**Advanced Data Base**

**Assignment #1**

**Safa Shubbar**

**sshubbar@kent.edu**

**Please answer the following questions:**

1. **Which one is faster, Multidimensional OLAP or Relational OLAP? Why?**

Multidimensional OLAP is faster than Relational OLAP for these reasons:

|  |  |  |
| --- | --- | --- |
|  | **Multidimensional OLAP** | **Relational OLAP** |
| 1 | * Excellent performance: MOLAP cubes are built for fast data retrieval, and are optimal for slicing and dicing operations. | Performance can be slow because of large size of data sets |
| 2 | Uses sparse array to store data-sets. | Uses relational table. |
| 3 | MOLAP is best suited for inexperienced users, since it is very easy to use. | ROLAP is best suited for experienced users. |
| 4 | Maintains a separate database for data cubes. | It may not require space other than available in the Data warehouse. |
| 5 | DBMS facility is weak. | DBMS facility is strong. |

1. **What is the difference between view and materialized view?**

|  |  |  |
| --- | --- | --- |
|  | **View** | **Materialized view** |
| 1 | uses a SQL query to pull data from the underlying tables, takes the output of a query and makes it appear like a virtual table | A materialized view is a table on disk that contains the result set of a query |
| 2 | It doesn't store anything | Stores data physically and get updated periodically. |

**Referring to google cloud documentation about “Visualizing BigQuery Data in a Jupyter Notebook” (**[**https://cloud.google.com/bigquery/docs/visualize-jupyter**](https://cloud.google.com/bigquery/docs/visualize-jupyter)**). Still use the natality dataset same as the example. And write query and plot the following data.**

1. **Chart of new baby births number of top five high states with the birth number.**
2. **Histogram of mother’s age while giving birth between 1998-2008.**
3. **Average of father’s age for the new birth over the years (1998-2008).**
4. **Chart of new baby births number of top five high states with the birth number.**

%load\_ext google.cloud.bigquery

%%bigquery top\_five\_states

SELECT

state AS state,

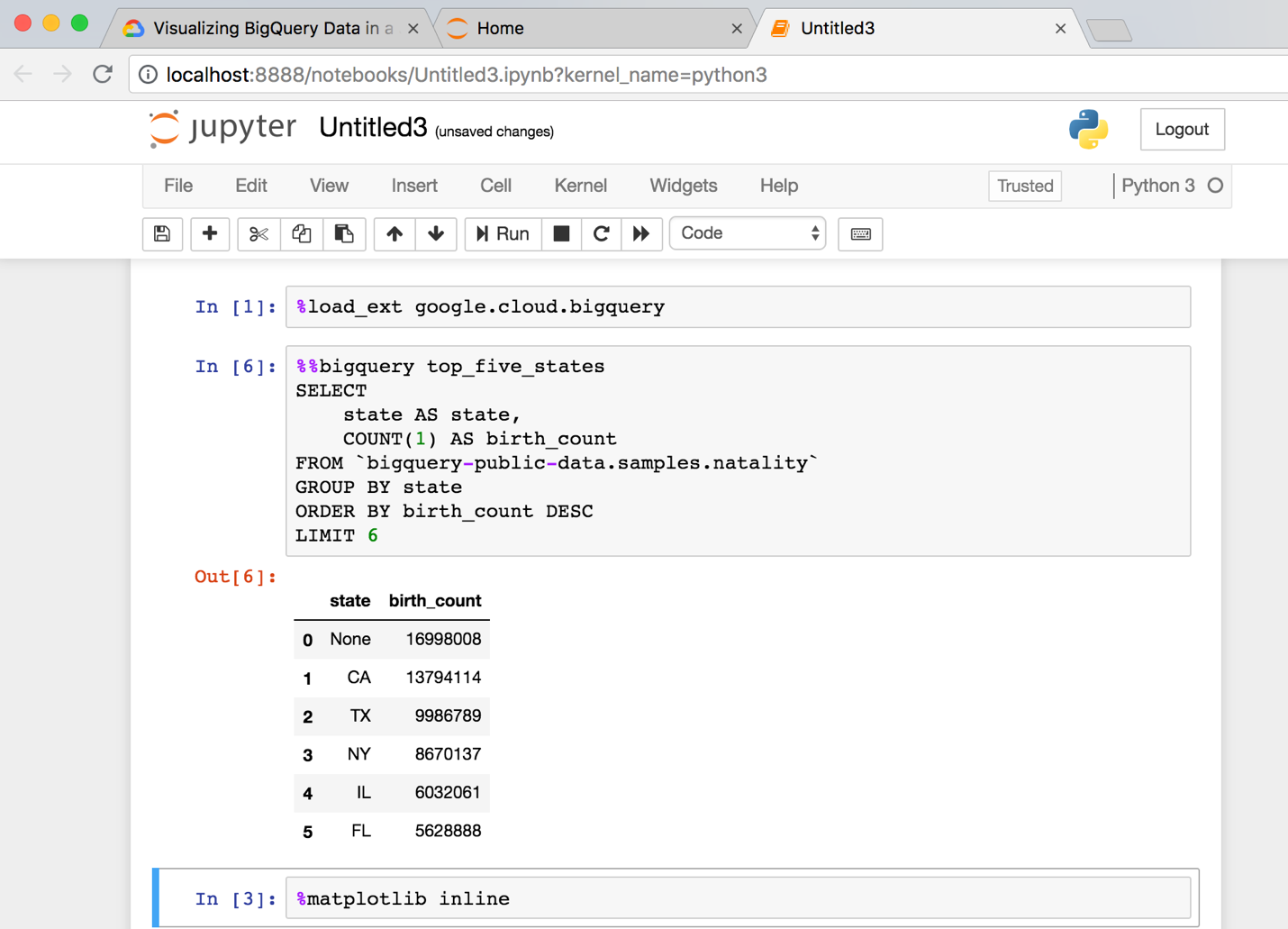
COUNT(1) AS birth\_count

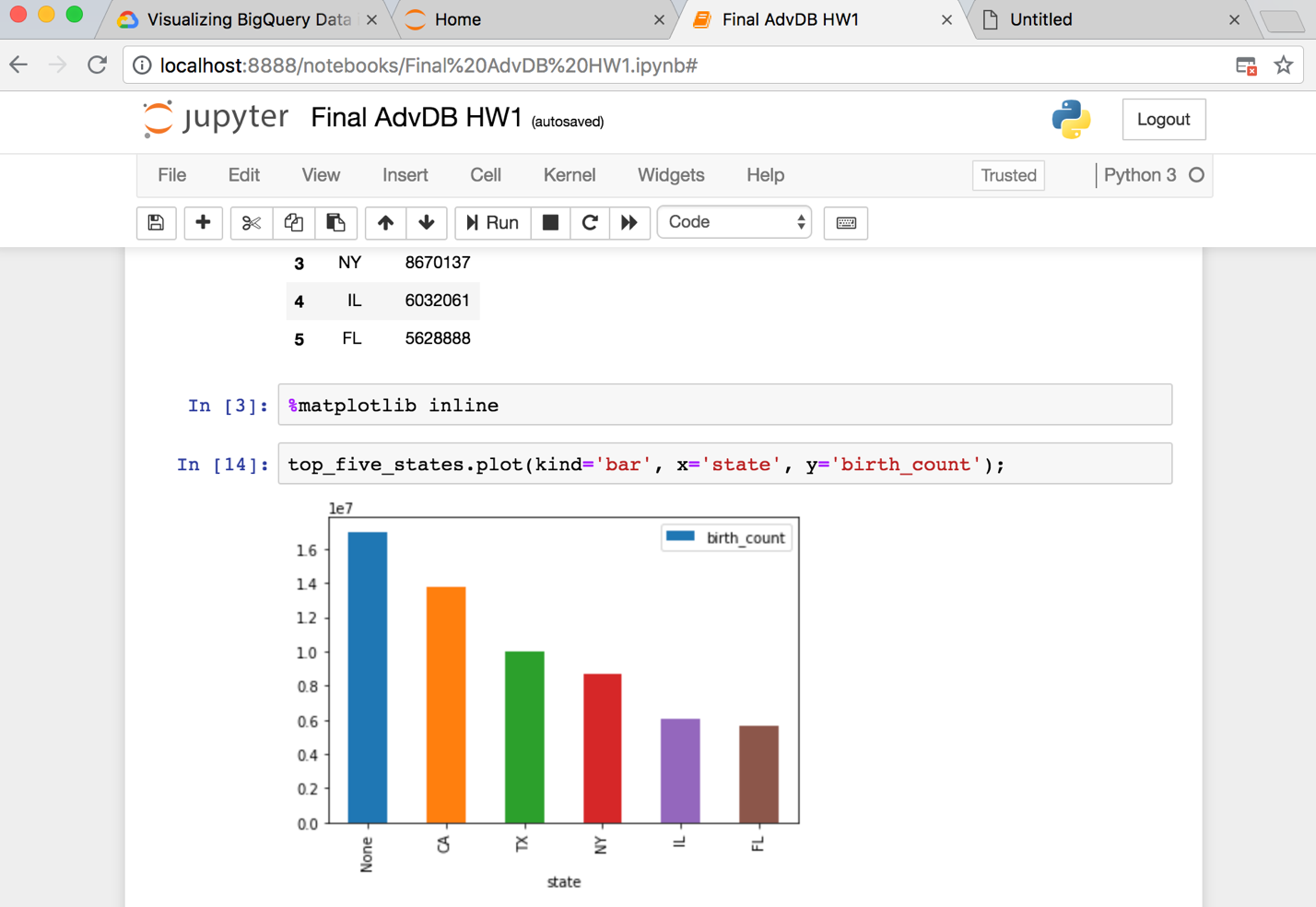
FROM `bigquery-public-data.samples.natality`

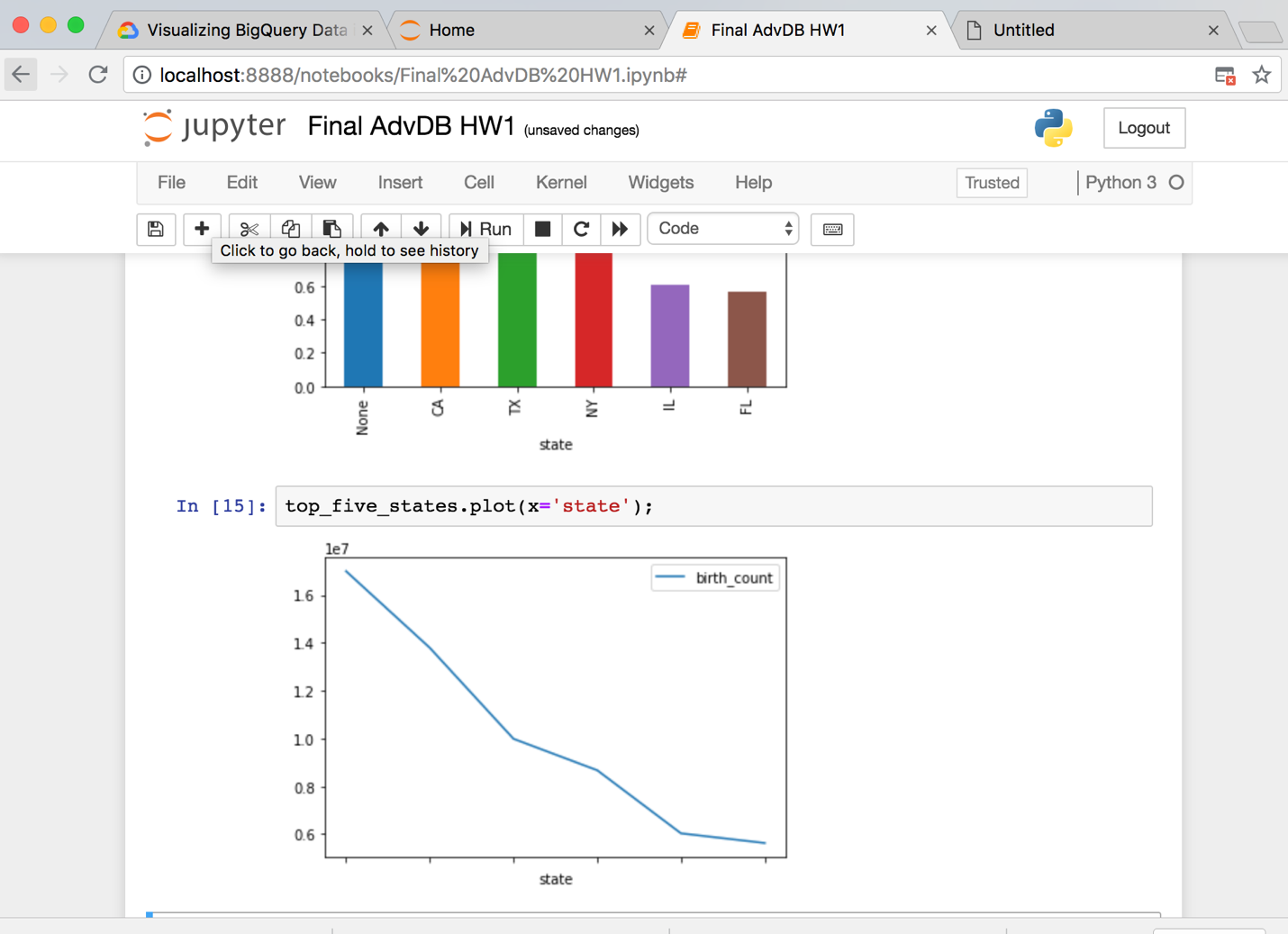
GROUP BY state

ORDER BY birth\_count DESC

LIMIT 6







%%bigquery top\_five\_states\_with\_Ohio

SELECT

state AS state, birth\_count from (SELECT state,

COUNT(1) AS birth\_count

FROM `bigquery-public-data.samples.natality`

GROUP BY state

ORDER BY birth\_count DESC

LIMIT 7)

UNION ALL

SELECT

state,

COUNT(1) AS birth\_count

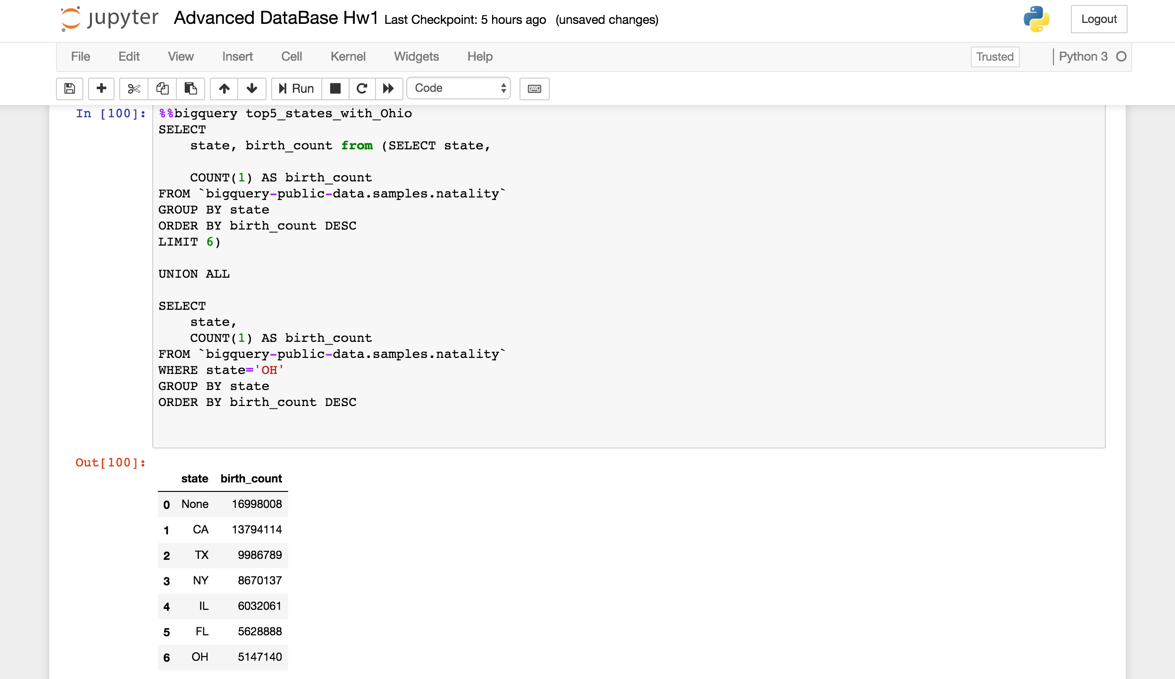
FROM `bigquery-public-data.samples.natality`

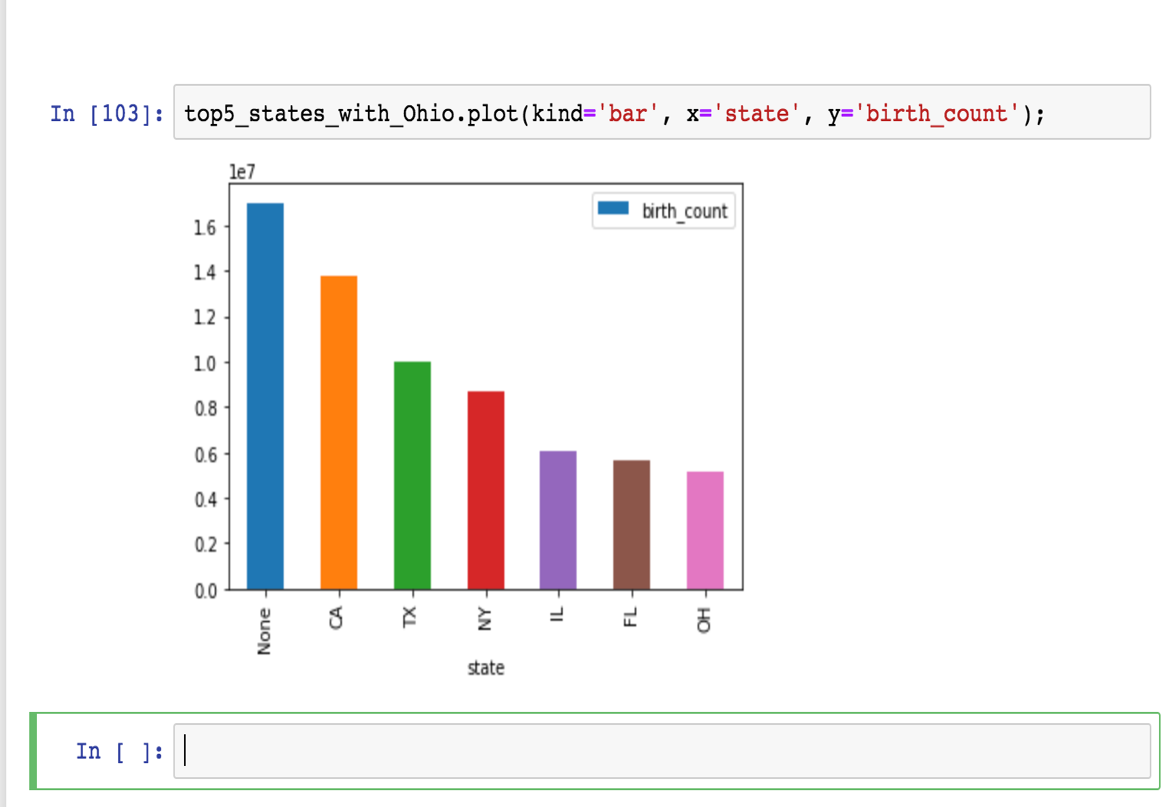
WHERE state='OH'

GROUP BY state

ORDER BY birth\_count DESC

LIMIT 7





1. **Histogram of mother’s age while giving birth between 1998-2008.**

from google.cloud import bigquery

client = bigquery.Client()

sql = """

SELECT

mother\_age,

COUNT(1) AS count,

year

FROM

`bigquery-public-data.samples.natality`

WHERE

year>1997 AND year < 2009

GROUP BY

mother\_age, year

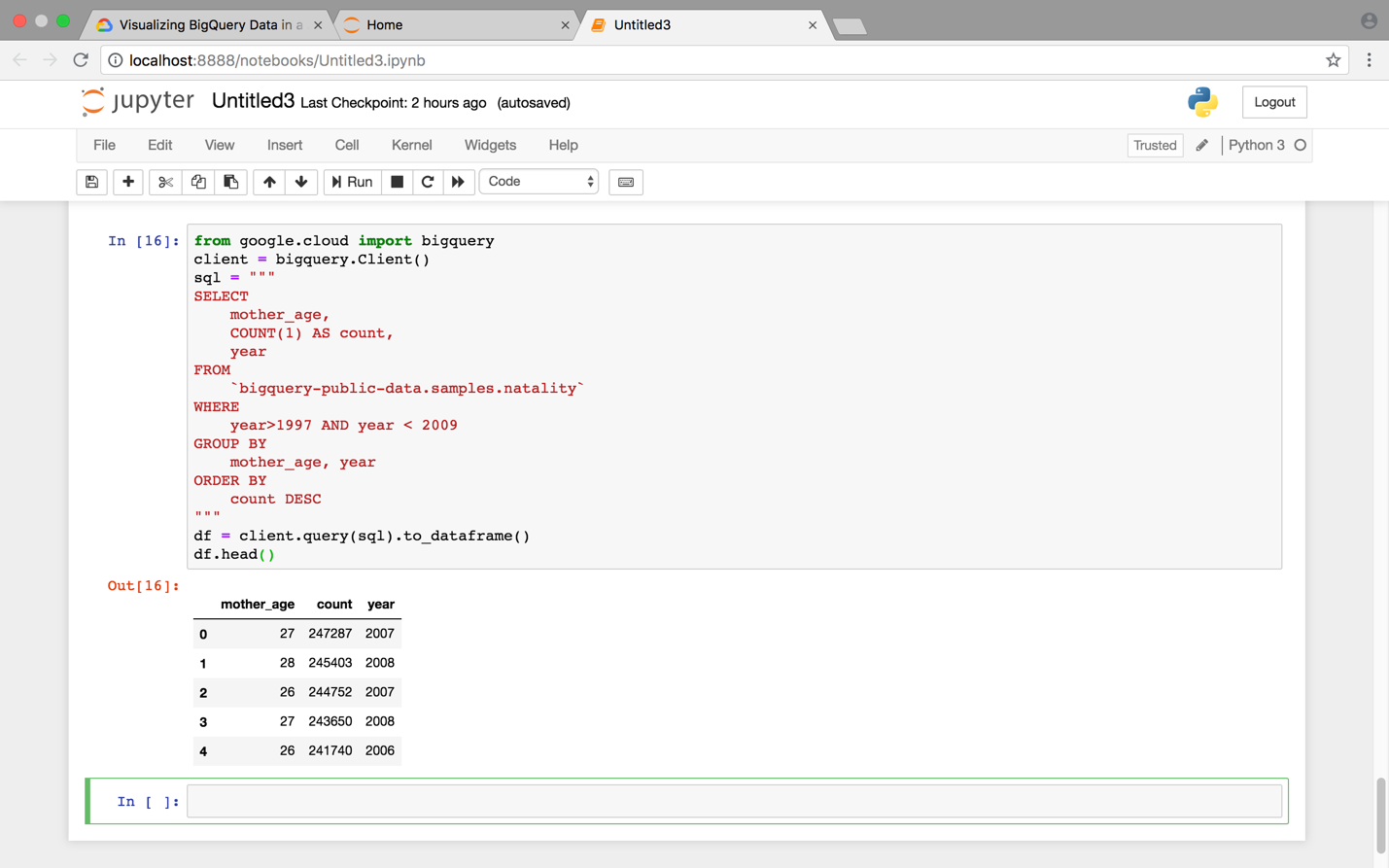
ORDER BY

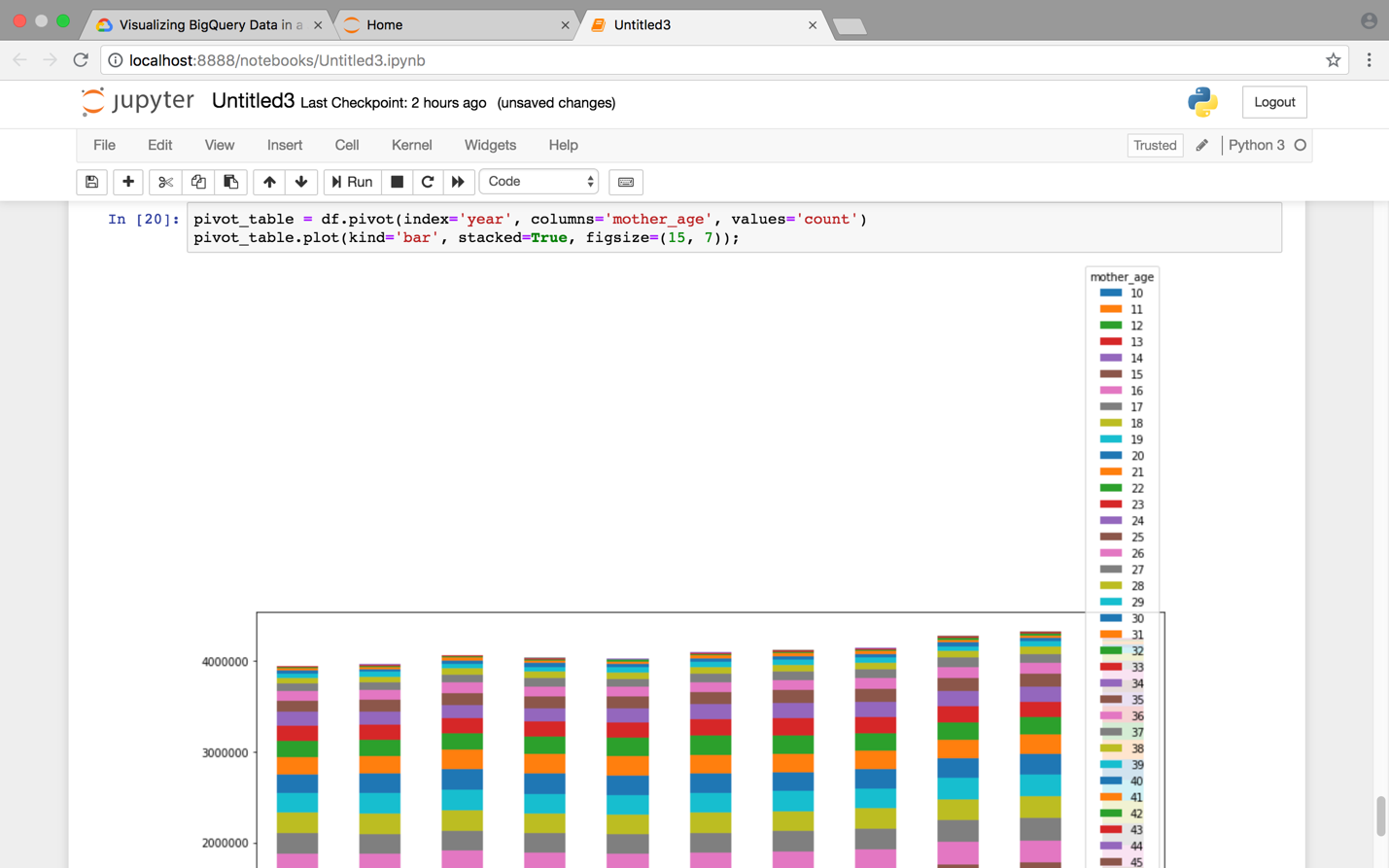
count DESC

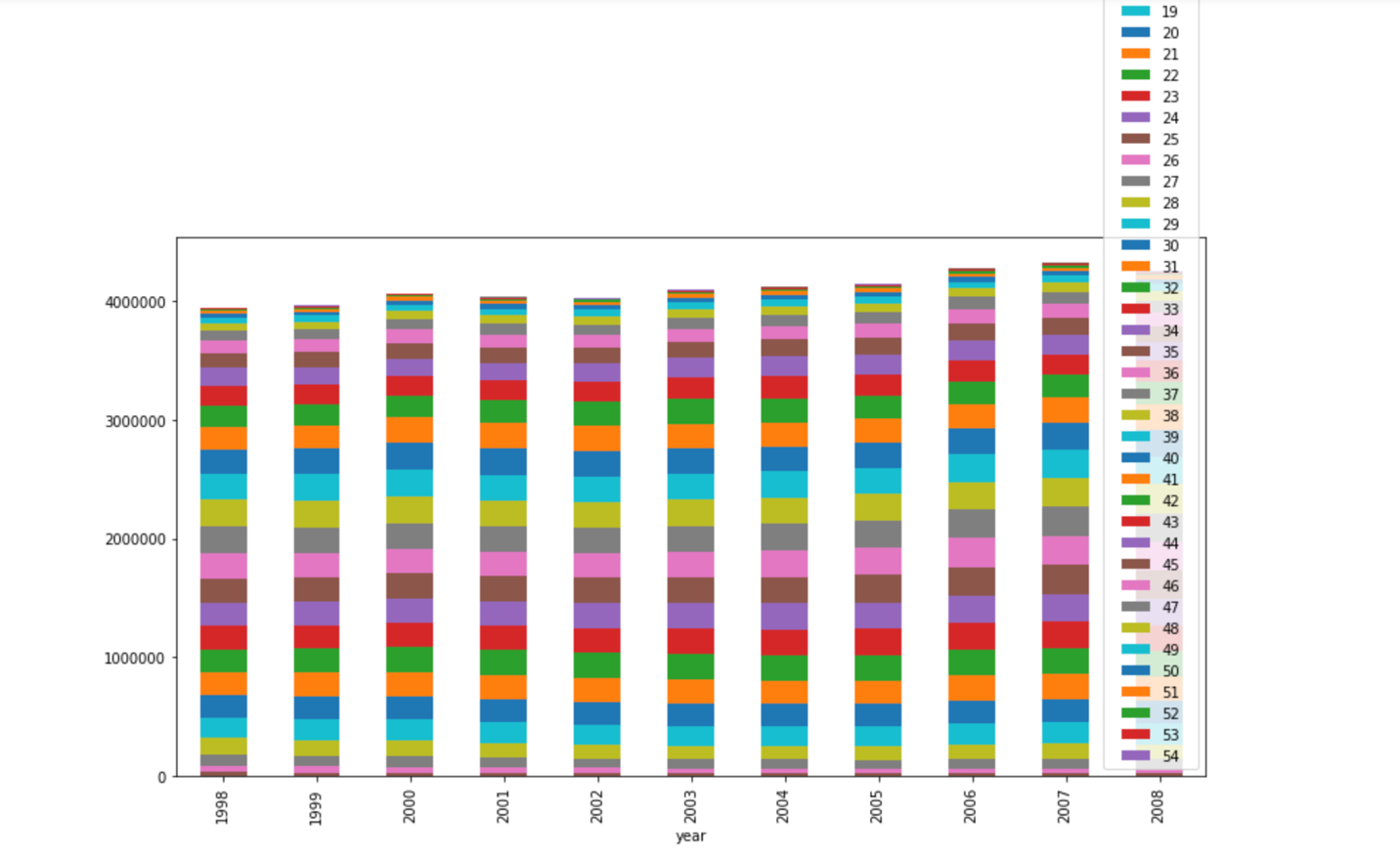
"""

df = client.query(sql).to\_dataframe()

df.head()







1. **Average of father’s age for the new birth over the years (1998-2008).**

%%bigquery father\_age

SELECT

year,

AVG(father\_age) AS AVG\_father\_age

FROM

`bigquery-public-data.samples.natality`

WHERE

year>1997 AND year < 2009

GROUP BY year

ORDER BY

year DESC

LIMIT 11

