```
| Vietnam          |   8.83786184736842E7 |
| Ethiopia         |  8.460339989473683E7 |
| Germany          |  8.168817173684208E7 |
| Egypt            |  8.064017099999999E7 |
| Iran             |  7.427240431578948E7 |
| Turkey           |  7.389499394736844E7 |
| Congo (Kinshasa) |   6.82958565263158E7 |
| Thailand         |  6.619103463157895E7 |
+------------------+----------------------+
```

In the preceding query, the fields retrieved using the SELECT command are passed through an aggregation function to give the average of the mid-year population for the years between 2000 and 2018 inclusive. In order to mix aggregated field and non-aggregated fields, we need the GROUP BY command to group the result by one or more columns, or else only a single result will be returned because of the aggregated function.

Joins

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 resulting table is grouped by country name and year and arranged in descending order.

```
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
  `bigquery-public-data.census_bureau_international.midyear_population` AS
  midyearpop
JOIN
  `bigquery-public-data.census_bureau_international.mortality_life_
  expectancy` AS mortality
ON
  midyearpop.country_name = mortality.country_name
```

```
WHERE
  midyearpop.year = 2018
GROUP BY
  country, year
ORDER BY
  population DESC
LIMIT
  20'

Waiting on bqjob_r4ecdb3f115b3f5d3_0000016628b526ea_1 ... (0s) Current
status: DONE
+------------------+------+--------------+--------------------+
|     country      | year |  population  |  life_expectancy   |
+------------------+------+--------------+--------------------+
| China            | 2018 | 1.384688986E9 |  75.58754098360653 |
| India            | 2018 | 1.296834042E9 |  69.15033333333334 |
| United States    | 2018 |  3.29256465E8 |  82.25324324324323 |
| Indonesia        | 2018 |  2.62787403E8 |  70.89647887323946 |
| Brazil           | 2018 |  2.08846892E8 |  71.26444444444446 |
| Pakistan         | 2018 |  2.07862518E8 |  66.57942857142856 |
| Nigeria          | 2018 |  2.03452505E8 | 53.483061224489774 |
| Bangladesh       | 2018 |  1.59453001E8 |  69.93685714285715 |
| Russia           | 2018 |  1.42122776E8 |  71.61112903225805 |
| Japan            | 2018 |  1.26168156E8 |   85.6562295081967 |
| Mexico           | 2018 |  1.25959205E8 |              75.22 |
| Ethiopia         | 2018 |  1.08386391E8 | 59.355633802816925 |
| Philippines      | 2018 |  1.05893381E8 |  69.13042253521127 |
| Egypt            | 2018 |   9.9413317E7 |   73.8963636363636 |
| Vietnam          | 2018 |   9.7040334E7 |   74.0014516129032 |
| Congo (Kinshasa) | 2018 |   8.5281024E7 | 56.483376623376614 |
| Iran             | 2018 |   8.3024745E7 |  72.58799999999997 |
| Turkey           | 2018 |   8.1257239E7 |  73.33577464788735 |
| Germany          | 2018 |   8.0457737E7 |  80.61900000000001 |
| Thailand         | 2018 |   6.8615858E7 |  75.35032786885246 |
+------------------+------+--------------+--------------------+
```

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
```