Using BigQuery to do Analysis

## Overview

In this lab you analyze 2 different public datasets, run queries on them, separately and then combined, to derive interesting insights.

### **What you'll learn**

In this lab, you will:

* Carry out interactive queries on the BigQuery console.
* Combine and run analytics on multiple datasets.

## Introduction

This lab uses two public datasets in BigQuery: weather data from the US National Oceanic and Atmospheric Administration (NOAA), and bicycle rental data from New York City.

You will encounter, for the first time, several aspects of Google Cloud Platform that are of great benefit to scientists:

1. **Serverless.** No need to download data to your machine in order to work with it - the dataset will remain on the cloud.
2. **Ease of use.** Run ad-hoc SQL queries on your dataset without having to prepare the data, like indexes, beforehand. This is invaluable for data exploration.
3. **Scale.** Carry out data exploration on extremely large datasets interactively. You don't need to sample the data in order to work with it in a timely manner.
4. **Shareability**. You will be able to run queries on data from different datasets without any issues. BigQuery is a convenient way to share datasets. Of course, you can also keep your data private, or share them only with specific persons -- not all data need to be public.

The end-result is that you will find if there are lesser bike rentals on rainy days.

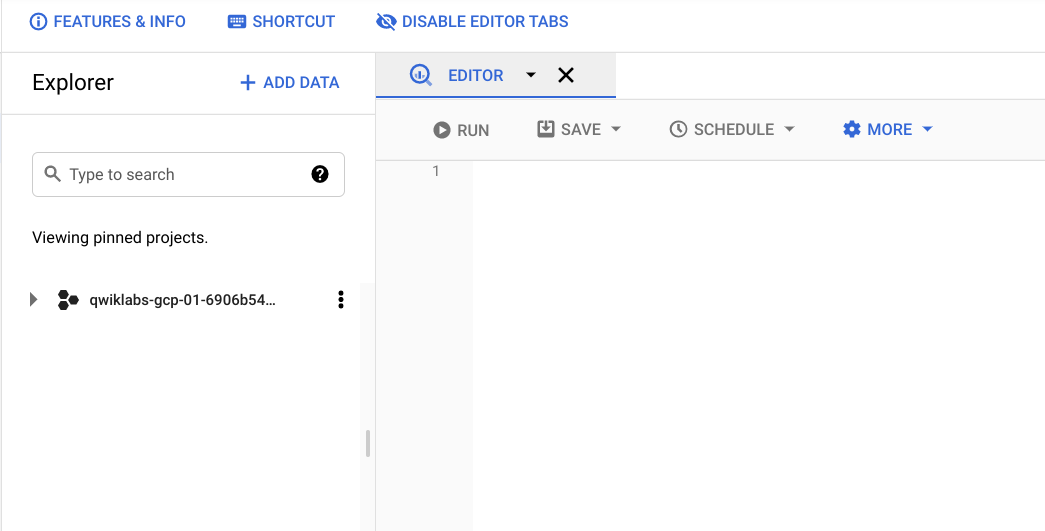
## Explore bicycle rental data

### **Open BigQuery Console**

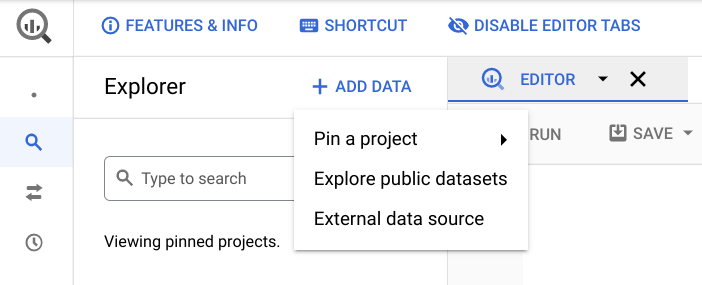
1. In the Google Cloud Console, select **Navigation menu** > **BigQuery**.

The **Welcome to BigQuery in the Cloud Console** message box opens. This message box provides a link to the quickstart guide and lists UI updates.

1. Click **Done**.



1. In the left pane, click **ADD DATA** > **Explore public datasets**.

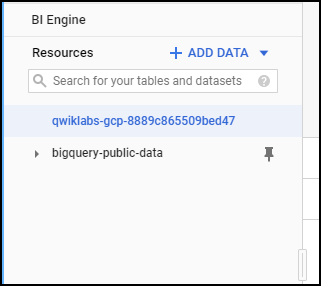


The Datasets window opens.

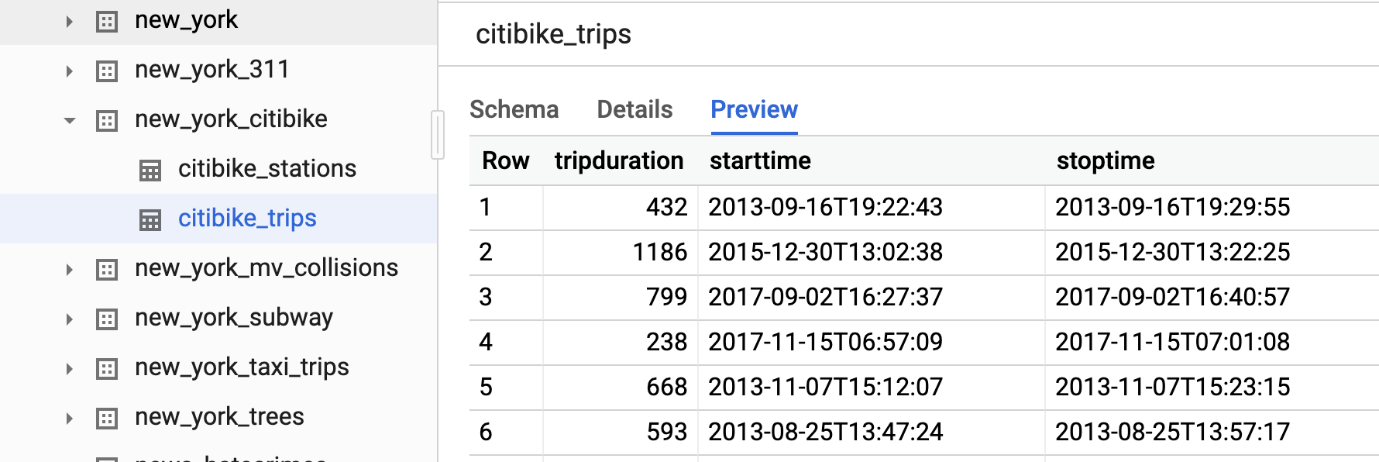
1. In the **Search** bar, type "NYC bike" then press **Enter**.
2. One result **NYC Citi Bike Trips** is returned. Click on the dataset name and then **View Dataset**.

Your The Google BigQuery console opens in a new browser tab.To keep your workspace organized, close this new browser tab and refresh the first tab.

1. In the BigQuery console (in the first browser tab) you see two projects in the left pane, one named your Qwiklabs project ID, and one named **bigquery-public-data**.



1. In the left pane of the BigQuery console, select **bigquery-public-data** > **new\_york\_citibike** > **citibike\_trips** table.
2. In the Table (citibike\_trips) window, click the Preview tab.



1. Examine the columns and some of the data values.

Click **Compose New Query** and enter the following:

SELECT

MIN(start\_station\_name) AS start\_station\_name,

MIN(end\_station\_name) AS end\_station\_name,

APPROX\_QUANTILES(tripduration, 10)[OFFSET (5)] AS typical\_duration,

COUNT(tripduration) AS num\_trips

FROM

`bigquery-public-data.new\_york\_citibike.citibike\_trips`

WHERE

start\_station\_id != end\_station\_id

GROUP BY

start\_station\_id,

end\_station\_id

ORDER BY

num\_trips DESC

LIMIT

10

Copied!

content\_copy

Click **Run**. Look at the result and try to determine what this query does ? (Hint: typical duration for the 10 most common one-way rentals)

1. Next, run the below to find another interesting fact: total distance travelled by each bicycle in the dataset. Note that the query limits the results to only top 5.

WITH

trip\_distance AS (

SELECT

bikeid,

ST\_Distance(ST\_GeogPoint(s.longitude,

s.latitude),

ST\_GeogPoint(e.longitude,

e.latitude)) AS distance

FROM

`bigquery-public-data.new\_york\_citibike.citibike\_trips`,

`bigquery-public-data.new\_york\_citibike.citibike\_stations` as s,

`bigquery-public-data.new\_york\_citibike.citibike\_stations` as e

WHERE

start\_station\_id = s.station\_id

AND end\_station\_id = e.station\_id )

SELECT

bikeid,

SUM(distance)/1000 AS total\_distance

FROM

trip\_distance

GROUP BY

bikeid

ORDER BY

total\_distance DESC

LIMIT

5

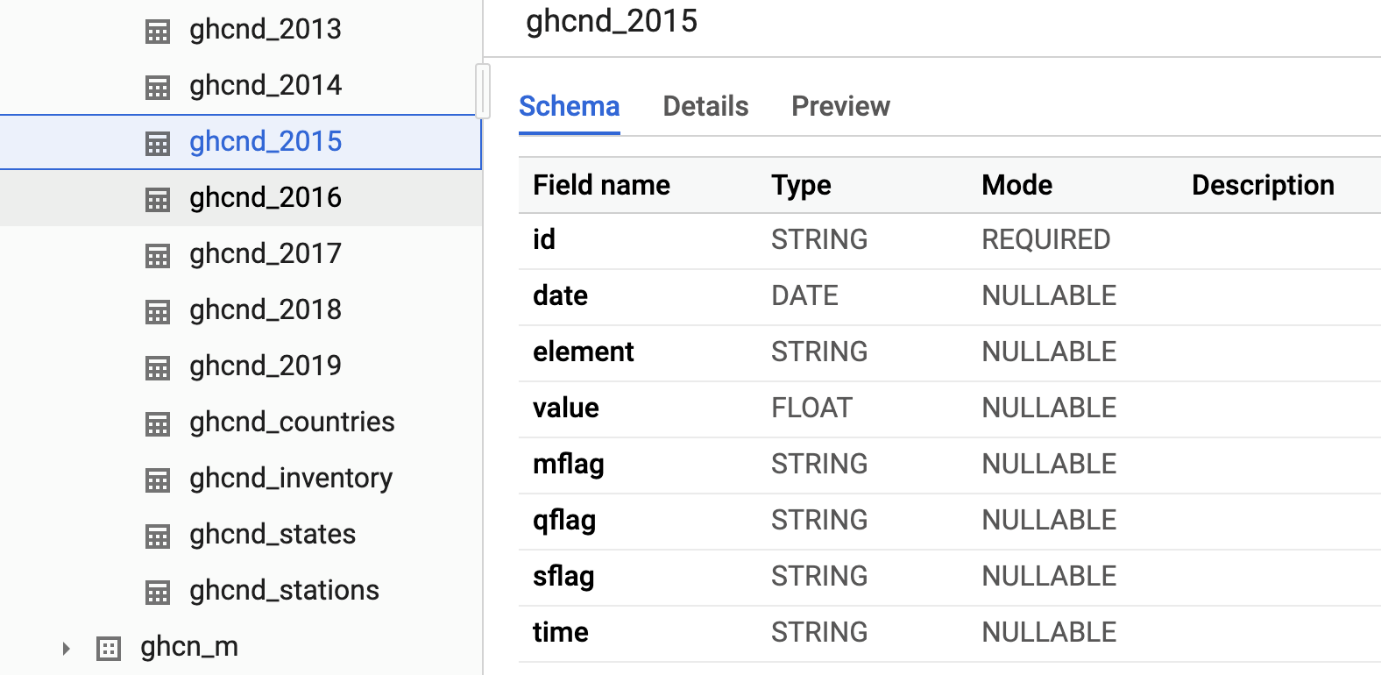
Copied!

content\_copy

**Note:** For this query, we also used the other table in the dataset called **citibike\_stations** to get bicycle station information.

## Explore the weather dataset

In the left pane of the BigQuery Console, select the newly added bigquery-public-data project and select **ghcn\_d** > **ghcnd\_2015**. Then click on the **Preview** tab. Your console should resemble the following:



Examine the columns and some of the data values.

Click **Compose New Query** and enter the following:

SELECT

wx.date,

wx.value/10.0 AS prcp

FROM

`bigquery-public-data.ghcn\_d.ghcnd\_2015` AS wx

WHERE

id = 'USW00094728'

AND qflag IS NULL

AND element = 'PRCP'

ORDER BY

wx.date

Copied!

content\_copy

Click **Run**.

This query will return rainfall (in mm) for all days in 2015 from a weather station in New York whose id is provided in the query (the station corresponds to NEW YORK CNTRL PK TWR )

## Find correlation between rain and bicycle rentals

How about joining the bicycle rentals data against weather data to learn whether there are fewer bicycle rentals on rainy days?

Click **Compose New Query** and enter the following:

WITH bicycle\_rentals AS (

SELECT

COUNT(starttime) as num\_trips,

EXTRACT(DATE from starttime) as trip\_date

FROM `bigquery-public-data.new\_york\_citibike.citibike\_trips`

GROUP BY trip\_date

),

rainy\_days AS

(

SELECT

date,

(MAX(prcp) > 5) AS rainy

FROM (

SELECT

wx.date AS date,

IF (wx.element = 'PRCP', wx.value/10, NULL) AS prcp

FROM

`bigquery-public-data.ghcn\_d.ghcnd\_2015` AS wx

WHERE

wx.id = 'USW00094728'

)

GROUP BY

date

)

SELECT

ROUND(AVG(bk.num\_trips)) AS num\_trips,

wx.rainy

FROM bicycle\_rentals AS bk

JOIN rainy\_days AS wx

ON wx.date = bk.trip\_date

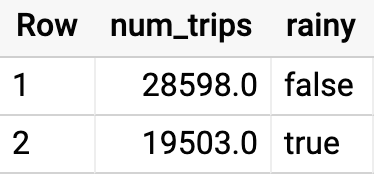
GROUP BY wx.rainy

Copied!

content\_copy

Click **Run**.

Now you can see the results of joining the bicycle rental dataset with a weather dataset that comes from a completely different source.



Running the query yields that, yes, New Yorkers ride the bicycle 47% fewer times when it rains.

## Summary

In this lab you did ad-hoc queries on two datasets. You were able to query the data without setting up any clusters, creating any indexes, etc. You were also able to mash up the two datasets and get some interesting insights. All without ever leaving your browser!

1.

Which of the following statements are true? (Choose TWO)

checkCloud SQL is optimized for high-throughput writes

Correct

checkBigQuery is optimized for high-read data

Correct



BigQuery is a row-based storage



Cloud SQL is optimized for high-read data

check

2.

Which of the following are the jobs of a data engineer? (Choose all that apply)

checkGet the data to where it can be useful

Correct

checkGet the data into a usable condition

Correct

checkAdd new value to the data

Correct

checkProductionize data processes

Correct

checkManage the data

Correct