**Big Data and Machine Learning on Google Cloud.**

**Introduction.**

welcome to the first section of the big data and machine learning fundamentals course here you'll explore the google infrastructure through compute and storage and see how innovation has enabled big data and machine learning capabilities after that you'll explore the history of big data and ml products which will help you understand the relevant product categories and to put it all together you'll see an example of a customer who adopted google cloud for their big data and machine learning needs finally you'll get hands-on practice using big data tools to analyze a public data set google has been working with data and artificial intelligence since its early days as a company in 1998. ten years later in 2008 the google cloud platform was launched to provide secure and flexible cloud computing and storage services you can think of the google cloud infrastructure in terms of three layers at the base layer is networking and security which lays the foundation to support all of google's infrastructure and applications on the next layer sit compute and storage google cloud separates or decouples as it's technically called compute and storage so they can scale independently based on need and on the top layer set the big data and machine learning products which enable you to perform tasks to ingest store process and deliver business insights data pipelines and ml models



and thanks to google cloud these tasks can be accomplished without needing to manage and scale the underlying infrastructure in the videos that follow we'll focus on the middle layer compute and storage in the top layer big data and machine learning products network and security fall outside of the focus of this course but if you're interested in learning more you can explore cloud.google.com forward slash training for more options

**Compute.**

let's focus our attention on the middle layer of the google cloud infrastructure compute and storage we'll begin with compute organizations with growing data needs often require lots of compute power to run big data jobs and as organizations design for the future the need for compute power only grows

00:20

google offers a range of computing services the first is compute engine compute engine is an is offering or infrastructure as a service which provides raw compute storage and network capabilities organized virtually into resources that are similar to physical data centers it provides maximum flexibility for those who prefer to manage server

00:44

instances themselves the second is google kubernetes engine or gke gke runs containerized applications in a cloud environment as opposed to on an individual virtual machine like compute engine a container represents code packaged up with all its dependencies the third computing service offered by google is app engine a fully managed

01:09

pass offering or platform as a service pass offerings bind code to libraries that provide access to the infrastructure application needs this allows more resources to be focused on application logic and then there is cloud functions which executes code in response to events like when a new file is uploaded to cloud

01:31

storage it's a completely serverless execution environment often referred to as functions as a service let's look at an example of a technology that requires a lot of compute power google photos offers a feature called automatic video stabilization this takes an unstable video like one captured while riding on the back of a

01:52

motorbike and stabilizes it to minimize movement for this feature to work as intended you need the proper data this includes the video itself which is really a large collection of individual images along with time series data on the camera's position in orientation from the onboard gyroscope and motion

02:21

from the camera lens a short video can require over a billion data points to feed the ml model to create a stabilized version as of 2020 roughly 28 billion photos and videos were uploaded to google photos every week with more than 4 trillion photos in total stored in the service

02:41

to ensure that this feature works as intended and accurately the google photos team needed to develop train and serve a high performing machine learning model on millions of videos that's a large training data set just as the hardware on a standard personal computer might not be powerful enough to process a big data job for an

03:01

organization the hardware on a smartphone is not powerful enough to train sophisticated ml models that's why google trains production machine learning models on a vast network of data centers only to then deploy smaller trained versions of the models to smartphone and personal computer hardware but where does all the processing power

03:21

come from according to stanford university's 2019 ai index report before 2012 artificial intelligence results tracked closely with moore's law with compute power doubling every two years the report states that since 2012 computing power has been doubling approximately every three and a half months this means that hardware manufacturers

03:47

have run up against limitations and cpus which are central processing units and gpus which are graphics processing units can no longer scale to adequately reach the rapid demand for machine learning to help overcome this challenge in 2016 google introduced the tensor processing unit or tpu tpus are google's custom developed

04:09

application specific integrated circuits used to accelerate machine learning workloads tpus act as domain specific hardware as opposed to general purpose hardware with cpus and gpus this allows for higher efficiency by tailoring architecture to meet the computation needs in a domain such as the matrix multiplication in machine learning

04:33

with tpus the computing speed increases more than 200 times this means that instead of waiting 26 hours for results with a single state-of-the-art gpu you'll only need to wait 7.9 minutes for a full cloud tpu v2 pod to deliver the same results cloud tpus have been integrated across

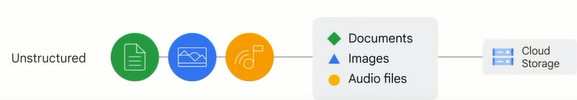
04:54

google products and this state-of-the-art hardware and supercomputing technology is available with google cloud products and services

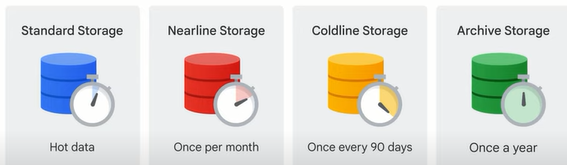
**Storage.**

now that we've explored compute and why it's needed for big data and ml jobs let's now examine storage for proper scaling capabilities compute and storage are decoupled this is one of the major differences between cloud computing and desktop computing with cloud computing processing limitations aren't attached to storage disks most applications require a database and storage solution of some kind with compute engine for example which was mentioned in the previous video you can install and run a database on a virtual machine just as you would do in a data center alternatively google cloud offers fully managed database and storage services these include cloud storage cloud bigtable cloud sql cloud spanner and firestore the goal of these products is to reduce the time and effort needed to store data this means creating an elastic storage bucket directly in a web interface or through a command line google cloud offers relational and non-relational databases in worldwide object storage we'll explore those options in more detail soon choosing the right option to store and process data often depends on the data type that needs to be stored in the business need

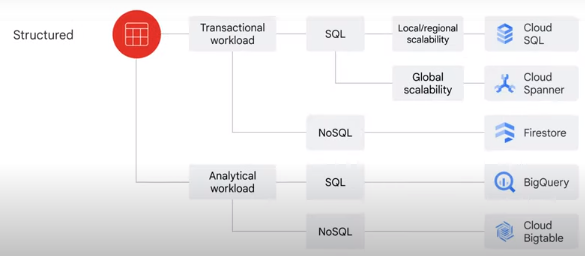
let's start with unstructured versus structured data unstructured data is information stored in a non-tabular form such as documents images and audio files unstructured data is usually best suited to cloud storage



cloud storage has four primary storage classes the first is standard storage standard storage is considered best for frequently accessed or hot data it's also great for data that is stored for only brief periods of time the second storage class is nearline storage this is best for storing infrequently accessed data like reading or modifying data once per month or less on average examples include data backups long tail multimedia content or data archiving the third storage class is cold line storage this is also a low cost option for storing infrequently accessed data however as compared to nearline storage cold line storage is meant for reading or modifying data at most once every 90 days the fourth storage class is archive storage this is the lowest cost option used ideally for data archiving online backup and disaster recovery it's the best choice for data that you plan to access less than once a year because it has higher costs for data access and operations and a 365-day minimum storage duration

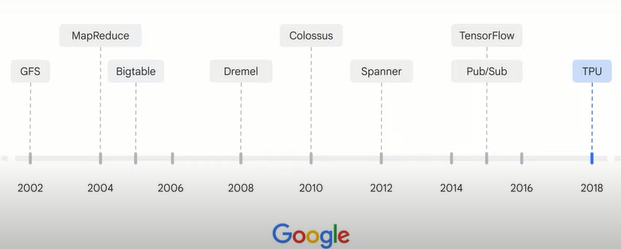


alternatively there is structured data which represents information stored in tables rows and columns structured data comes in two types transactional workloads and analytical workloads transactional workloads stem from online transactional processing systems which are used when fast data inserts and updates are required to build row-based records this is usually to maintain a system snapshot they require relatively standardized queries that impact only a few records then there are analytical workloads which stem from online analytical processing systems which are used when entire data sets need to be read they often require complex queries for example aggregations once you've determined if the workloads are transactional or analytical you'll need to identify whether the data will be accessed using sql or not so if your data is transactional and you need to access it using sql then cloud sql and cloud spanner are two options cloud sql works best for local to regional scalability while cloud spanner is best to scale a database globally if the transactional data set will be accessed without sql firestore might be the best option firestore is a transactional nosql document-oriented database if you have analytical workloads that require sql commands bigquery is likely the best option bigquery google's data warehouse solution lets you analyze petabyte-scale data sets alternatively cloud bigtable provides a scalable no-sql solution for analytical workloads it's best for real-time throughput applications that require only millisecond latency.



**The history of big data and ML products.**

the final layer of the google cloud infrastructure that is left to explore is big data and machine learning products in this video we'll examine the evolution of data processing frameworks through the lens of product development understanding the chronology of products can help address typical big data and machine learning challenges historically speaking google experienced challenges related to big data quite early mostly with large data sets fast changing data and varied data this was the result of needing to index the world wide web and as the internet grew google needed to invent new data processing methods so in 2002 google released the google file system or gfs gfs was designed to handle data sharing and petabyte storage at scale it served as the foundation for cloud storage and also what would become the managed storage functionality in bigquery a challenge that google was facing around this time was also how to index the exploding volume of content on the web to solve this in 2004 google wrote a report that introduced mapreduce mapreduce was a new style of data processing designed to manage large-scale data processing across big clusters of commodity servers as google continued to grow new challenges arose specifically with recording and retrieving millions of streaming user actions with high throughput the solution was the release in 2005 of cloud bigtable a high performance nosql database service for large analytical and operational workloads with mapreduce available some developers were restricted by the need to write code to manage their infrastructure which prevented them from focusing on application logic as a result from 2008 to 2010 google started to move away from mapreduce as the solution to process and query large data sets so in 2008 dremel was introduced dremel took a new approach to big data processing by breaking the data into smaller chunks called shards and then compressing them dremel then uses a quick optimizer to share tasks between the many shards of data in the google data centers which processed queries and delivered results the big innovation was that dremel auto scaled to meet query demands dremel became the query engine behind bigquery google continued innovating to solve big data and machine learning challenges some of the technology solutions released include colossus in 2010 which is a cluster level file system and successor to the google file system spanner in 2012 which is a globally consistent scalable relational database pub sub in 2015 which is a service used for streaming analytics and data integration pipelines to ingest and distribute data and tensorflow also in 2015 which is a free and open source software library for machine learning and artificial intelligence 2018 brought the release of the tensor processing unit or tpu which you'll recall from earlier in this course and it's thanks to these technologies that the big data and machine learning product line is now robust this includes cloud storage dataproc cloud bigtable bigquery dataflow firestore pub sub looker cloud spanner automl and vertex ai the unified platform these products and services are made available through google cloud and you'll get hands-on practice with some of them as part of this course



**Big data and ML product categories.**

as we explored in the last video google offers a range of big data and machine learning products so how do you know which is best for your business needs let's take a closer look at the list of products which can be divided into four general categories along the data to ai workflow ingestion and process storage analytics and machine learning understanding these product categories can help narrow down your choice



**the first category is ingestion and process** which include products that are used to digest both real-time and batch data the list includes pub sub data flow data proc and cloud data fusion you'll explore how dataflow and pub sub can ingest streaming data later in this course

**the second product category is data storage** and you'll recall from earlier that there are five storage products cloud storage cloud sql cloud spanner cloud bigtable and firestore cloud sql and cloud spanner are relational databases while bigtable and firestore are nosql databases

**the third product category is analytics** the major analytics tool is bigquery bigquery is a fully managed data warehouse that can be used to analyze data through sql commands in addition to bigquery you can analyze data and visualize results using google data studio and looker you'll explore bigquery looker and data studio in this course

**and the final product category is machine learning or ml** ml products include both the ml development platform and the ai solutions the primary product of the ml development platform is vertex ai which includes automl vertex ai workbench and tensorflow ai solutions are built on the ml development platform and includes state-of-the-art products to meet both horizontal and vertical market needs these include document ai contact center ai retail product discovery and healthcare data engine these products unlock insights that only large amounts of data can provide we'll explore the machine learning options and workflow together with these products in greater detail later.

**Customer example: Gojek.**

with many big data and machine learning product options available it can be helpful to see an example of how an organization has leveraged google cloud to meet their goals in this video you'll learn about a company called gojek and how they were able to find success through google cloud's data engineering and machine learning offerings the story starts in jakarta indonesia traffic congestion is a fact of life for most indonesian residents to minimize delays many rely heavily on motorcycles including motorcycle taxis known as ojx to travel to and from work or personal engagements founded in 2010 and headquartered in jakarta a company called gojek started as a call center for ojek bookings the organization has written demand for the service to become one of the few unicorns in southeast asia a unicorn is a privately held startup business valued at over 1 billion us dollars since its inception gojek has collected data to understand

01:03

customer behavior and in 2015 launched a mobile application that bundled ride hailing food delivery and grocery shopping they hit hyper growth very quickly according to the q2 2021 gojek fact sheet the gojek app has been downloaded over 190 million times and they have 2 million driver partners and about 900 000 merchant partners

01:31

the business has relied heavily on the skills and expertise of the technology team and on selecting the right technologies to grow and to expand into new markets gojek chose to run its applications and data in google cloud gojek's goal is to match the right driver with the right request as quickly

01:49

as possible in the early days of the app a driver would be pinged every 10 seconds which meant 6 million pings per minute which turned out to be 8 billion pings per day across their driver partners they generated around 5 terabytes of data each day leveraging information from this data

02:09

was vital to meeting their company goals but gojek faced challenges along the way let's explore two of them to see how google cloud was able to solve them the first challenge was data latency when they wanted to scale their big data platform they found that most reports were produced one day later so they

02:28

couldn't identify problems immediately to help solve this go-jek migrated their data pipelines to google cloud the team started using dataflow for streaming data processing and bigquery for real-time business insights another challenge was quickly determining which location had too many or too few drivers to meet demand gojek was able to use dataflow to build

02:53

a streaming event data pipeline this let driver locations ping pub sub every 30 seconds and dataflow would process the data the pipeline would aggregate the supply pings from the drivers against the booking requests this would connect to gojek's notification system to alert drivers where they should go this process required a system that was

03:15

able to scale up to handle times of high throughput and then back down again dataflow was able to automatically manage the number of workers processing the pipeline to meet demand the gojek team was able to visualize and identify supply and demand issues they discovered that the areas with the

03:32

highest discrepancy between supply and demand came from train stations often there were far more booking requests than there were available drivers since using google cloud's big data and machine learning products the gojek team has been able to actively monitor requests to ensure the drivers are in the areas with the highest demand

03:54

this brings faster bookings for riders and more work for the drivers.

**Lab introduction: Exploring a BigQuery Public Dataset.**

now it's time for you to take a break from course videos and get hands-on practice with one of the big data and machine learning products that was introduced earlier bigquery in the lab that follows this video you'll use bigquery to explore a public data set you'll practice querying a public data set creating a custom table loading data into a table and querying a table please note that this exercise involves leaving the current learning platform and opening quick labs qwiklabs offers a clean google cloud environment for a fixed period of time you'll have multiple attempts at each lab so if you don't complete it the first time or if you want to experiment more with it later you can return and start a new instance.

**Lab: Exploring a BigQuery Public Dataset.**

### **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.

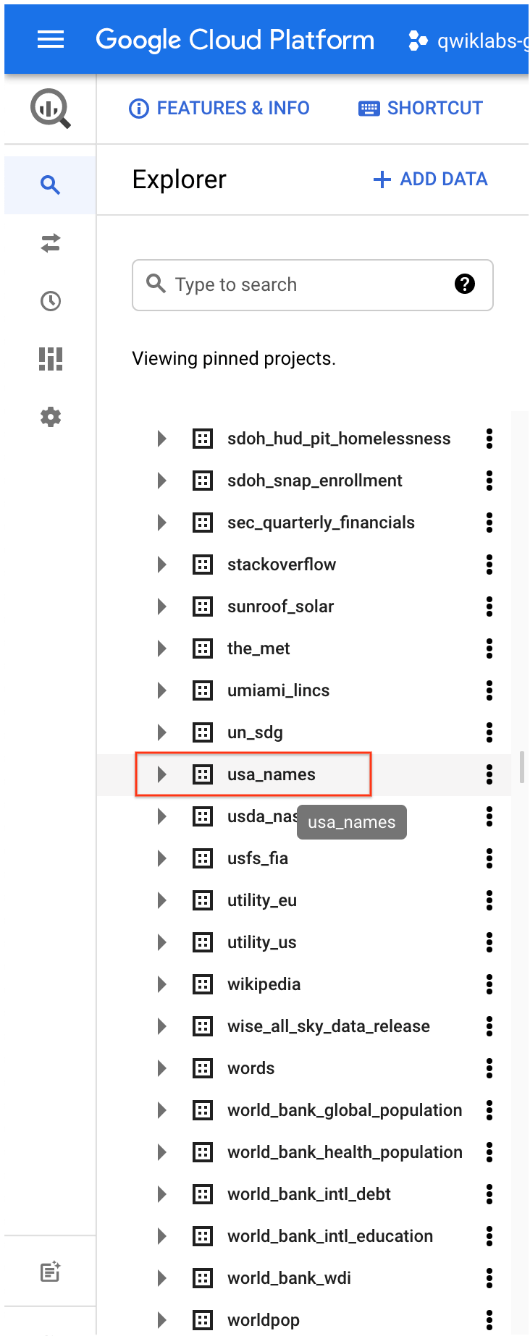
1. Click **Done**.

## Task 1. Query a public dataset

In this task, you load a public dataset, USA Names, into BigQuery, then query the dataset to determine the most common names in the US between 1910 and 2013.

### **Load the USA Names dataset**

1. In the left pane, click **ADD DATA** > **Pin a project**.
2. Click **Enter project name**.
3. Enter bigquery-public-data and click **Pin**.
4. Click bigquery-public-data in the pinned project list to expand it.
5. Scroll down the list of public datasets, clicking **More Results** until you find **usa\_names**.



1. Click **usa\_names** to expand the dataset.
2. Click **usa\_1910\_2013** to open that table.
3. Click **Query** above the schema and click **In new tab** to open a new Query editor tab.

**Note:** You can also search and explore the available datasets va the **ADD DATA** > **Explore public datasets** menu.

### **Query the USA Names dataset**

Query bigquery-public-data.usa\_names.usa\_1910\_2013 for the name and gender of the babies in this dataset, and then list the top 10 names in descending order.

1. Copy and paste the following query into the **Query editor** text area, replacing the existing query:

SELECT

name, gender,

SUM(number) AS total

FROM

`bigquery-public-data.usa\_names.usa\_1910\_2013`

GROUP BY

name, gender

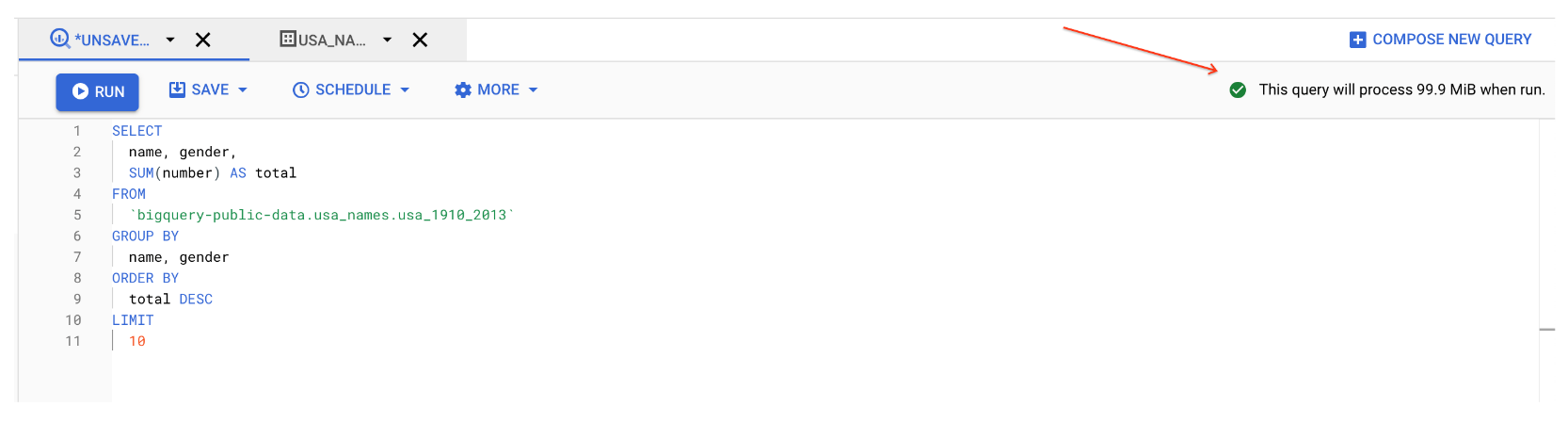
ORDER BY

total DESC

LIMIT

10

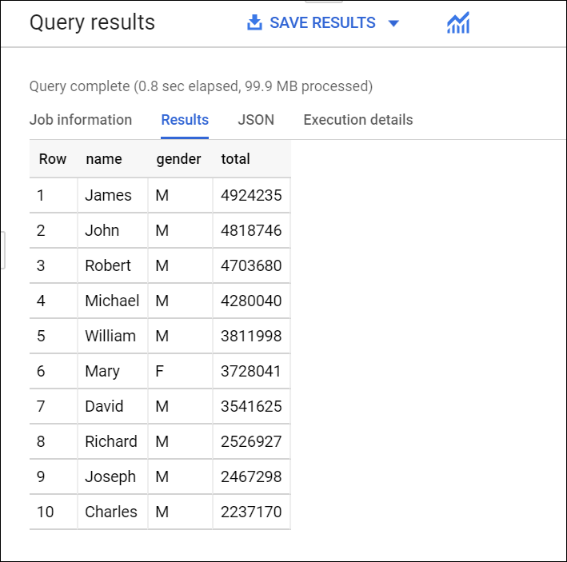
1. In the upper right of the window, view the query validator.



BigQuery displays a green check mark icon if the query is valid. If the query is invalid, a red exclamation point icon is displayed. When the query is valid, the validator also shows the amount of data the query processes when you run it. This helps to determine the cost of running the query.

1. Click **Run**.

The query results opens below the Query editor. At the top of the Query results section, BigQuery displays the time elapsed and the data processed by the query. Below the time is the table that displays the query results. The header row contains the name of the column as specified in GROUP BY in the query.



## Task 2. Create a custom table

In this task, you create a custom table, load data into it, and then run a query against the table.

### **Download the data to your local computer**

The file you're downloading contains approximately 7 MB of data about popular baby names, and it is provided by the US Social Security Administration.

1. Download the [baby names zip file](https://www.ssa.gov/OACT/babynames/names.zip) to your local computer.

**Note:** If this download link fails please copy the baby names zip file from the student resources on the left pane of the instruction guide.

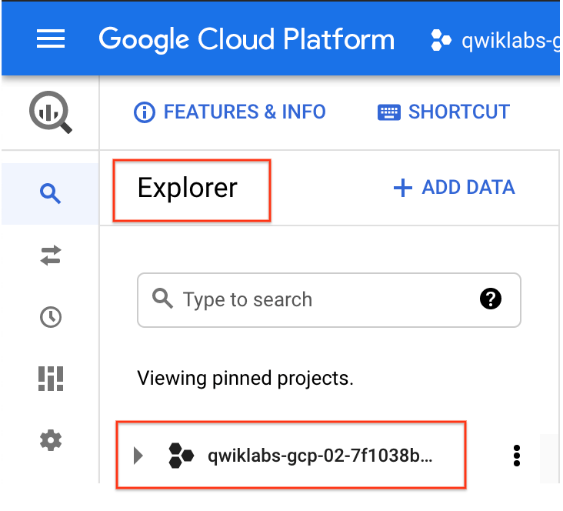
1. Unzip the file onto your computer.
2. The zip file contains a NationalReadMe.pdf file that describes the dataset. [Learn more about the dataset](https://www.ssa.gov/OACT/babynames/background.html).
3. Open the file named yob2014.txt to see what the data looks like. The file is a comma-separated value (CSV) file with the following three columns: name, sex (M or F), and number of children with that name. The file has no header row.
4. Note the location of the yob2014.txt file so that you can find it later.

## Task 3. Create a dataset

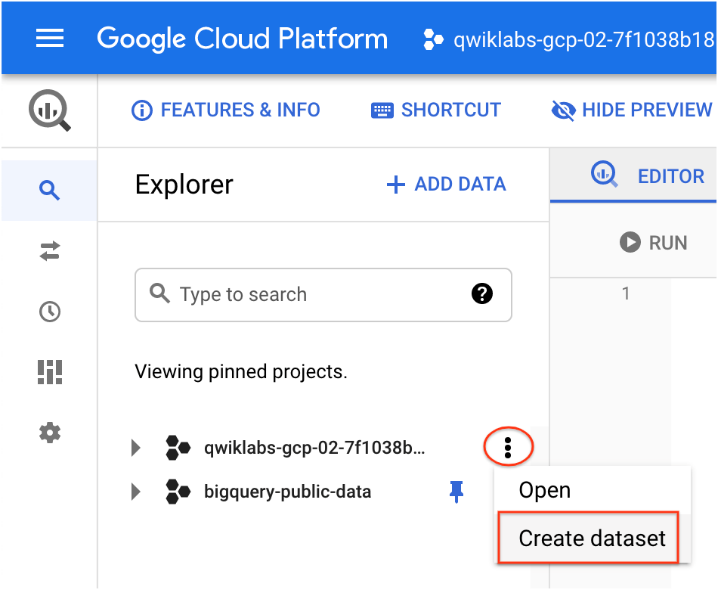
In this task, you create a dataset to hold your table, add data to your project, then make the data table you'll query against.

Datasets help you control access to tables and views in a project. This lab uses only one table, but you still need a dataset to hold the table.

1. Back in the Cloud Console, in the left pane, in the **Explorer** section, click your Project ID (it will start with qwiklabs).



1. Click on the three dots next to your project ID and then click **Create dataset**.



1. On the **Create dataset** page:
   * For **Dataset ID**, enter babynames.
   * For **Data location**, choose **us (multiple regions in United States)**.
   * For **Default table expiration**, leave the default value.
   * For **Encryption**, leave the default value.
2. Click **Create dataset** at the bottom of the pane.

## Task 4. Load the data into a new table

In this task, you load data into the table you made.

1. Click on the three dots next to **babynames** found in the left pane in the **Explorer** section, and then click **Create table**.

Use the default values for all settings unless otherwise indicated.

1. On the **Create table** page:
   * For **Source**, choose **Upload** from the Create table from: dropdown menu.
   * For **Select file**, click **Browse**, navigate to the yob2014.txt file and click **Open**.
   * For **File format**, choose **CSV** from the dropdown menu.
   * For **Table name**, enter names\_2014.
   * In the **Schema** section, click the **Edit as text** toggle and paste the following schema definition in the text box.

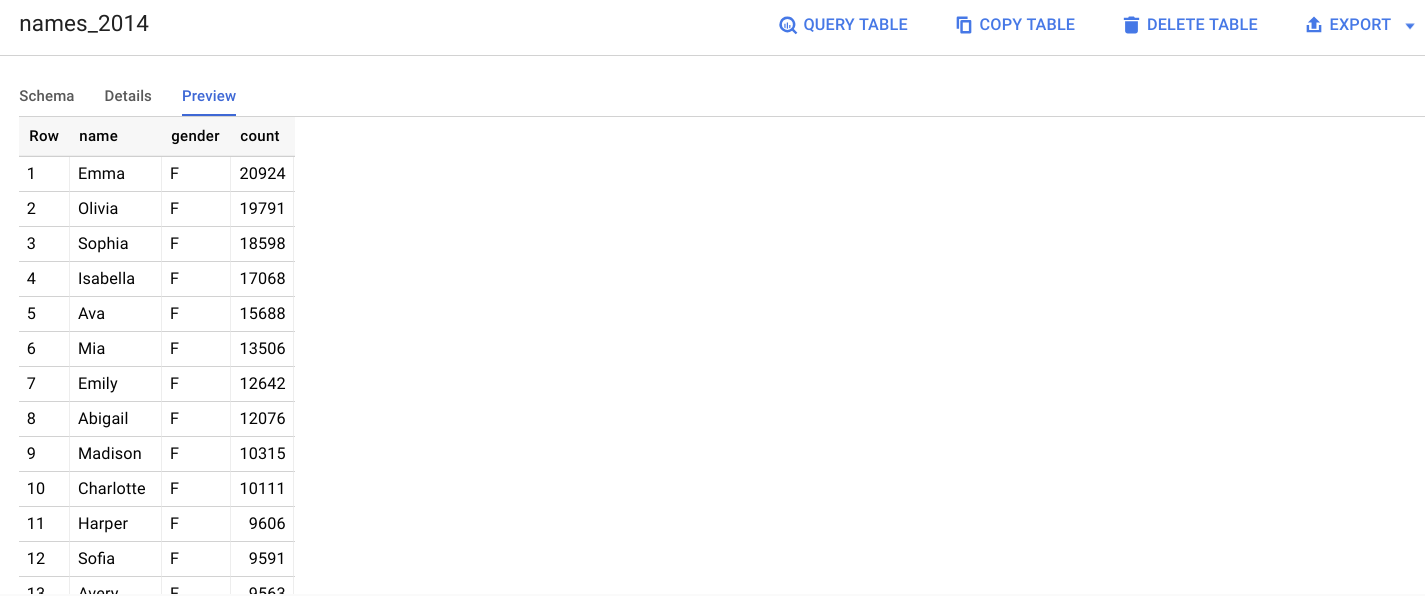
name:string,gender:string,count:integer

1. Click **Create table** (at the bottom of the window).

**Note:** If you see an import error, your data should still have been imported. To clear the error click **Close** , then click **Cancel** to exist the import dialog, and finally click **Yes, quit** in response to the warning that your changes will not be saved.

### **Preview the table**

1. In the left pane, select **babynames** > **names\_2014** in the navigation pane.
2. In the details pane, click the **Preview** tab.



## Quick quiz. You need a table to hold the dataset.

## 

## True

## **False**

## Task 5. Query the table

Now that you've loaded data into your table, you can run queries against it. The process is identical to the previous example, except that this time, you're querying your table instead of a public table.

1. In the Query editor, click **Compose new query**.
2. Copy and paste the following query into the **Query editor**. This query retrieves the top 5 baby names for US males in 2014.

Note: Inside '' it does distinguish upper vs. lower case, therefore make sure to align exactly the names of the dataset and the table you created.

SELECT

name, count

FROM

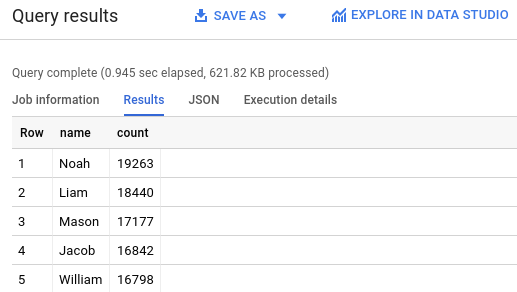
`babynames.names\_2014`

WHERE

gender = 'M'

ORDER BY count DESC LIMIT 5

Click **Run**. The results are displayed below the query window.



Quick quiz. Check which you can use to access BigQuery.

**Third-party tools**

**Command-line tool**

**Make calls to BigQuery REST API**

**Web UI**

### **Congratulations!**

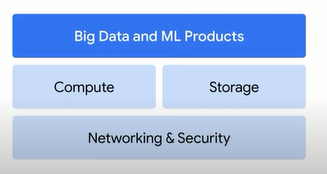
You queried a public dataset, then created a custom table, loaded data into it, and then ran a query against that table.

**Summary.**

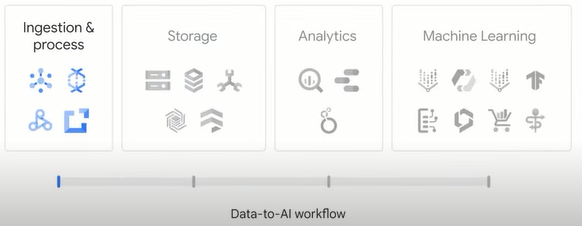
this brings us to the end of the first section of the big data and machine learning fundamentals course before we move forward let's review what we've covered so far you began by exploring the google cloud infrastructure through three different layers at the base layer is networking and security which makes up the foundation to support all of google cloud's infrastructure and applications on the next layer sit compute and storage google cloud decouples compute and storage so they can scale independently based on need and on the top layer sit the big data in machine learning products

in the next section you learned about the history of big data and machine learning technologies and then explored the four major product categories ingestion and process storage analytics and machine learning after that you saw an example of how go-jek the indonesian on-demand multi-service platform and digital payment technology group leveraged google cloud big data and machine learning products to expand their business and finally you got hands-on practice with bigquery by analyzing a public data set.

Google Cloud Infraestructure:



BigData and ML and the four major product categories :



**Quiz**

1. Compute Engine, Google Kubernetes Engine, App Engine, and Cloud Functions represent which type of services?

Database and storage

Networking

**Compute**

Machine learning

2. Which Google hardware innovation tailors architecture to meet the computation needs on a domain, such as the matrix multiplication in machine learning?

CPUs (central processing units)

**TPUs (Tensor Processing Units)**

GPUs (graphic processing units)

DPUs (data processing units)

3. Pub/Sub, Dataflow, Dataproc, and Cloud Data Fusion align to which stage of the data-to-AI workflow?

**Ingestion and process**

Analytics

Storage

Machine learning

4. Cloud Storage, Cloud Bigtable, Cloud SQL, Cloud Spanner, and Firestore represent which type of services?

Machine learning

**Database and storage**

Networking

Compute

5.Which data storage class is best for storing data that needs to be accessed less than once a year, such as online backups and disaster recovery?

Standard storage

Coldline storage

Nearline storage

**Archive storage**

6. AutoML, Vertex AI Workbench, and TensorFlow align to which stage of the data-to-AI workflow?

Ingestion and process

Analytics

Storage

**Machine learning**