "Rating",
          "access_type": "index",
          "possible_keys": ["PRIMARY", "DishId"],
          "key": "PRIMARY",
          "key_length": "8",
          "used_key_parts": ["UserId", "DishId"],
          "r_loops": 1,
          "rows": 1024,
          "r_rows": 1024,
          "r_table_time_ms": 0.248030006,
          "r_other_time_ms": 0.191626447,
          "filtered": 100,
          "r_filtered": 100
```



```
  "subqueries": [
    {
      "expression_cache": {
        "state": "disabled",
        "r_loops": 200,
        "r_hit_ratio": 0,
        "query_block": {
          "select_id": 2,
          "r_loops": 625,
          "r_total_time_ms": 1.57491001,
          "table": {
            "table_name": "Rating",
            "access_type": "ref",
            "possible_keys": ["PRIMARY"],
            "key": "PRIMARY",
            "key_length": "4",
            "used_key_parts": ["UserId"],
            "ref": ["func"],
            "r_loops": 625,
            "rows": 1,
            "r_rows": 1.6048,
            "r_table_time_ms": 1.092010707,
            "r_other_time_ms": 0.215285462,
            "filtered": 100,
            "r_filtered": 100
          }
```

Rating Score



```
{
  "query_block": {
    "select_id": 1,
    "r_loops": 1,
    "r_total_time_ms": 5.942882653,
    "having_condition": "averageRating > 6",
    "filesort": {
      "sort_key": "(subquery#2)",
      "r_loops": 1,
      "r_total_time_ms": 0.316330065,
      "r_used_priority_queue": false,
      "r_output_rows": 148,
      "r_buffer_size": "15Kb",
      "r_sort_mode": "sort_key,rowid",
      "temporary_table": {
        "table": {
          "table_name": "Rating",
          "access_type": "index",
          "possible_keys": ["PRIMARY", "DishId"],
          "key": "PRIMARY",
          "key_length": "8",
          "used_key_parts": ["UserId", "DishId"],
          "r_loops": 1,
          "rows": 1024,
          "r_rows": 1024,
          "r_table_time_ms": 0.24151391,
          "r_other_time_ms": 0.17754788,
          "filtered": 100,
          "r_filtered": 100
```

```
  "subqueries": [
    {
      "expression_cache": {
        "state": "disabled",
        "r_loops": 200,
        "r_hit_ratio": 0,
        "query_block": {
          "select_id": 2,
          "r_loops": 625,
          "r_total_time_ms": 1.605329395,
          "table": {
            "table_name": "Rating",
            "access_type": "ref",
            "possible_keys": ["PRIMARY"],
            "key": "PRIMARY",
            "key_length": "4",
            "used_key_parts": ["UserId"],
            "ref": ["func"],
            "r_loops": 625,
            "rows": 1,
            "r_rows": 1.6048,
            "r_table_time_ms": 1.108701537,
            "r_other_time_ms": 0.221891794,
            "filtered": 100,
            "r_filtered": 100
```

Query 2:

```
SELECT u.UserFN, u.UserLN, (Select Count(*) from Creates c1 where u.UserId = c1.UserId)
as dishCount
   FROM User u,Creates c
   GROUP BY u.UserId
   Having dishCount > 1
   ORDER BY dishCount desc
```

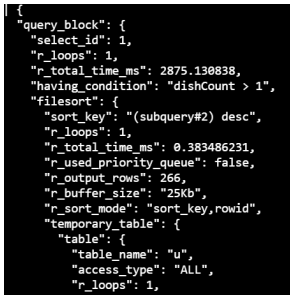
We first indexed on userId from the user table and this improved the run time by 4ms for the main query and made the run time of the subquery slower by 0.002 ms. This clearly led to an efficiency improvement in the query.

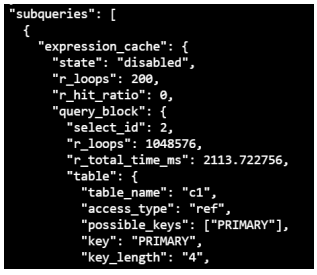
We then tried to index on userId from the creates table and this greatly improved the performance of the query. We observed that the query ran almost a 100 ms faster than the default index and the subquery performance was the same.

We finally tried to index by the firstname(UserFN) of the users in the user table. This improved efficiency by around 90 ms but increased the subquery run time by 0.01 ms.

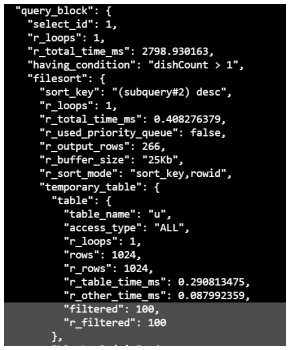
Thus we decided that the most efficient index on this was the userId on the creates table as it gave us the best performance numbers. The read rows, filtered rows and loops once again produce similar results so the main metric to choose from is the run time here.

Screenshots of Runtimes:
No Index





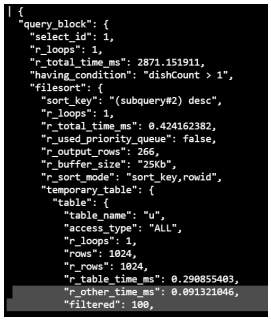
User UserId



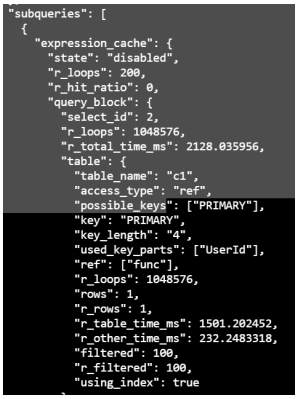
```
"query_block": {
  "select_id": 1,
  "r_loops": 1,
  "r_total_time_ms": 2798.930163,
  "having_condition": "dishCount > 1",
  "filesort": {
    "sort_key": "(subquery#2) desc",
    "r_loops": 1,
    "r_total_time_ms": 0.408276379,
    "r_used_priority_queue": false,
    "r_output_rows": 266,
    "r_buffer_size": "25Kb",
    "r_sort_mode": "sort_key,rowid",
    "temporary_table": {
      "table": {
        "table_name": "u",
        "access_type": "ALL",
        "r_loops": 1,
        "rows": 1024,
        "r_rows": 1024,
        "r_table_time_ms": 0.290813475,
        "r_other_time_ms": 0.087992359,
        "filtered": 100,
        "r_filtered": 100
      },
```

```
"subqueries": [
  {
    "expression_cache": {
      "state": "disabled",
      "r_loops": 200,
      "r_hit_ratio": 0,
      "query_block": {
        "select_id": 2,
        "r_loops": 1048576,
        "r_total_time_ms": 2078.728927,
        "table": {
          "table_name": "c1",
          "access_type": "ref",
          "possible_keys": ["PRIMARY"],
          "key": "PRIMARY",
          "key_length": "4",
          "used_key_parts": ["UserId"],
          "ref": ["func"],
          "r_loops": 1048576,
          "rows": 1,
          "r_rows": 1,
          "r_table_time_ms": 1473.027638,
          "r_other_time_ms": 218.7600763,
          "filtered": 100,
          "r_filtered": 100,
          "using_index": true
```

User UserFN

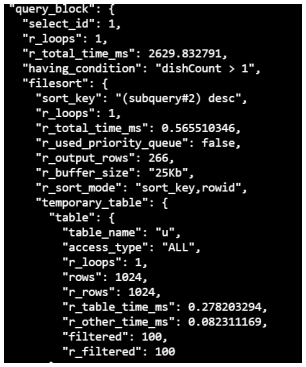


```
| {
  "query_block": {
    "select_id": 1,
    "r_loops": 1,
    "r_total_time_ms": 2871.151911,
    "having_condition": "dishCount > 1",
    "filesort": {
      "sort_key": "(subquery#2) desc",
      "r_loops": 1,
      "r_total_time_ms": 0.424162382,
      "r_used_priority_queue": false,
      "r_output_rows": 266,
      "r_buffer_size": "25Kb",
      "r_sort_mode": "sort_key,rowid",
      "temporary_table": {
        "table": {
          "table_name": "u",
          "access_type": "ALL",
          "r_loops": 1,
          "rows": 1024,
          "r_rows": 1024,
          "r_table_time_ms": 0.290855403,
          "r_other_time_ms": 0.091321046,
          "filtered": 100,
```

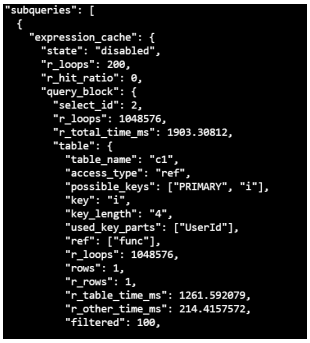


```
"subqueries": [
  {
    "expression_cache": {
      "state": "disabled",
      "r_loops": 200,
      "r_hit_ratio": 0,
      "query_block": {
        "select_id": 2,
        "r_loops": 1048576,
        "r_total_time_ms": 2128.035956,
        "table": {
          "table_name": "c1",
          "access_type": "ref",
          "possible_keys": ["PRIMARY"],
          "key": "PRIMARY",
          "key_length": "4",
          "used_key_parts": ["UserId"],
          "ref": ["func"],
          "r_loops": 1048576,
          "rows": 1,
          "r_rows": 1,
          "r_table_time_ms": 1501.202452,
          "r_other_time_ms": 232.2483318,
          "filtered": 100,
          "r_filtered": 100,
          "using_index": true
```

Creates UserId



```
"query_block": {
  "select_id": 1,
  "r_loops": 1,
  "r_total_time_ms": 2629.832791,
  "having_condition": "dishCount > 1",
  "filesort": {
    "sort_key": "(subquery#2) desc",
    "r_loops": 1,
    "r_total_time_ms": 0.565510346,
    "r_used_priority_queue": false,
    "r_output_rows": 266,
    "r_buffer_size": "25Kb",
    "r_sort_mode": "sort_key,rowid",
    "temporary_table": {
      "table": {
        "table_name": "u",
        "access_type": "ALL",
        "r_loops": 1,
        "rows": 1024,
        "r_rows": 1024,
        "r_table_time_ms": 0.278203294,
        "r_other_time_ms": 0.082311169,
        "filtered": 100,
        "r_filtered": 100
```



```
"subqueries": [
  {
    "expression_cache": {
      "state": "disabled",
      "r_loops": 200,
      "r_hit_ratio": 0,
      "query_block": {
        "select_id": 2,
        "r_loops": 1048576,
        "r_total_time_ms": 1903.30812,
        "table": {
          "table_name": "c1",
          "access_type": "ref",
          "possible_keys": ["PRIMARY", "i"],
          "key": "i",
          "key_length": "4",
          "used_key_parts": ["UserId"],
          "ref": ["func"],
          "r_loops": 1048576,
          "rows": 1,
          "r_rows": 1,
          "r_table_time_ms": 1261.592079,
          "r_other_time_ms": 214.4157572,
          "filtered": 100,
```