

The JOIN command is used to bring together or concatenate data from two or more tables by matching their respective rows. The command uses the ON clause to determine what column will be used for the matching.

Subselect

The following query selects the average population for each country and their life expectancy for the year 2018. The data is joined from the 'midyear_population' table and the 'mortality_life_expectancy' table in the 'census_bureau_international' dataset. The query uses a subselect statement in the first FROM clause to filter by year and specific countries. The resulting table is grouped by country name and year and arranged in descending order. The general idea of a subselect statement is to be able to create more complex queries without using intermediate tables.

```
bq query --use_legacy_sql=false 'SELECT
  midyearpop.country_name AS country,
  midyearpop.year AS year,
  AVG(midyearpop.midyear_population) AS population,
  AVG(mortality.life_expectancy) AS life_expectancy
FROM (
  SELECT
    country_name,
    year,
    midyear_population
  FROM
    `bigquery-public-data.census_bureau_international.midyear_population`
  WHERE
    year = 2018
    AND (country_name LIKE "Nigeria"
    OR country_name LIKE "Egypt")) AS midyearpop
JOIN
  `bigquery-public-data.census_bureau_international.mortality_life_
  expectancy` AS mortality
```

```

ON
  midyearpop.country_name = mortality.country_name
GROUP BY
  country,
  year
ORDER BY
  population DESC
LIMIT
  20'

```

Waiting on bqjob_r5d381c26fcb6480e_0000016628e220c3_1 ... (0s) Current status: DONE

```

+-----+-----+-----+-----+
| country | year | population | life_expectancy |
+-----+-----+-----+-----+
| Nigeria | 2018 | 2.03452505E8 | 53.483061224489774 |
| Egypt   | 2018 | 9.9413317E7  | 73.8963636363636 |
+-----+-----+-----+-----+

```

The Case Against Running Select *

In BigQuery, it is ill-advised to run the `SELECT *` command, which is used in SQL to retrieve all the columns from the table. This command is rather expensive in BigQuery especially if your table contains terabytes of data. If instead you want to have a feel for the columns and their entries in your dataset, you can execute the command `'bq head [table_name]'` to retrieve the first few rows of the table. As an example, we used the command in the following example listing to retrieve the first few rows of the 'market' table we earlier loaded from GCS in the 'crypto_data' dataset.