**ETL Project: TEAM 10**

Project Description

The goal of the project was to build the database of “confirmed” planets – exoplanets across galaxies and explore their attributes.

In addition to this, we made an attempt to explore how they were discovered: what telescope was used, and which observatory contributed into the discovery.

1. **Extract:** sources of data the information was extracted from:
2. The Extrasolar Planets Encyclopaedia: <http://exoplanet.eu/catalog/>

* We have uploaded the main catalog, listing the names of “confirmed planets” as the csv file
* Key file attributes: 98 columns, 4302 rows

2. NASA Exoplanet Archive: <https://exoplanetarchive.ipac.caltech.edu/cgi-bin/TblView/nph-tblView?app=ExoTbls&config=planets>

* Uploaded the file as .csv file
* Key file attributes: 4,197 records, 256 columns

3. List of major observatories: <https://en.wikipedia.org/wiki/List_of_astronomical_observatories>

- uploaded html table to pandas, cleansed the data and uploaded directly to Postgres SQL database

4. List of large optical telescopes: <https://en.wikipedia.org/wiki/List_of_large_optical_telescopes>

- uploaded html table to pandas, cleansed the data and uploaded directly to Postgres SQL database

1. **T**ransform: what data cleaning or transformation was required

We decided to create 3 tables associated with the following objects:

1. **Planets** and their attribute (primary key – planet id)
2. **Observatories:** major observatories, including, but not limited to the ones that participated in discovery of exoplanets
3. Telescopes: list of major optical telescopes, including ones participated in the discovery (bummer: the data turned out to be in quite different form from the *planet* files!)
4. **Planet** table:

* we read with Jupiter notebook “The Extrasolar Planets Encyclopedia” and “Nasa Archive” .csv files and created 2 pd.dataframes (sources #1 and #2)
* Most of the cleaning involved dropping unnecessary columns as well as columns that mostly consisted of null values. After organizing the columns, we renamed them to make it more readable as well as have a common column name to join the exoplanet dataframe to the planet dataframe. We then repeated this cleaning process with the Nasa planet dataframe. Once they were both cleaned, we joined the two datasets together, and then exported the combined dataframe as a csv to then upload to sql postgres.

- we ended up with the 25-column .csv file – output of Jupiter notebook. The file was added to git repo

* Csv out file: [planets\_exoplanets.csv](https://github.com/LanaPal/ETL_Team_10/blob/master/ETL_Project/planets_exoplanets.csv)
* Jupiter notebook file used for data cleansing and prepping: [planet\_exoplanet\_data.ipynb](https://github.com/LanaPal/ETL_Team_10/blob/master/ETL_Project/planet_exoplanet_data.ipynb)

1. **Observatories** table:

* we used a Wikipedia html page, containing numerous tables as a source
* scraped and merged 26(!) tables with the Jupiter notebook
* renamed the columns and uploaded directly to postgres database
* just in case (for QA and if we would choose to import data to the database as csv, the csv file was created (as an output of Jupiter notebook)
* both files are located in “jupiter\_prep\_file” folder: [observ\_file.ipynb](https://github.com/LanaPal/ETL_Team_10/blob/master/ETL_Project/jupiter_prep_file/observ_file.ipynb) (Jupiter) and [list\_of\_observatories.csv](https://github.com/LanaPal/ETL_Team_10/blob/master/ETL_Project/jupiter_prep_file/list_of_observatories.csv) (csv output file)

1. **Telescopes** table:
2. we used a Wikipedia html page, containing numerous tables as a source
3. scraped and merged 3 tables with the Jupiter notebook
4. a lot of manipulations were required, since the table from one page turned out to be in a different format (different # of columns). To avoid losing data, we had to split the “Aperture Column” for the first table into meters and inches measurements, drop and rename some columns, etc.
5. just in case (for QA and if we would choose to import data to the database as csv, the csv file was created (as an output of Jupiter notebook)
6. both files are located in “jupiter\_prep\_file” folder: [tele\_file\_prep\_final.ipynb](https://github.com/LanaPal/ETL_Team_10/blob/master/ETL_Project/jupiter_prep_file/tele_file_prep_final.ipynb) (Jupiter) and [list\_of\_telescopes.csv](https://github.com/LanaPal/ETL_Team_10/blob/master/ETL_Project/jupiter_prep_file/list_of_telescopes.csv) (csv output file)
7. **Load:** the final database, tables/collections, and why this was chosen

We decided on the following database structure (the picture below). The database was expected to be “relational” – connected by the telescopes and observatory id.

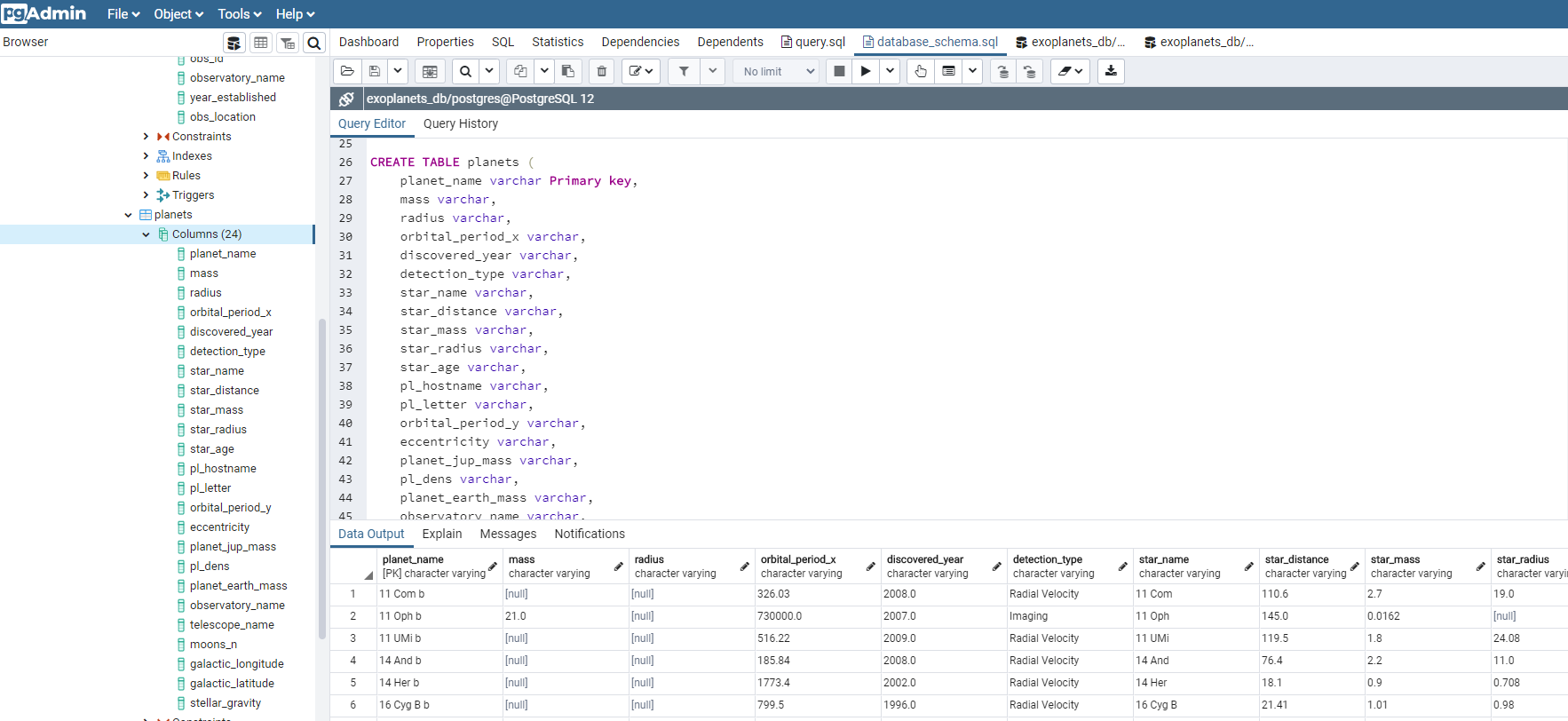
A screenshot of a computer

Description automatically generated

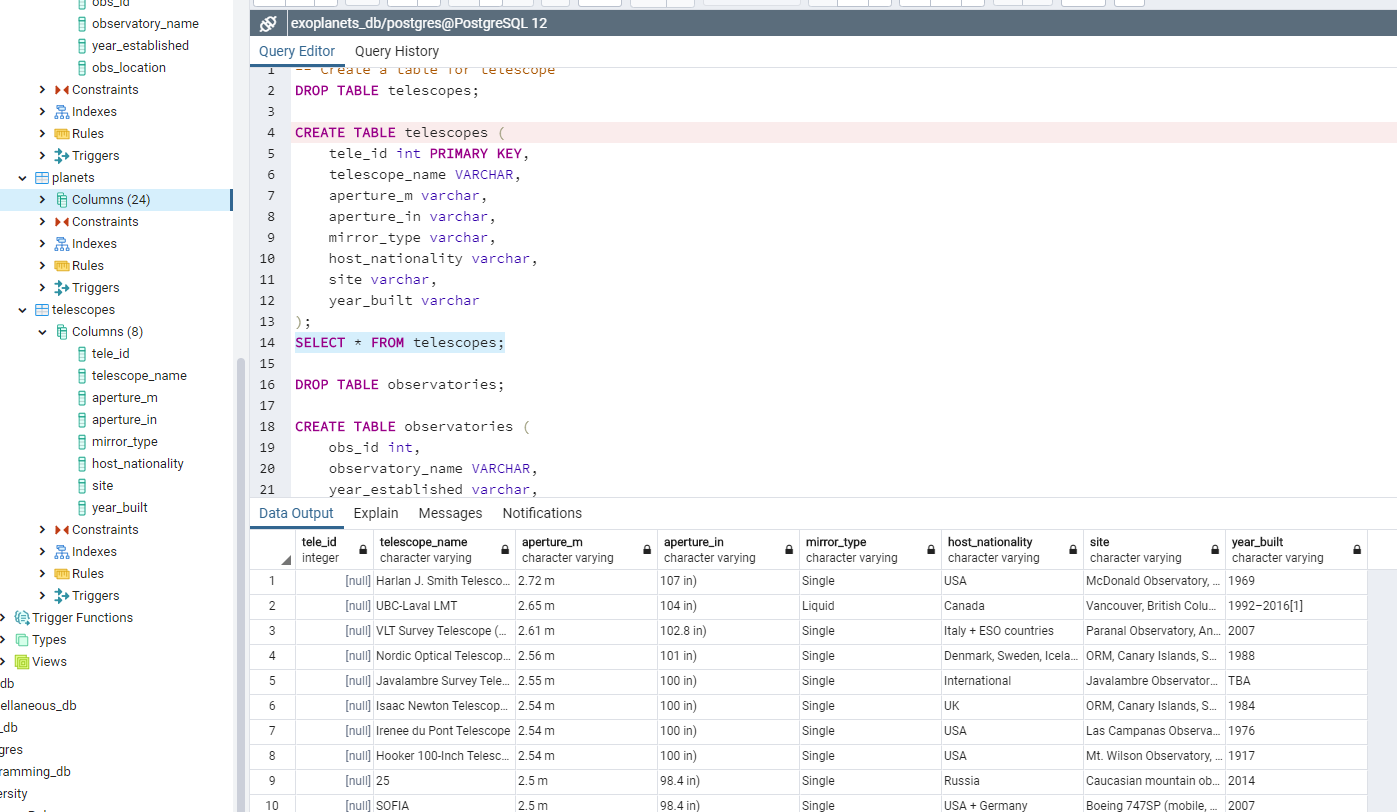
Loading method:

1. The schema of tables was created (see the schema file: P\_schema -> [database\_schema.sql](https://github.com/LanaPal/ETL_Team_10/blob/master/ETL_Project/P_schema/database_schema.sql). The key challenge was associated with creating the ids / primary keys, since the panda df numeration starts from 0, and the primary key is supposed to be a non-null value
2. For the Planet table, we uploaded data from csv file
3. Data for Observatories and Telescopes table were uploaded directly from Jupiter notebook.

Screenshot of the **Planets** table:



Screenshot of the **Telescopes** table:



Screenshot of the **Observatories** table:

