

Using JOINS in SQL (BigQuery)



Sheba Morgan

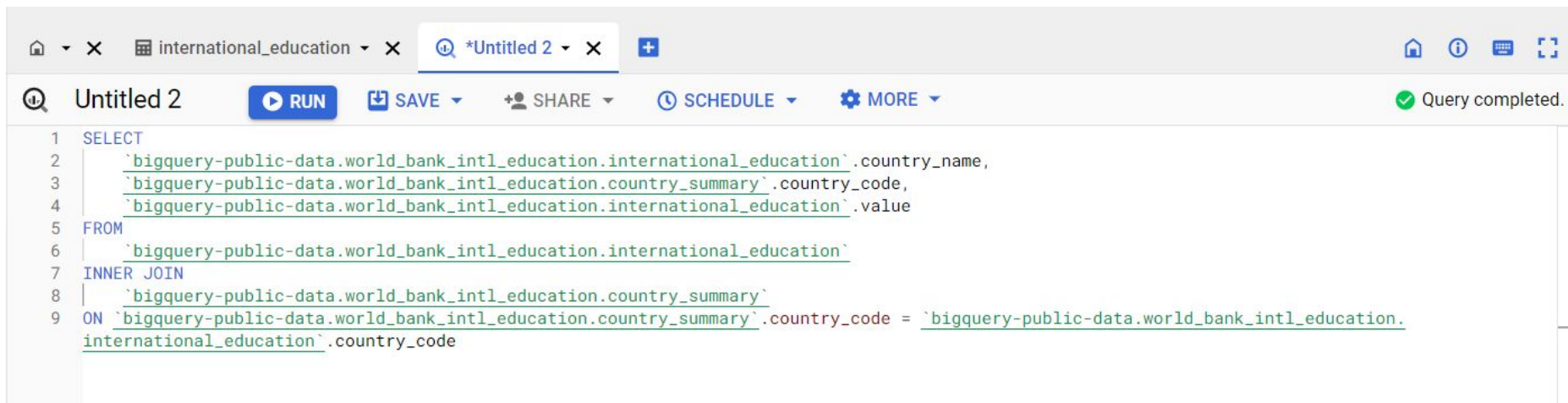
INTRODUCTION

For this data analysis project, I will be working with the International Education public dataset available in BigQuery. The dataset includes multiple tables, including the `international_education` table and the `country_summary` table. Both of these tables contain the column “`country_code`”, according to the datasets schema.



To combine the data from these tables based on the shared "country_code" column, I will be utilizing the INNER JOIN clause in my query. This allows me to merge the relevant information from both tables and gain comprehensive insights for analysis.

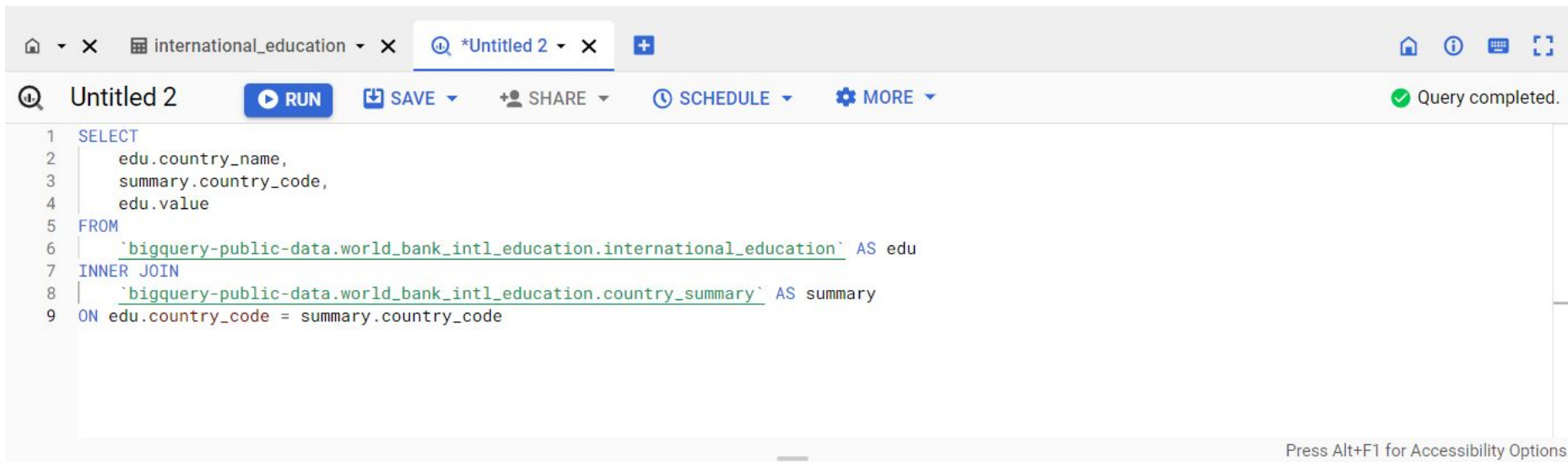
This was my initial query..



The screenshot shows a web browser window with a Google Cloud BigQuery console. The browser's address bar shows the URL `international_education`. The console's header bar includes a search icon, the text "Untitled 2", and buttons for "RUN", "SAVE", "SHARE", "SCHEDULE", and "MORE". A green checkmark and the text "Query completed." are visible on the right side of the header. The main area of the console displays a SQL query with line numbers 1 through 9. The query uses an INNER JOIN to combine data from two tables in the `bigquery-public-data.world_bank_intl_education` dataset, joining on the `country_code` column. The first table is `international_education` and the second is `country_summary`. The query selects `country_name` from the first table and `country_code` and `value` from the second table.

```
1 SELECT
2   `bigquery-public-data.world_bank_intl_education.international_education`.country_name,
3   `bigquery-public-data.world_bank_intl_education.country_summary`.country_code,
4   `bigquery-public-data.world_bank_intl_education.international_education`.value
5 FROM
6   `bigquery-public-data.world_bank_intl_education.international_education`
7 INNER JOIN
8   `bigquery-public-data.world_bank_intl_education.country_summary`
9 ON `bigquery-public-data.world_bank_intl_education.country_summary`.country_code = `bigquery-public-data.world_bank_intl_education.international_education`.country_code
```

Because that query was too long and a bit hard to work with, I decided to use an alias for each table instead..



The screenshot shows the Google Cloud BigQuery console interface. At the top, there's a tab bar with 'international_education' and '*Untitled 2'. Below the tab bar, the query editor shows a SQL query. The query selects 'edu.country_name', 'summary.country_code', and 'edu.value' from two tables: 'bigquery-public-data.world_bank_intl_education.international_education' (aliased as 'edu') and 'bigquery-public-data.world_bank_intl_education.country_summary' (aliased as 'summary'). The tables are joined using an INNER JOIN on the 'country_code' column. The interface includes a 'RUN' button, 'SAVE', 'SHARE', 'SCHEDULE', and 'MORE' options. A status message at the top right indicates 'Query completed.'.

```
1 SELECT
2   edu.country_name,
3   summary.country_code,
4   edu.value
5 FROM
6   `bigquery-public-data.world_bank_intl_education.international_education` AS edu
7 INNER JOIN
8   `bigquery-public-data.world_bank_intl_education.country_summary` AS summary
9 ON edu.country_code = summary.country_code
```

Press Alt+F1 for Accessibility Options.

Here are the results of my query..

Query results

[SAVE RESULTS](#)[EXPLORE DATA](#)[JOB INFORMATION](#)[RESULTS](#)[JSON](#)[EXECUTION DETAILS](#)[EXECUTION GRAPH](#)[PREVIEW](#)

Row	country_name	country_code	value
1	Chad	TCD	135079.0
2	Chad	TCD	92857.0
3	Chad	TCD	97603.0
4	Chad	TCD	158735.0
5	Chad	TCD	302478.0
6	Chad	TCD	181979.0
7	Chad	TCD	49917.0
8	Chad	TCD	148229.0
9	Chad	TCD	74299.0
10	Chad	TCD	7174.0
11	Chad	TCD	7637.0
12	Chad	TCD	39966.0
13	Chad	TCD	105822.0
14	Chad	TCD	25141.0

Results per page:

50

1 – 50 of 5080415

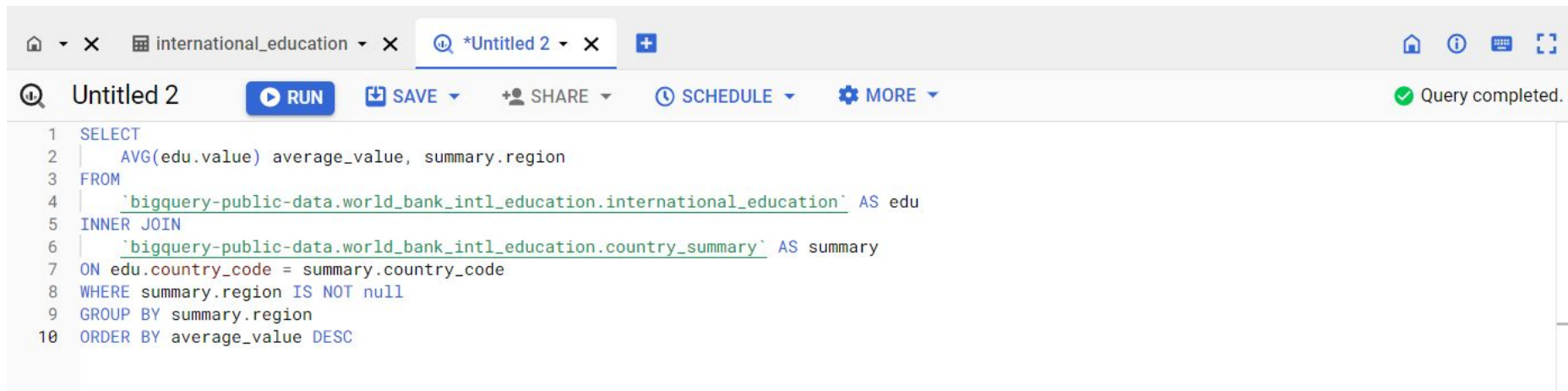


DATA QUESTION

What is the average amount of money spent per region on education?



I was able to answer the data question using this query..



The screenshot displays the Google BigQuery web interface. At the top, there's a navigation bar with tabs for 'international_education' and '*Untitled 2'. Below this, the main header shows 'Untitled 2' with a search icon, and action buttons for 'RUN', 'SAVE', 'SHARE', 'SCHEDULE', and 'MORE'. A green checkmark and the text 'Query completed.' are visible on the right. The SQL query is entered in the main text area, consisting of 10 lines of code that select the average value and region from two BigQuery public datasets, joining them on country code and ordering by average value in descending order.

```
1 SELECT
2 |     AVG(edu.value) average_value, summary.region
3 FROM
4 |     `bigquery-public-data.world_bank_intl_education.international_education` AS edu
5 INNER JOIN
6 |     `bigquery-public-data.world_bank_intl_education.country_summary` AS summary
7 ON edu.country_code = summary.country_code
8 WHERE summary.region IS NOT null
9 GROUP BY summary.region
10 ORDER BY average_value DESC
```

These were the results..

Query results

 SAVE RESULTS ▾

 EXPLORE DATA ▾



JOB INFORMATION

RESULTS

JSON

EXECUTION DETAILS

EXECUTION GRAPH

PREVIEW

Row	average_value ▾	region ▾
1	41653437607.78...	North America
2	3882406275.688...	East Asia & Pacific
3	2696534813.883...	South Asia
4	2374149029.045...	Europe & Central Asia
5	973477665.3314...	Middle East & North Africa
6	965999068.5475...	Latin America & Caribbean
7	211692028.7065...	Sub-Saharan Africa

An alias was also set to give the AVG(edu.value) a more descriptive name for the temporary table the query returned.

The WHERE statement also excluded any rows with null information.

CONCLUSION

In conclusion, the project aimed to determine the average amount of money spent per region on education. By employing aliases and excluding null values, valuable insights into regional spending patterns were obtained. The results contribute to a better understanding of education expenditures and showcase the effectiveness of the analysis techniques employed.

