The Search for Beauty

SI 507

Arshia Will | Aywill

Github Repository: <https://github.com/aywill/507final>

Demo Recording:https://www.loom.com/share/7fa77729da5843bea1d7e74d90eb78c1

I changed the nature of my final project. I will be building an interactive program that scrapes the sephora.com, providing the user with information on brands and products carried on the website.

Data Sources:

Sephora.com: <https://www.sephora.com/>

The program crawls and scrapes multiple pages on sephora.com dictated by the user’s response. The information pulled is eventually used to generate python class object for ‘Brands’ and ‘Products’ and its attributes. Caching is used every time a call is made to Sephora. There are a lot of relevant pieces but here are the class objects attributes all(but cruelty free status) are generated from the scraped data.

class brand:

    def \_\_init\_\_(self, name, url, products):

        self.name=name

        self.url=url

        self.products= products

class product:

    def \_\_init\_\_(self, name, url, brand\_name,price=None, number=None ,category=None, in\_stock=None, ingredients=None, details=None, size=None , use=None , cruelty\_free=None):

        self.name=name

        self.url=url

        self.brand\_name=brand\_name

        self.price=price

        self.category=category

        self.in\_stock=in\_stock

        self.ingredients=ingredients

        self.details=details

        self.size=size

        self.use=use

        self.number=number

        self.cruelty\_free=cruelty\_free

**Records:** An estimate is around 3600. Sephora carries 0ver 300 brands and the HTML will allow for web scraping up to 12 products per brand. The program only retrieves the record is it is called.

Crueltyfreekitty.com: <https://www.crueltyfreekitty.com/cruelty-free-sephora-brands/>

The program scrapes crueltyfreekitty to get the cruelty free status of cosmetic brands sold at Sephora. I chose to scrape the site oppose to building a list because the cruelty free status of brands and brands Sephora sells changes constantly. By webscraping the site each time, the program runs will give the most updated list of cruelty free brands. Caching is not used to guarantee the user has the most up to date list.

**Records:** The one HTML file. Most interested in the ul element that contains the cruelty free brands.

Sephora-reviews.CSV

I have been working on compiling Sephora reviews independently of this project. You cannot scrape them directly through a web crawling program. The CSV does not have every review for every product but enough to highlight in the program. Relevant columns include Name(Name of product reviewed),Brand Name(Brand of product reviewed), Rating(numerical rating on a scale of 1-4 the reviewer gave the product), and Review Text(self-explanatory.)

**Records:** Currently around 3200.

Cache File:

A close up of a newspaper

Description automatically generated

Database:

The program generates a database called sephora.sqlite and creates and populates 2 tables based on the user’s interaction with the program. They are populated through calling class attributes of objects generated from web scraping. I have not added relational aspects yet because I am having difficulty getting that to work.

**Brands Table:** displays products, url to product page, and count of how many products were found.

    create\_brands\_sql = '''

        CREATE TABLE IF NOT EXISTS "Brands" (

            "Id" INTEGER PRIMARY KEY AUTOINCREMENT,

            "Name" TEXT NOT NULL,

            "Url" TEXT NOT NULL,

            "ProductCount" TEXT NOT NULL

        )

    '''

A screenshot of a cell phone

Description automatically generated

**Products Table:** displays Name of Product, Brand Name, Category, price, size, item no, Cruelty Free status, URL, Ingredients, Details, and Use.

    create\_products\_sql = '''

        CREATE TABLE IF NOT EXISTS 'Products'(

            'Id' INTEGER PRIMARY KEY AUTOINCREMENT,

            "Name" TEXT NOT NULL,

            'BrandName' TEXT NOT NULL,

            'Category' TEXT NOT NULL,

            'Price' TEXT NOT NULL,

            'Size' TEXT,

            'ItemNumber' TEXT,

            'CrueltyFree' TEXT NOT NULL,

            'URL' TEXT NOT NULL,

            'Ingredients' TEXT,

            'Details' TEXT,

            'Use' TEXT )

    '''

**A screenshot of a cell phone

Description automatically generated**

Interaction and Presentation Plans

The user interacts with the program through command line prompts. Here is a brief walkthrough:

Main Menu:

User must select to search a brand directly, search a brand by first letter, or quit the program.

Searching Brands:

**Explore Brand by First Letter:** User inputs a first letter then a list of corresponding brands is displayed. User can then choose to search a brand from that list.

**Search a Brand:** User Bypass the brand list display and searches the full brand name.

Product List:

Once the user searches a brand the program returns a numbered list with up to 12 products the brand sells. Users are then asked to select a product to know more.

Product Details:

Prints out relevant information about the product selected. User is then prompted with a menu on what they want to do next.

**Read Product Review:** Displays up to 10 reviews for selected products. Reviews contain the numeric rating and review text. For privacy reasons reviews are all anonymous. If no reviews are found the program prints ‘Sorry, there are no reviews for this product.’ The user is then asked if they want check for reviews on sephora.com. If yes, the webpage is launched in the user’s browser and brings the user back to the main menu.

**View product on Sephora.com:** The item URL is launched in a web browser and the user returns to the main menu.

**Back to Product List:** the user is brought back to the product list and is able to select another product.

**Back to Main Menu:** user is returned to the main menu