# Google Query

## Web UI Tutorial & Managing Stackoverflow Database

Athena Li
Lancy Mao
Elena Lopez

# Agenda

1. Introduction
2. Basic Tutorial
3. Specific Application
4. Question & Answer

Google BigQuery

# Introduction

# Introduction

**What is BigQuery?**

- Externalization of Dremel
  - High performance and scalable query engine on the cloud
- First to identify the drawbacks of MapReduce
  - Not ideal to query large, distributed datasets in real time
  - Extremely slow
- Supports a subset of SQL for querying and retrieving data
- End of the Big Data pipeline

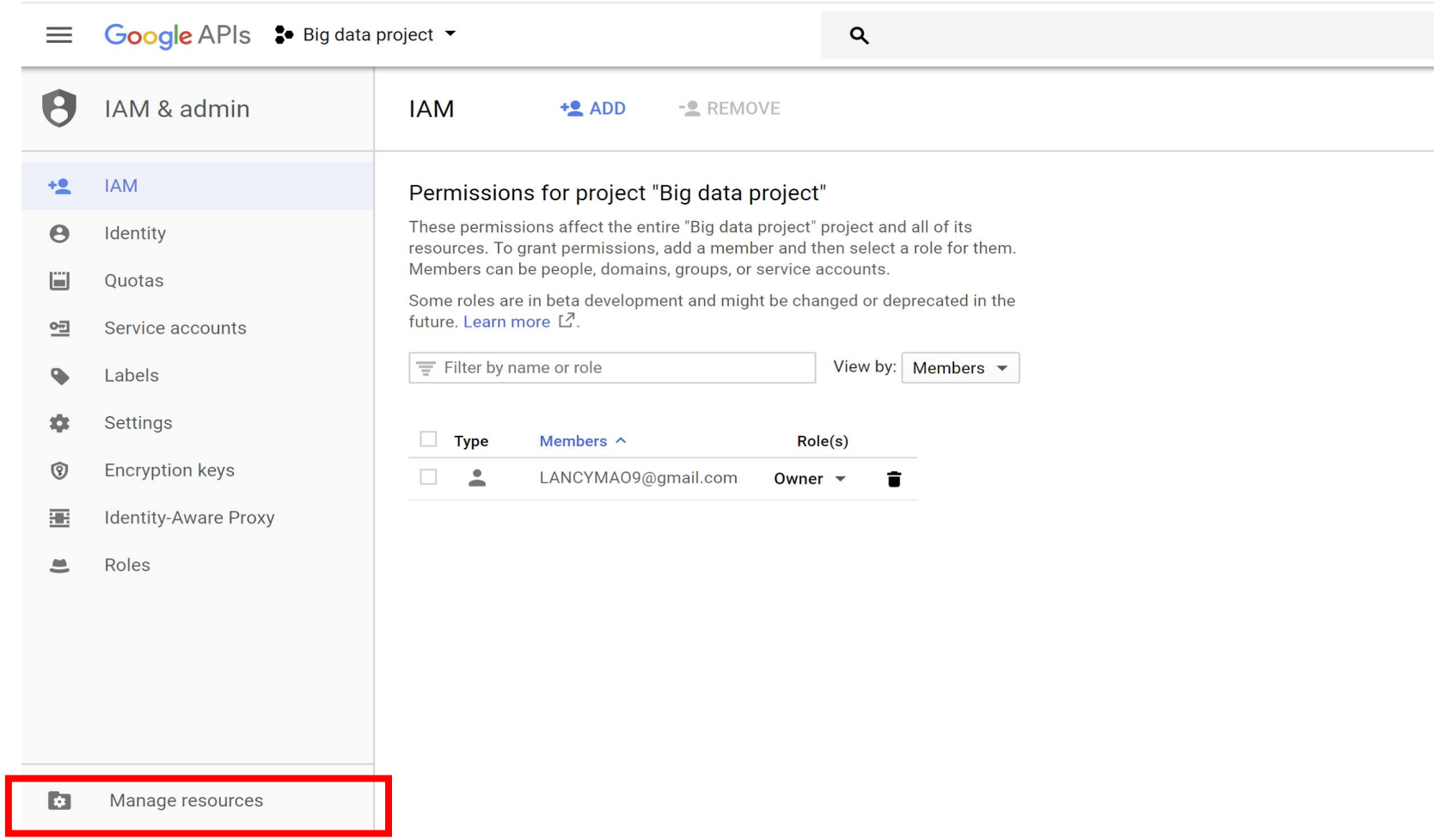
**BigQuery Strengths**

- Deliver blazing fast query performance on distributed data sets that are stored across thousand of servers
- Interactive data analysis tool for large datasets
- Explore record level data in real time

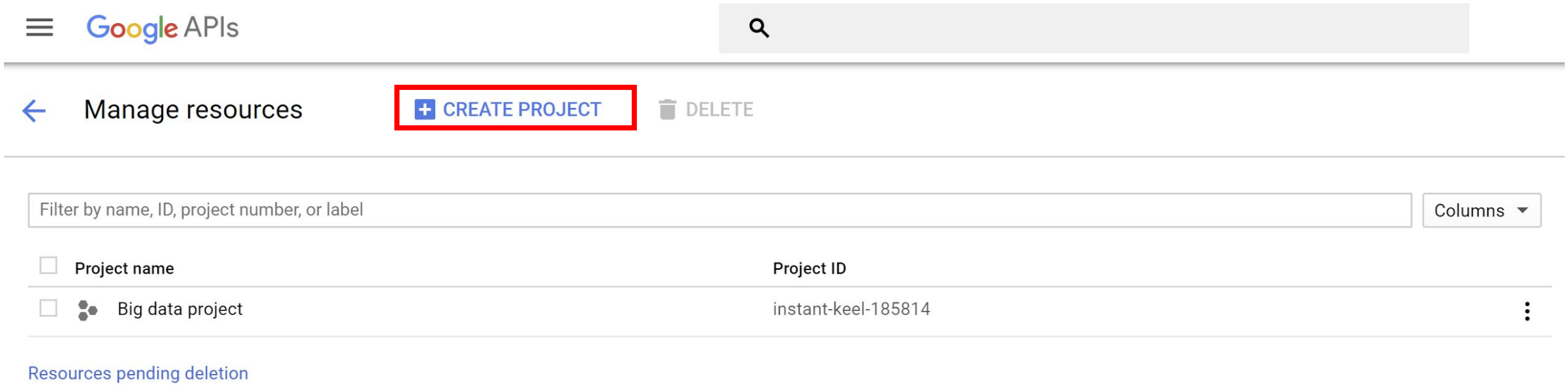
# Basic Tutorial

Google BigQuery

# Create project

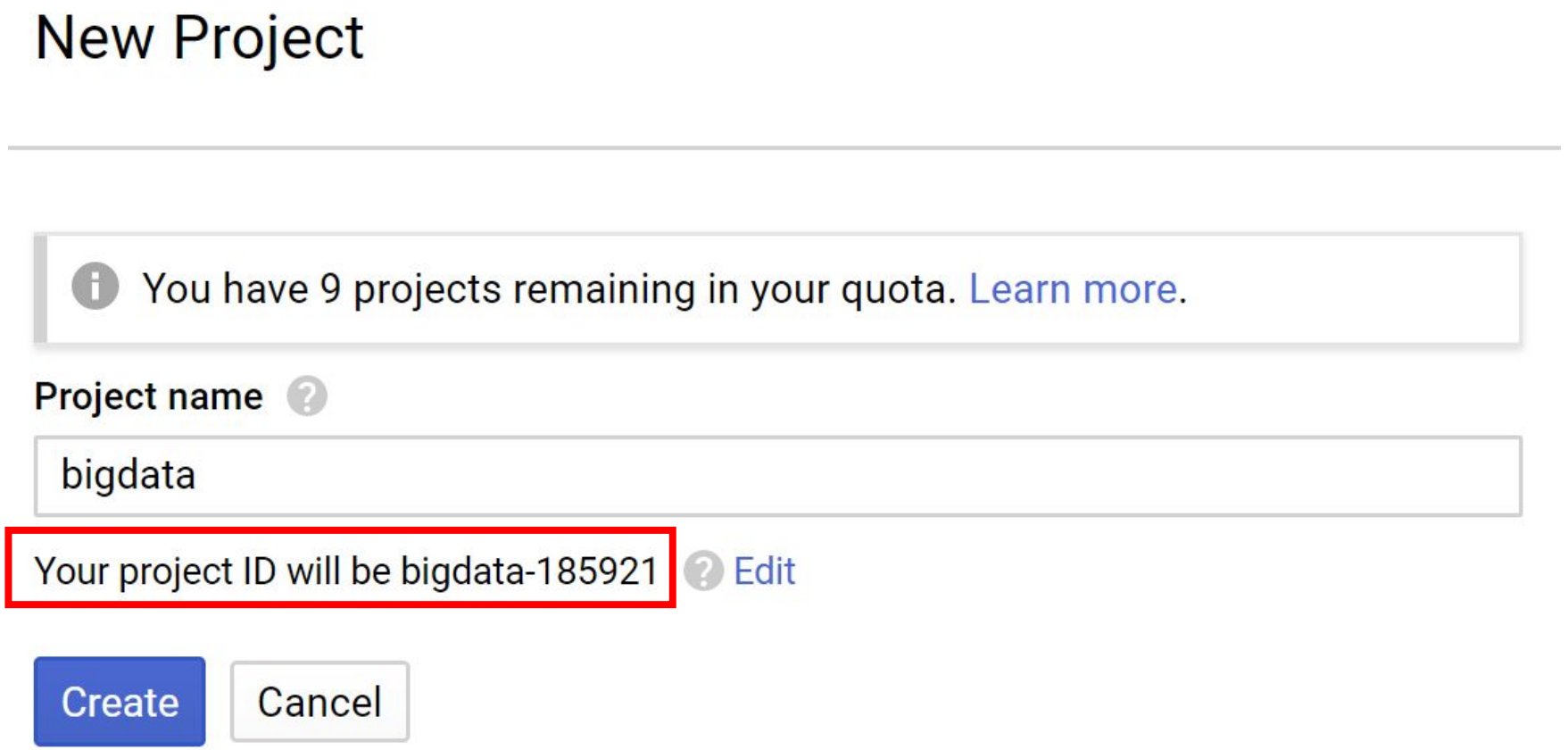


# Create project



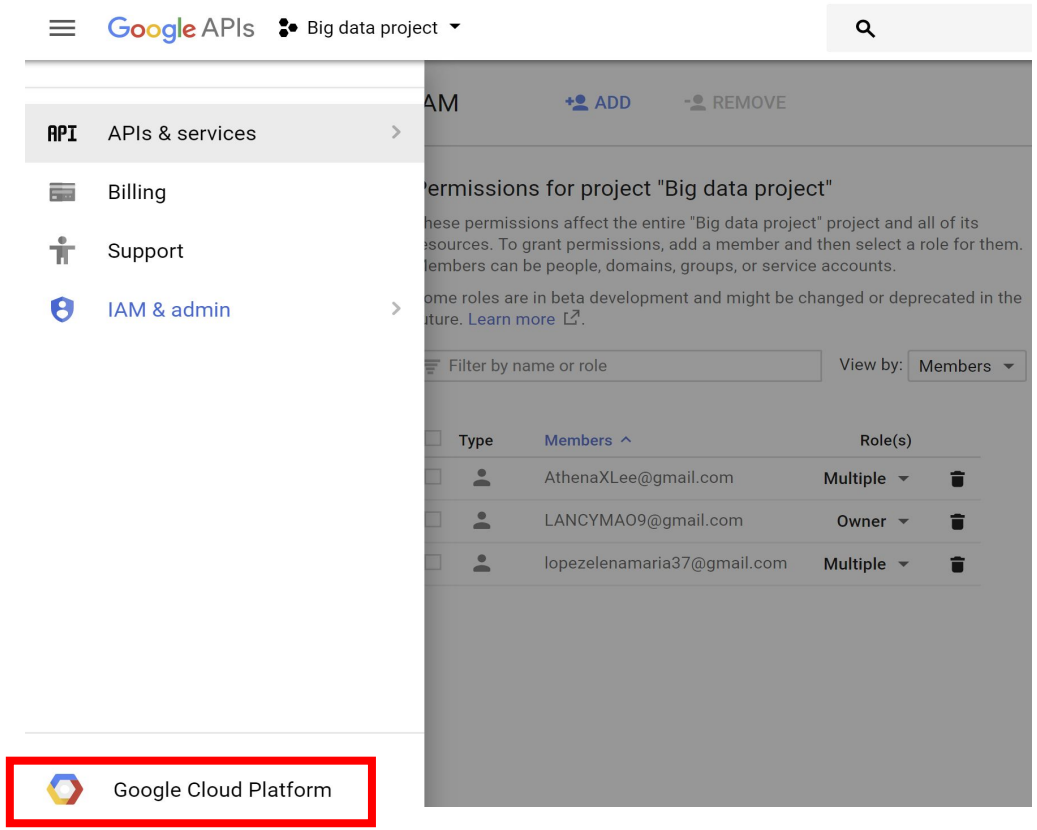
# Create project

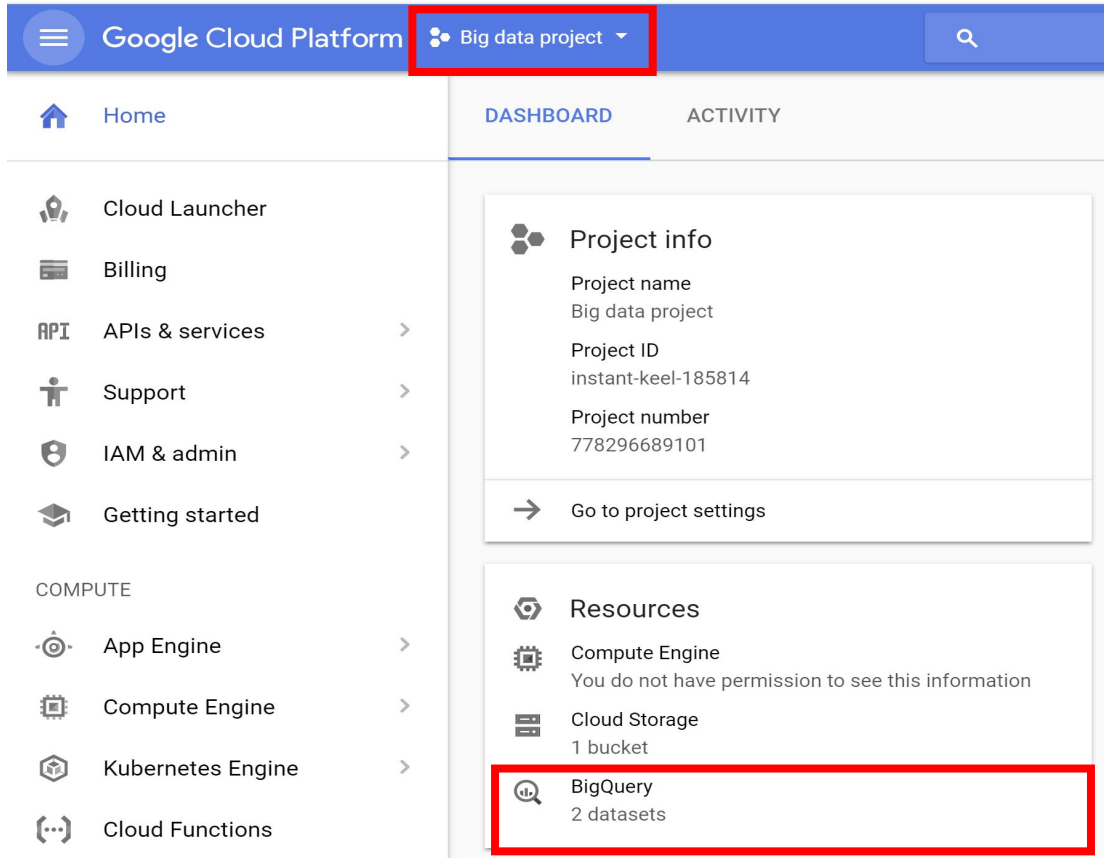
The project ID is the globally unique identifier for your project. You cannot change the project ID after the project is created.  It will be used in SQL query to refer to the database and table.



# Create project

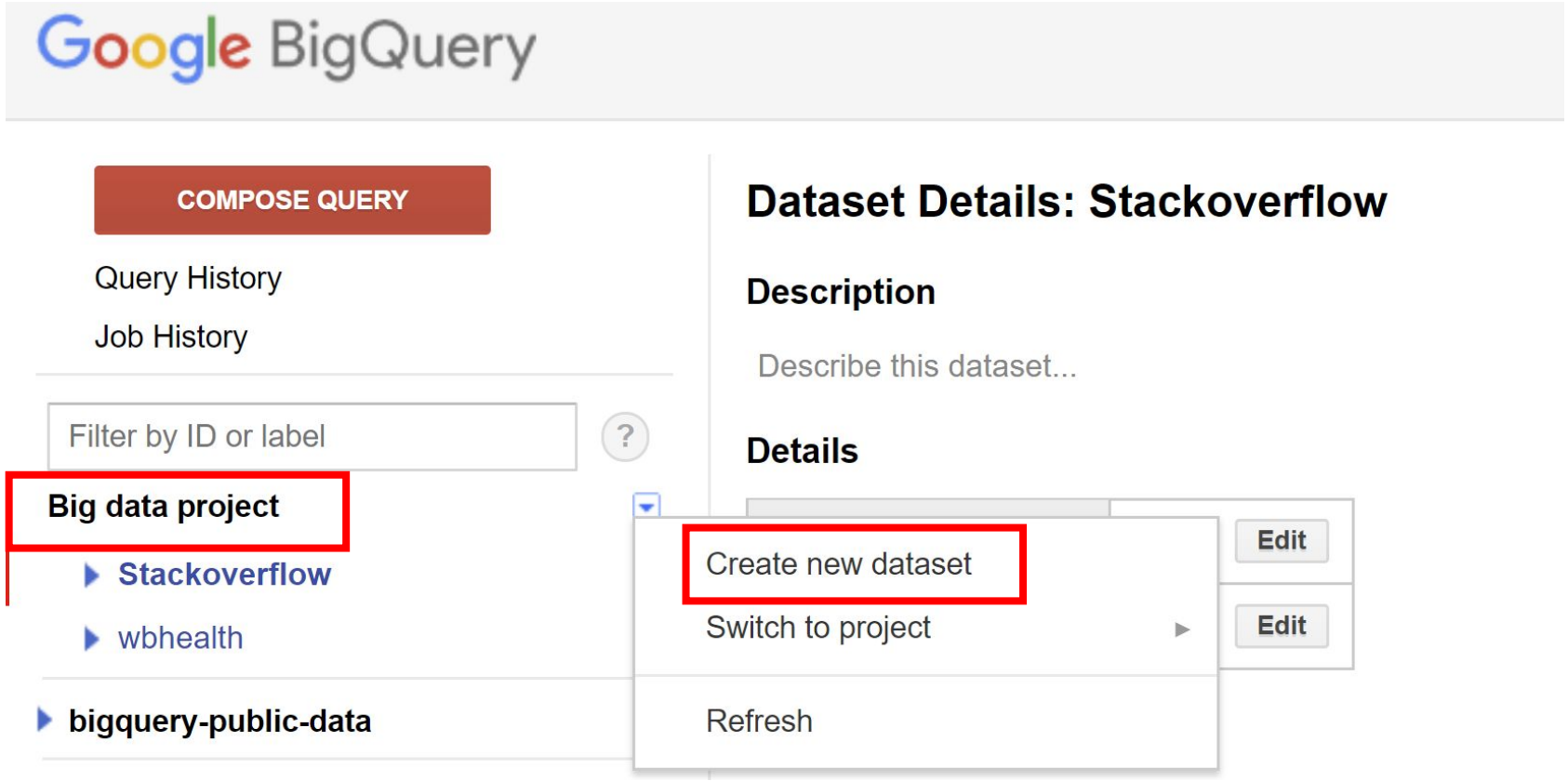
After creating project, go to Google Cloud Platform, and click BigQuery in Resources section to go to query page.



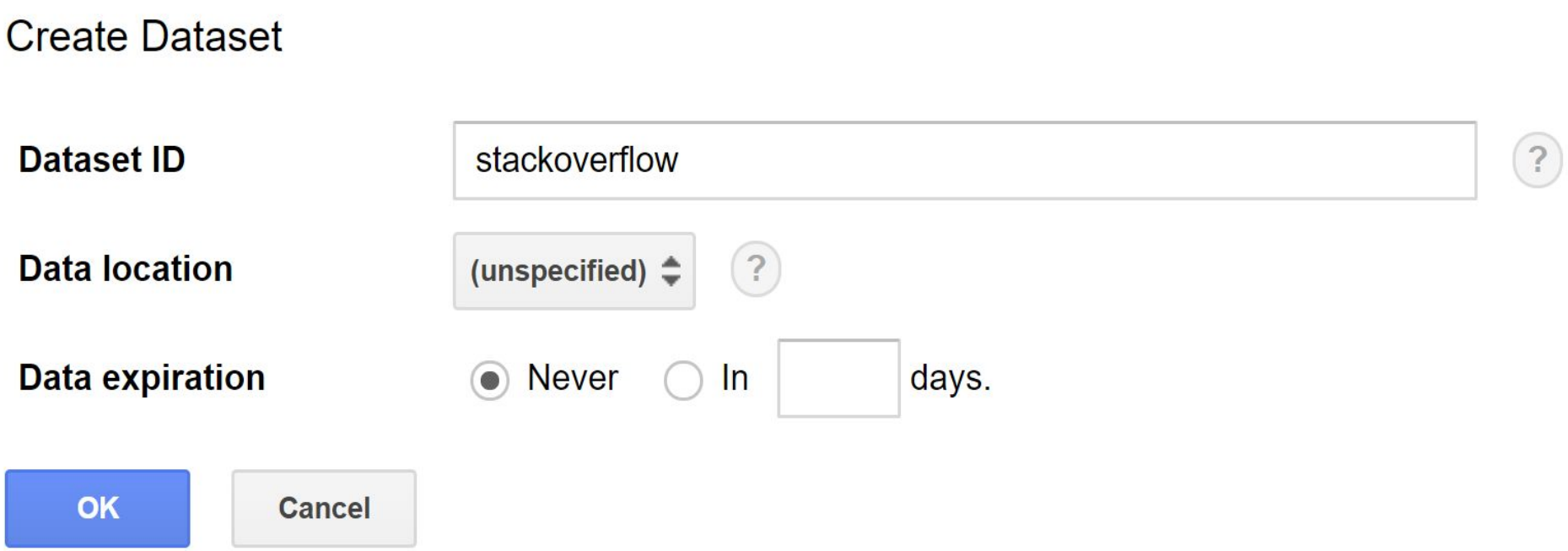



# Create database

After creating project, you can click “create new dataset” to easily create a blank database in our project.

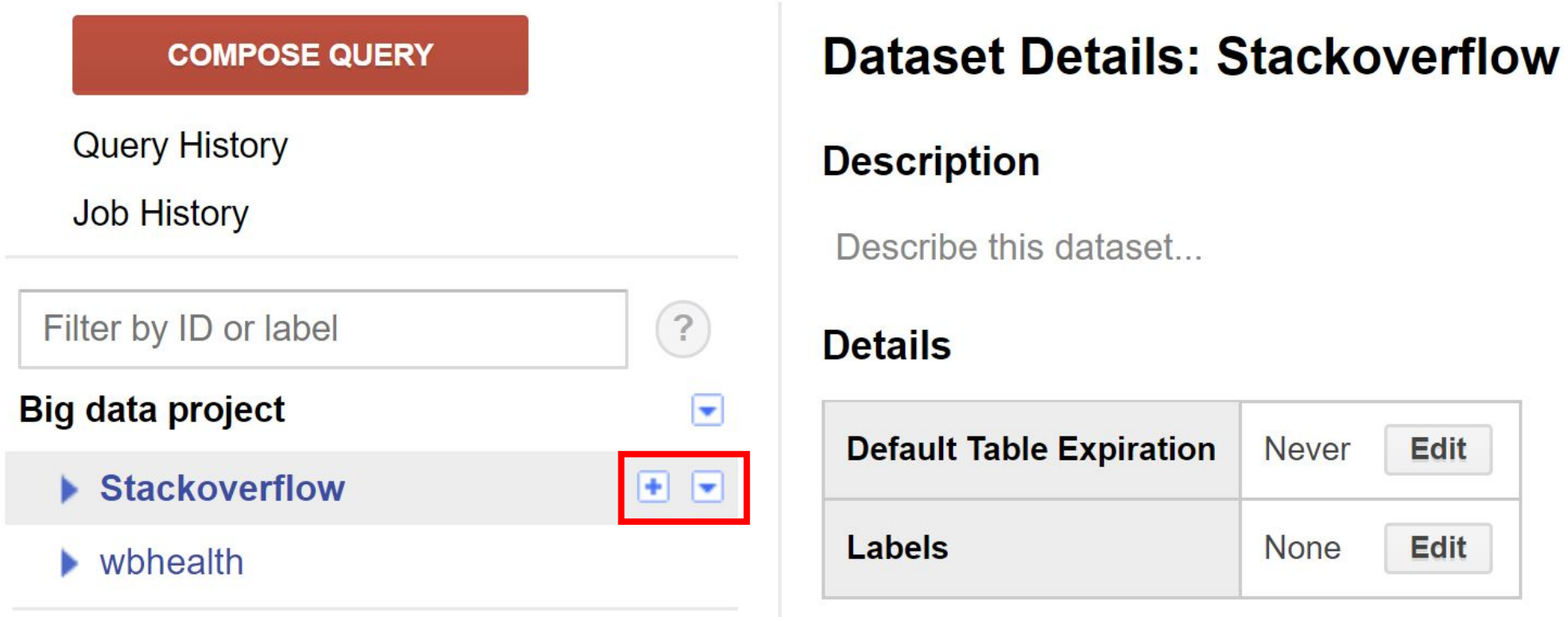


# Create database



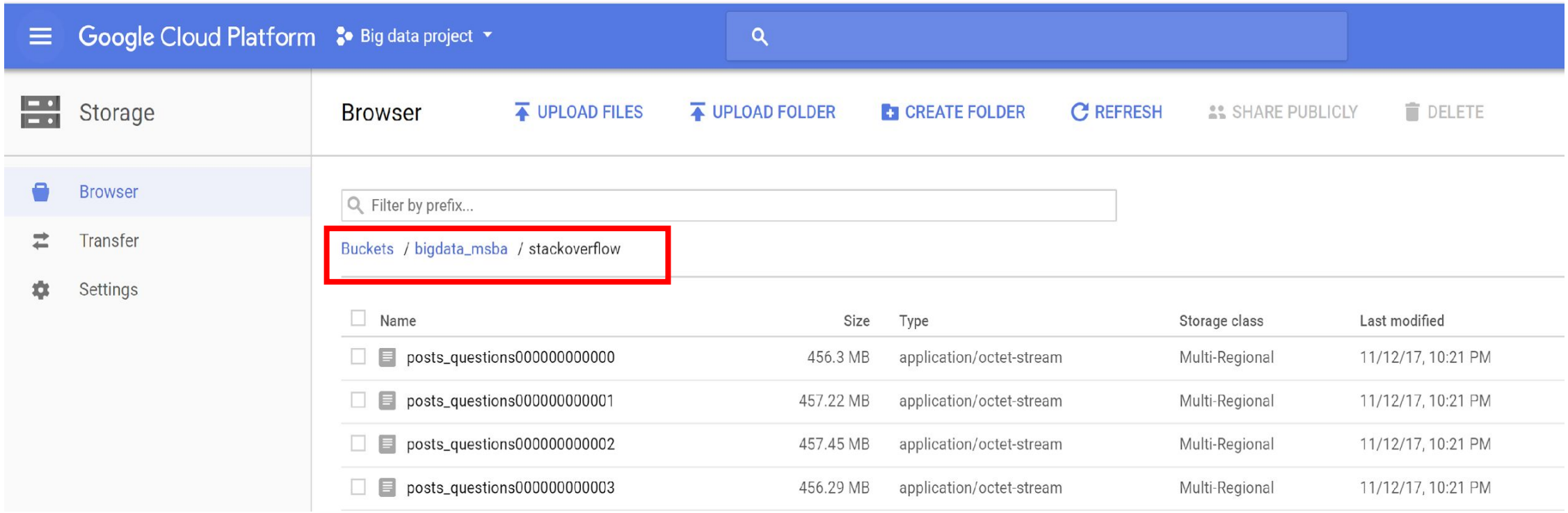
# Create table

After database is created, you can click “+” after the database name to create as many table as you want.

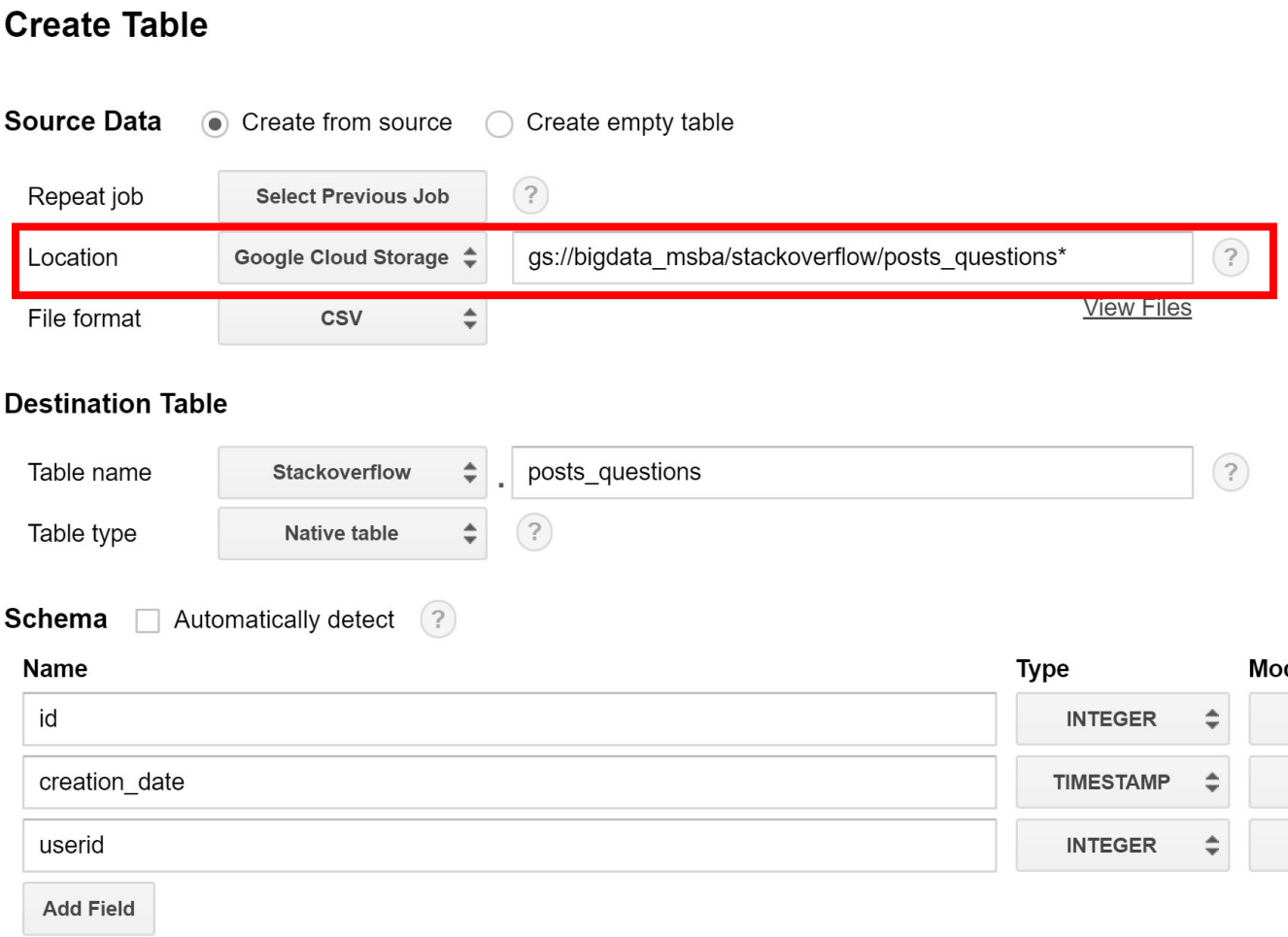


# Create table

This is the platform where I store raw data which is data source for my new table, remember the path to the file and paste it.



# Create table




There are several sources you can choose, for example, file from your local machine or from Google Cloud Platform. For example, we want to import data which stored in Google Cloud Platform.

Then we should specify the file location in the Google Cloud Platform.

Google Cloud Storage URIs begin with "gs://" and specify the bucket and object you want to load. **Example**: gs://mybucket/path/to/mydata.csv. You can use a wildcard to load multiple files

# Several Remarks on Loading Data

**Query data without loading it**

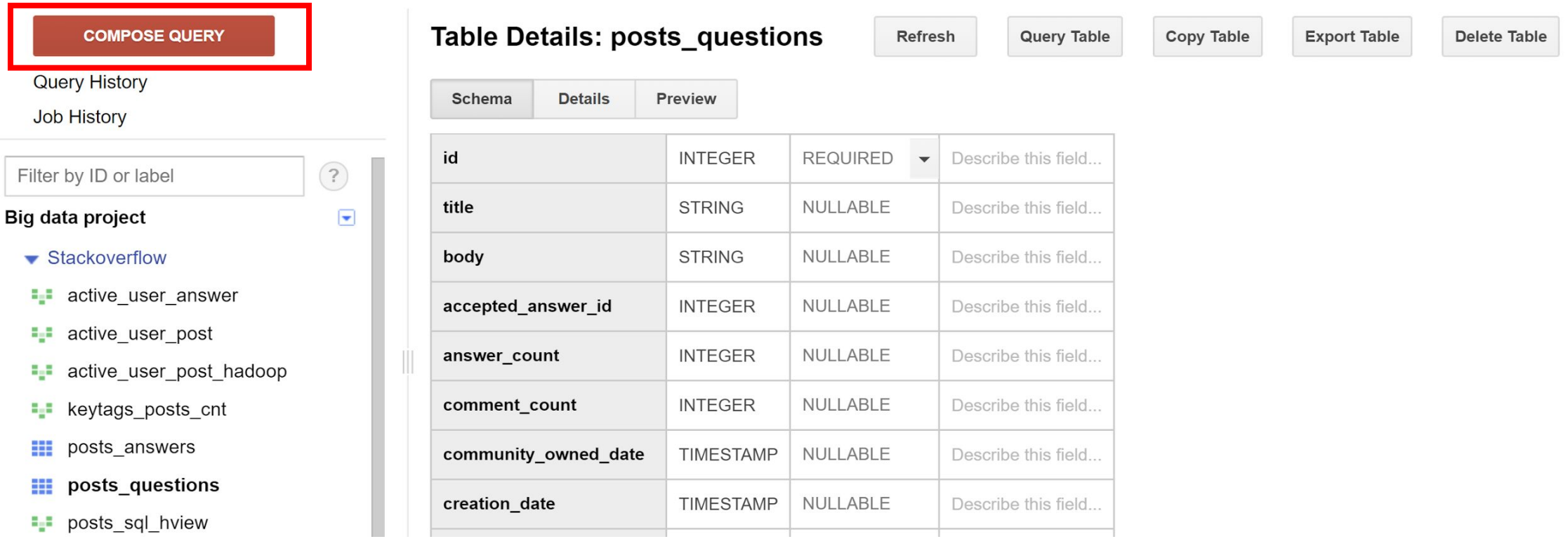
- Public datasets:
  - Public datasets are datasets stored in BigQuery and shared with the public.
- Shared datasets:
  - You can share datasets stored in BigQuery. If someone has shared a dataset with you, you can run queries on that dataset without loading the data.
- External data sources:
  - You can skip the data loading process by creating a table that is based on an external data source.
- Stackdriver log files:
  - Cloud Logging provides an option to export log files into BigQuery.
- Stream the data one record at a time.
  - Typically used when you need the data to be available immediately.

**Query data by loading it:**

- From Google Cloud Storage
- From a readable data source (such as your local machine)
- By inserting individual records using streaming inserts
- Using DML statements to perform bulk inserts
- Using a Google Cloud Dataflow pipeline to write data to BigQuery

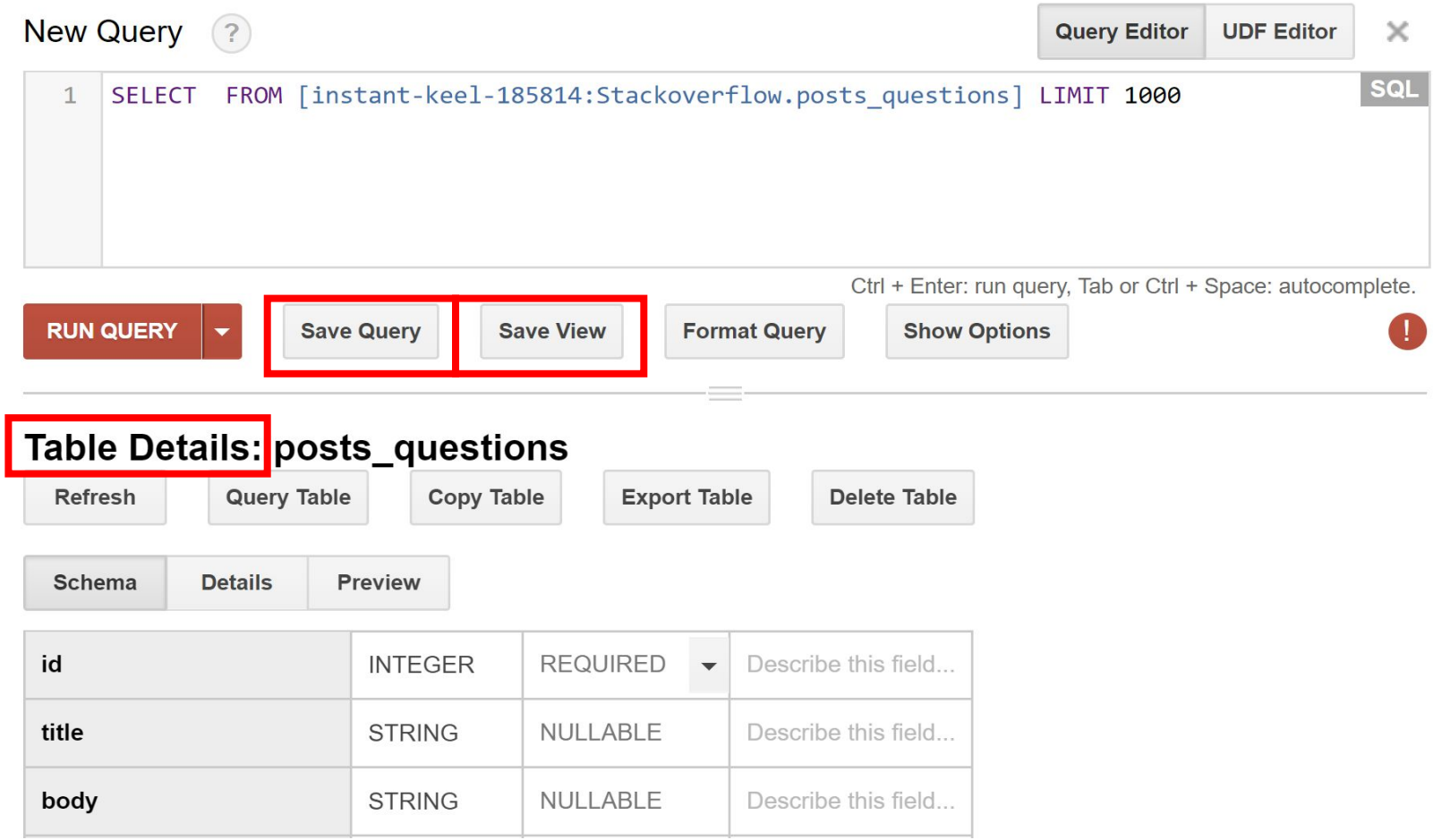
# Write, Execute, and Save Queries

Just click “Compose Query”



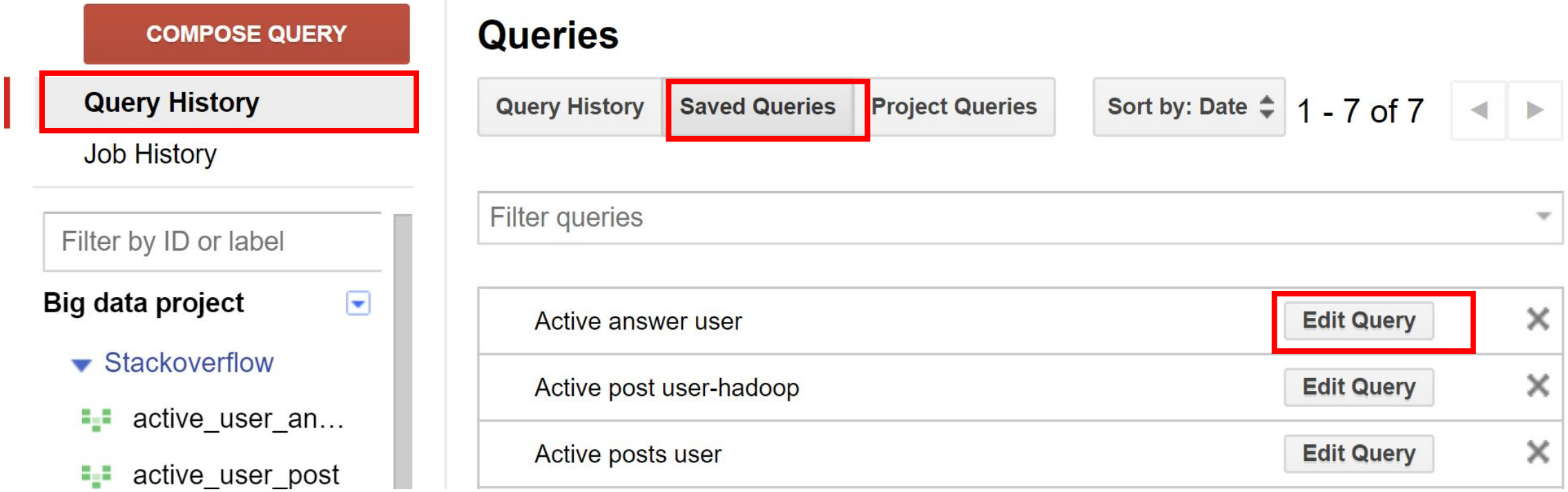
# Write, Execute, and Save Queries

New Query will be on the top of the screen with table schema below, so it will be more convenient to check fields in tables while writing queries. And we can save query and view.



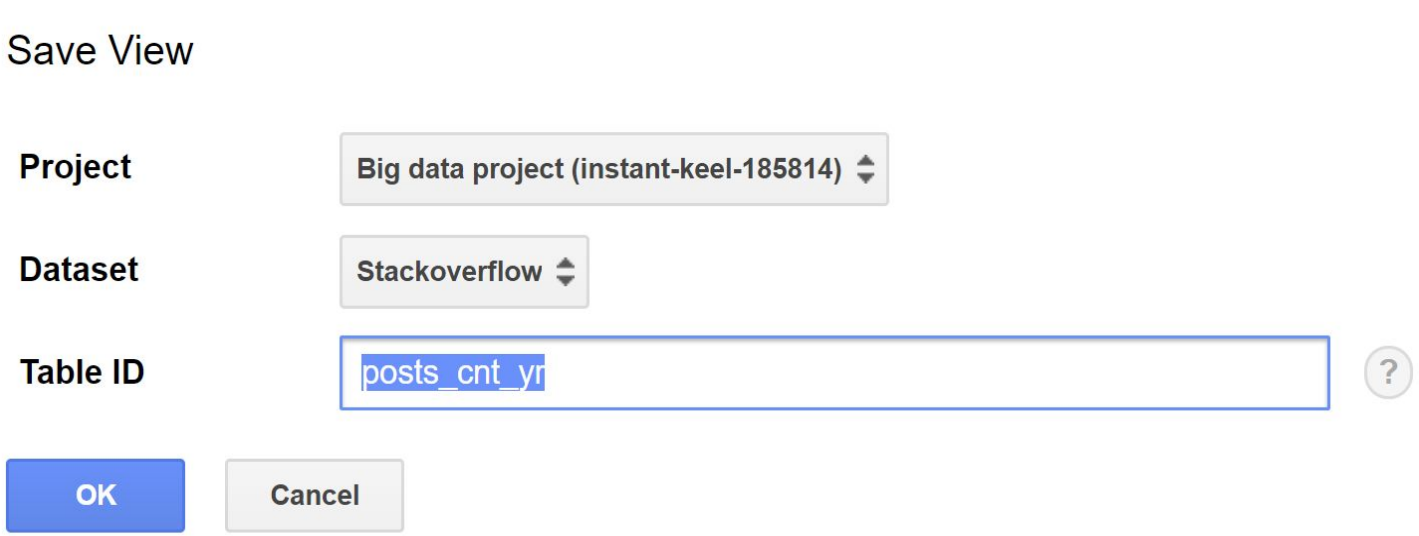
# Write, Execute, and Save Queries

You can also **save query** for future use. If you click “Query history” on the upper left side, you can see query history, saved queries, and project queries. Edit the saved queries and re-run them.



# Write, Execute, and Save Queries

**Save view**: if you want to create a table based on queries on other tables, you can just click “save view” to create a view without coding.




# Write, Execute, and Save Queries

Error message will change once you correct the error, so it will avoid running incorrect queries several times.
Also you can see how much data the query will process.

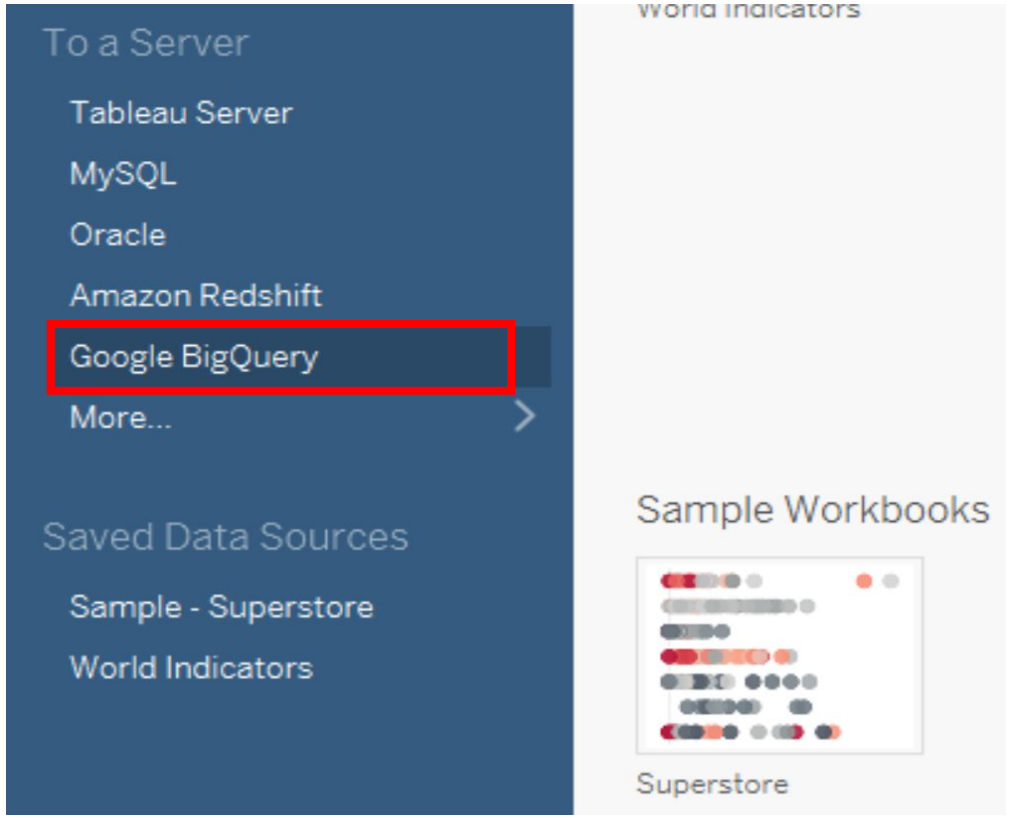
```
1  SELECT
2    u.display_name,
3    pq.owner_user_id,
4    COUNT(id) AS post_cnt
5  FROM
6    [instant-keel-185814:Stackoverflow.post
7  JOIN
8    [instant-keel-185814:Stackoverflow.user
9  ON
10   pq.owner_user_id = u.id
11 WHERE
```

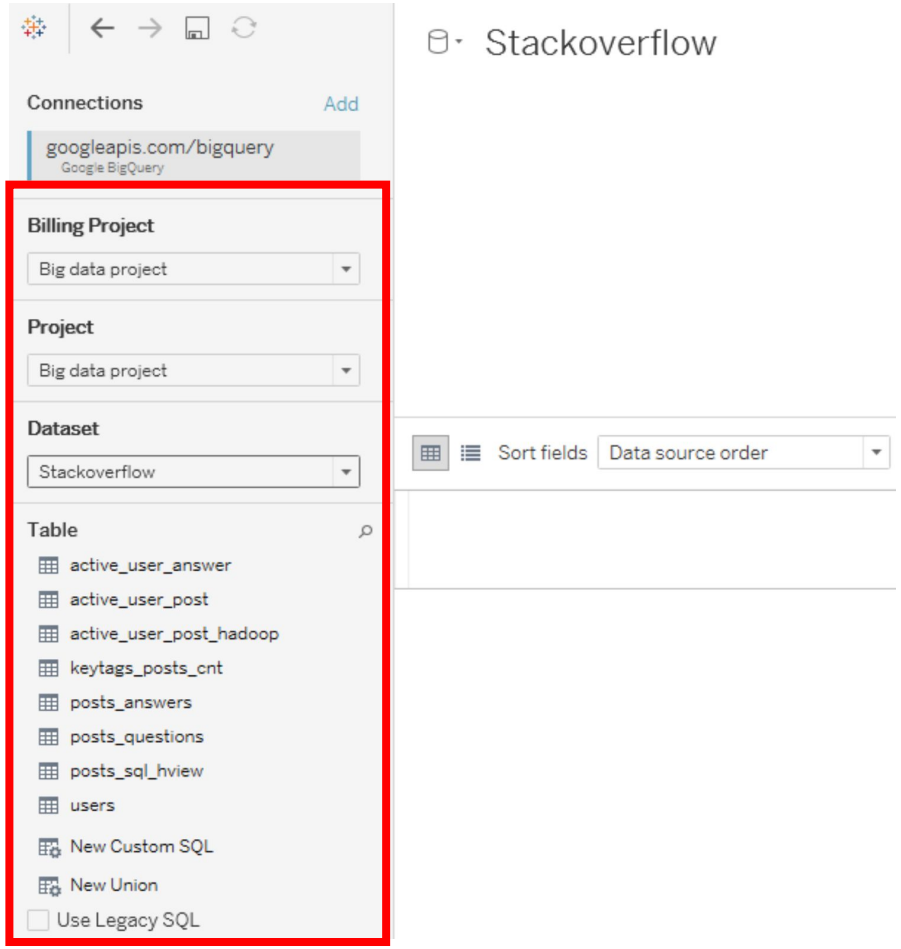
**Error:** 3.9 - 3.10: Ambiguous field reference id.

```
1  SELECT
2    u.display_name,
3    pq.owner_user_id,
4    COUNT(pq.id) AS post_cnt
5  FROM
6    [instant-keel-185814:Stackoverflow.posts_questions] pq
7  JOIN
8    [instant-keel-185814:Stackoverflow.users] u
9  ON
10   pq.owner_user_id = u.id
11 WHERE
```

**Valid:** This query will process 476 MB when run.

# Link with other software, e.g. Tableau



# Stackoverflow Application

# Stackoverflow Database

This dataset was last updated August, 2017. Originally, there are 16 tables, and we choose the following three to include in our database:

| Table ID | posts_questions | posts_answers | users |
|---|---|---|---|
| Table Size | 21.8 GB | 17.7 GB | 1.10 GB |
| Number of Rows | 14,458,875 | 22,668,556 | 7,617,191 |

# Post questions count by tags by year-month

```
1  SELECT * FROM
2  (SELECT 'SQL' AS tag, YEAR(creation_date) AS Year, MONTH(creation_date) AS Month, COUNT(id) AS post_cnt
3  FROM Stackoverflow.posts_questions
4  WHERE tags LIKE '%sql%' GROUP BY tag,Year,Month),
5  (SELECT 'Hadoop' AS tag, YEAR(creation_date) AS Year, MONTH(creation_date) AS Month,COUNT(id) AS post_cnt
6  FROM Stackoverflow.posts_questions
7  WHERE tags LIKE '%hadoop%'GROUP BY tag,Year,Month),
8  (SELECT 'MySQL' AS tag, YEAR(creation_date) AS Year, MONTH(creation_date) AS Month,COUNT(id) AS post_cnt
9  FROM Stackoverflow.posts_questions
10 WHERE tags LIKE '%mysql%'GROUP BY tag,Year,Month),
11 (SELECT 'Hive' AS tag, YEAR(creation_date) AS Year, MONTH(creation_date) AS Month,COUNT(id) AS post_cnt
```

**Valid:** This query will process 591 MB when run.

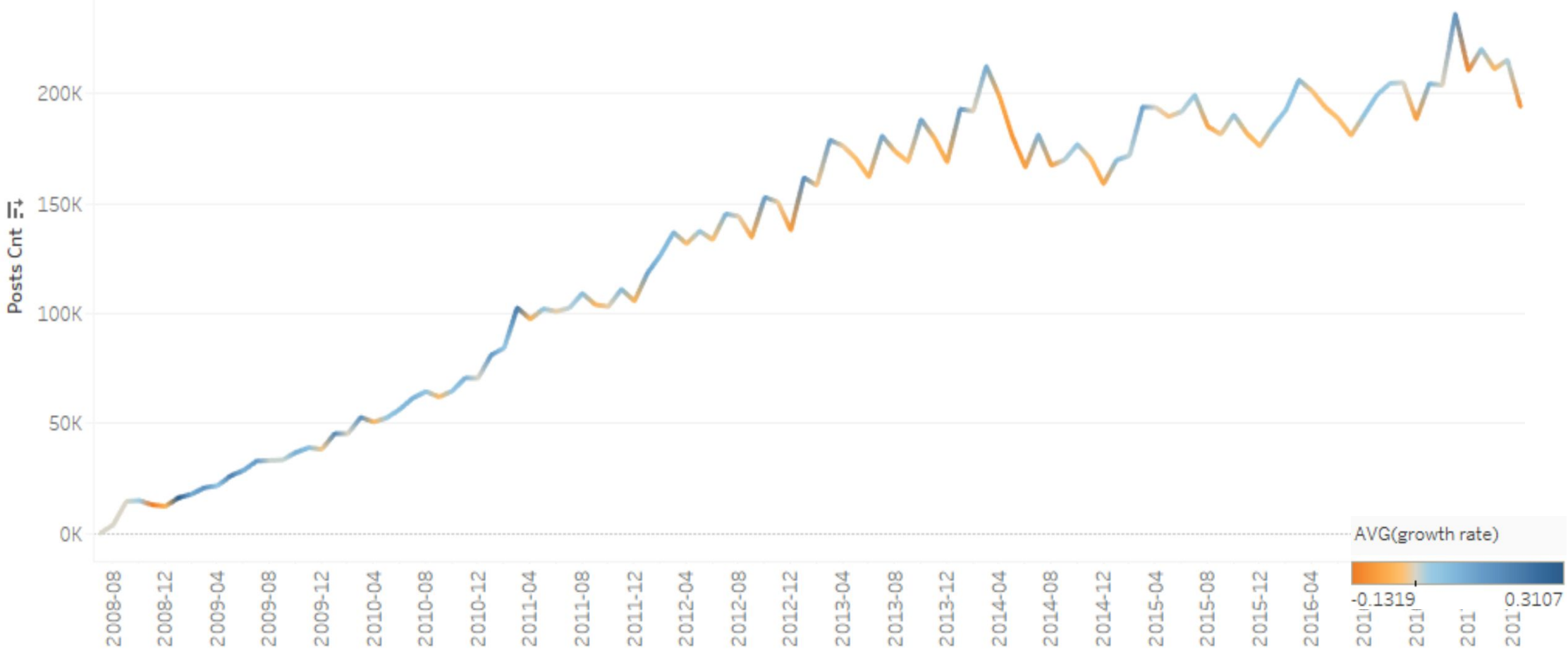
RUN QUERY | Save Query | Save View | Format Query | Show Options | Query complete (1.5s elapsed, cached)

Results | Explanation | Job Information | Download as CSV

| Row | tag | Year | Month | post_cnt |
|---|---|---|---|---|
| 1 | AWS | 2008 | 8 | 4 |
| 2 | AWS | 2008 | 9 | 5 |
| 3 | AWS | 2008 | 10 | 14 |
| 4 | AWS | 2008 | 11 | 4 |
| 5 | AWS | 2008 | 12 | 2 |
| 6 | AWS | 2009 | 1 | 10 |

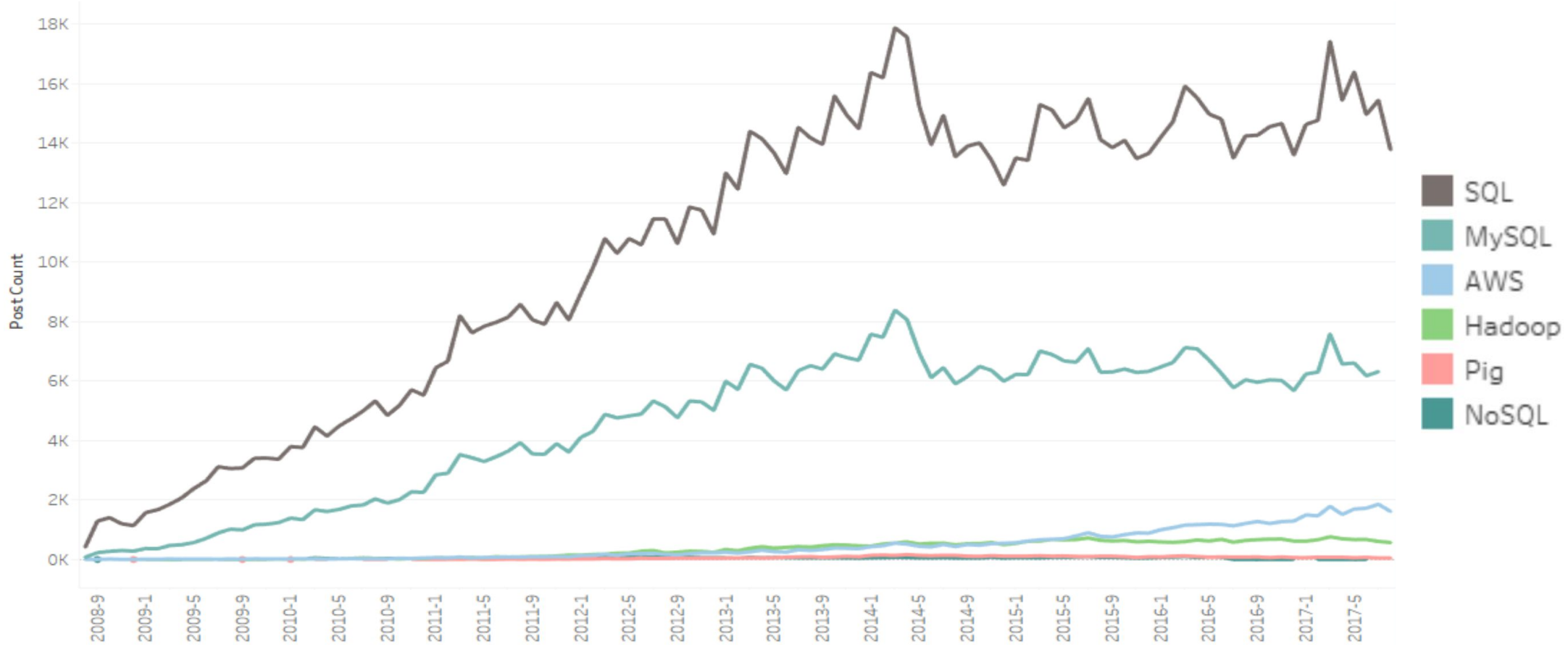
# Post questions count by year

The trend is going upward overtime, but the rate of posting is slowing down



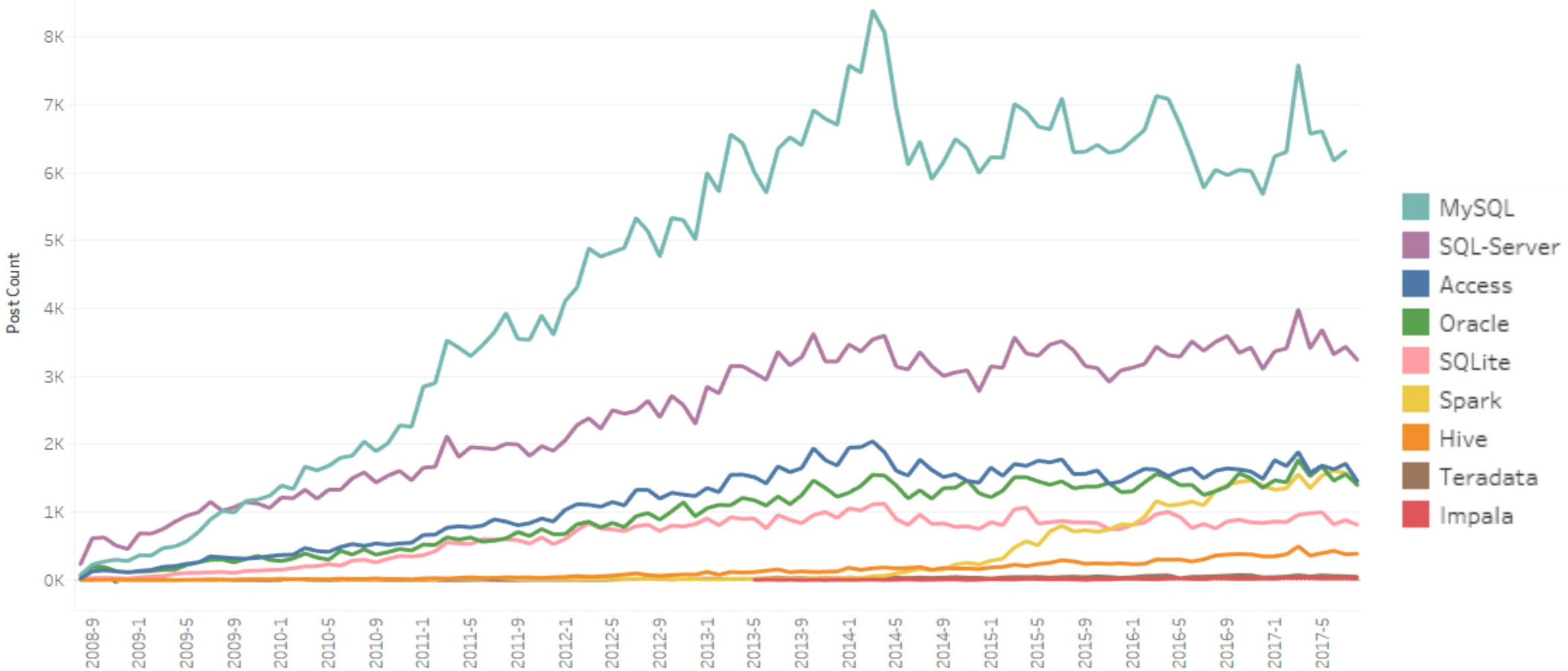
# Question count by tags:
## Tools covered in class

SQL and MySQL are still the most popular tools



# Question count by tags:

## Popular relational DBMS



# SQL-Related Qs with highest view in 2017

```sql
SELECT
  id,title,view_count
FROM
  [instant-keel-185814:Stackoverflow.posts_questions]
WHERE
  YEAR(creation_date)==2017 AND
  tags LIKE '%sql%'
ORDER BY
  view_count DESC
LIMIT 1;
```

**Valid:** This query will process 1.41 GB when run.

RUN QUERY | Save Query | Save View | Format Query | Show Options | Query complete (1.0s elapsed, cached)

Results | Explanation | Job Information | Download as CSV

| Row | id | title | view_count |
|---|---|---|---|
| 1 | 41645309 | MySQL Error: : 'Access denied for user 'root'@'localhost' | 60176 |

# The Post Question

## MySQL Error: : 'Access denied for user 'root'@'localhost'


asked 10 months ago

viewed 95,430 times

active 20 days ago

8

7

$ **./mysqladmin -u root -p** '*redacted*'
Enter password:

> mysqladmin: connect to server at 'localhost' failed error:
> **'Access denied for user 'root'@'localhost' (using password: YES)'**

How can I fix this?

mysql sql database database-connection

BLOG

The Cliffs of Insa
Technologies on

Looking f

# Hadoop-related Qs with most favorite in 2017

```sql
SELECT
  id,title,favorite_count
FROM
  [instant-keel-185814:Stackoverflow.posts_questions]
WHERE
  YEAR(creation_date)==2017 AND
  tags LIKE '%hadoop%'
ORDER BY
  favorite_count DESC
LIMIT 1;
```

**Valid:** This query will process 1.33 GB when run.

RUN QUERY | Save Query | Save View | Format Query | Show Options | Query complete (0.5s elapsed, cached)

Results | Explanation | Job Information | Download as CSV

| Row | id | title | favorite_count |
|---|---|---|---|
| 1 | 43270820 | How to restart a failed task on Airflow | 5 |

# The Post Question

## How to restart a failed task on Airflow



asked 7 months ago
viewed 1,565 times
active 1 month ago

7

5

I am using a **LocalExecutor** and my dag has **3 tasks** where task(C) is dependant on task(A). Task(B) and task(A) can run in parallel something like below

A-->C

B

**So task(A) has failed** and but **task(B) ran fine**. Task(C) is yet to run as task(A) has failed.

My question is **how do i re run Task(A) alone so Task(C) runs** once Task(A) completes and Airflow UI marks them as success.

python hadoop airflow apache-airflow bigdata

BLOG

The Cliffs of Ins
Technologies or



Google BigQuery

# Active user with most Qs in 2017

```
SELECT
  pq.owner_user_id,u.display_name,u.reputation,u.location,COUNT(pq.id) AS post_cnt
FROM
  [instant-keel-185814:Stackoverflow.posts_questions] pq
JOIN
  [instant-keel-185814:Stackoverflow.users] u
ON pq.owner_user_id = u.id
WHERE YEAR(pq.creation_date)==2017
GROUP BY u.display_name, pq.owner_user_id, u.reputation,u.location
ORDER BY post_cnt DESC
LIMIT 10;
```

**Valid:** This query will process 572 MB when run.

| Row | userid | u_display_name | u_reputation | u_location | post_cnt |
|---|---|---|---|---|---|
| 1 | 1223975 | Alexander Mills | 8041 | San Francisco, CA, United States | 243 |
| 2 | 156458 | Tim | 23388 | | 237 |
| 3 | 258483 | Dims | 7573 | Moscow, Russia | 227 |
| 4 | 1235929 | Dave | 1181 | | 209 |
| 5 | 5421539 | Zhao Yi | 2516 | | 207 |

# Active user with most Hadoop-related Qs in 2017

```sql
SELECT
  pq.owner_user_id, u.display_name, u.reputation, u.location, u.up_votes, u.down_votes, COUNT(pq.id) AS post_cnt
FROM
  [instant-keel-185814:Stackoverflow.posts_questions] pq
JOIN
  [instant-keel-185814:Stackoverflow.users] u
ON pq.owner_user_id = u.id
WHERE YEAR(pq.creation_date)==2017 AND pq.tags LIKE '%hadoop%'
GROUP BY u.display_name, pq.owner_user_id, u.reputation, u.up_votes, u.down_votes, u.location
ORDER BY post_cnt DESC
LIMIT 20;
```

**Valid:** This query will process 1.03 GB when run.

| Row | pq_owner_user_id | u_display_name | u_reputation | u_location | post_cnt |
|---|---|---|---|---|---|
| 1 | 2235335 | Jain Hemant | 62 | Ahmedabad, Gujarat, India | 31 |
| 2 | 4287344 | oula alshiekh | 114 | | 28 |
| 3 | 7119501 | Sidhartha | 143 | Bangalore, Karnataka, India | 27 |
| 4 | 5636416 | earl | 130 | | 23 |
| 5 | 3544612 | Saurab | 125 | | 22 |
| 6 | 3438473 | CuriousMind | 1302 | | 21 |
| 7 | 1460514 | SUDARSHAN | 137 | Bangalore, Karnataka, India | 20 |
| 8 | 4751033 | Basil Paul | 62 | Chennai, Tamil Nadu, India | 19 |
| 9 | 3454410 | Shafiq | 1551 | | 15 |
| 10 | 3956731 | KayV | 1608 | Gurgaon, India | 15 |

# Active user with most answers in 2017

```
SELECT
  pa.owner_user_id AS userid, u.display_name, u.reputation, u.location, COUNT(pa.id) AS answer_cnt
FROM
  [instant-keel-185814:Stackoverflow.posts_answers] pa
JOIN
  [instant-keel-185814:Stackoverflow.users] u
ON pa.owner_user_id = u.id
WHERE YEAR(pa.creation_date)==2017
GROUP BY userid, u.display_name, pa.owner_user_id, u.reputation, u.location
ORDER BY answer_cnt DESC
LIMIT 20;
```

| Row | userid | u_display_name | u_reputation | u_location | answer_cnt |
|---|---|---|---|---|---|
| 1 | 1144035 | Gordon Linoff | 579631 | New York, United States | 5940 |
| 2 | 2901002 | jezrael | 153048 | Bratislava, Slovakia | 3084 |
| 3 | 3732271 | akrun | 285053 | | 2632 |
| 4 | 3832970 | Wiktor Stribiżew | 211872 | Warsaw, Poland | 2084 |
| 5 | 2336654 | piRSquared | 84859 | Bellevue, WA, United States | 2042 |

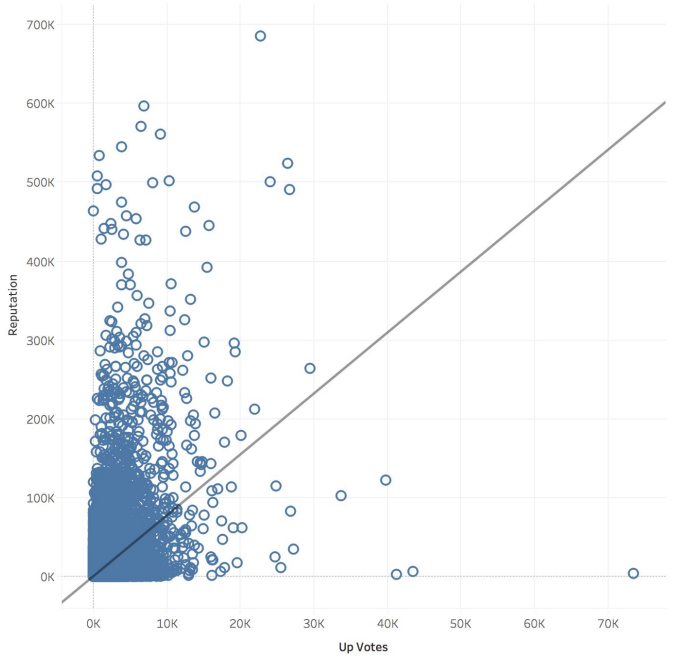
# Accepted answers & User Reputation

```
SELECT acc.userid, acc.name, acc.u.creation_date AS creation_date, acc.u.location AS location,
acc.u.up_votes AS up_votes,
acc.u.reputation AS reputation, acc.acc_cnt AS accepted_cnt, a.a_cnt AS answer_cnt
FROM
(SELECT pa.owner_user_id AS userid, u.display_name AS name, u.creation_date, u.location,
u.up_votes, u.reputation, COUNT(pq.accepted_answer_id) AS acc_cnt
FROM [instant-keel-185814:Stackoverflow.posts_questions] pq
JOIN [instant-keel-185814:Stackoverflow.posts_answers] pa
ON pq.accepted_answer_id = pa.id
JOIN [instant-keel-185814:Stackoverflow.users] u
ON pa.owner_user_id   = u.id
GROUP BY userid, name, u.creation_date, u.location, u.up_votes, u.reputation
ORDER BY acc_cnt DESC) acc
JOIN
(SELECT pa.owner_user_id AS userid, COUNT(pa.id) AS a_cnt
FROM [instant-keel-185814:Stackoverflow.posts_answers] pa
GROUP BY userid
ORDER BY a_cnt DESC) a ON acc.userid=a.userid
ORDER BY acc.acc_cnt DESC;
```

# Accepted answers & User Reputation

| Row | acc_userid | acc_name | creation_date | location | up_votes | reputation | accepted_cnt | answer_cnt |
|---|---|---|---|---|---|---|---|---|
| 1 | 22656 | Jon Skeet | 2008-09-26 12:05:05 UTC | Reading, United Kingdom | 15865 | 969386 | 20898 | 33911 |
| 2 | 1144035 | Gordon Linoff | 2012-01-11 19:53:57 UTC | New York, United States | 8129 | 579631 | 17235 | 37988 |
| 3 | 29407 | Darin Dimitrov | 2008-10-19 16:07:47 UTC | Sofia, Bulgaria | 1946 | 751190 | 12702 | 21503 |
| 4 | 100297 | Martijn Pieters | 2009-05-03 14:53:57 UTC | Cambridge, United Kingdom | 5480 | 579284 | 12695 | 17571 |
| 5 | 115145 | CommonsWare | 2009-05-31 16:20:08 UTC | Pennsylvania, United States | 9059 | 666690 | 11723 | 19573 |

Number of UpVotes v. User Reputation



# Questions?