

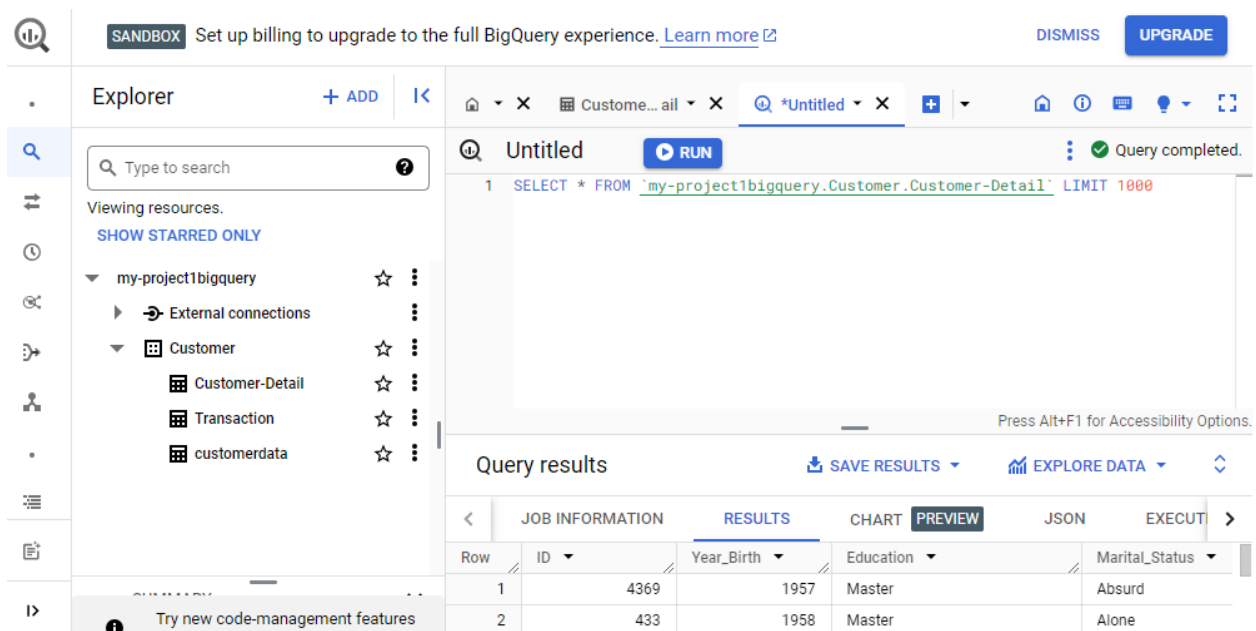
Customer Analyses with Big query

A. Go to google cloud then make project then use big query to add datasets

B.DML

1.Select from

customer DETAIL



The screenshot displays the Google Cloud BigQuery interface. On the left, the Explorer pane shows the project structure with a dataset named 'Customer' containing a table 'Customer-Detail'. The main editor shows a SQL query: `SELECT * FROM my-project1bigquery.Customer.Customer-Detail LIMIT 1000`. The query has been executed successfully, as indicated by the 'Query completed.' message. Below the query editor, the 'Query results' section is visible, showing a table with columns: Row, ID, Year_Birth, Education, and Marital_Status. The first two rows of data are displayed.

Row	ID	Year_Birth	Education	Marital_Status
1	4369	1957	Master	Absurd
2	433	1958	Master	Alone

CUSTOMER DATA

SANDBOX Set up billing to upgrade to the full BigQuery experience. [Learn more](#) **DISMISS** **UPGRADE**

Explorer + ADD <

Viewing resources.
[SHOW STARRED ONLY](#)

- my-project1bigquery
 - External connections
 - Customer
 - Customer-Detail
 - Transaction
 - customerdata

Untitled RUN Query completed.

```

1 SELECT * FROM `my-project1bigquery.Customer.Customer-Detail` LIMIT 1000;
2
3 SELECT * FROM `my-project1bigquery.Customer.customerdata` LIMIT 1000

```

Press Alt+F1 for Accessibility Options.

Query results SAVE RESULTS EXPLORE DATA

JOB INFORMATION RESULTS CHART PREVIEW JSON EXECUTE

Row	customer_id	customer_account	first_name
1	US-269345	MA253269345717	Michael
2	US-267098	MA371267098704	James

Try new code-management features of BigQuery Studio in Preview.

Transactions

my-project1bigquery

- External connections
- Customer
 - Customer-Detail
 - Transaction
 - customerdata

Untitled RUN

```

4
5 SELECT * FROM `my-project1bigquery.Customer.Transaction` LIMIT 1000

```

Press Alt+F1 for Accessibility Options.

Query results SAVE RESULTS EXPLORE DATA

JOB INFORMATION RESULTS CHART PREVIEW JSON EXECUTE

Row	ID	transactionId
1	5524	ORD201708011814
2	4141	ORD201708012479
3	5324	ORD201708012570
4	965	ORD201708012186
5	4855	ORD201708012305
6	1994	ORD201708012323
7	2125	ORD201708012322
8	2569	ORD201708012348

Results per page: 50 1 - 43 of 43

Try new code-management features of BigQuery Studio in Preview: version history, sharing with IAM, and creating Python notebooks with Colab.

2LEFTJOIN BETWEEN CUSTOMER DETAIL, TRANSACTION

The SQL query you've provided is used to retrieve all records from the **Transaction** table (**source**) that do not have a corresponding match in the **Customer-Detail** table (**target**). The **LEFT JOIN** is employed to include all records from the left (source) table

The screenshot shows the BigQuery Studio interface. On the left, a sidebar displays the project hierarchy: **my-project1bigquery** (expanded) containing **External connections**, **Customer** (expanded), **Customer-Detail**, **Transaction**, and **customerdata**. A notification banner at the bottom left reads: "Try new code-management features of BigQuery Studio in Preview: version history, sharing with IAM, and creating Python notebooks with Colab." The main editor displays a SQL query:

```

3 SELECT * FROM `my-project1bigquery.Customer.customerdata` LIMIT 1000;
4
5 SELECT * FROM `my-project1bigquery.Customer.Transaction` LIMIT 1000;
6
7 SELECT *
8 FROM `my-project1bigquery.Customer.Transaction` AS source
9 LEFT JOIN `my-project1bigquery.Customer.Customer-Detail` AS target
10 ON source.ID = target.ID;
11

```

Below the query editor, the "Query results" section is visible, showing tabs for **JOB INFORMATION**, **RESULTS** (selected), **CHART**, **PREVIEW**, **JSON**, and **EXECUT**. The results table displays the following data:

Row	ID	transactionId	ID_1	Year_Birth	E
1	5524	ORD201708011814	5524	1957	C
2	4141	ORD201708012479	4141	1965	C
3	5324	ORD201708012570	5324	1981	F

3.INNERJOIN

In this query:

- **INNER JOIN**: This type of join returns only the rows where there is a match between the **ID** column in the **Transaction** table (**source**) and the **Customer-Detail** table (**target**).
- **SELECT ***: Retrieves all columns from the result set.
- **FROM my-project1bigquery.Customer. Transaction AS source**: Specifies the source table as **Transaction** and aliases it as **source**.
- **INNER JOIN my-project1bigquery.Customer. Customer-Detail AS target ON source.ID = target.ID**: Performs an inner join based on the common **ID** column.

This query will return rows where there is a match between the **ID** columns in both tables. If there is no match, those rows won't be included in the result set. If you want to include all rows from the left table (**Transaction**) regardless of whether there is a match in the right table (**Customer-Detail**), you would use a **LEFT JOIN** instead, as shown in your original query.

The screenshot shows the BigQuery Studio interface with the same sidebar and notification banner as the previous image. The main editor displays a SQL query:

```

10 ON source.ID = target.ID;
11
12
13
14 SELECT *
15 FROM `my-project1bigquery.Customer.Transaction` AS source
16 INNER JOIN `my-project1bigquery.Customer.Customer-Detail` AS target
17 ON source.ID = target.ID;
18

```

Below the query editor, the "Query results" section is visible, showing tabs for **JOB INFORMATION**, **RESULTS** (selected), **CHART**, **PREVIEW**, **JSON**, and **EXECUT**. The results table displays the following data:

Row	ID	transactionId	ID_1	Year_Birth	E
1	5524	ORD201708011814	5524	1957	C
2	4141	ORD201708012479	4141	1965	C
3	5324	ORD201708012570	5324	1981	F

At the bottom of the results section, it shows "Results per page: 50" and "1 - 43 of 43".

4.SUM OF INCOME

```
SELECT SUM(INCOME) AS total_income
```

```
FROM `my-project1bigquery.Customer.Customer-Detail`;
```

The screenshot shows the BigQuery Studio interface. On the left, the 'Customer' dataset is expanded, showing 'Customer-Detail', 'Transaction', and 'customerdata'. The SQL editor contains the following query:

```
23 SELECT SUM(INCOME) AS total_income
24
25 FROM `my-project1bigquery.Customer.Customer-Detail`;
26
27
```

Below the editor, the 'Query results' section is displayed with the 'RESULTS' tab selected. The results table has one row:

Row	total_income
1	115779909

A notification box on the left side of the results area states: 'Try new code-management features of BigQuery Studio in Preview: version history, sharing with IAM, and creating Python notebooks with Colab.'

5.min-INCOME

```
SELECT min(INCOME) AS MIN_income
```

```
FROM `my-project1bigquery.Customer.Customer-Detail`;
```

The screenshot shows the BigQuery Studio interface. On the left, the 'Customer' dataset is expanded, showing 'Customer-Detail', 'Transaction', and 'customerdata'. The SQL editor contains the following query:

```
27 SELECT min(INCOME) AS MIN_income
28
29 FROM `my-project1bigquery.Customer.Customer-Detail`;
30
```

Below the editor, the 'Query results' section is displayed with the 'RESULTS' tab selected. The results table has one row:

Row	MIN_income
1	1730

6.count of female & males

The image displays two screenshots of a data query interface, likely from a tool like Power BI or DAX Studio. Both screenshots show a SQL query and its results.

Top Screenshot:

- Query:**

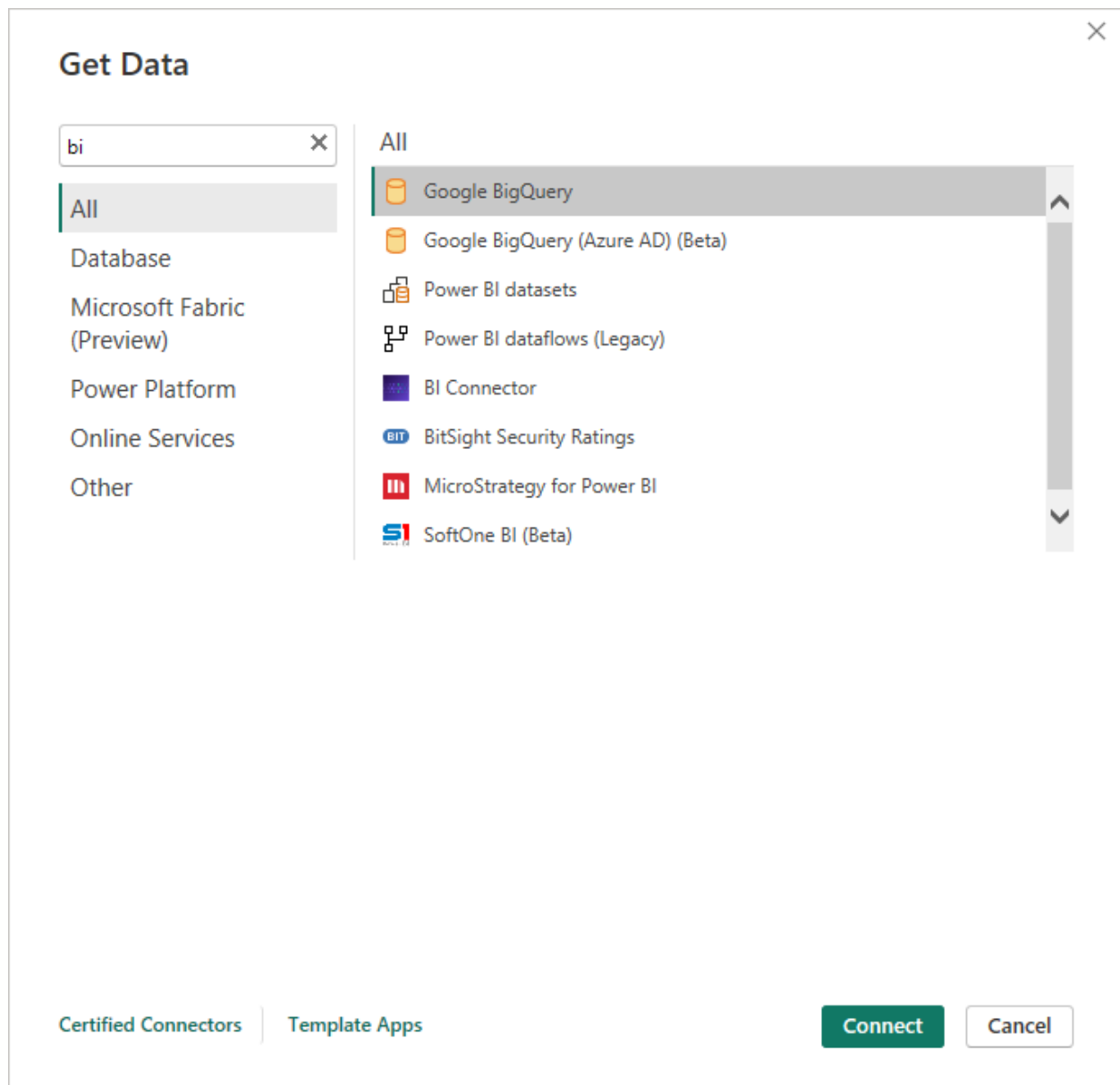
```
SELECT COUNT(gender) AS F
FROM my-project1bigquery.Customer.customerdata
WHERE gender = 'F';
```
- Results:** A table with 1 row and 2 columns. The first column is labeled 'F' and contains the value 6.

Bottom Screenshot:

- Query:**

```
SELECT COUNT(gender) AS F
FROM my-project1bigquery.Customer.customerdata
WHERE gender = 'M';
```
- Results:** A table with 1 row and 2 columns. The first column is labeled 'F' and contains the value 5.

Go to powerbi to analyses more and make active dashboard



REFERENCES

<https://towardsdatascience.com/merging-tables-using-sql-a2e60ff687e9>