Data Analytics Bootcamp

ETL Project

Mens and Womens Shoe Database Creation

Introduction

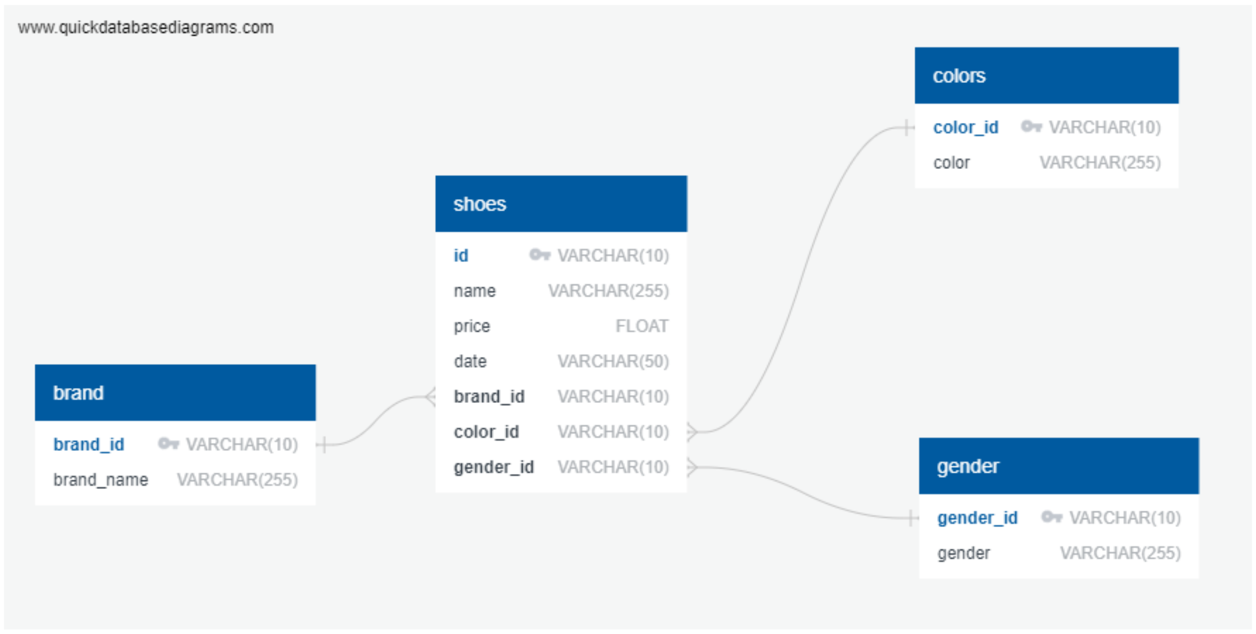
Our team was tasked to Extract, Transform and Load a number of disparate datasets into a single database. As part of the process, we needed to clean the data and get it ready for use in a production database, and be able to query the data.

We found two datasets on Kaggle.com, relating to the sale of mens and womens shoes over a period of time. We thought this would be appropriate data for our project because

- The two datasets could be related to each other (i.e. mens and womens shoes have similar properties - sizes, prices, colours, brands etc)
- There was a lot of data in the datasets - the Mens dataset had over 19,000 rows, and the Women's dataset had approximately 10,000 rows
- The data looked to be relatively dirty and required substantial cleaning prior to loading into the database
- Some of the data in the datasets looked to be redundant, and we thought it would be good practice for us to transform the redundant data into relational data for SQL.

Design

Our Team went through the available datasets and decided on a resulting database schema that looked as follows:



Data Cleaning

We downloaded CSV files from Kaggle.com and set to work. We initially had three datasets - 1x Mens Shoes dataset, and 2x Womens Shoes dataset. The first Womens Shoes dataset had a high level of redundant data, once duplicates had been removed only 653 of 10,000 records remained, so we switched to the other Womens dataset.

We loaded the CSV files into Pandas, and our team figured out what cleaning tasks were required, and drafted a basic schema for our database.

Our Data Cleaning tasks were divided between the three of us, and were as follows:

- Search the Name column for items that contain shoe-like entries, and drop rows that do not contain a shoe-like word in their name (e.g. boot, sandal, heel, etc). This removed 6504 rows from the Mens Shoes dataset, and 1921 rows from the Women's Shoes dataset.
- Remove all rows that had a price not listed in US Dollars. This removed 751 rows from the Men's Shoes dataset.
- Filter out shoes that are on Sale (where the isSale column = False). At the 11th hour we found a bug in our code that meant this cleaning aspect did not work correctly, however we do not think it substantially affected the quality of our final dataset, as we chose the highest price of a given item anyway.
- Sort or Group the shoes by their Shoe ID, and filter out duplicate entries. To make the decision on which shoes to filter out, we chose to get the MAX price for each Shoe ID. This removed 1179 rows from the Men's dataset.
- Delimit the Colour column, and keep the first entry only. Many shoes had colour entries like "Brown,Tan,Gold,White,Pearl" for example. We decided it was too complex to retain all this information, and just chose to keep a single colour. Rows without a colour entry were dropped. This removed 2261 rows from the Men's dataset.
- Date Read-in. The "Date" columns had an unusual syntax, for example "2017-01-09T20:25:29Z". Some coding was required to clean the date strings up, to make them readable by DateTime. Dates that could not

be adequately read were removed. This removed 2153 rows from the Women's Shoes dataset.

Overall, our datasets were reduced substantially. We were all surprised by the amount of data that got dropped.

- The Men's Shoes Dataset was reduced from 19,315 rows, down to 3015 rows, an 85% reduction.
- The Women's Shoes Dataset was reduced from 10,000 rows, down to 5698 rows, a 43% reduction.

One of the outcomes of our Data Cleaning was a decision that we were searching for the highest RRP price of a listed shoe out of the dataset, if there were multiple entries for the same item. Another decision from our cleaning process was to reduce the complexity regarding Shoe Colours, and have only one colour assigned to a particular Shoe.

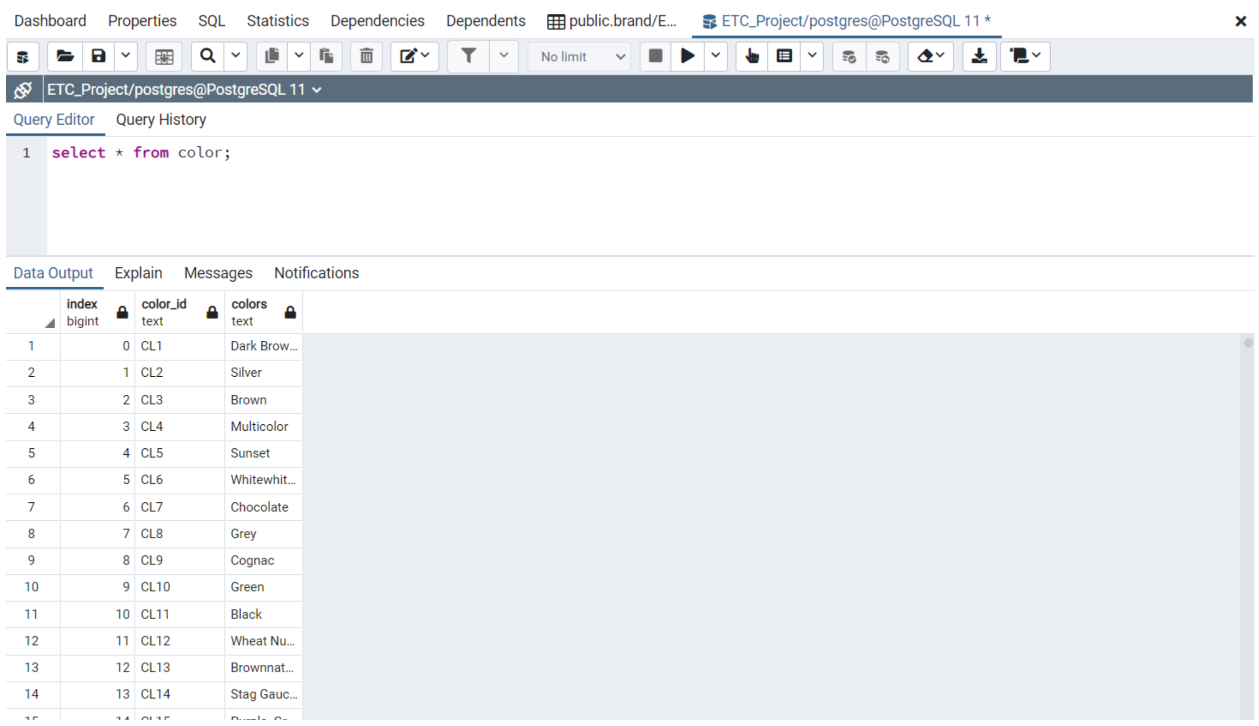
Final Clean & Load into PostgreSQL

Prior to loading into Postgres, we ran a final cleanup. This did some tasks such as:

- Setting Title Case on all the Brands and Colours. This reduced the overall brand and colour count by about 10%.
- Setting Lower Case on the column names and table names
- Renaming the price column for easier referencing in queries

We used SQLAlchemy to create a connection to the PostgreSQL database, and uploaded our data into the database. There were possibly some bugs in our upload code, as the resulting tables contained an index number column which may not have been intended.

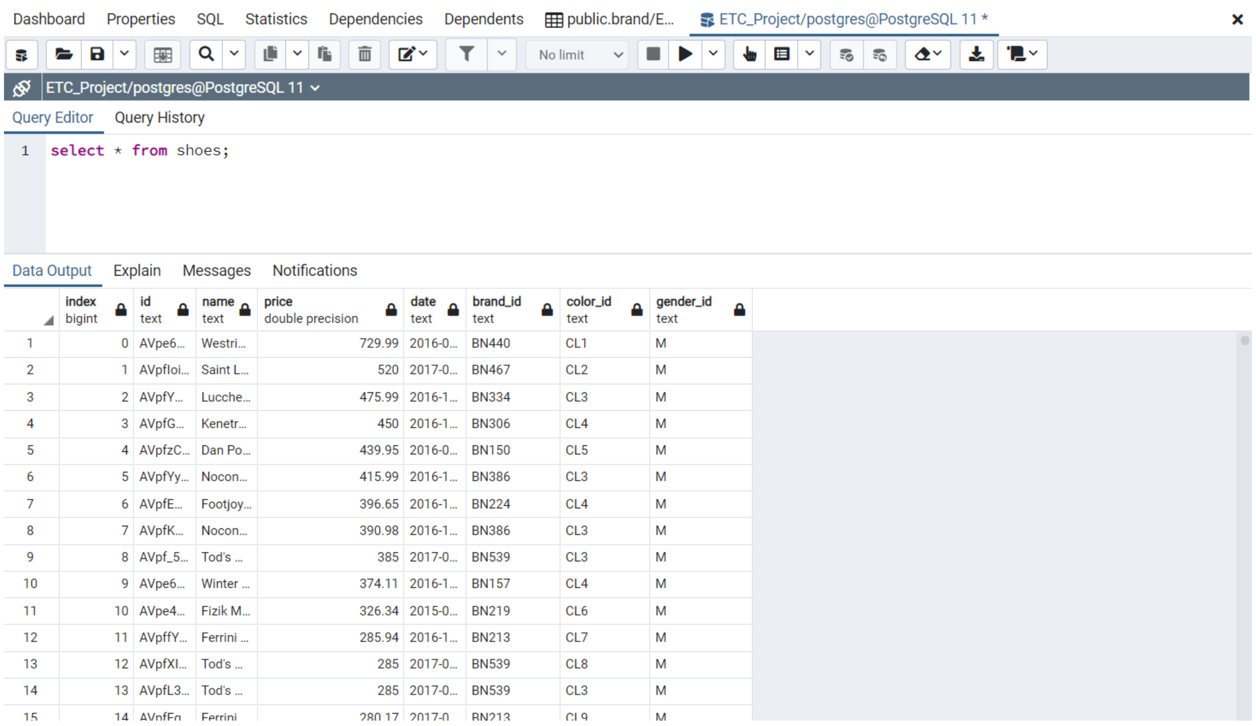
Color Table



```
select * from color;
```

	index bigint	color_id text	colors text
1	0	CL1	Dark Brow...
2	1	CL2	Silver
3	2	CL3	Brown
4	3	CL4	Multicolor
5	4	CL5	Sunset
6	5	CL6	Whitewhit...
7	6	CL7	Chocolate
8	7	CL8	Grey
9	8	CL9	Cognac
10	9	CL10	Green
11	10	CL11	Black
12	11	CL12	Wheat Nu...
13	12	CL13	Brownnat...
14	13	CL14	Stag Gauc...

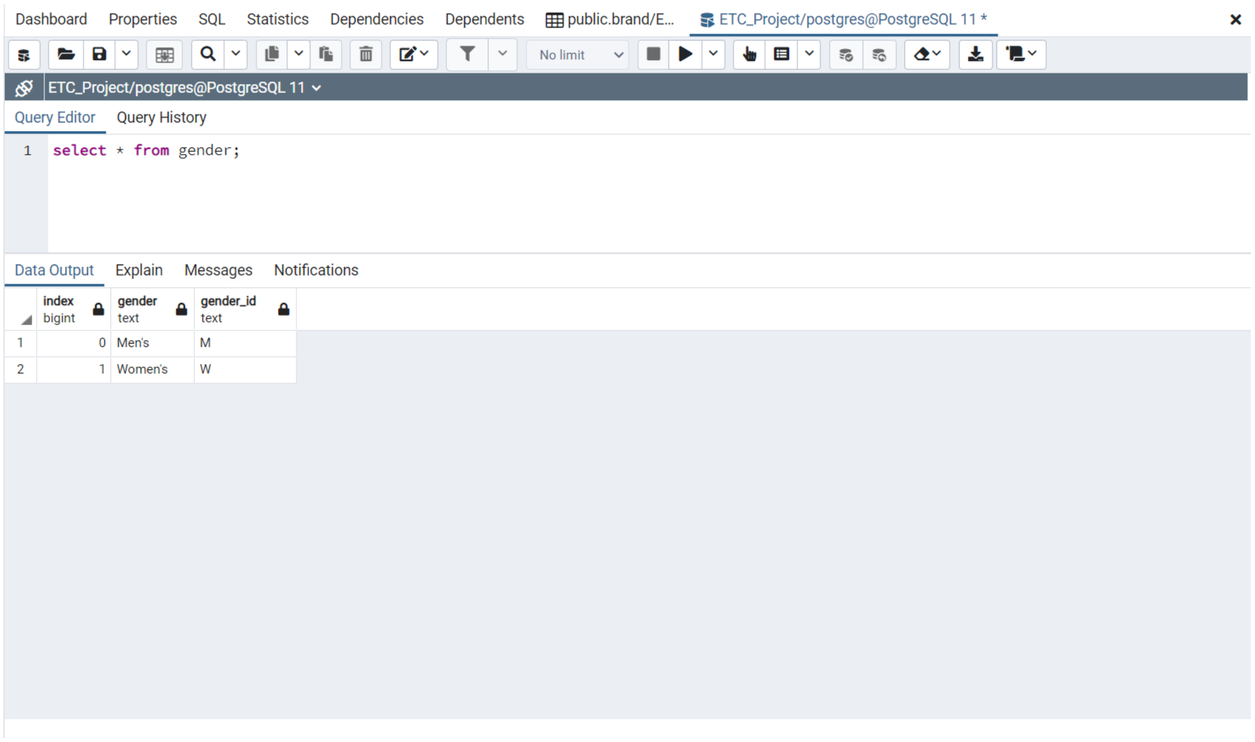
Shoes Table



```sql
select * from shoes;
```

	index bigint	id text	name text	price double precision	date text	brand_id text	color_id text	gender_id text
1	0	AVpe6...	Westri...	729.99	2016-0...	BN440	CL1	M
2	1	AVpfloi...	Saint L...	520	2017-0...	BN467	CL2	M
3	2	AVpfY...	Lucche...	475.99	2016-1...	BN334	CL3	M
4	3	AVpfG...	Kenetr...	450	2016-1...	BN306	CL4	M
5	4	AVpfzC...	Dan Po...	439.95	2016-0...	BN150	CL5	M
6	5	AVpfYy...	Nocon...	415.99	2016-1...	BN386	CL3	M
7	6	AVpfE...	Footjoy...	396.65	2016-1...	BN224	CL4	M
8	7	AVpfK...	Nocon...	390.98	2016-1...	BN386	CL3	M
9	8	AVpf_5...	Tod's ...	385	2017-0...	BN539	CL3	M
10	9	AVpe6...	Winter ...	374.11	2016-1...	BN157	CL4	M
11	10	AVpe4...	Fizik M...	326.34	2015-0...	BN219	CL6	M
12	11	AVpffY...	Ferrini ...	285.94	2016-1...	BN213	CL7	M
13	12	AVpfXI...	Tod's ...	285	2017-0...	BN539	CL8	M
14	13	AVpfL3...	Tod's ...	285	2017-0...	BN539	CL3	M
15	14	AVpfFq	Ferrini	280.17	2017-0	BN213	CL9	M

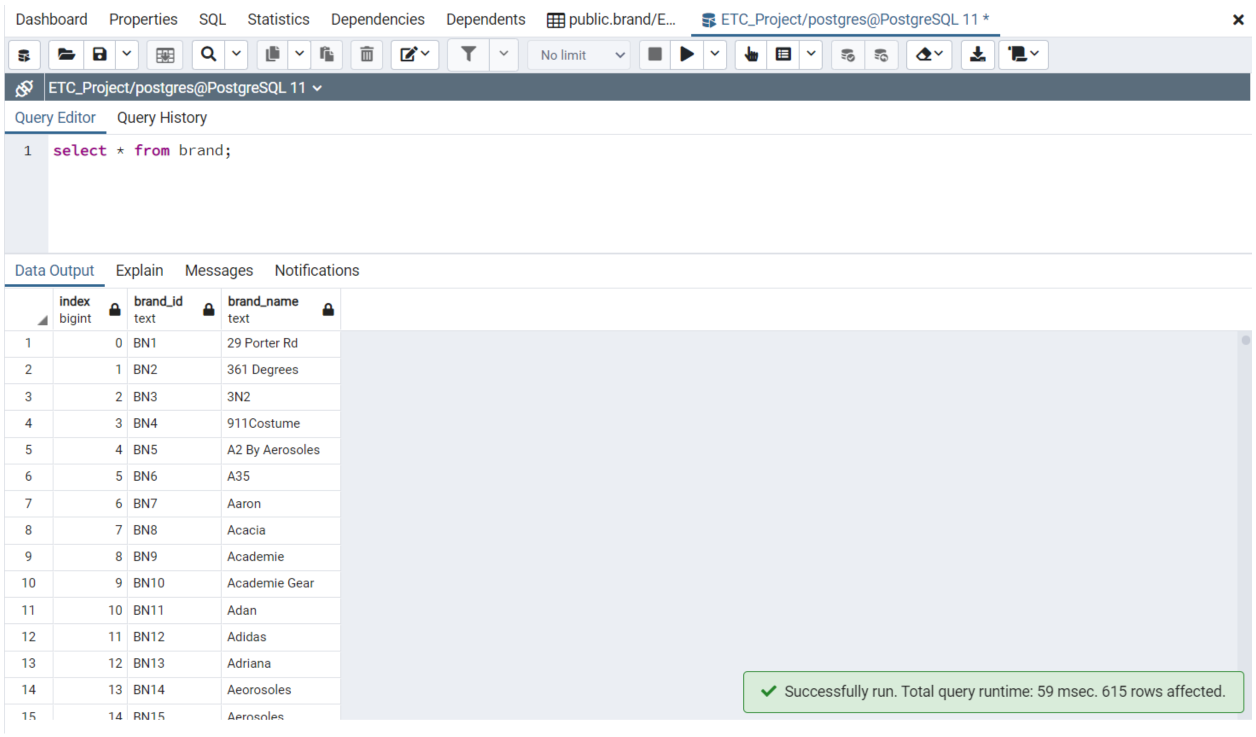
Gender table:



```sql
select * from gender;
```

	index bigint	gender text	gender_id text
1	0	Men's	M
2	1	Women's	W

Brand table



Dashboard Properties SQL Statistics Dependencies Dependents public.brand/E... ETC_Project/postgres@PostgreSQL 11 *

ETC_Project/postgres@PostgreSQL 11

Query Editor Query History

```
select * from brand;
```

Data Output Explain Messages Notifications

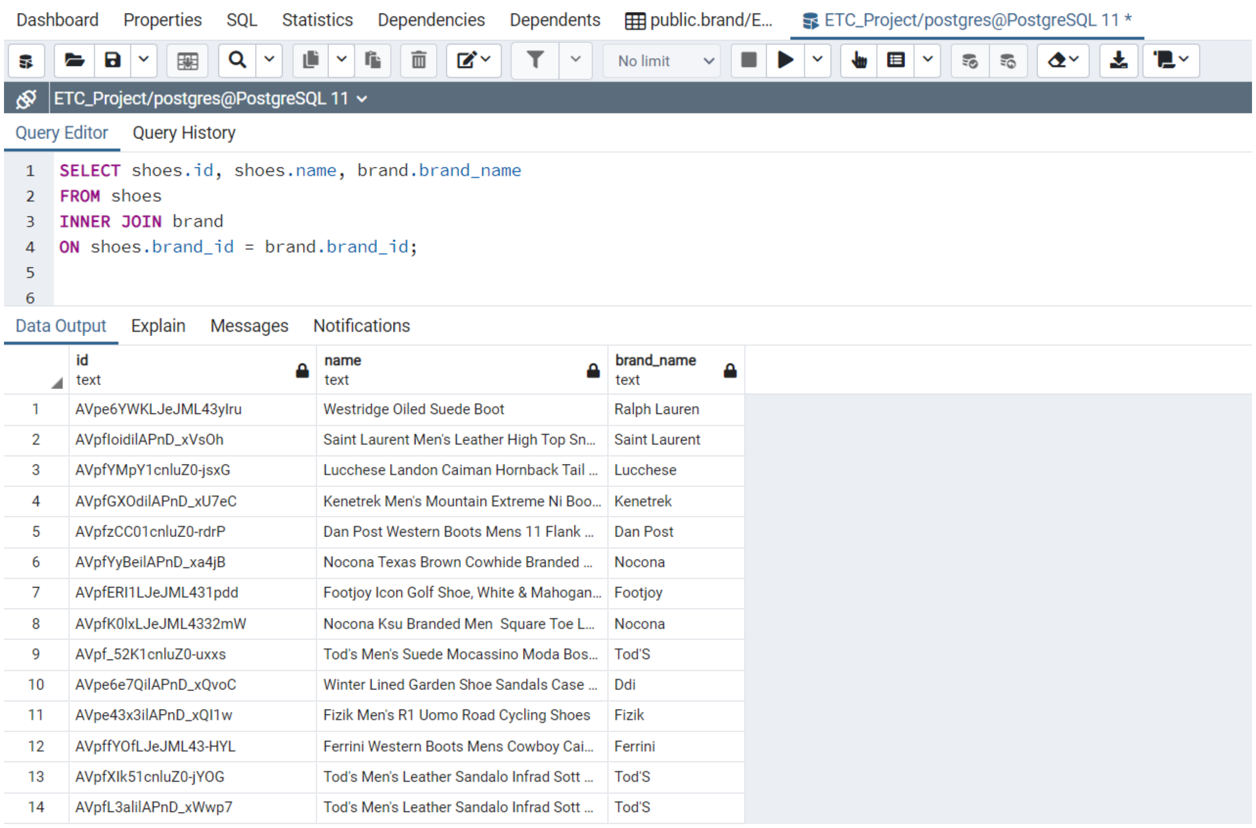
	index bigint	brand_id text	brand_name text
1	0	BN1	29 Porter Rd
2	1	BN2	361 Degrees
3	2	BN3	3N2
4	3	BN4	911Costume
5	4	BN5	A2 By Aerosoles
6	5	BN6	A35
7	6	BN7	Aaron
8	7	BN8	Acacia
9	8	BN9	Academie
10	9	BN10	Academie Gear
11	10	BN11	Adan
12	11	BN12	Adidas
13	12	BN13	Adriana
14	13	BN14	Aeorosoles
15	14	BN15	Aerosoles

Successfully run. Total query runtime: 59 msec. 615 rows affected.

Queries and Analysis

We ran a few queries on the database to check it, as follows.

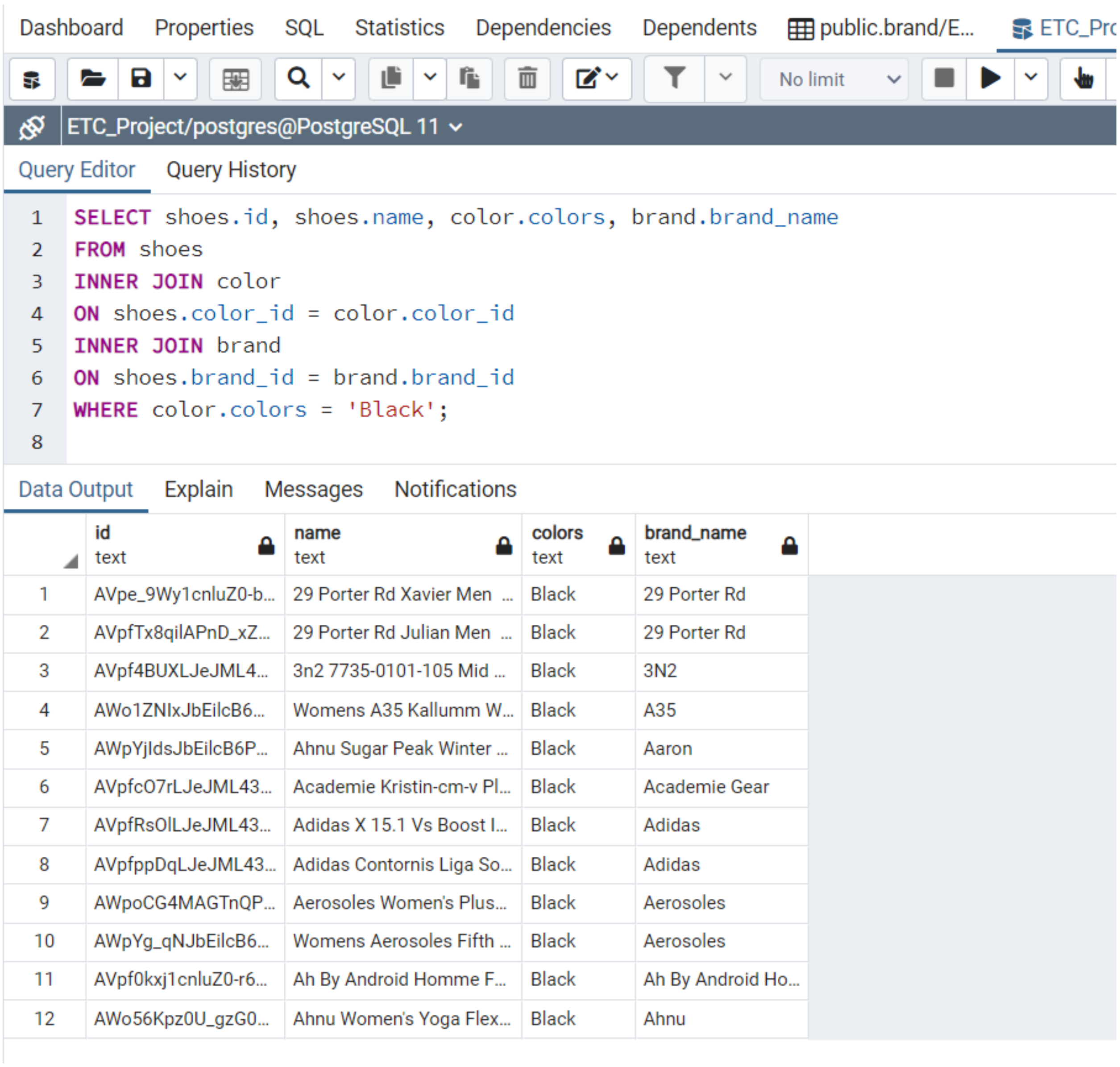
Join shoes and brand table and display all the shoes id, name and their brand name



```sql
SELECT shoes.id, shoes.name, brand.brand_name
FROM shoes
INNER JOIN brand
ON shoes.brand_id = brand.brand_id;
```

	id text	name text	brand_name text
1	AVpe6YWKLJeJML43ylru	Westridge Oiled Suede Boot	Ralph Lauren
2	AVpfloidilAPnD_xVsOh	Saint Laurent Men's Leather High Top Sn...	Saint Laurent
3	AVpfYMpY1cnluZ0-jsxG	Lucchese Landon Caiman Hornback Tail ...	Lucchese
4	AVpfGXOdilAPnD_xU7eC	Kenetrek Men's Mountain Extreme Ni Boo...	Kenetrek
5	AVpfzCC01cnluZ0-rdrP	Dan Post Western Boots Mens 11 Flank ...	Dan Post
6	AVpfYyBeilAPnD_xa4jB	Nocona Texas Brown Cowhide Branded ...	Nocona
7	AVpfERI1LJeJML431pdd	Footjoy Icon Golf Shoe, White & Mahogan...	Footjoy
8	AVpfK0lxLJeJML4332mW	Nocona Ksu Branded Men Square Toe L...	Nocona
9	AVpf_52K1cnluZ0-uxxs	Tod's Men's Suede Mocassino Moda Bos...	Tod'S
10	AVpe6e7QilAPnD_xQvoC	Winter Lined Garden Shoe Sandals Case ...	Ddi
11	AVpe43x3ilAPnD_xQl1w	Fizik Men's R1 Uomo Road Cycling Shoes	Fizik
12	AVpffYOfLJeJML43-HYL	Ferrini Western Boots Mens Cowboy Cai...	Ferrini
13	AVpfXIk51cnluZ0-jYOG	Tod's Men's Leather Sandalo Infrad Sott ...	Tod'S
14	AVpfL3alilAPnD_xWwp7	Tod's Men's Leather Sandalo Infrad Sott ...	Tod'S

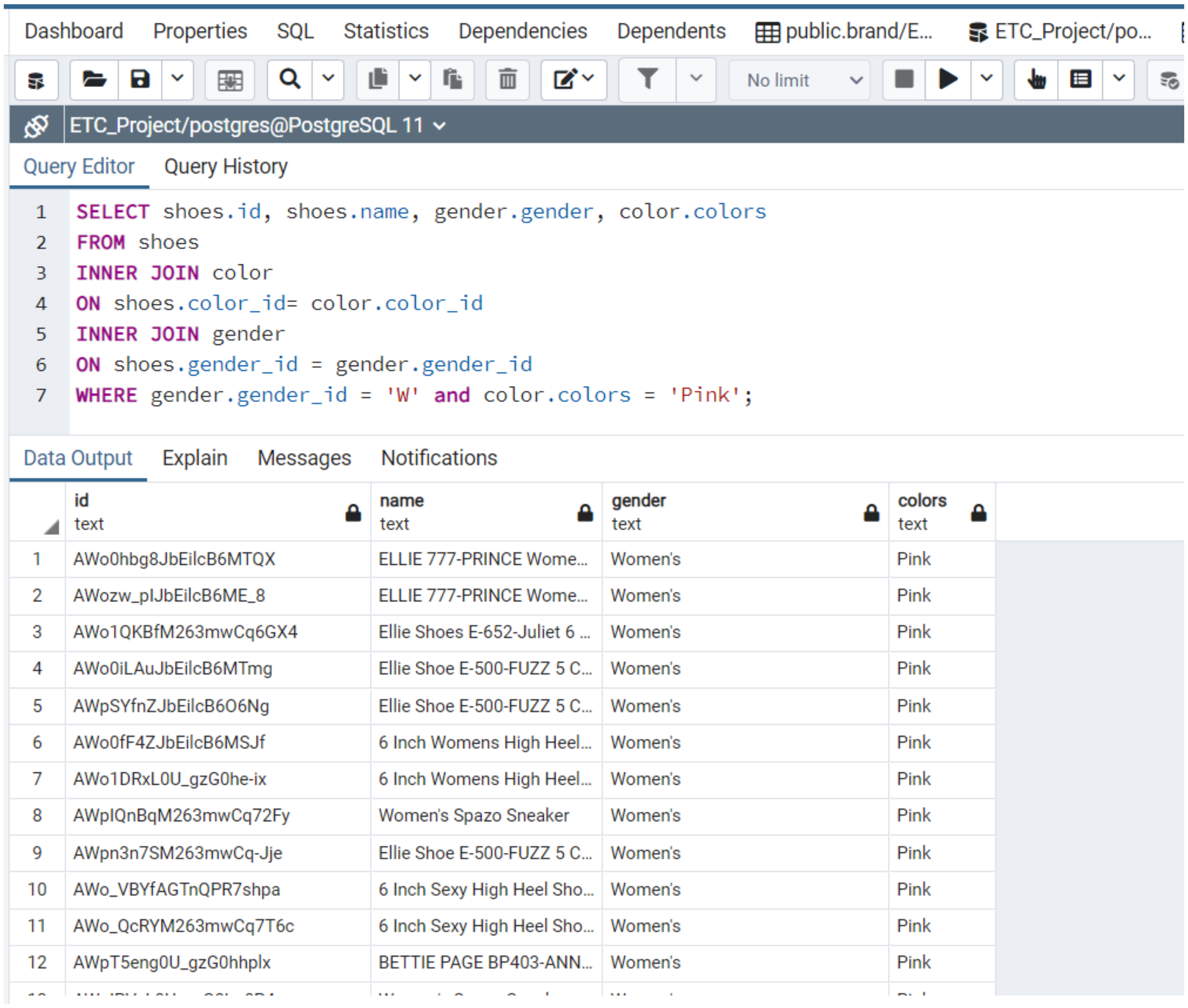
Join shoe, brand and color table, display shoe id, name, color and brand_name for all shoes with 'Black' color.



```sql
SELECT shoes.id, shoes.name, color.colors, brand.brand_name
FROM shoes
INNER JOIN color
ON shoes.color_id = color.color_id
INNER JOIN brand
ON shoes.brand_id = brand.brand_id
WHERE color.colors = 'Black';
```

	id text	name text	colors text	brand_name text
1	AVpe_9Wy1cnluZ0-b...	29 Porter Rd Xavier Men ...	Black	29 Porter Rd
2	AVpfTx8qilAPnD_xZ...	29 Porter Rd Julian Men ...	Black	29 Porter Rd
3	AVpf4BUXLJeJML4...	3n2 7735-0101-105 Mid ...	Black	3N2
4	AWo1ZNIxJbEilcB6...	Womens A35 Kallumm W...	Black	A35
5	AWpYjldsJbEilcB6P...	Ahnu Sugar Peak Winter ...	Black	Aaron
6	AVpfcO7rLJeJML43...	Academie Kristin-cm-v Pl...	Black	Academie Gear
7	AVpfRsOlLJeJML43...	Adidas X 15.1 Vs Boost I...	Black	Adidas
8	AVpfppDqLJeJML43...	Adidas Contornis Liga So...	Black	Adidas
9	AWpoCG4MAGTnQP...	Aerosoles Women's Plus...	Black	Aerosoles
10	AWpYg_qNJbEilcB6...	Womens Aerosoles Fifth ...	Black	Aerosoles
11	AVpf0kxj1cnluZ0-r6...	Ah By Android Homme F...	Black	Ah By Android Ho...
12	AWo56Kpz0U_gzG0...	Ahnu Women's Yoga Flex...	Black	Ahnu

Display only Women's shoes and color 'Pink'.



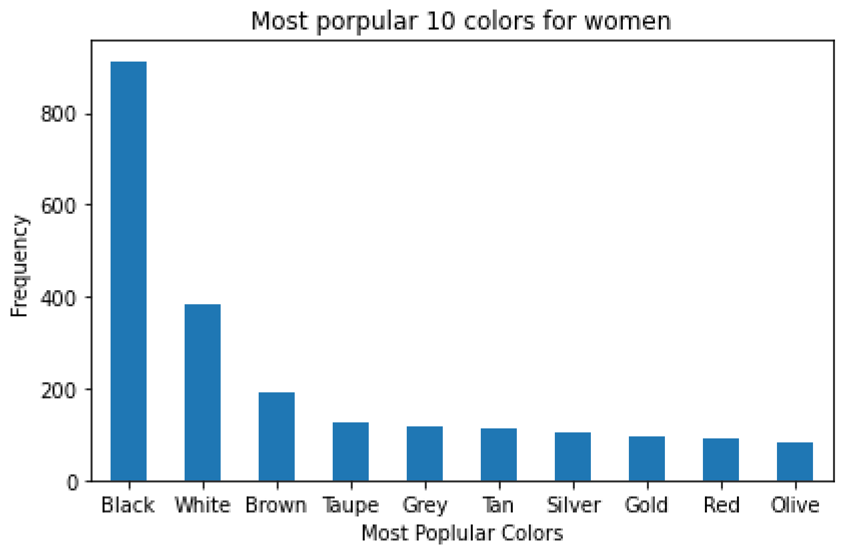
```
SELECT shoes.id, shoes.name, gender.gender, color.colors
FROM shoes
INNER JOIN color
ON shoes.color_id= color.color_id
INNER JOIN gender
ON shoes.gender_id = gender.gender_id
WHERE gender.gender_id = 'W' and color.colors = 'Pink';
```

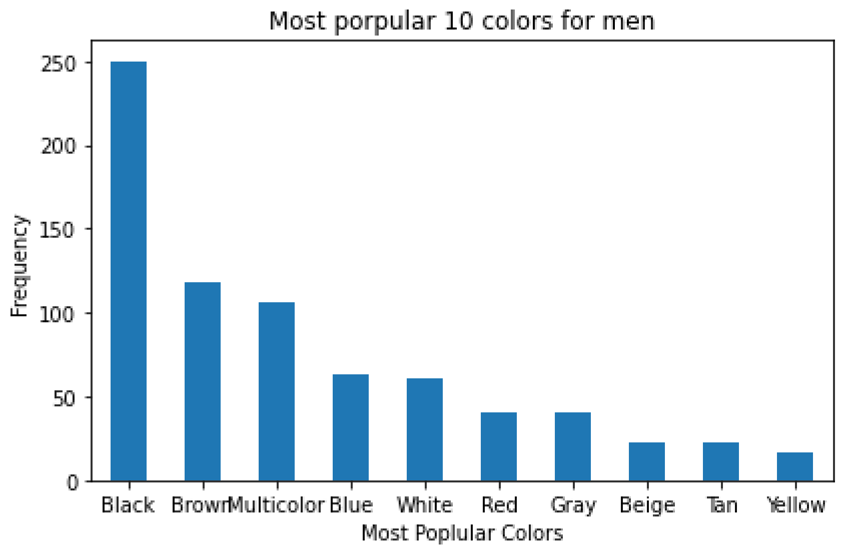
	id text	name text	gender text	colors text
1	AWo0hbg8JbEilcB6MTQX	ELLIE 777-PRINCE Wome...	Women's	Pink
2	AWozw_plJbEilcB6ME_8	ELLIE 777-PRINCE Wome...	Women's	Pink
3	AWo1QKBfM263mwCq6GX4	Ellie Shoes E-652-Juliet 6 ...	Women's	Pink
4	AWo0iLAuJbEilcB6MTmg	Ellie Shoe E-500-FUZZ 5 C...	Women's	Pink
5	AWpSYfnZJbEilcB6O6Ng	Ellie Shoe E-500-FUZZ 5 C...	Women's	Pink
6	AWo0fF4ZJbEilcB6MSJf	6 Inch Womens High Heel...	Women's	Pink
7	AWo1DRxL0U_gzG0he-ix	6 Inch Womens High Heel...	Women's	Pink
8	AWplQnBqM263mwCq72Fy	Women's Spazo Sneaker	Women's	Pink
9	AWpn3n7SM263mwCq-Jje	Ellie Shoe E-500-FUZZ 5 C...	Women's	Pink
10	AWo_VBYfAGTnQPR7shpa	6 Inch Sexy High Heel Sho...	Women's	Pink
11	AWo_QcRYM263mwCq7T6c	6 Inch Sexy High Heel Sho...	Women's	Pink
12	AWpT5eng0U_gzG0hhplx	BETTIE PAGE BP403-ANN...	Women's	Pink

We have also imported the database from SQL back to pandas and completed some analysis.

Most popular shoe colors for men and women in bar chart

(SQL_Plot.ipynb)





Project documentary

ETLProject_Combined_All.ipynb

Run this file to:

- ➢ Import from csv to python
- ➢ Clean up men's and women's shoe data separately, and save to separate csv file
- ➢ Create 3 reference tables: color, gender, brand
- ➢ Combine the 2 dataset
- ➢ Store in PostgressSQL:
    - ❖ There is no need to run a schema before importing
    - ❖ A config.py contain Postgres password is required

SQL_Plot.ipynb

Run this file to:

- ➢ Export tables from PostgresSQL and save as pandas dataframe
- ➢ Analyse data using pandas
- ➢ Plot bar charts

REFERENCES

Women’s Shoe Prices

https://www.kaggle.com/datafiniti/womens-shoes-prices

Men’s Shoe Prices

https://www.kaggle.com/datafiniti/mens-shoe-prices