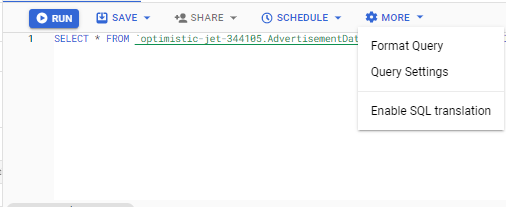
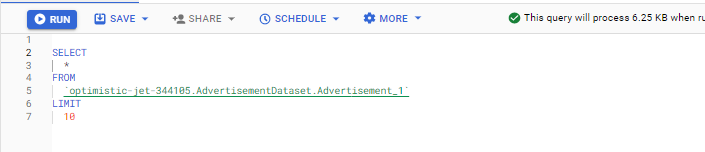
BigQuery : Format query and Query Settings

**Format Query**

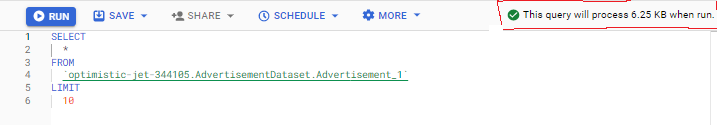
Select Format Query option under the MORE drop down menu.

Format query is used to make the queries more readable and easy to understand

Here is the how the formatted query looks like

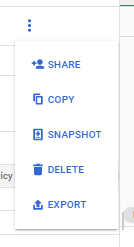


When we type a query in the ‘query editor’, BigQuery tells us in advance how much data the query will process

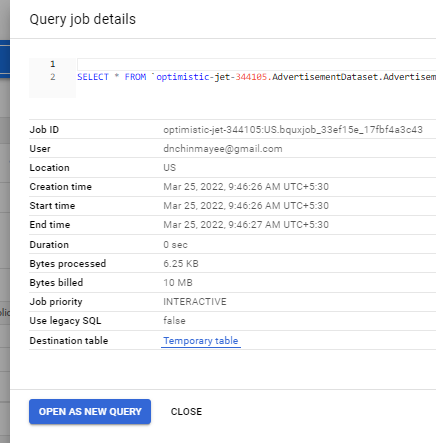


**Writing query results to a permanent table**

Under trmporary table we can preview the table and can export the data Google cloud storage.



BigQuery uses cached results that is within 24 hrs time period whether the underlining data has received any change or not. If not then you won’t occur any query costs.



**Temporary tables** are available only if the user wants to run the query. But if we want to share query results then it is need to save the query results to a **permanent table**.

Use the following procedure to write the query results to a permanent table.

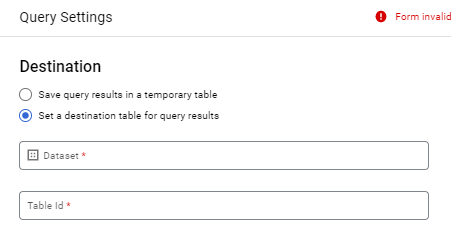
Open the BigQuery page in the cloud Console

Open the **explorer** and select project name then dataset name then table name.

Open the **query console** and enter a valid SQL query.

Click **MORE** and then select **Query Settings**

Select the **Set a destination table** for query results option.



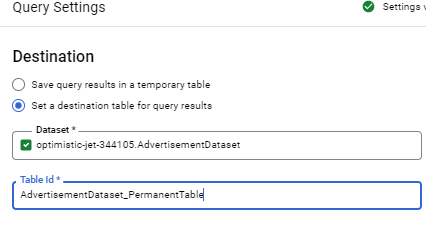
In the Destination section, select the Dataset name where the table will be created, and choose a Table name.

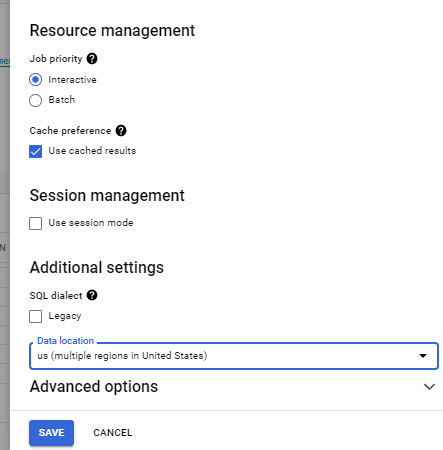
In the Destination table from preference section, choose one of the following:

* Write if empty — Writes the query results to the table only if the table is empty.
* Append to table — Appends the query results to an existing table.
* Overwrite table — Overwrites an existing table with the same name using the query results.

We can use any previous project and dataset name to save the result to a Permanent table or else we can create a new destination to save the result. For example:

Save the result in a previous project link……



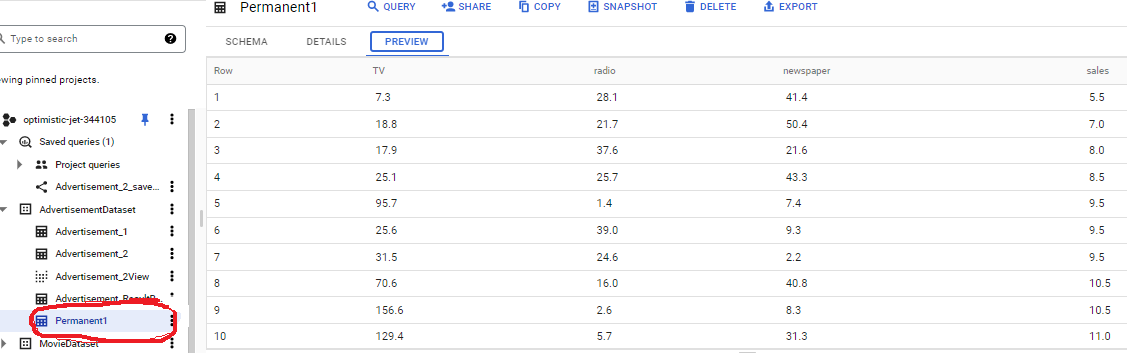


Use the data location as US.

OR else create a new destination and then give <Table name> to save the result or to a permanent table.

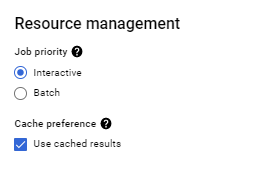
**Run** the query again.

Finally you will get a new table under left-hand side of the panel.



Another option is there to **SAVE RESULT tool** which is used to save results immediately to Google drive or local drive.

**Resource management**

****

**Running interactive and batch query jobs**

Two modes of query exist in BigQuery. They are 'interactive' and 'batch' queries.

* Interactive queries are executed immediately but have limits (runs concurrently 20 at a time).
* If those limits are hit, the query will fail immediately. This is because BigQuery assumes that an interactive query is something you need run immediately.
  + Example: join and Group by
* Batch queries can be queued and are limitless in the number of queries.
* If one fails it will be retried later, whereas an interactive query will fail immediately if the limits are hit.
* There are two primary reasons to use batch priority
  + It queues up your jobs.
  + It run low priority queries in a way that doesn't impact high priority ones.

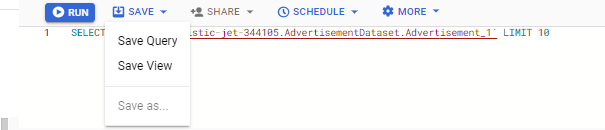
**Views tool**

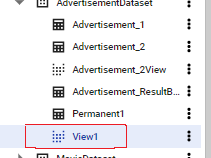
* A view is a virtual table defined by a query.
* A view can be queried just like a query table.
* Views are listed next to tables in a data set denoted with a different icon.
* When running a query that references a view, BigQuery will also run the query that defines the view.

**What is the advantage of View?**

* Views help to hide the complexity of data able allowing to join multiple tables into a single virtual table ready to query.
* Views does not require any storage since they are virtual tables and so there is not a storage cost associated with them.

**Create a view**

****

****

Choose the project and dataset where you want to create a view. The view is populated under your destination dataset and we can query it just like a query table.

**View limitations**

BigQuery views are subject to the following limitations:

* Views are read-only. You cannot run DML (insert, update, delete) queries against a view.
* The dataset that contains your view and the dataset that contains the tables referenced by the view must be in the same location.
* You cannot run a BigQuery job that exports data from a view.
* You cannot reference query parameters in views.
* You cannot reference a view in a wildcard table query.
* You cannot include a temporary user-defined function or a temporary table in the SQL query that defines a view.

**SCHEDULE**

* Schedule a queries runs queries in recurring basis.
* Scheduled queries must be written in standard SQL, which can include data definition language (DDL) and data manipulation language (DML) statements.
* You can organize query results by date and time.

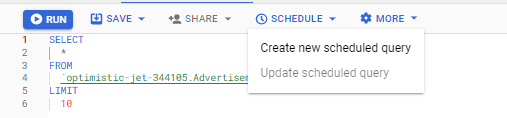
Steps needed before use SCHEDULE queries

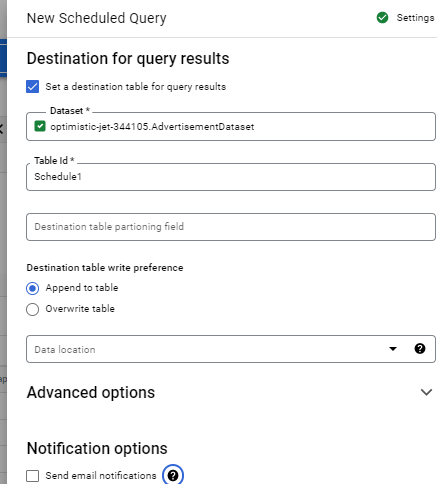
1. Creating a project and enabling the BigQuery API

Before using data transfer service one must need to create a new project or can use an existing project. In both the cases it is needed to create to enable the BigQuery API.

1. To create a project or use an existing projct and enable the BigQuery API

* Go to project selector and create a new Google Cloud project.
* For new projcets, BigQuery is automatically enabled.
* For an existing project, enable the BigQuery API.

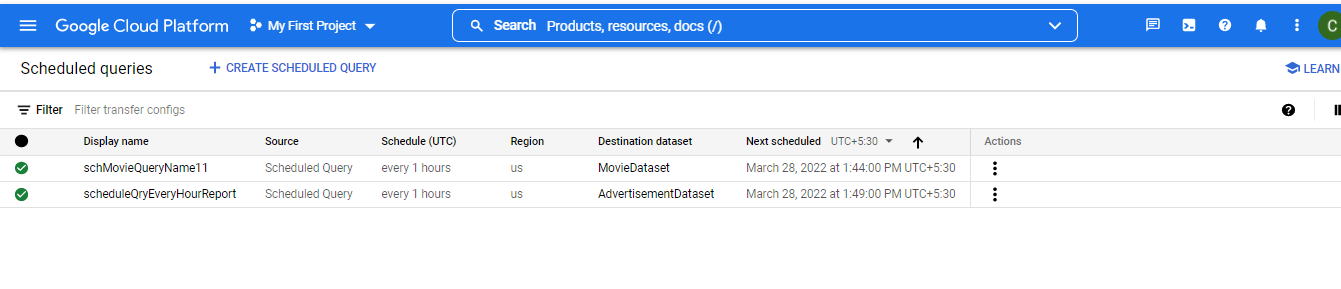
****

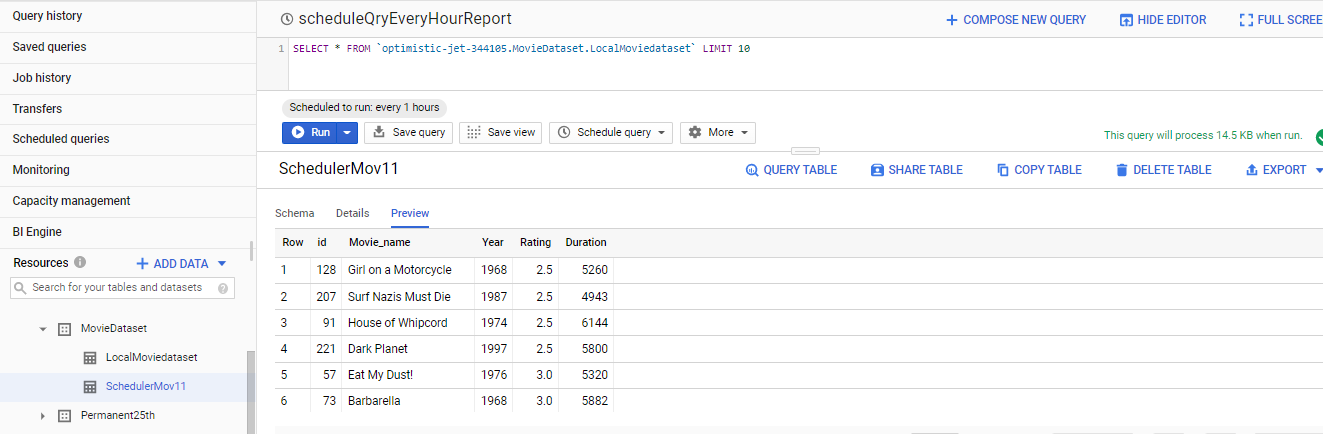
****

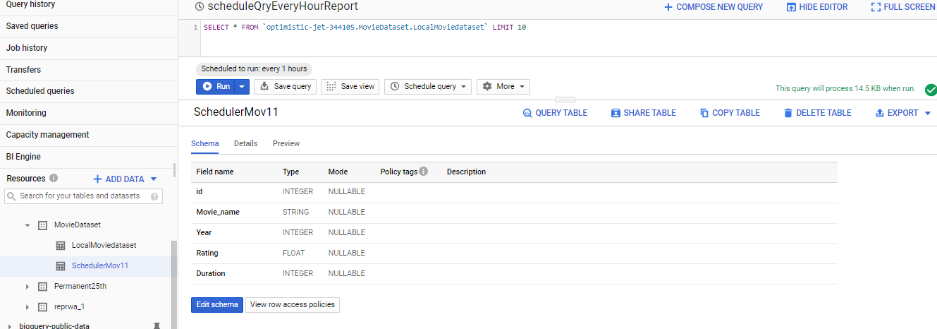
And **save** the schedule query.

(Add previous day data to every day.)

Schedule query is created. Go to SCHEDULE QUERY on left side of panel and click on SCHEDULE QUERY.







We can check the run history and Configuration.

https://prwatech.in/blog/google-cloud-platform/creating-scheduled-query-in-bigquery/

**Options**

https://cloud.google.com/bigquery/docs/scheduling-queries#:~:text=You%20can%20schedule%20queries%20to,query%20string%20and%20destination%20table.

* + - Ingestion time(when BigQuery ingests the data)
      * Daily partitioning
        + Apply when data is continuously added over time or data is spread out over a wide range of dates.
      * Hourly partitioning
        + High volume of data with timestamp value is less than 6 months.
      * Monthly/yearly partitioning
    - Date/Timestamp column
      * Tables are partitioned based on a TIMESTAMP, DATE, or DATETIME column in the table.