MASTER IN ENTREPRENEURSHIP
INNOVATION MANAGEMENT
IN COLLABORATION WITH MIT SLOAN

IN COLLABORATION WITH
MIT MANAGEMENT SLOAN SCHOOL

UNIVERSITÀ DEGLI STUDI DI NAPOLI PARTHENOPE

MASTER MEIM 2021-2022

Digital Strategies - Clustering use cases Google Cloud Platform

Lesson given by prof. Alessio Ferone

Digital Strategies

Machine Learning with BigQuery ML

Agenda

- **Google Cloud Platform**
- Google BigQuery
- Google BigQuery SQL
- Google BigQuery ML

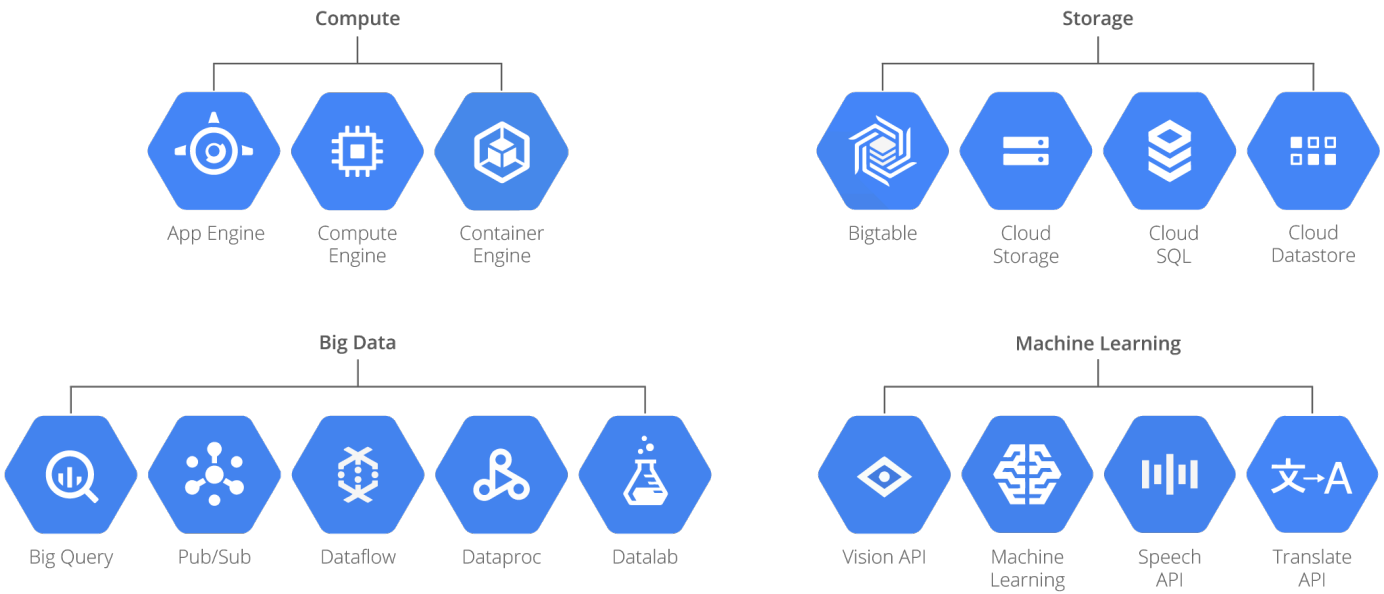
Introduction

- The adoption of the **public cloud** enables companies and users to access **innovative technologies**
- In particular for **Big Data** and **Artificial Intelligence** (AI)
- To be effective the new **AI capabilities** need to be **shared** between **different roles**
- Most cloud providers are currently addressing the challenge of **democratizing AI**
- In this context, **Google Cloud** provides several services to handle and process large amount of data

Google Cloud Platform

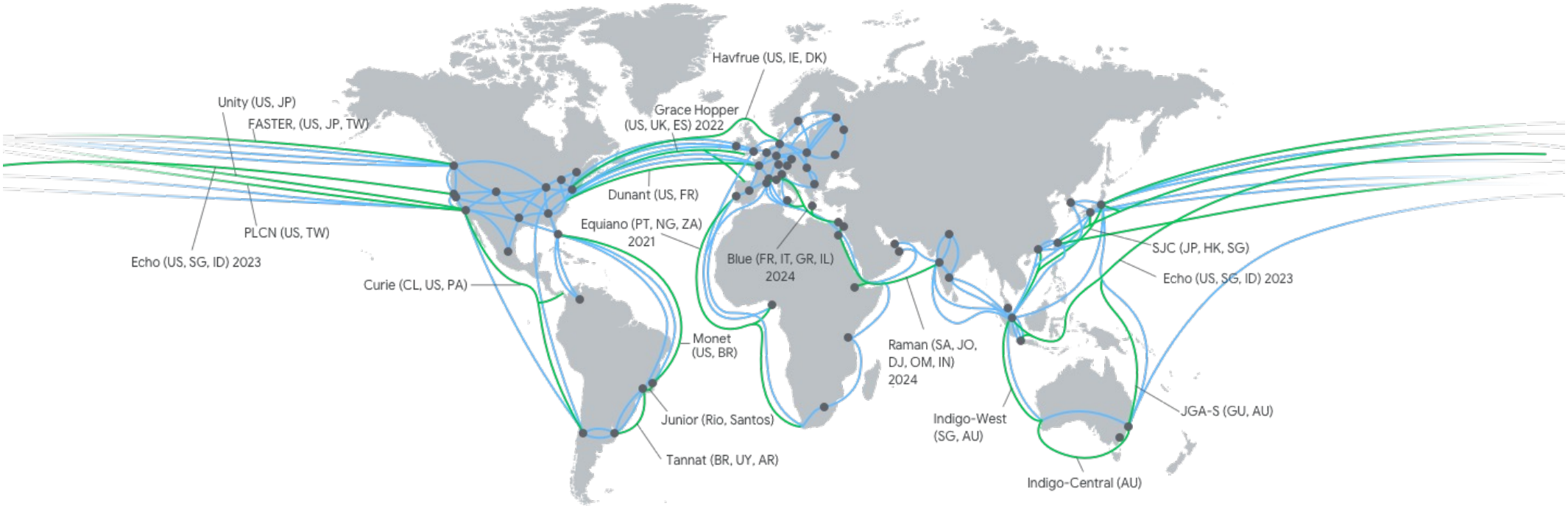
- Starting from 1998 with Google Search, Google developed one of the largest and powerful infrastructures in the world
- Gmail, YouTube, Maps
- In 2008 the infrastructure is opened to business customers launching **Google Cloud Platform** (**GCP**) -> **https://console.cloud.google.com**
- Services comprise:
    - **Compute**: virtual machines, containers, app engines
    - **Storage** and **Database**: Cloud Storage, Cloud SQL
    - **Networking**: Virtual Private Clouds
    - **Big Data**: DataProc, Hadoop, BigQuery
    - **AI** and **Machine Learning**: TensorFlow, AutoML, BQML
    - **IoT**: IoT Core

Google Cloud Platform



Google Cloud Platform: regions

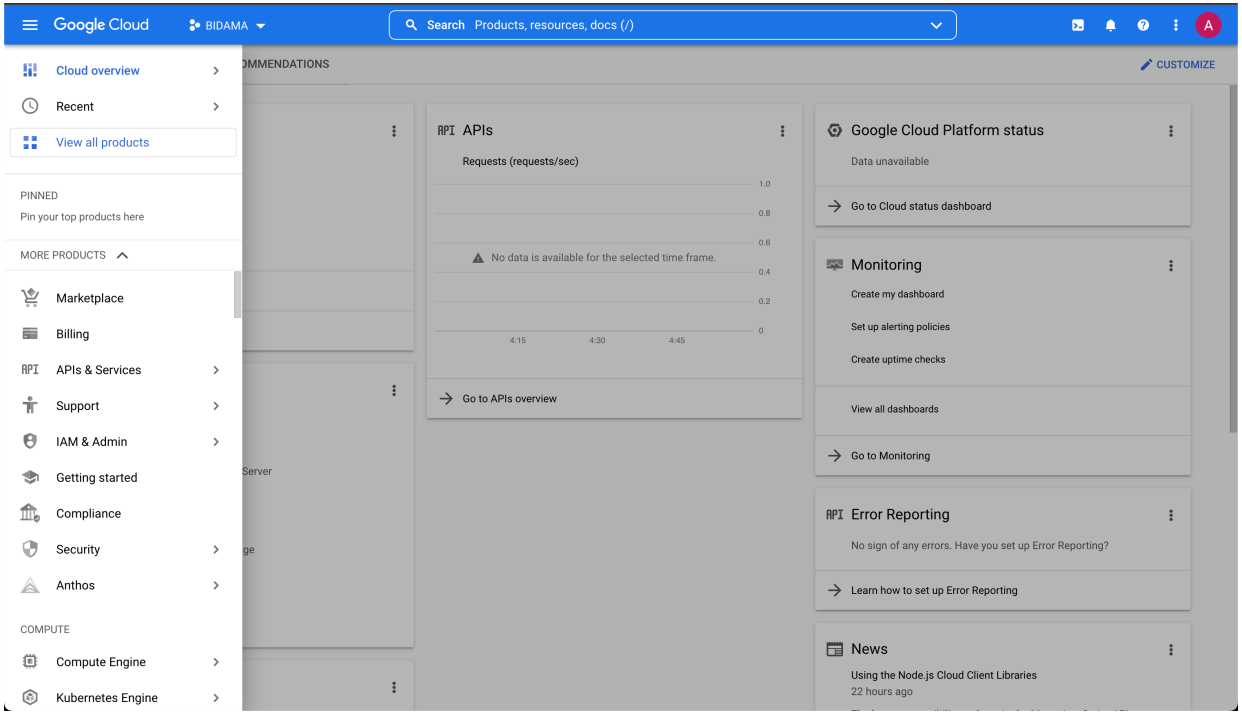
Google Cloud Platform: network



Google Cloud Platform: advantages

- **Fully managed** and **serverless** services: no maintenance
- Leading research in the AI and ML
- **Google Cloud Console** is a web-based interface accessible from compatible web browser

Google Cloud Platform: advantages

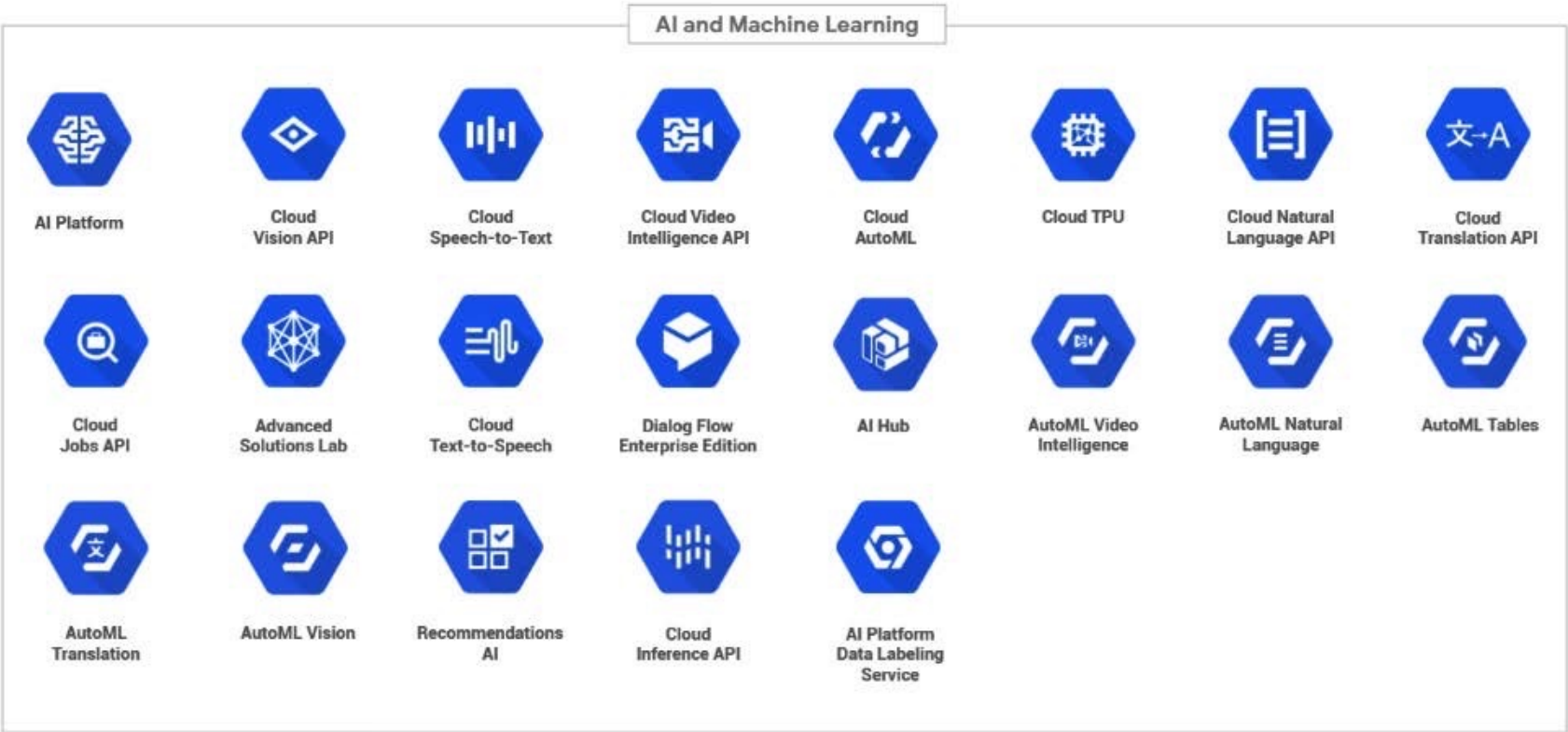


Google Cloud Platform: AI and ML

- AI and ML features are embedded within many Google product
  - **Google Maps**: predict arrival time
  - **Google Translate**
  - **YouTube**: recommend video to watch
  - **Google Photos**: recognize people, object, places

- GCP provides **services** to address all the steps in a typical **ML model**:
  - **Ingestion** and **preparation** of data
  - **Building** and **training** the model
  - **Evaluation** and **validation**
  - **Deployment** and **maintenance**

Google Cloud Platform: AI and ML



Google Cloud Platform: AI and ML

- **AI** and **ML services** can bedivided into **3 categories**
    - **Core platform**: Infrastructure-as-a-Service (IaaS) approach to provide different processing units (CPU, GPU, TPU), Deep Learning VM Image, AI Platform and AI Platform notebooks
    - **AI building blocks**: AutoML, BigQuery ML
    - **Solution**: AI Hub (marketplace for AI components), Document AI (extracting relevant informations from different type of documents)

Agenda

- Google Cloud Platform
- **Google BigQuery**
- Google BigQuery SQL
- Google BigQuery ML

BigQuery

- Google BigQuery is a **highly scalable**, **serverless**, **distributed** data warehouse technology
- It can **store** petabytes of **data** and **query** them with high performance
  - Giga:10^9 Tera: 10^{12} Peta: 10^{15}
- Being **serverless**, users who store and query data on BigQuery don't have to **manage** the underlying **infrastructure**
- BigQuery has a **distributed architecture** running on thousands of nodes across Google's data centers
  - datasets are chunked and replicated across different regions to guarantee **maximum performance** and **availability**
- The **storage** and **compute** layers are fully **decoupled** in BigQuery

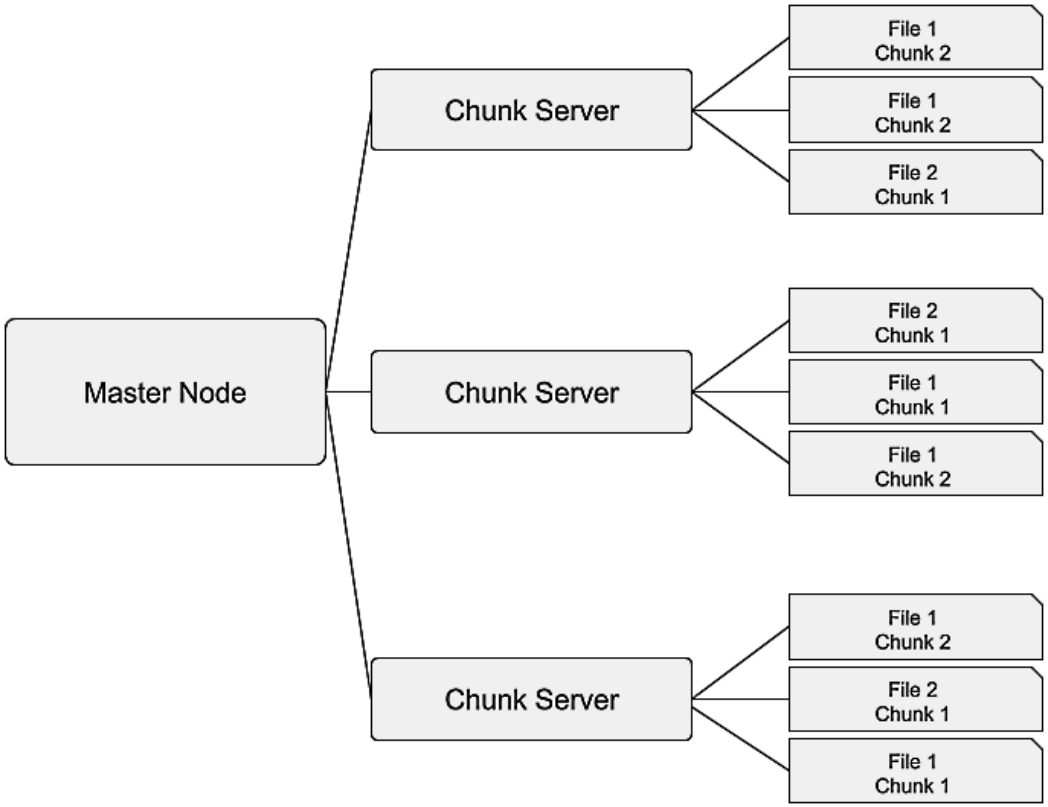
BigQuery: storage

- BigQuery stores data in columnar format rather than in row format
- **Data** is stored in Google's proprietary distributed filesystem named **Google File System** (codename Colossus)
- Google File System is based on two different server types

  - **Master servers**: Nodes that don't store data but are responsible for managing the metadata of each file, such as the location and available number of replicas of each chunk that compose a file

  - **Chunk servers**: Nodes that actually store the chunks of files that are replicated across different servers

BigQuery: storage



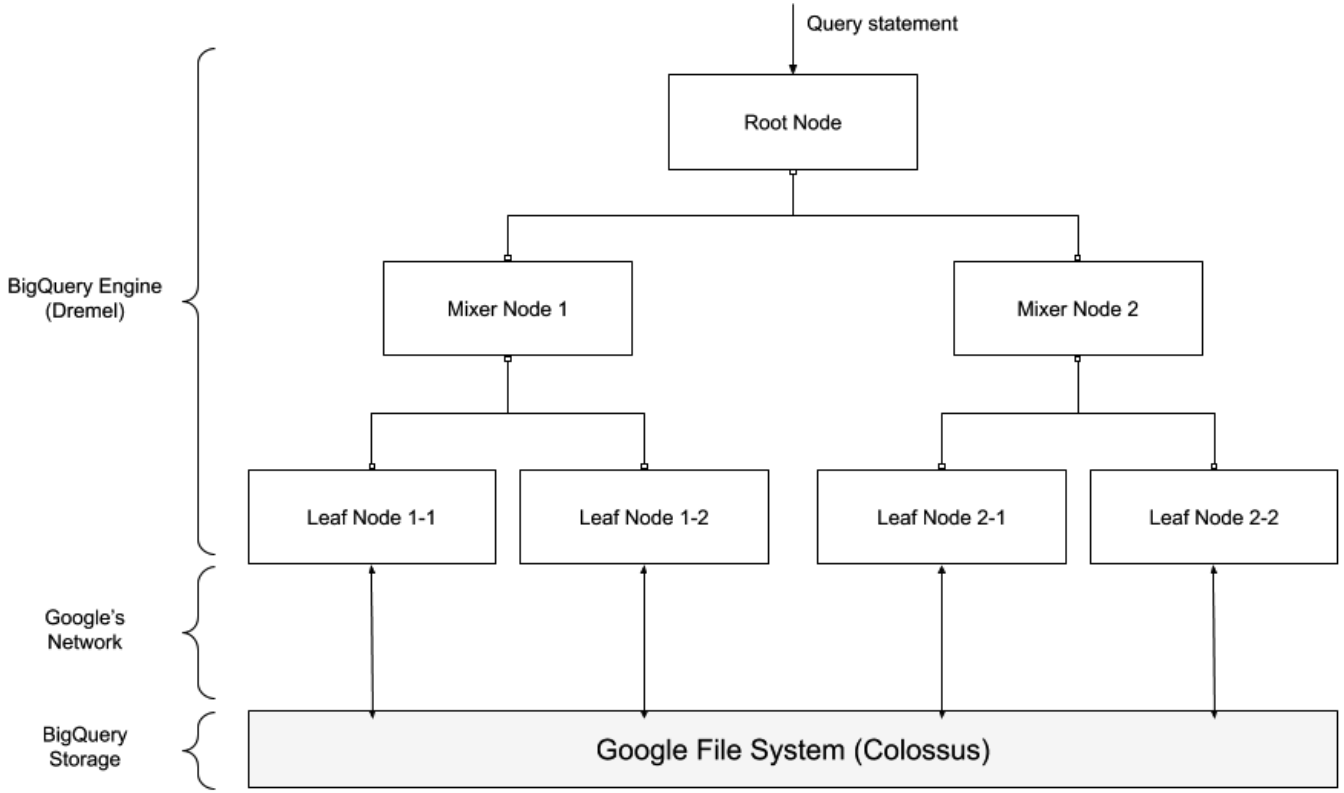
BigQuery: compute

- The **compute layer** is responsible for **receiving query** statements from BigQuery users and **executing** them in the fastest way
- The **query engine** is based on Dremel, a technology developed by Google, that leverages a **multi-level tree architecture**

BigQuery: compute

1. The **root node** of the tree **receives** the **query** to execute
2. The root node **splits** and **distributes** the query to **intermediate nodes** (mixers)
3. Mixer nodes rewrite queries before **passing** them to the **leaf nodes** or to other mixer nodes
4. **Leaf nodes** are responsible for parallelizing the **reading** of the chunks of **data** from Google File System
5. Leaf nodes **perform computations** on the data and eventually shuffle them across other leaf nodes
6. At the end of the computation, each leaf node **produces** a **result** that is returned to the parent node
7. When all the **results** are **returned** to the **root node**, the outcome of the query is sent to the user or application that requested the execution

BigQuery: compute

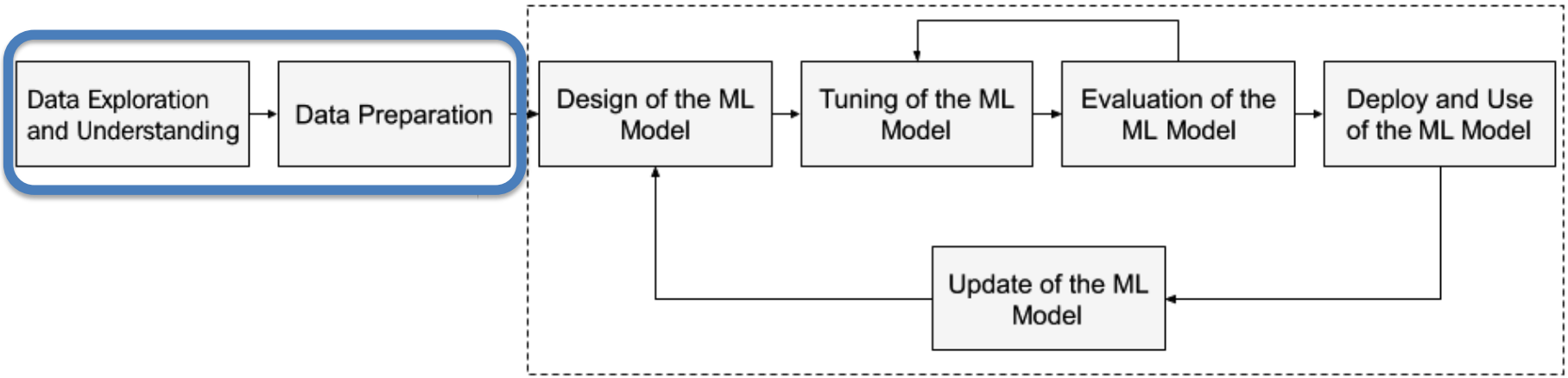


BigQuery structure

- BigQuery structures (tables, views and ML models) are organized in **datasets**
- Each dataset is a **container** for different **structures** and can be used to control access to underlying data structures
- A **dataset** is directly linked to the following:
  - A **GCP project** that hosts the dataset and is linked to the billing account
  - A geographic **location**
  - A **name** assigned to the dataset that should be unique in the GCP project

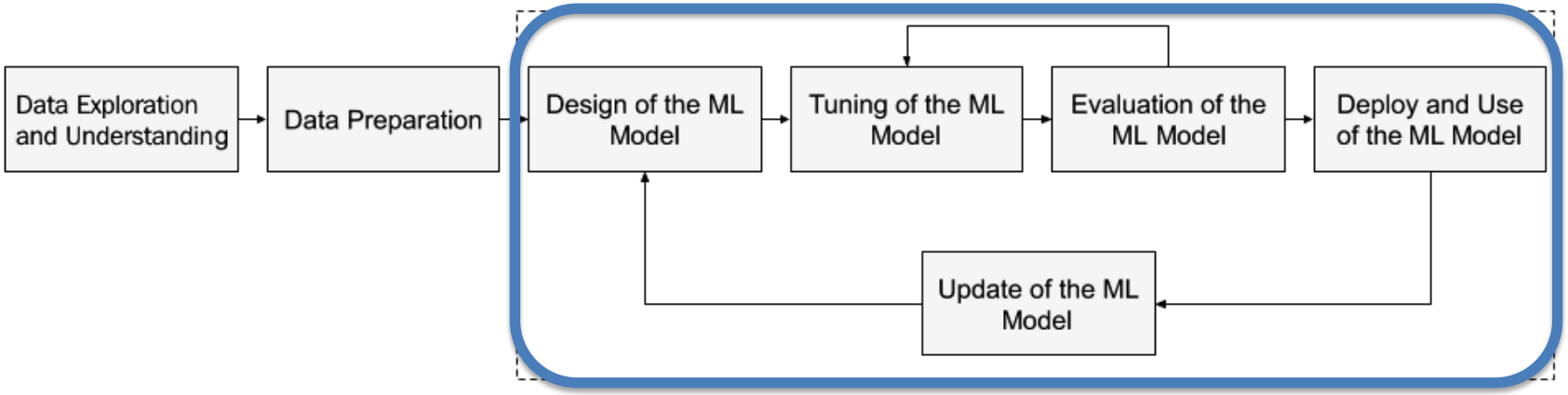
ML model development

- The first two steps involve preliminary raw data analyses and operations:
  - **Data Exploration and Understanding**: understand the meaning of all the columns in the dataset and select the fields to take into consideration
  - **Data Preparation**: filter, aggregate and clean up the dataset -> ready to use for the training phase



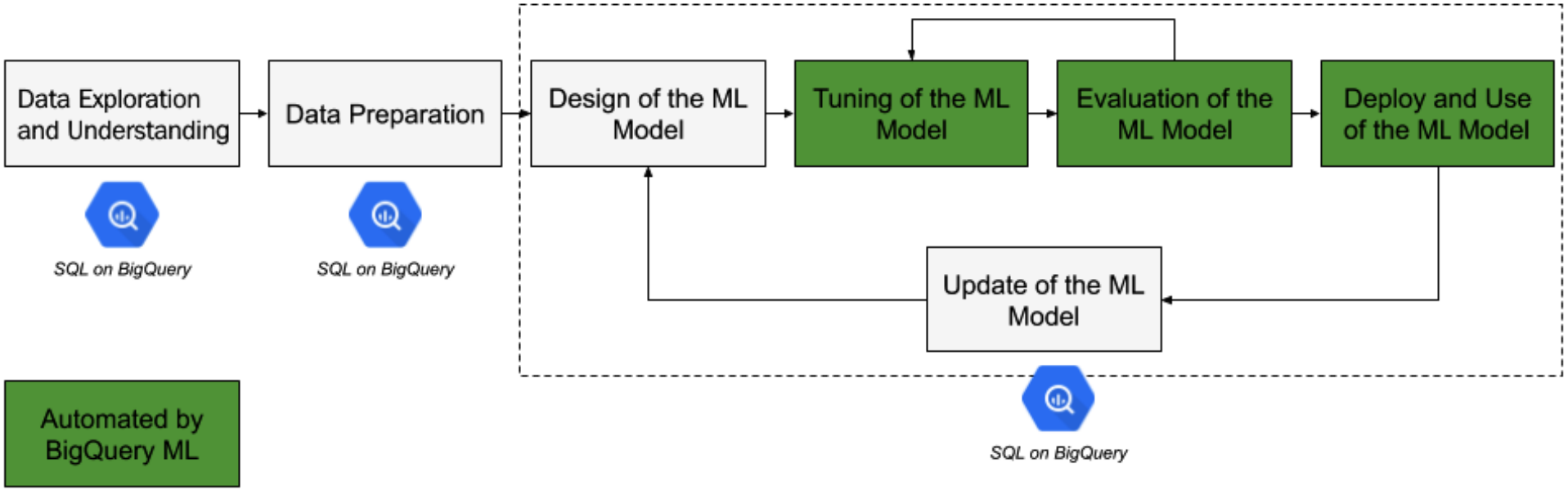
ML model development

- Developing ML model
  - **Experimenting** with different algorithms on the training dataset
  - Parameter **tuning** to get better performance out of the ML model
  - **Evaluating** the model on the test dataset (different from training)
  - **Deploying** and eventually **updating** the ML model



BigQuery ML

- BigQuery ML simplifies, accelerates and automates most of the activities involved in the development of a ML model

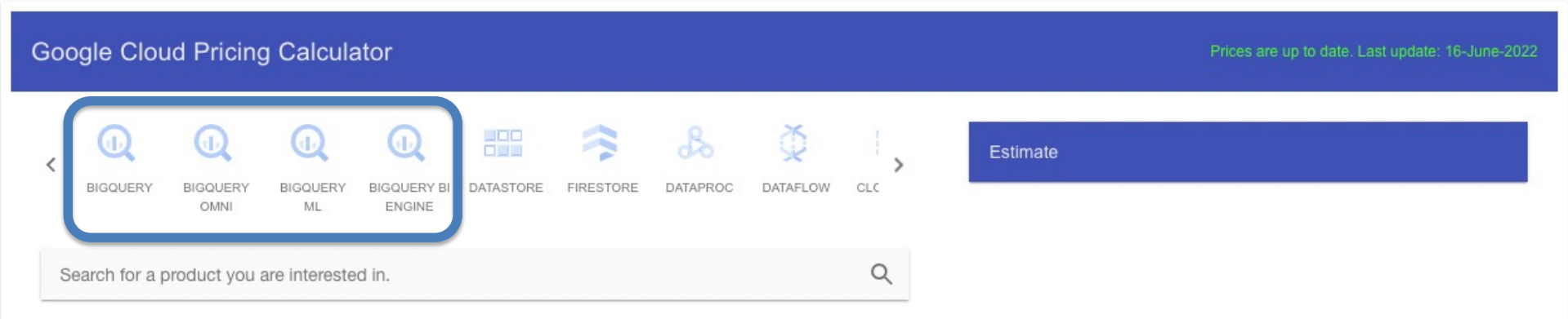


BigQuery ML: algorithms

- **Linear regression**: forecast numerical values with a linear model
- **Binary logistic regression**: binary classification (Yes or No, 1 or 0, True or False)
- **Multiclass logistic regression**: classification with multiple options
- **Matrix factorization**: for developing recommendation systems
- **Time series**: forecast leveraging timeseries data from the past
- **Boosted tree**: classification and regression
- **AutoML table**: AutoML capabilities from the BigQuery SQL interface
- **Deep Neural Network**: classification or regression
- **K-means clustering**: data segmentation of similar objects

BigQuery: pricing

- BigQuery operations have a cost
  - Storage
  - Compute
  - Training
  - Evaluation



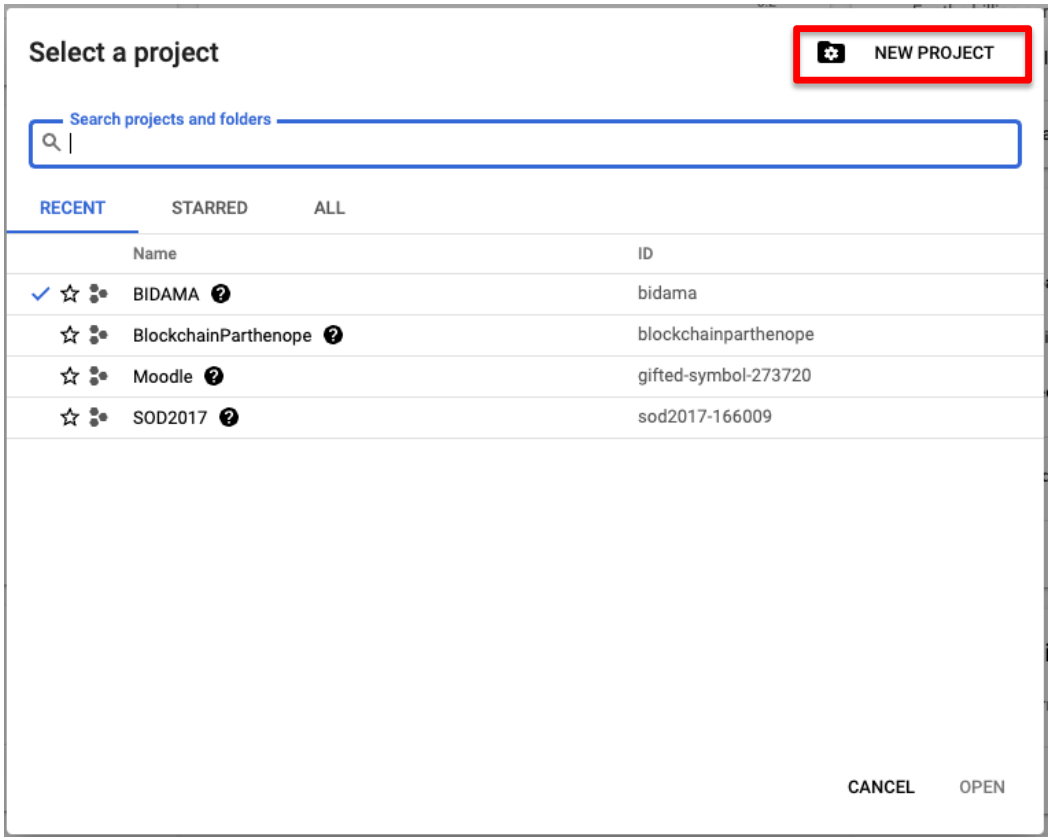
BEFORE CONTINUING

CREATE A GOOGLE CLOUD ACCOUNT

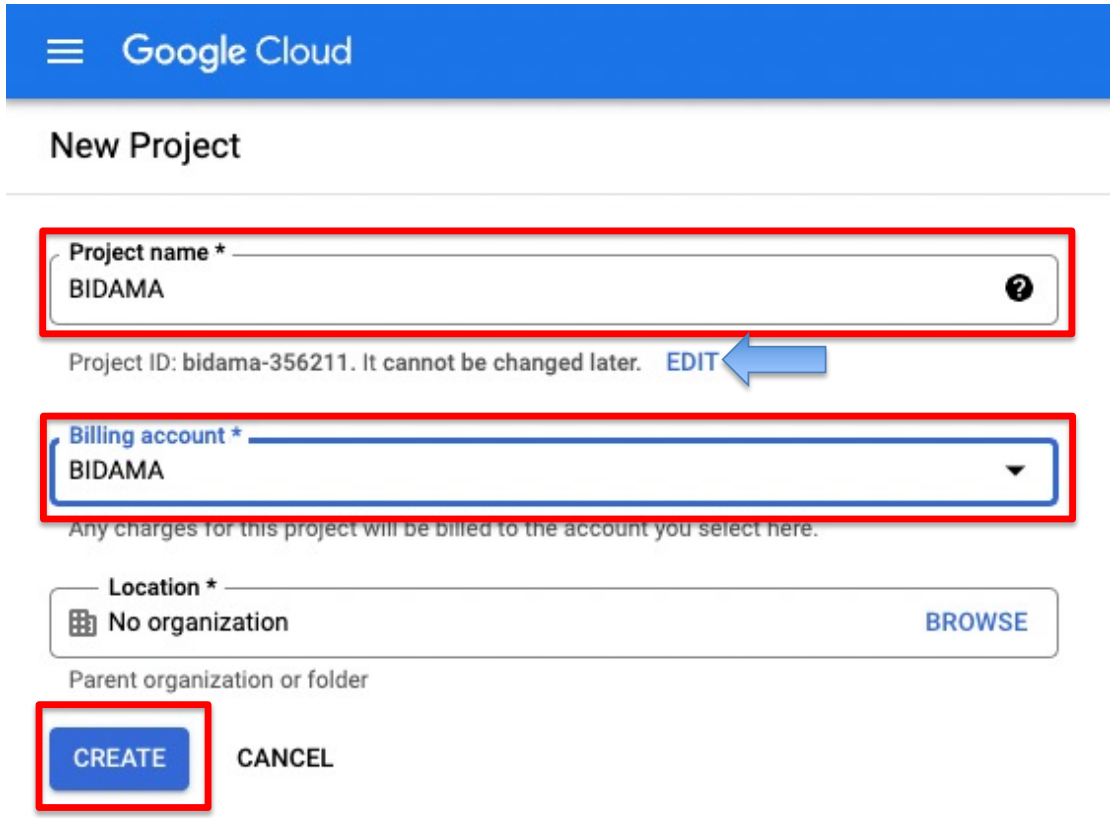
https://console.cloud.google.com

Redeem Google Cloud Coupon

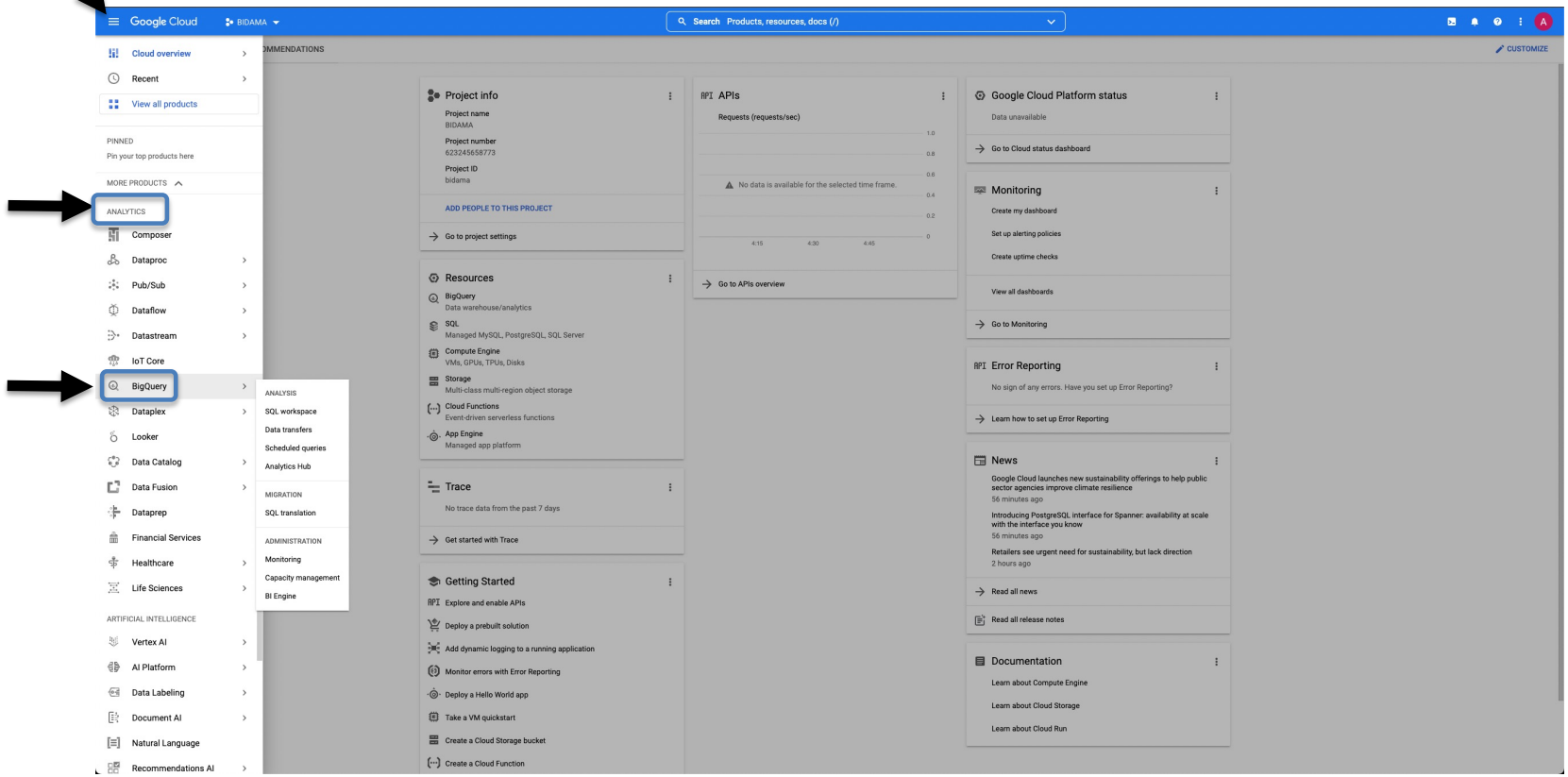
New Project


Select a project
NEW PROJECT
Search projects and folders
RECENT
STARRED
ALL
Name
ID
BIDAMA
bidama
BlockchainParthenope
blockchainparthenope
Moodle
gifted-symbol-273720
SOD2017
sod2017-166009
CANCEL
OPEN

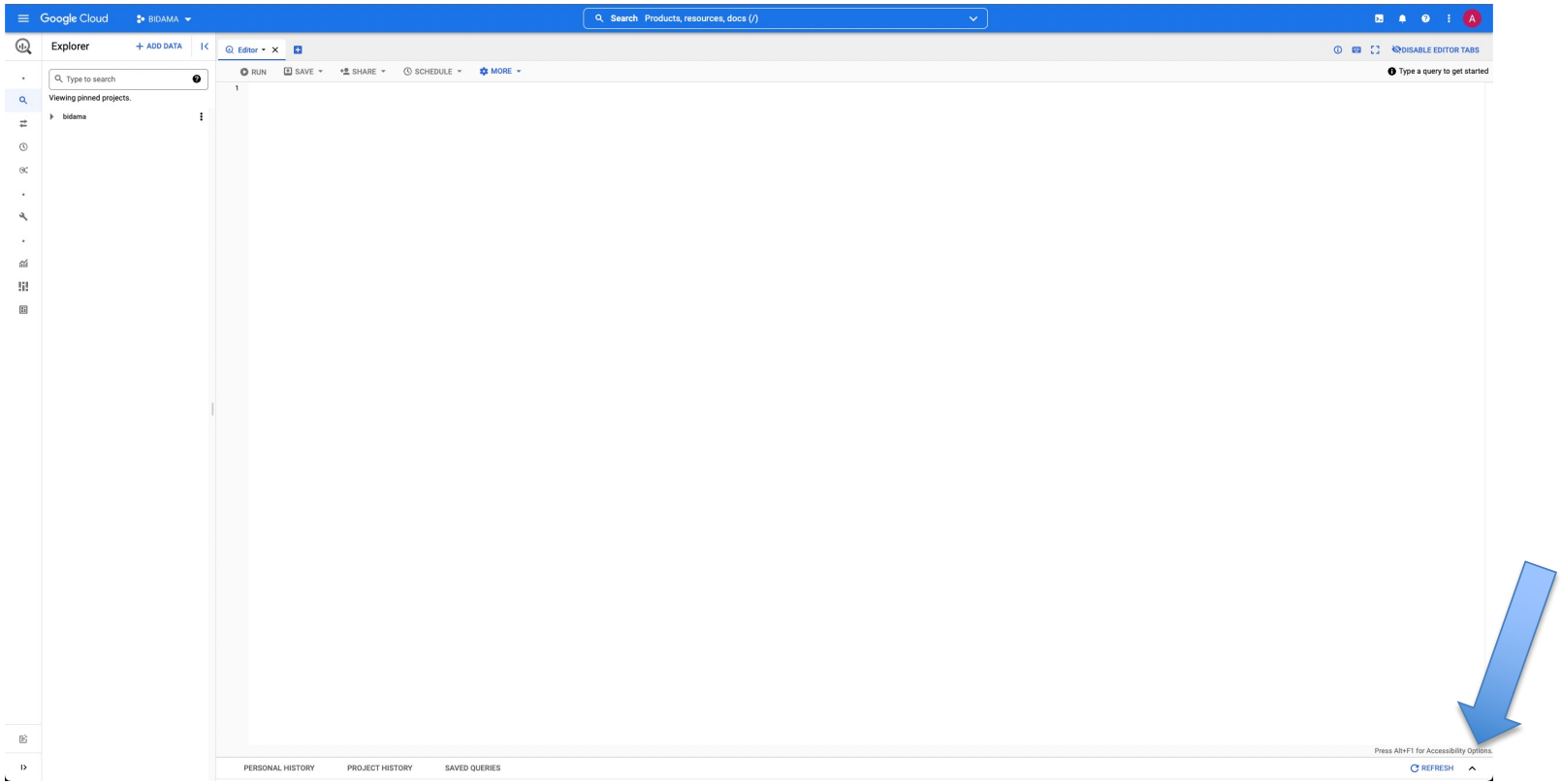
New Project


Google Cloud
New Project
Project name *
BIDAMA
Project ID: bidama-356211. It cannot be changed later.
EDIT
Billing account *
BIDAMA
Any charges for this project will be billed to the account you select here.
Location *
No organization
BROWSE
Parent organization or folder
CREATE
CANCEL

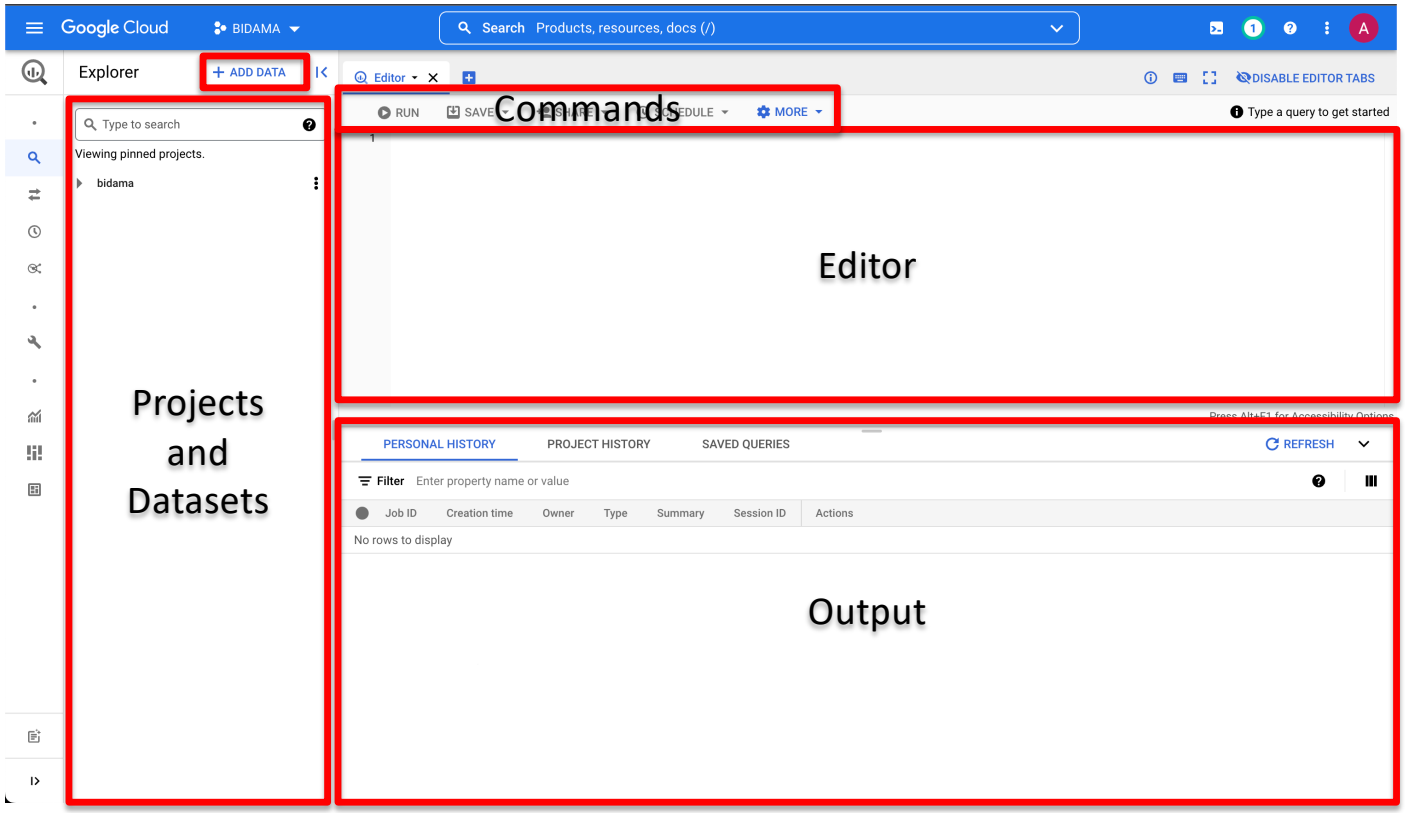
Interacting with BigQuery



Interacting with BigQuery



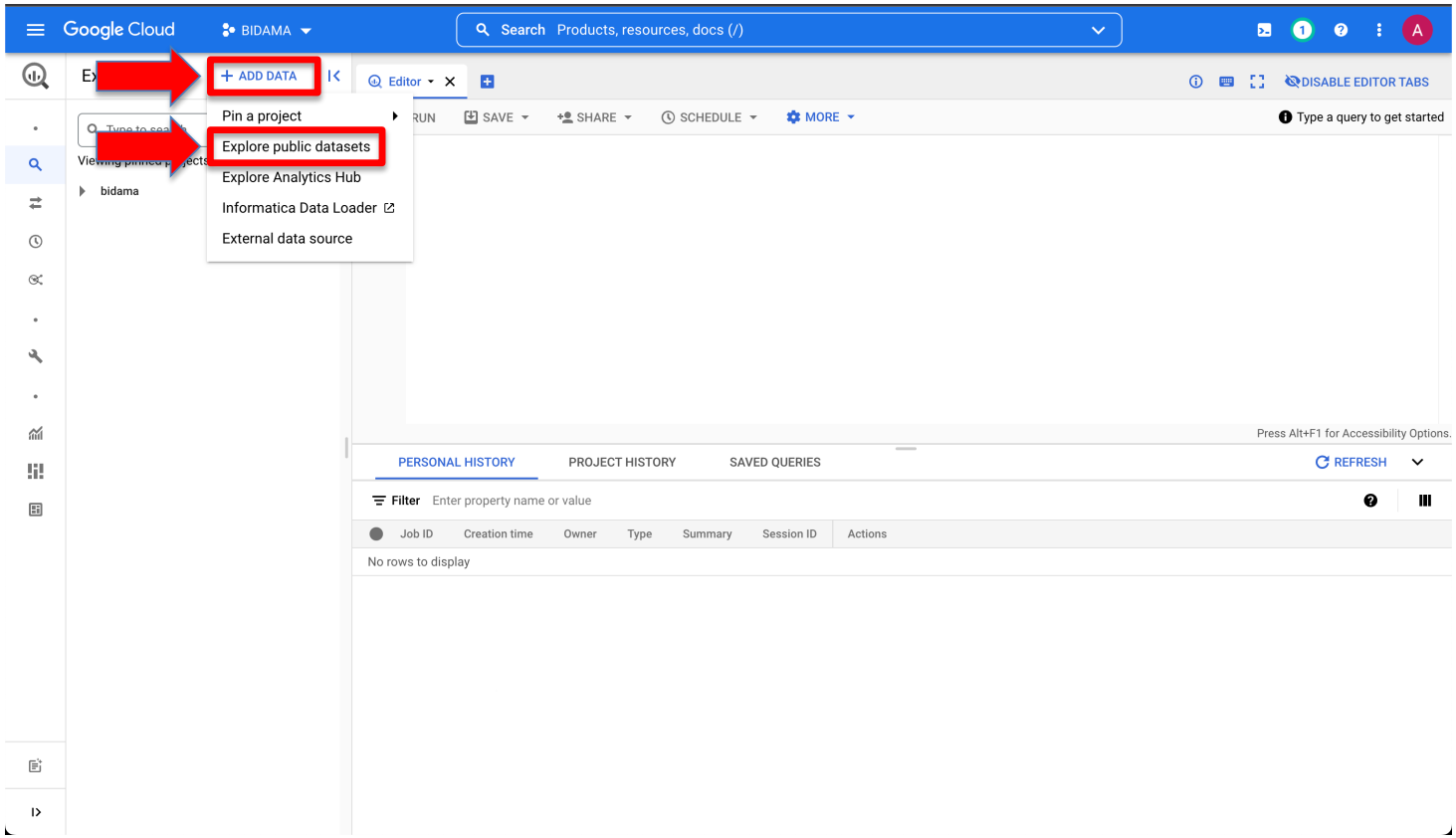
Interacting with BigQuery



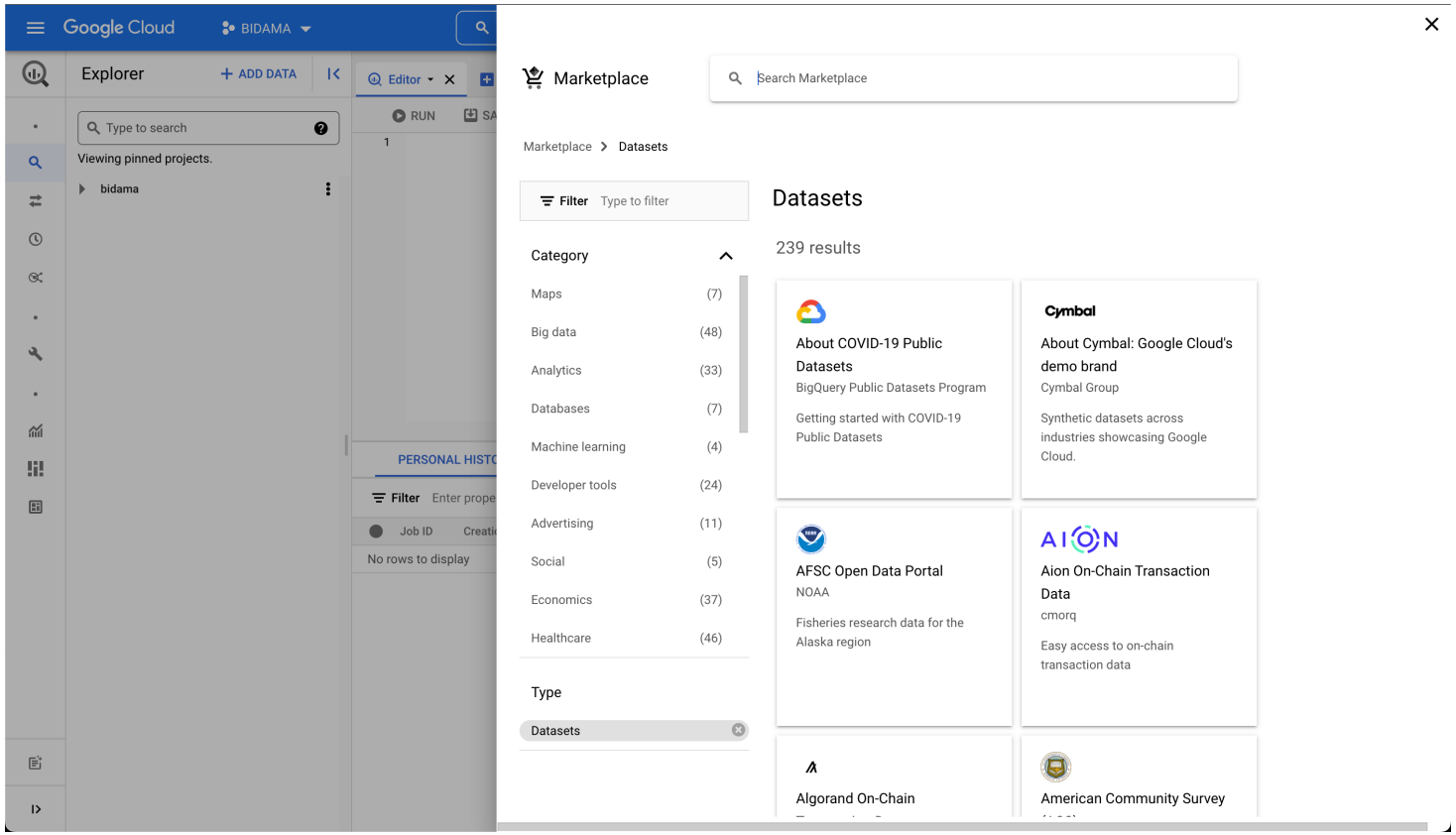
BigQuery: datasets

- (Big) Data is fundamental to exploit machine learning algorithms
- Collecting data and build large datasets is one of the most time-consuming and tedious task in the data management field
- **Cloud Public Datasets Program** allows to use data already collected and ingested into BigQuery
- The BigQuery public datasets are available in the Datasets section of the Google Cloud Marketplace

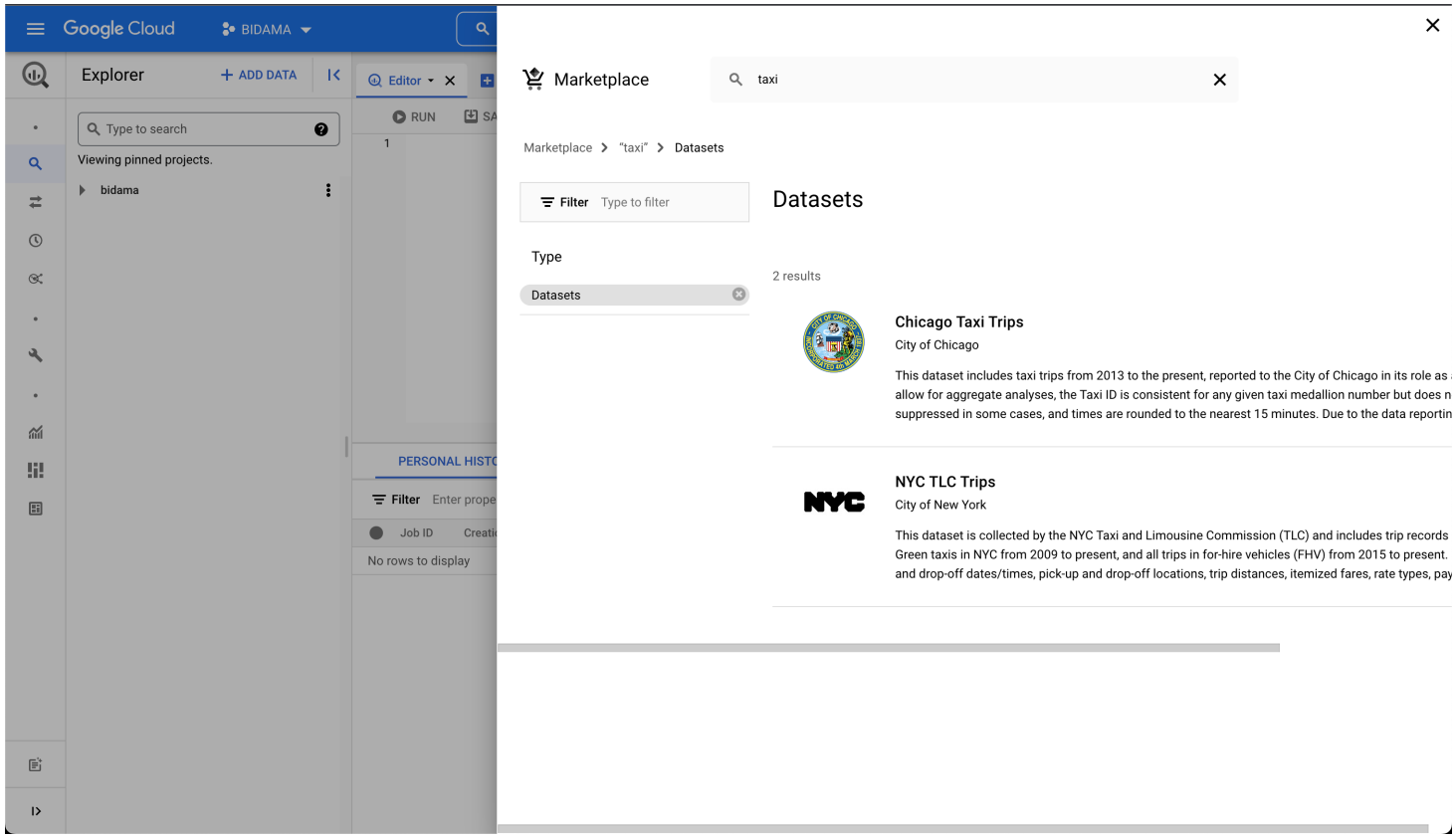
BigQuery: datasets



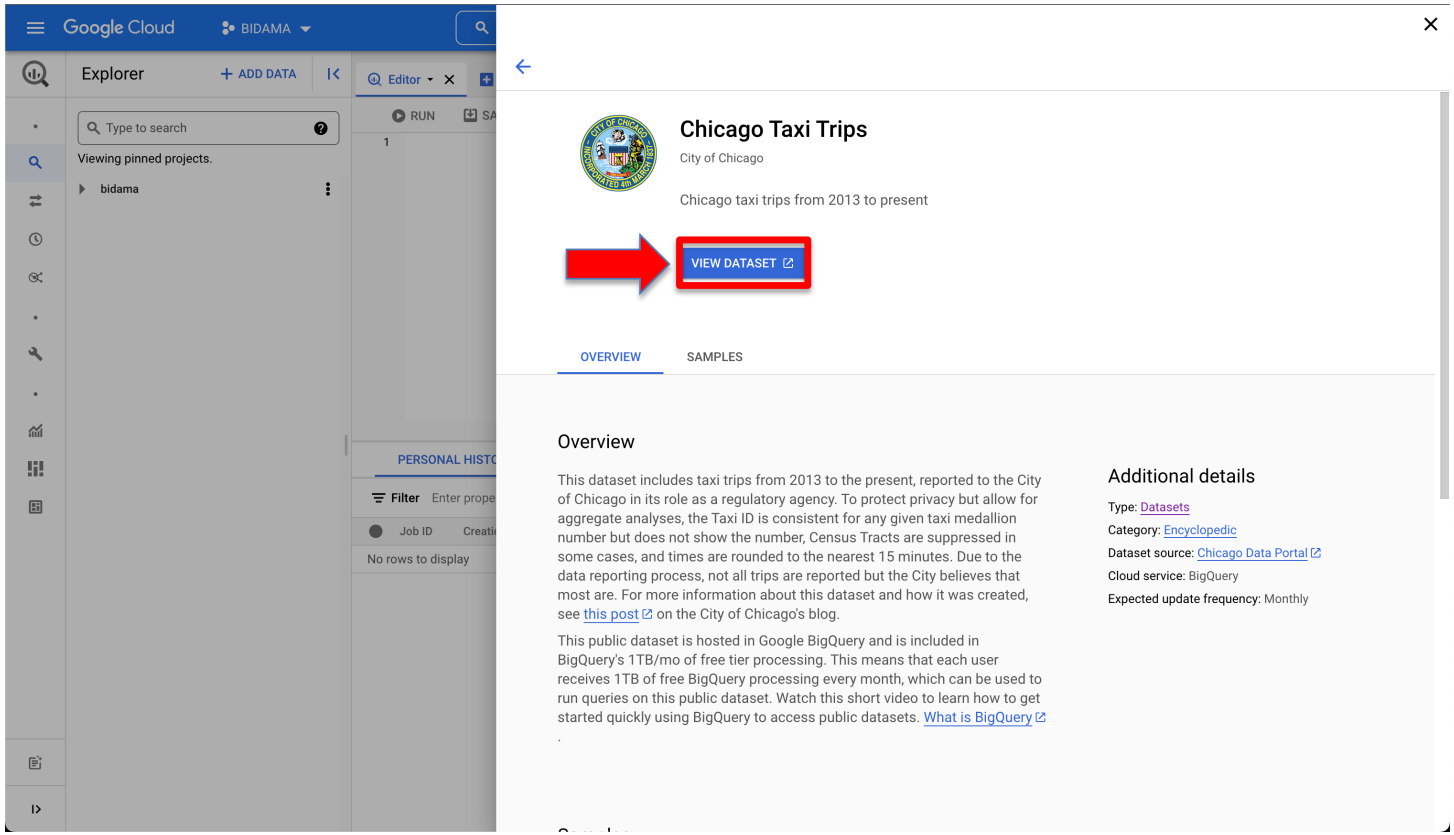
BigQuery: datasets



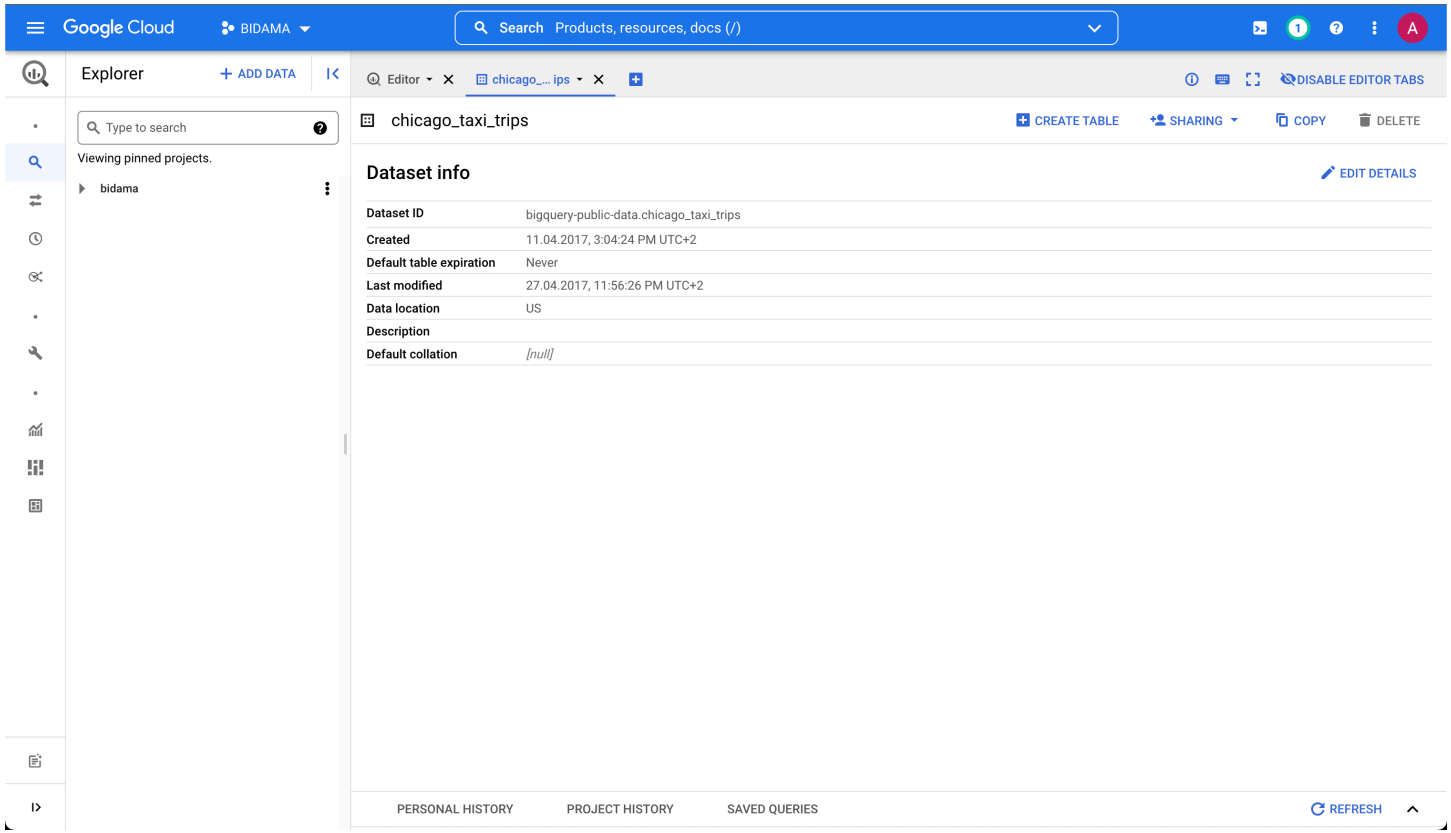
BigQuery: datasets



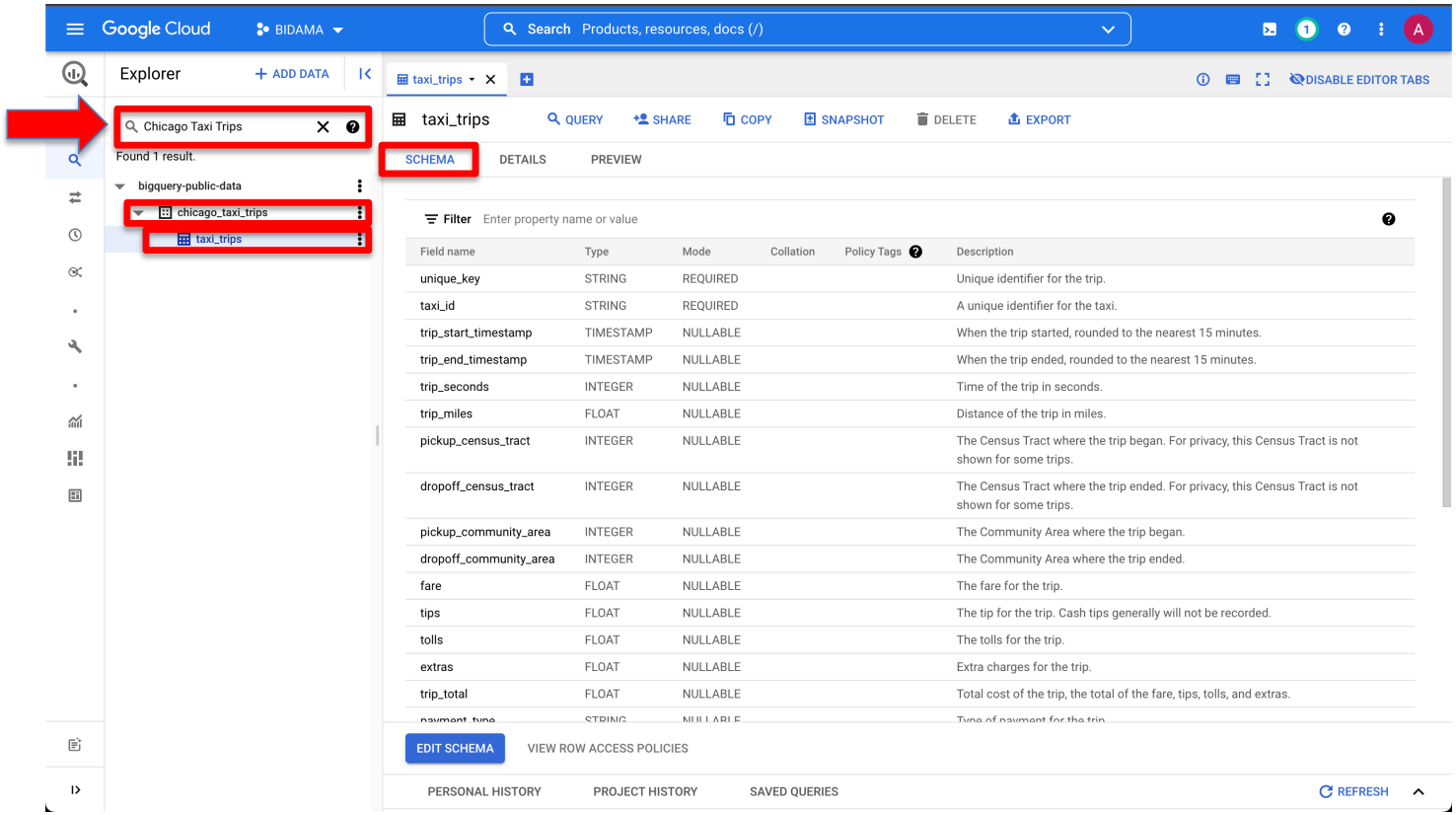
BigQuery: datasets



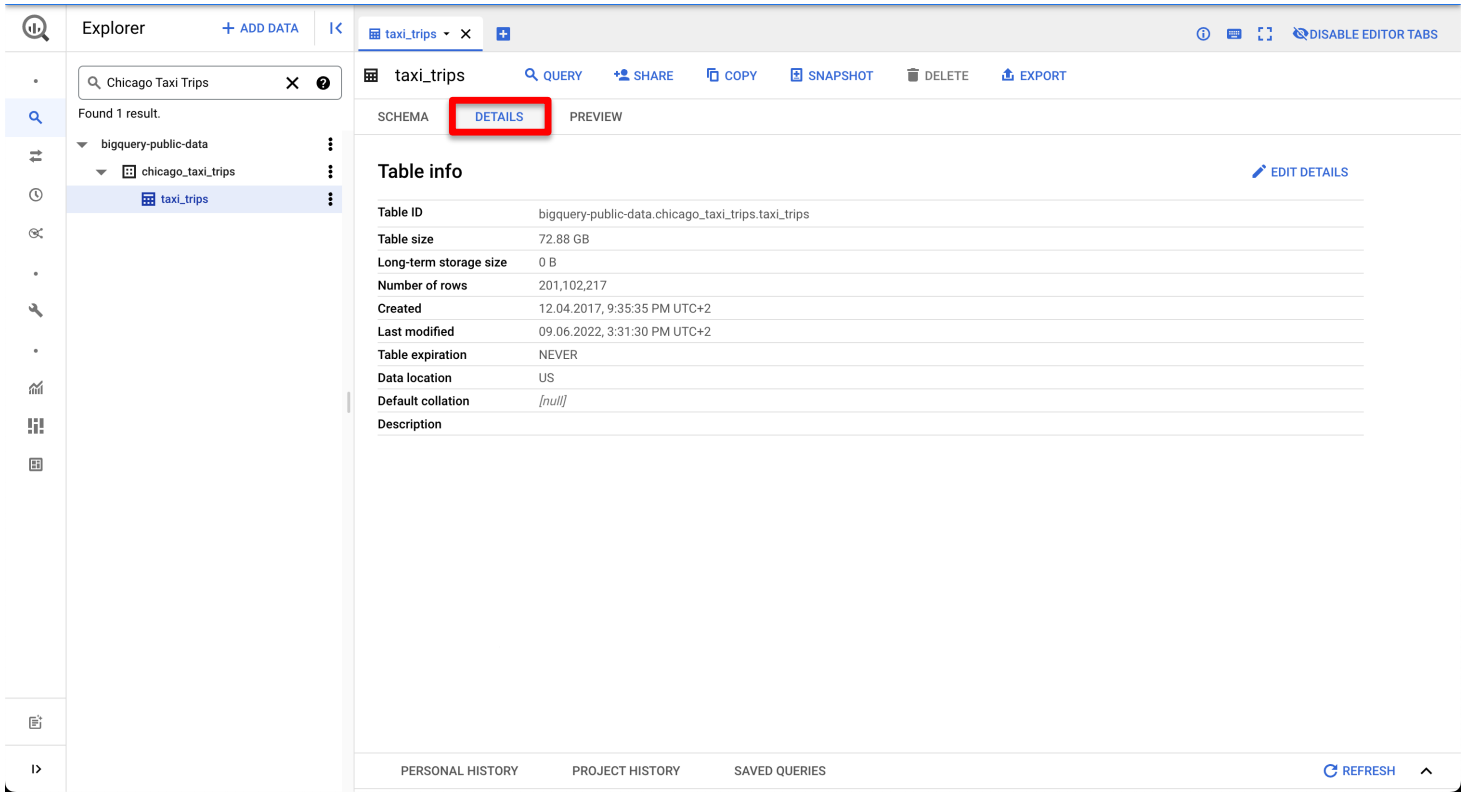
BigQuery: datasets



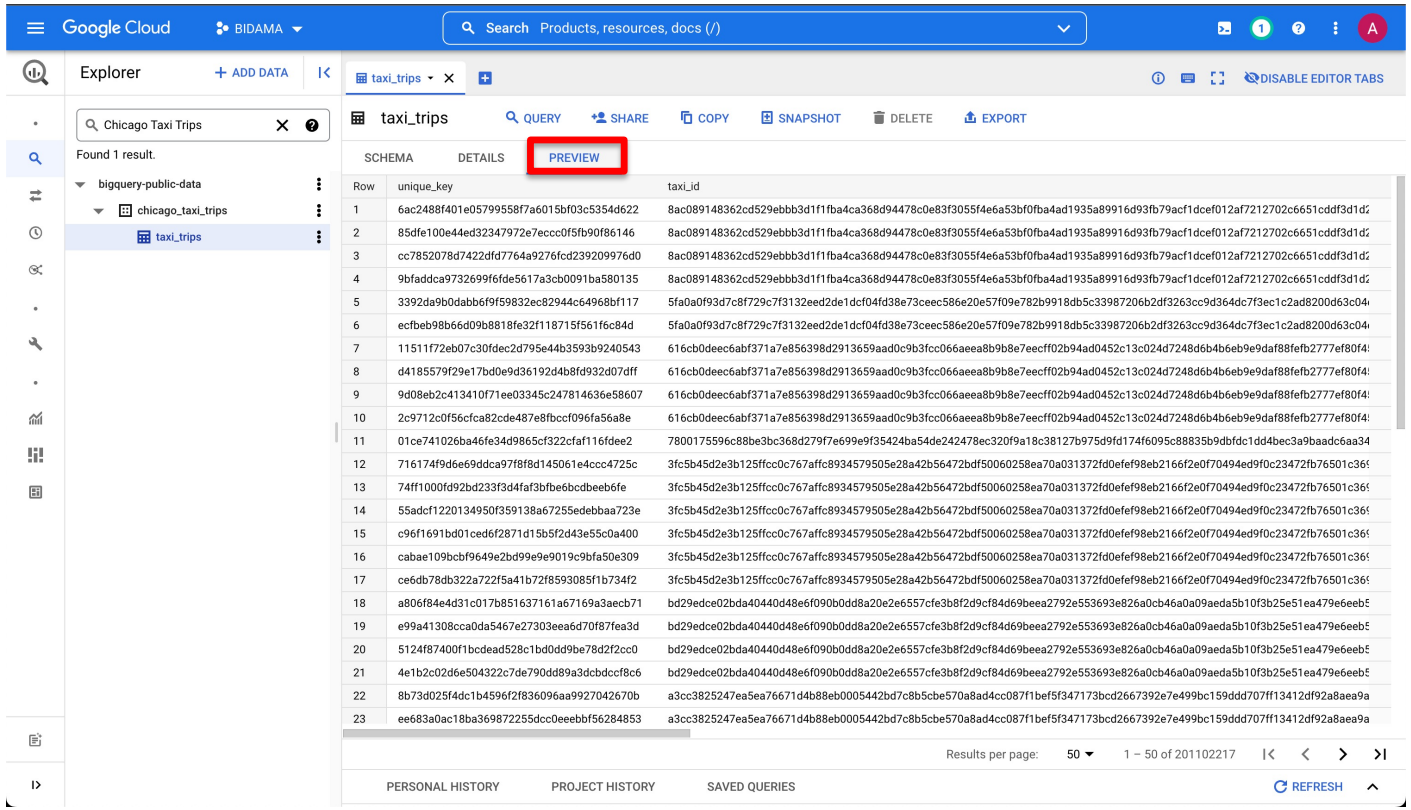
BigQuery: datasets



BigQuery: datasets



BigQuery: datasets



Agenda

- Google Cloud Platform
- Google BigQuery
- **Google BigQuery SQL**
- Google BigQuery ML

BigQuery: syntax

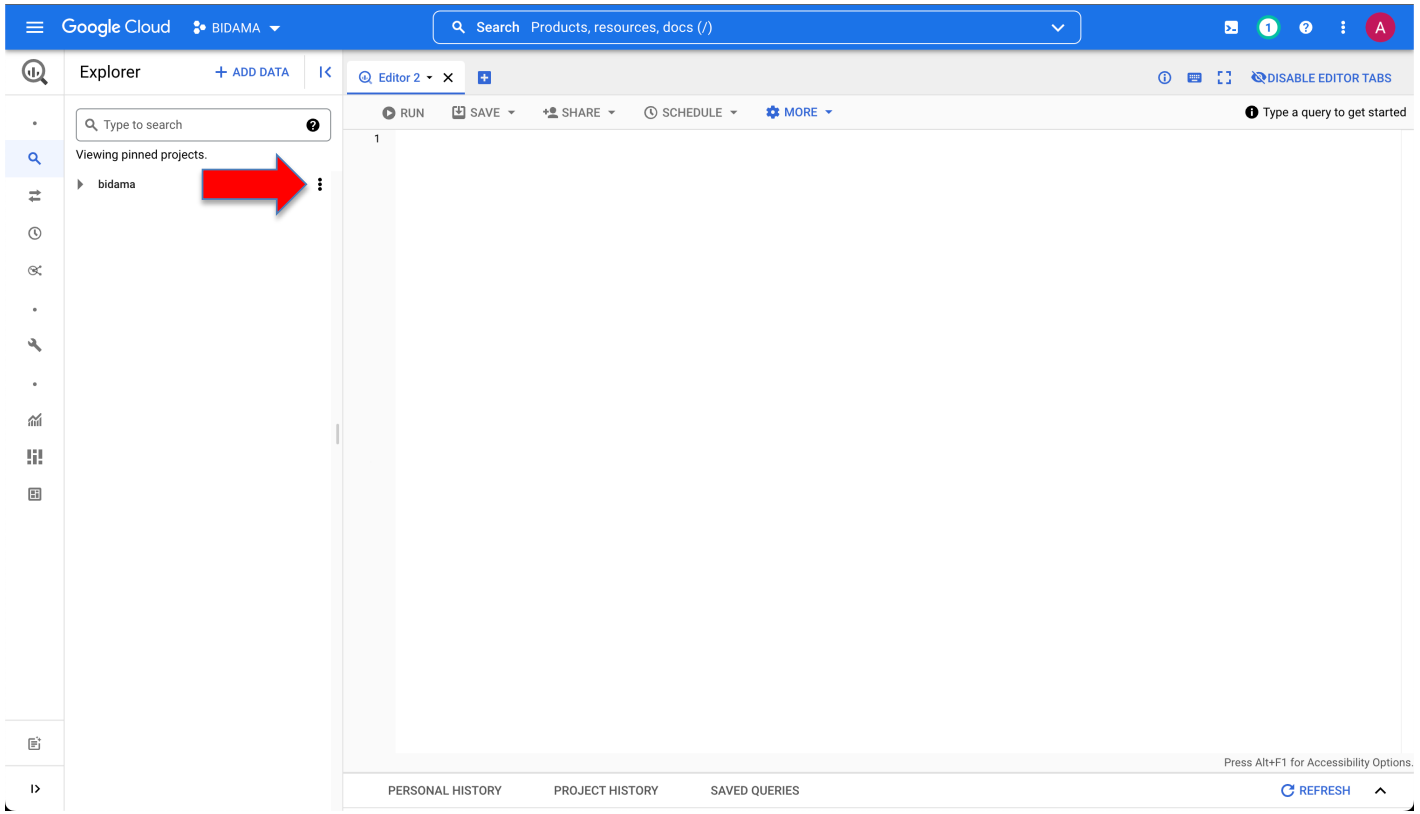
- **BigQuery** is based on **SQL** with extensions which allow to use Machine Learning features

- In particular the process will go through
  - BigQuery dataset
  - BigQuery SQL
  - BigQuery ML

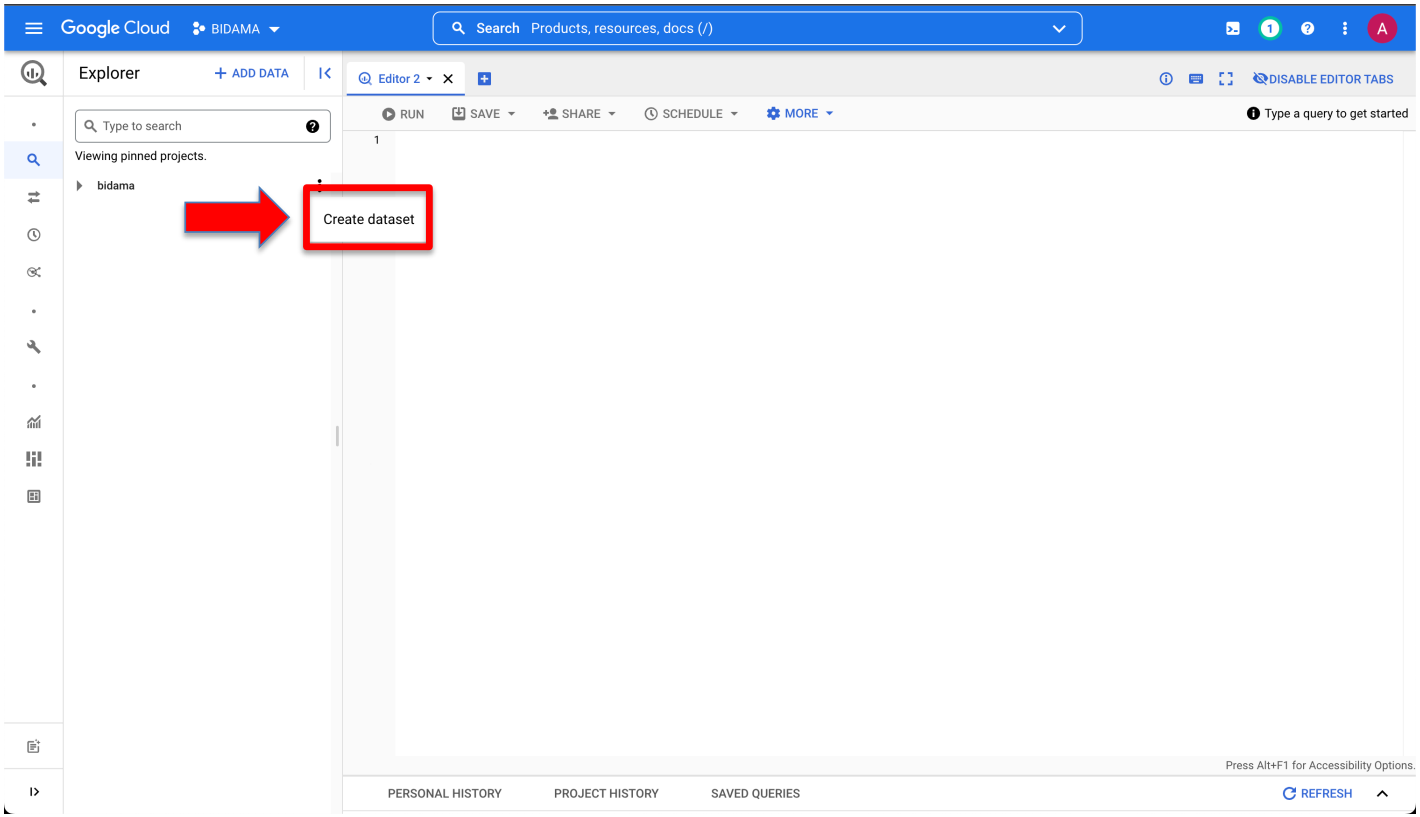
Creating BigQuery dataset

- First of all it is necessary to create a BigQuery dataset
- Select **BigQuery service** in GCP
- **Create Dataset**

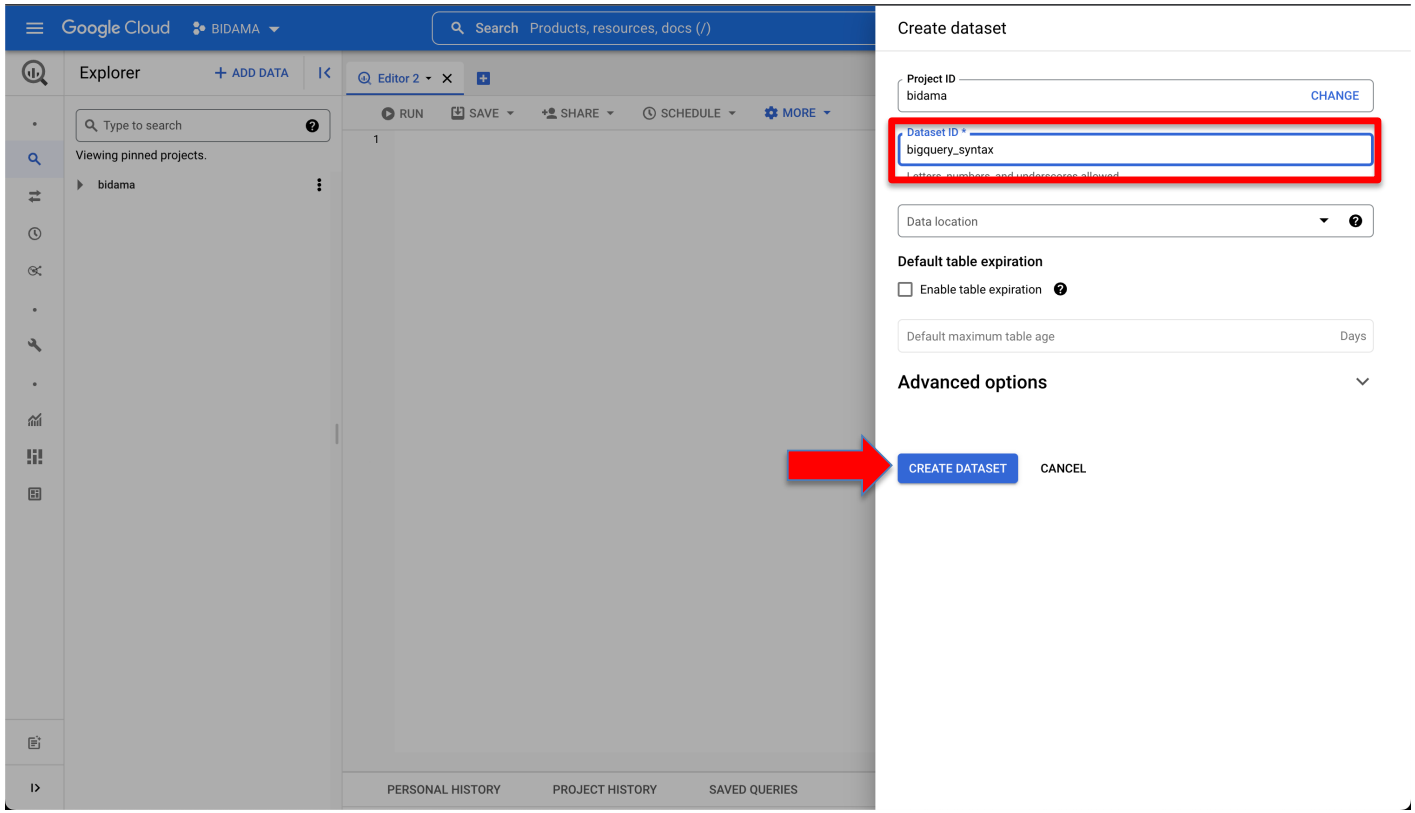
Creating BigQuery dataset



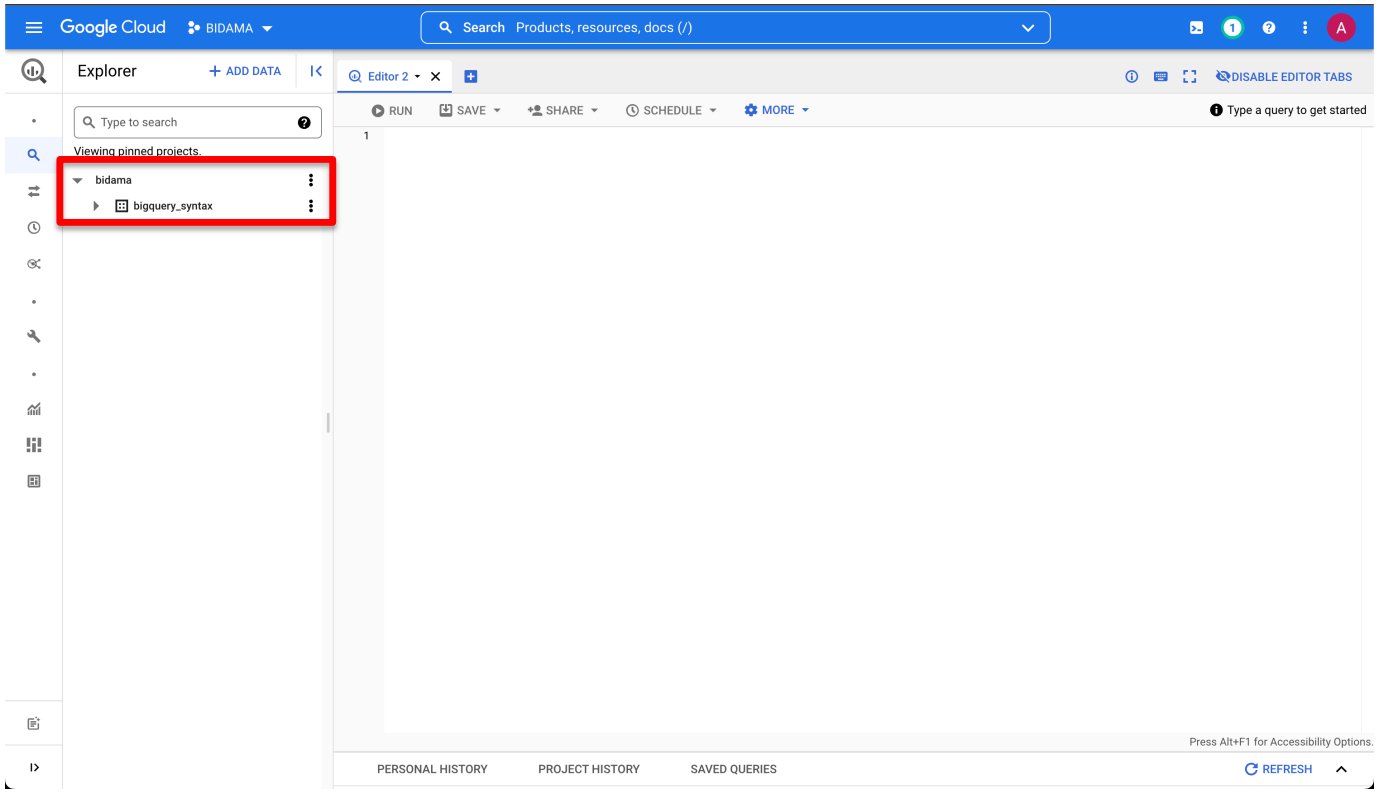
Creating BigQuery dataset



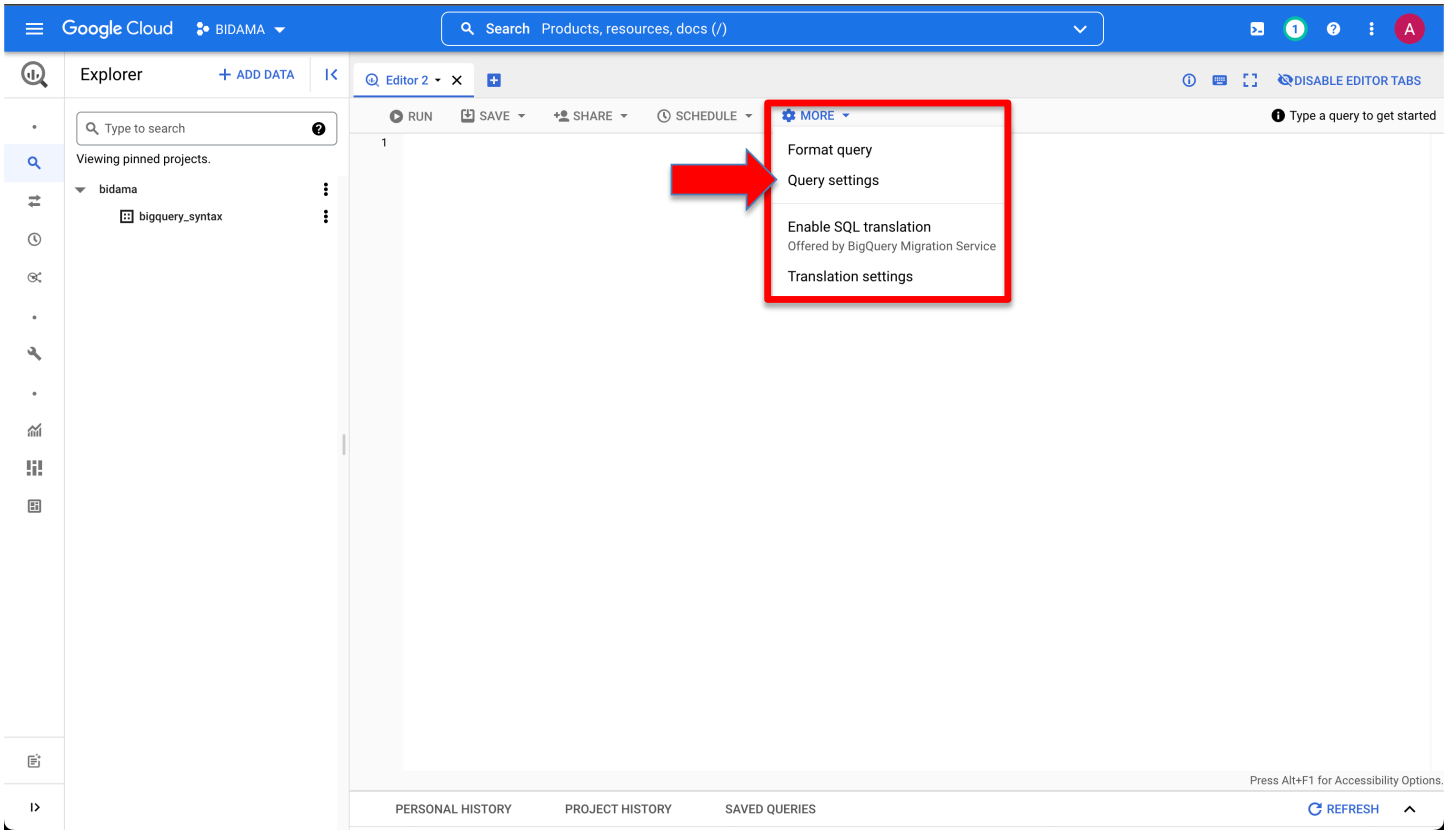
Creating BigQuery dataset



Creating BigQuery dataset



BigQuery SQL



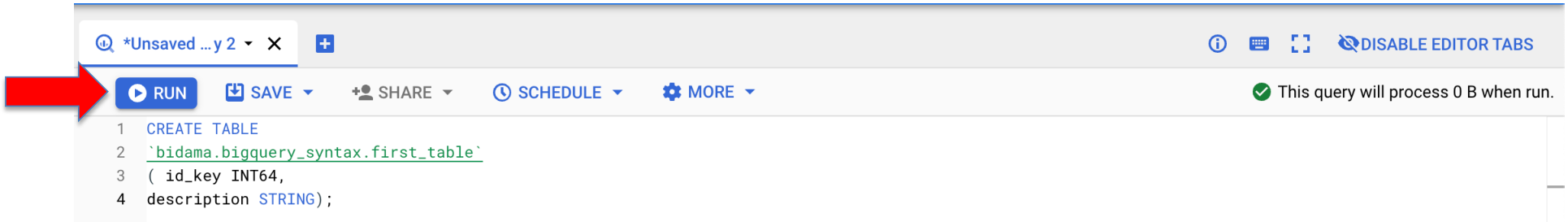
BigQuery SQL: CRUD

- **CRUD** stands for
  - Create
  - Read
  - Update
  - Delete

- That are the same basic operations for handling BigQuery objects

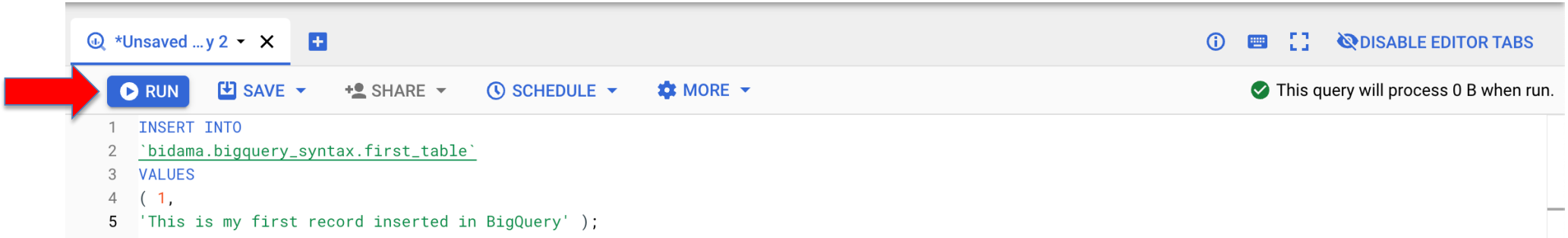
BigQuery SQL: CREATE

- The **CREATE** statement is used to create objects or to insert new items into an existing table
- The first two words of the query statement, **CREATE TABLE**, are used to start the creation of a new table
- Next, the id of the object is composed by the **project name**, the **dataset name** and the **table name** concatenated with the **.** symbol and enclosed by the backtick character `
- Last, the list of fields with their data type separated by the comma character



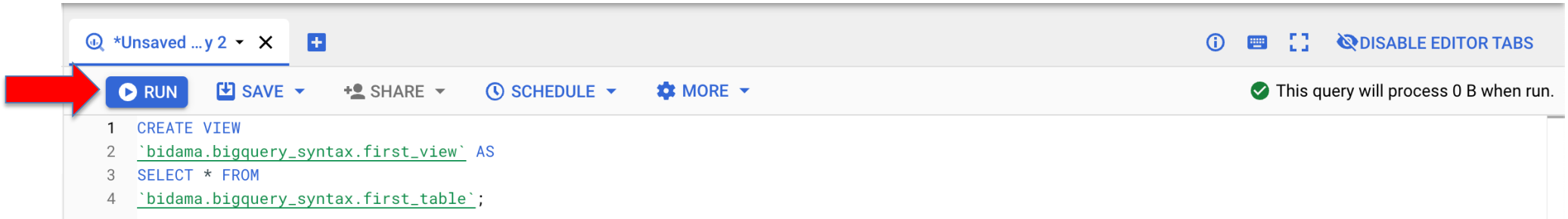
BigQuery SQL: INSERT

- The **INSERT** statement is used to insert values into the table



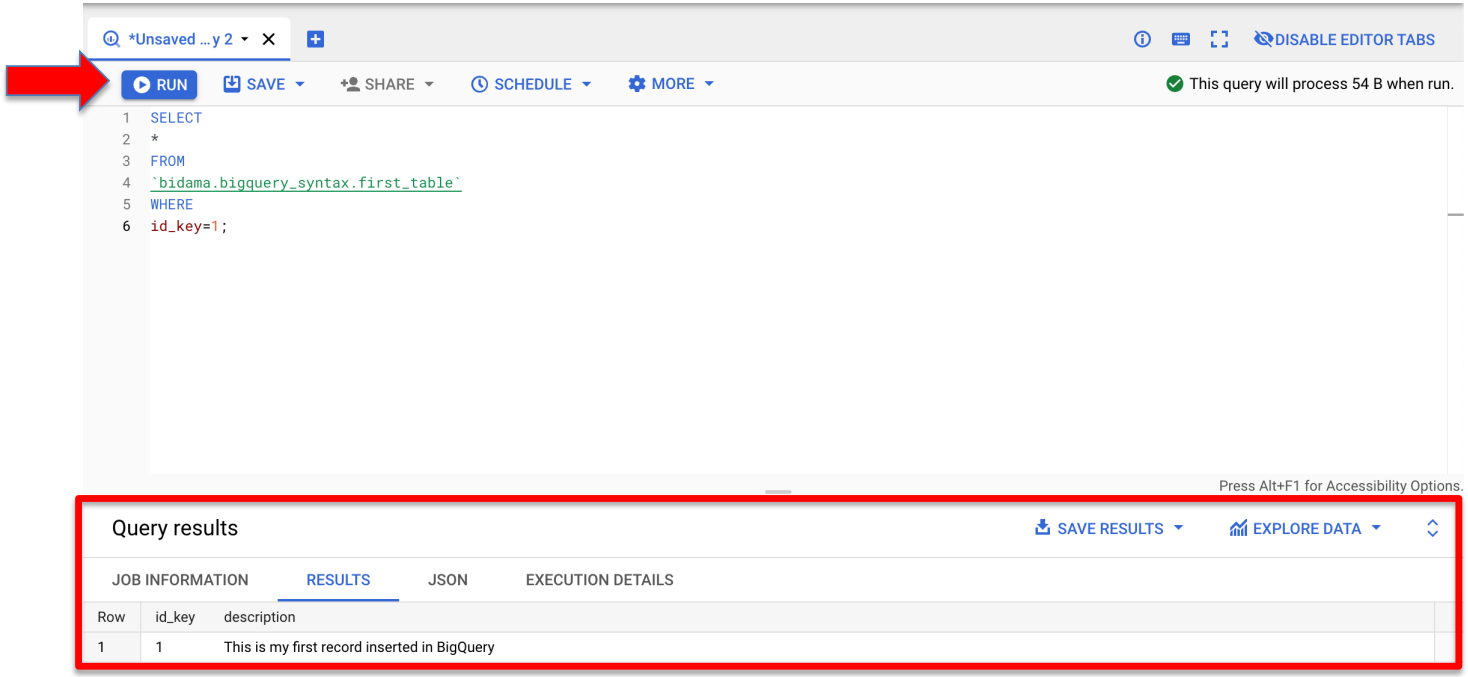
BigQuery SQL: VIEW

- The **CREATE VIEW** statement is used to create a **view**, i.e. to access records of an underlaying table



BigQuery SQL: READ

- **Read** operations are mainly based on **SELECT** statements and can be applied to different database objects such as tables and views



BigQuery SQL: READ

*Unsaved ... y 2 | first_table | DISABLE EDITOR TABS

RUN | SAVE | SHARE | SCHEDULE | MORE | This query will process 8 B when run.

```
SELECT COUNT(*) FROM (
SELECT
*
FROM
`bidama.bigquery_syntax.first_table`
WHERE
id_key=1
);
```

Press Alt+F1 for Accessibility Options.

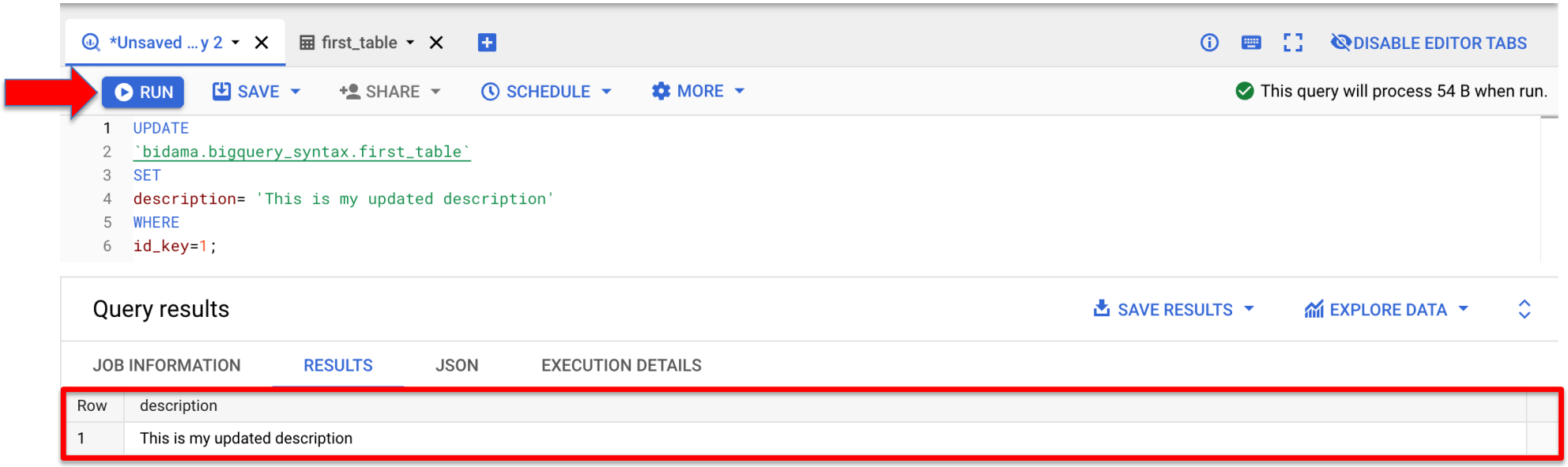
Query results | SAVE RESULTS | EXPLORE DATA

JOB INFORMATION | RESULTS | JSON | EXECUTION DETAILS

Row	f0_
1	1

BigQuery SQL: UPDATE

- Update operations such as **UPDATE** and **MERGE** are supported and can be used to change the value of a record or a set of records



BigQuery SQL: DELETE

- **DELETE** a record

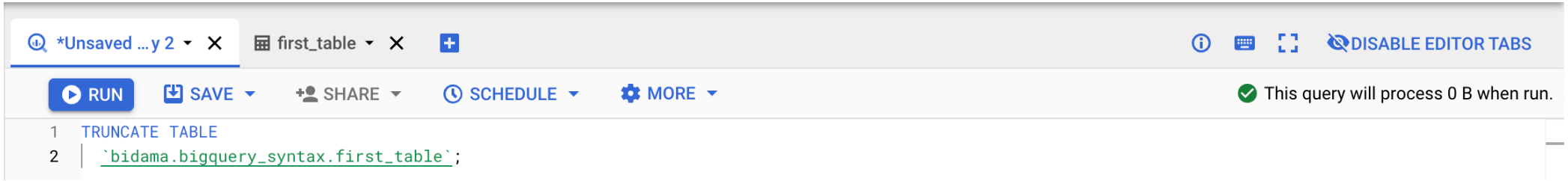
*Unsaved ... y 2 | first_table | DISABLE EDITOR TABS

RUN | SAVE | SHARE | SCHEDULE | MORE | This script will process 40 B when run.

```
DELETE
  `bidama.bigquery_syntax.first_table`
WHERE
  id_key=1;
```

BigQuery SQL: DELETE

- **DELETE** all record from a table



BigQuery SQL: DELETE

- **DELETE** a table

*Unsaved ... y 2 | first_table | DISABLE EDITOR TABS

RUN | SAVE | SHARE | SCHEDULE | MORE | This query will process 0 B when run.

```
1 DROP TABLE
2   `bidama.bigquery_syntax.first_table`;
```

BigQuery SQL: DELETE

- **DELETE** a view (only metadata are deleted)

*Unsaved ... y 2 | first_table | DISABLE EDITOR TABS

RUN | SAVE | SHARE | SCHEDULE | MORE | This query will process 0 B when run.

```
DROP VIEW
  `bidama.bigquery_syntax.first_view`;
```