**project proposal:**

Our group set out to create a user-friendly tool/database to address the need of consumers to identify the different ingredients that make up the personal hygiene/beauty products that we as consumers use daily. Whether the consumer has sensitive skin, dry skin, oily hair, allergies to a specific ingredient or simply wants to search by price or peer recommendations, our tool seeks to empower the user to be able to navigate the various product offerings with the confidence that this tool has done the homework for them and has narrowed the products to those that fit the customer’s needs.

**Initial project set up:**

When we visualized what we wanted to accomplish with the project and what the timeline was, we initially set out to obtain various variables (skin type, top grossing products, etc.) on a wider set of products/product categories (mascara, eye brows etc.), on two websites (sephora.com and ulta.com). Given our time constraints however, we decided to downscale the project to a narrower product and variable set. Importantly, however given the way that we set up the code, the project can easily be modified and scaled to extract any additional products and variables that the user may need/want by changing the search list.

|  |  |  |
| --- | --- | --- |
| **Initial targeted product categories:** |  | **Scaled down product categories:** |
| **Products:** |  | **Products:** |
| **Face:** |  | **Face:** |
| Foundation |  | Foundation |
| Blush |  | Blush |
| Contour |  | **Eyes:** |
| Highlight |  | Eyeshadow |
| **Eyes:** |  | Eyeliner |
| Eyeshadow |  | **Skin Care** |
| Mascara |  | Moisturizer |
| Eyeliner |  | Cleanser |
| Eyebrows |  | **Hair Care** |
| **Skin Care** |  | Shampoo |
| Moisturizer |  | Conditioner |
| Cleanser |  |  |
| Mask |  |  |
| **Hair Care** |  |  |
| Shampoo |  |  |
| Conditioner |  |  |
| Hair Treatments |  |  |

|  |  |  |
| --- | --- | --- |
| **Variables to keep out for:** |  | **What we’re web scraping:** |
| Type of makeup |  | Product Name |
| Ingredients |  | Product Brand |
| Skin type (oily, dry, normal, combo) |  | Rating |
| Skin fairness |  | Price |
| Top grossing products |  | Details |
| Highest rated |  | Ingredients |
| Hair type |  |  |
| Reviews |  |  |
| Pricing |  |  |

Given the quick turnaround (five days from start to finish) our timeline was compressed. As a group we agreed that all the web-scrapping needed to be done by Monday (day - 2), all the data transformation completed by Tuesday (day - 3), to go live and complete documentation by Wednesday (day - 4), and on Thursday analyze, submit and present our findings to the class.

**Technical write up of project:**

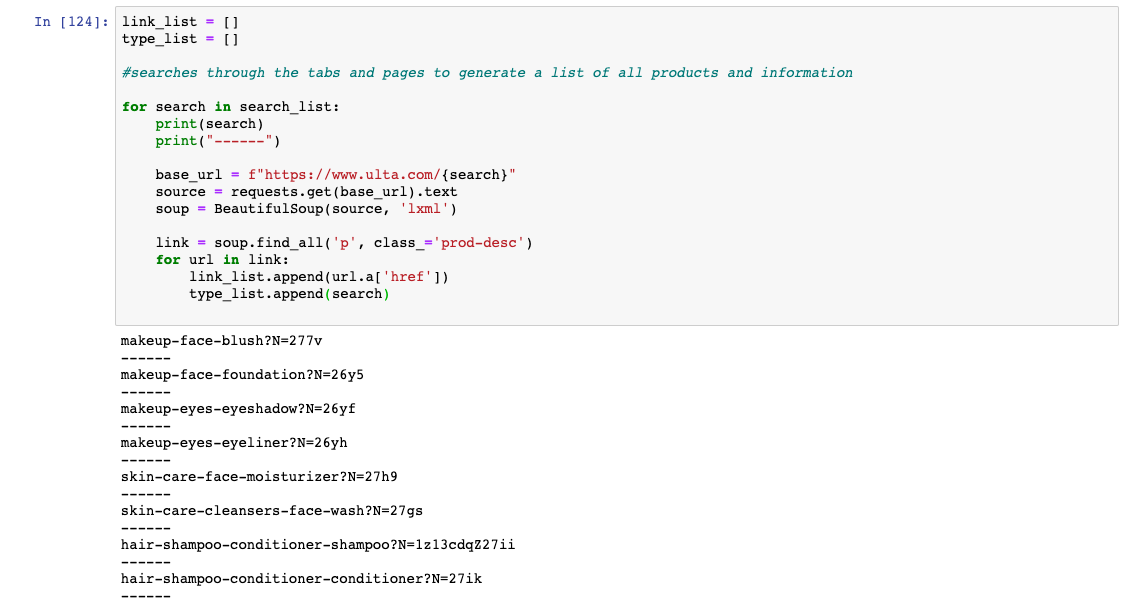
With the end project result in mind of being able to make product recommendations to consumers based on their preferences. The group set out to get the most relevant and up to date product information in order to accomplish this goal, next we transformed the data and later loaded to data in a usable format by the end consumer. We followed the Extract Transform Load (ETL) methodology. One can find a more detailed explanation on each step below, but in brief, we extracted the data through web scrapping from [www.sephora.com](http://www.sephora.com) and [www.ulta.com](http://www.ulta.com) websites, we then cleaned up the data and created dataframes in order to transform the data to a usuable format and then loaded the data to PostreSQL database for easy access.

**Web-scrapping (Data Extraction)**

For the web-scrapping part of the exercise we used Beautiful Soup and Chromedriver to scrape the links from the Sephora and Ulta.com websites. We started by creating separate list to store the data and then combined them into a dataframe. Part of the challenge that we encountered, was that part of the data was written in Javascript, which called for us to use Chromedriver and for us to write additional code to mandate a delay in the execution of the code to allow the webpage to load before we could extract the data that we needed.

Some of the code used to accomplish this step is below:

**A portion of the Ulta.com Web-scrapping code using Beautiful Soup**



**Similarly, a portion of the Sephora Web-Scrapping code using Beautiful Soup**

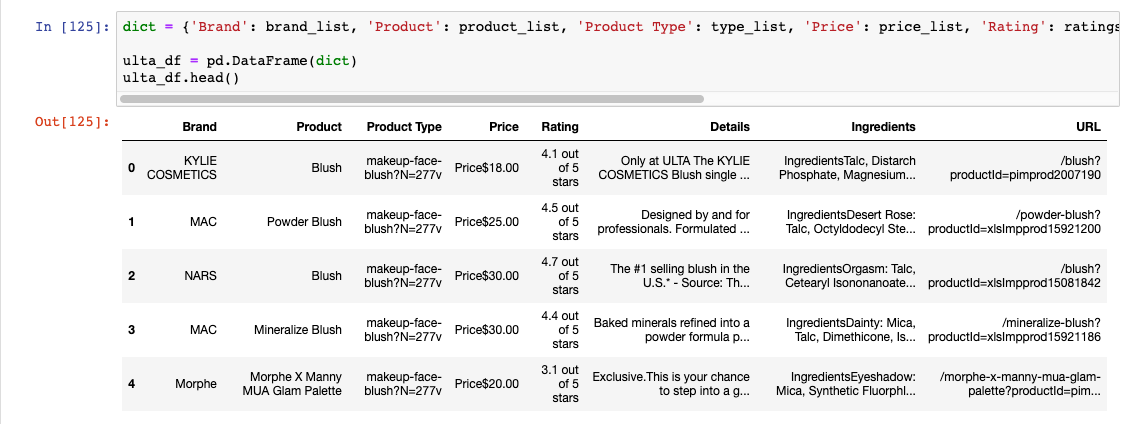


**Data Management – Cleanup – (Transformation)**

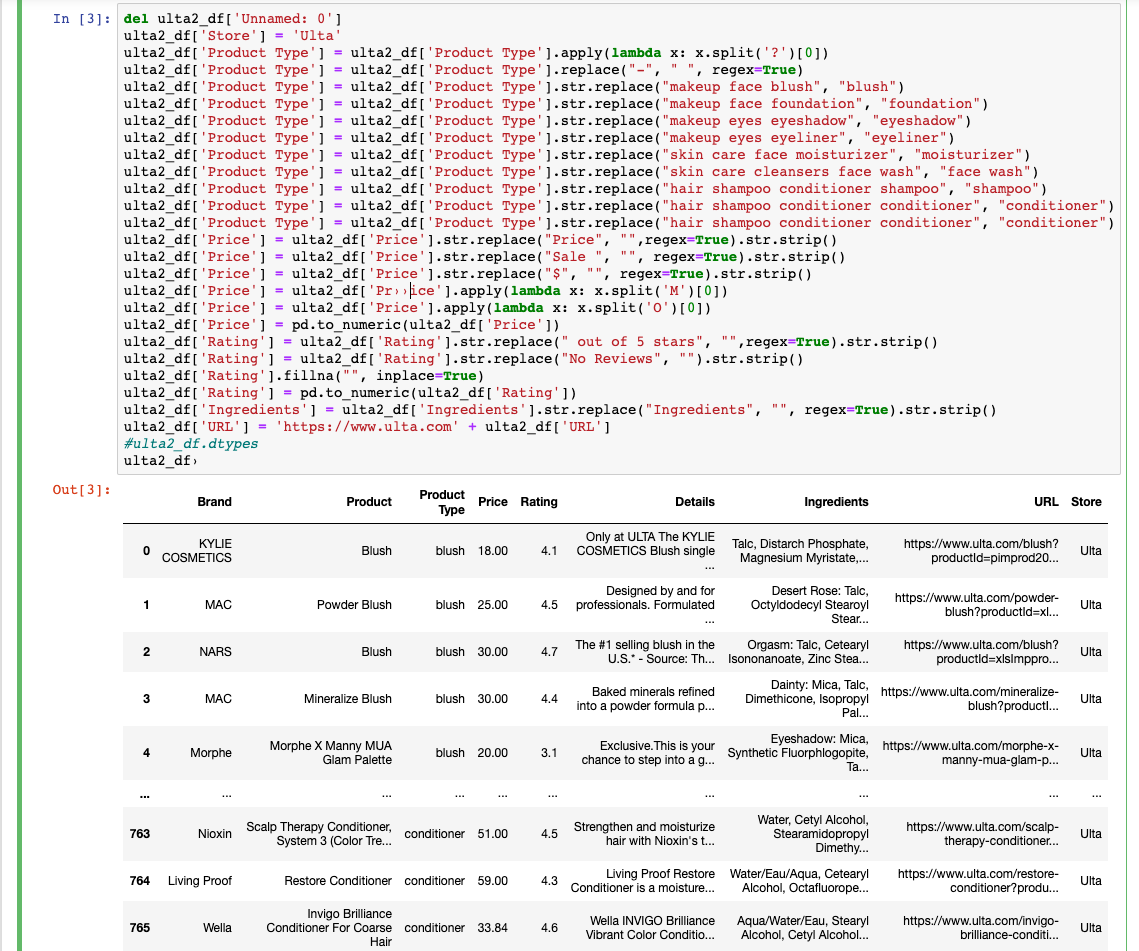
Once we had the data we used pandas to create dataframes in order to clean, determine data integrity and interpret the data. We needed to add the store that the product was available on (in this case Sephora and Ulta), rename the product types, change the price variable from a string to a numeric object, make the ratings numeric as well and append the domain URL to the URLs for the products.

Portion of the code used during this step is below:

**Code to store data in dataframes:**



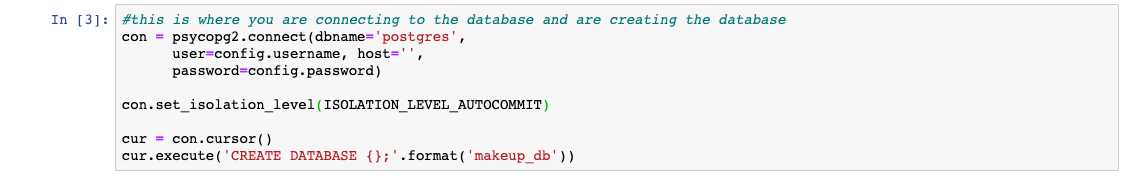
**Code to clean the data and turned it into usable format:**



**Storing the data for quick/uniformed access (load):**

For loading we imported the pandas dataframes into a new jupyter notebook, set up the connection to PostgreSQL database, merged the Sephora and Ulta table into a combined new master table. We opted to merge the two tables at this point, as it would improve use and accessibility of the data going forward. Following this step, we proceeded to import the data to PostgreSQL.

**Code snipped to stablish a connection with PostgreSQL**



**Code snipped to import the data into tables in PostgreSQL**

