

What is my goal in this project?

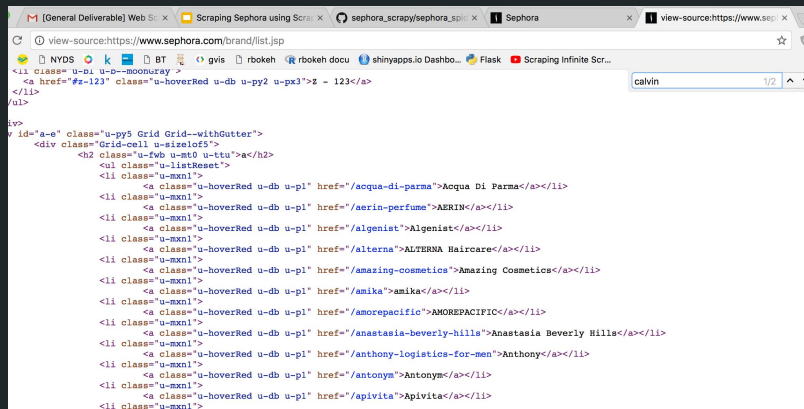
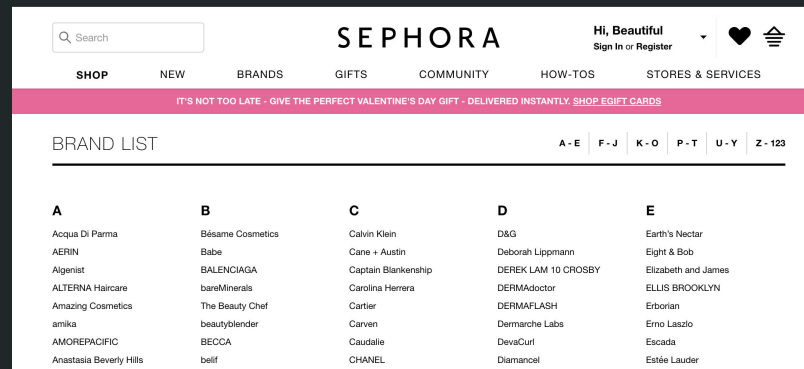
1. Scrape Sephora
2. Learning how to use Flask and creating an Application which can do simple queries of Sephora reviews data using Flask and sqlite3

How to Scrape Sephora's website

1. Getting the brand links
2. Getting the product links
3. Getting product details
4. Constructing the API link using information from (#1) and (#2)
5. Getting data from the API

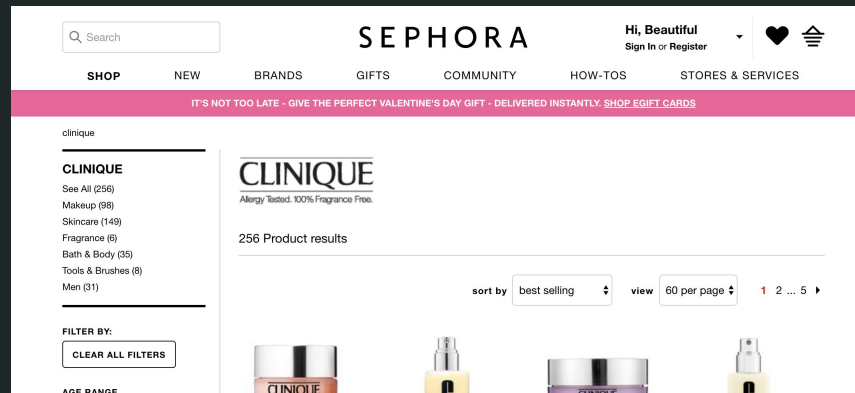
Getting the brand links

<https://www.sephora.com/brand/list.jsp>



Getting the product links

view-source:<https://www.sephora.com/clinique?products=all>



```
Apps NYDS k BT gvis rboket rboket docu shynapps.io Dashbo... Fisk Scrapping Infinite Scr...

//remove SephoraChat event, so we don't call this again when we click on a Search Page leftnav.
document.removeEventListener('SephoraChat', addChat, false);

});

// Listen for the event firing from sephora.js.
document.addEventListener('SephoraChat', addChat, false);

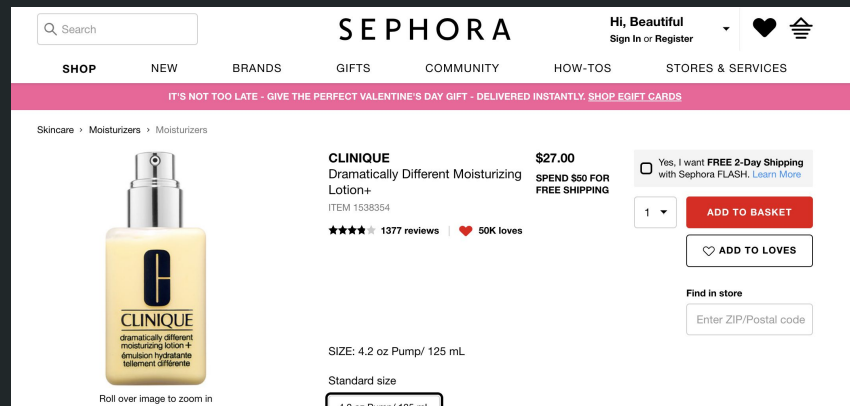
//Listen for the event firing from Signal.
document.addEventListener('SignalChat', addChat, false);

</script>
<!-- nth-level SEO searchResultsJSONScript droplet -->
<!-- end nth-level SEO searchResultsJSONScript droplet -->

<script id="searchResult" type="text/json" seph-pagdata>
  {
    "page_size":160,"refinements":{
      {("display_name":"Age Range","values":{
        {("display_name":"20s","status":1,"value":100119),
        {("display_name":"30s","status":1,"value":100120),
        {("display_name":"40s","status":1,"value":100121),
        {("display_name":"50s +","status":1,"value":100122),
        {("display_name":"Gens","status":1,"value":100118)},
        {("display_name":"Brand","values":{
        {("display_name":"CLINIQUE","status":4,"value":900162)},
        {("display_name":"Benefits","values":{
        {("display_name":"Hydrating","status":1,"value":9010005),
        {("display_name":"Lengthening","status":1,"value":100114),
        {("display_name":"Long-wearing","status":1,"value":9010004),
        {("display_name":"Plumping","status":1,"value":9010006),
        {("display_name":"Volumizing","status":1,"value":100115),
        {("display_name":"Waterproof","status":1,"value":100116)},
        {("display_name":"Bristle Type","values":{
        {("display_name":"Additive","status":1,"value":14578019),
        {("display_name":"Playful","status":1,"value":14578021),
        {("display_name":"Sporty","status":1,"value":14578023)},
        {("display_name":"Color Family","values":{
        {("display_name":"Berry","status":1,"value":9010109),
        {("display_name":"Black","status":1,"value":100010),
        {("display_name":"Blue","status":1,"value":100013),
        {("display_name":"Brown","status":1,"value":100002),
        {("display_name":"Coral","status":1,"value":9010111),
        {("display_name":"Gold","status":1,"value":100014),
        {("display_name":"Green","status":1,"value":100015),
        {("display_name":"Grey","status":1,"value":100001),
        {("display_name":"Multi","status":1,"value":100004),
        {("display_name":"Nude","status":1,"value":100016),
        {("display_name":"Pink","status":1,"value":100006),
        {("display_name":"Purple","status":1,"value":100011),
        {("display_name":"Red","status":1,"value":100007),
        {("display_name":"Unconventional","status":1,"value":9010128),
        {("display_name":"Universal","status":1,"value":9010110),
        {("display_name":"White","status":1,"value":100009)},
        {("display_name":"Colors","values":{
        {("display_name":"Concerns","values":{
        {("display_name":"Acne/Blemishes","status":1,"value":100042),
        {("display_name":"Anti-aging","status":1,"value":100033),
        {("display_name":"Blackheads","status":1,"value":100041),
        {("display_name":"Dark circles","status":1,"value":100040),
        {("display_name":"Dark spots","status":1,"value":100035),
        {("display_name":"Dryness","status":1,"value":100025),
        {("display_name":"Bulliness/Uneven texture","status":1,"value":104001),
```

Getting the product details

<https://www.sephora.com/product/dramatically-different-moisturizing-lotion-P381030?skuld=1538354&icid2=products%20grid:p381030>

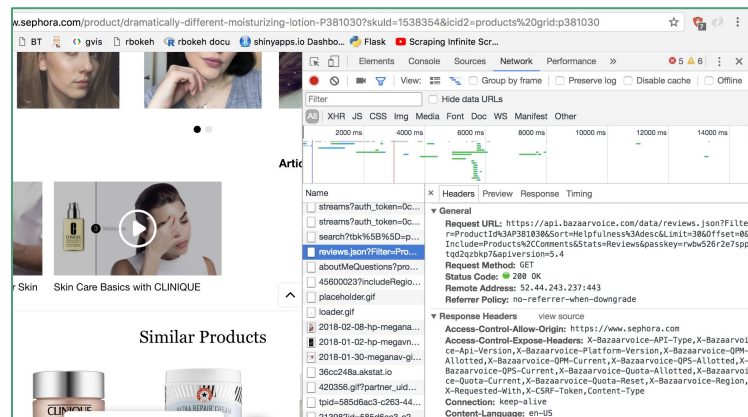
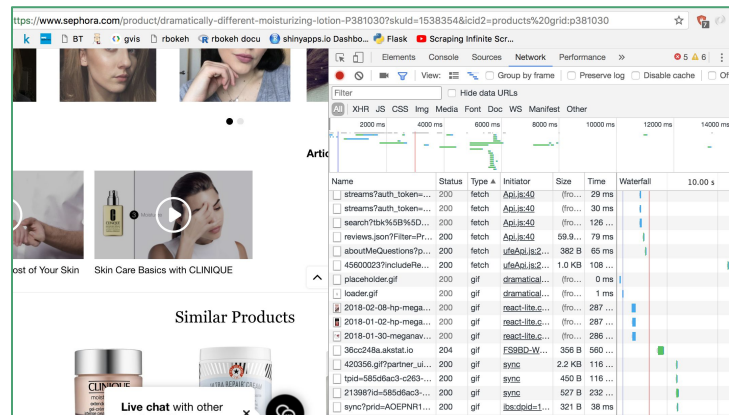


```
<html lang="en" class="css-lmk2x0" data-comp="Index"><head data-comp="Head"><title>Dramatically Different Moisturizing Lotion+ - CLINIQUE | Sephora</title>
<meta name="viewport" content="width=1024"/><meta name="description" content="Shop CLINIQUE's Dramatically Different Moisturizing Lotion+ at Sephora. This
fast-absorbing, lightweight daily moisturizer is for dry to combination skin types.</><link rel="canonical"
href="https://www.sephora.com/product/dramatically-different-moisturizing-lotion-P381030"/><link rel="alternate" media="only screen and (max-width: 640px)"
href="https://m.sephora.com/product/dramatically-different-moisturizing-lotion-P381030"/><meta name="og:description" content="Shop CLINIQUE's Dramatically
Different Moisturizing Lotion+ at Sephora. This fast-absorbing, lightweight daily moisturizer is for dry to combination skin types.</><meta name="og:title"
content="Dramatically Different Moisturizing Lotion+ - CLINIQUE | Sephora"><meta name="og:type" content="website"><meta name="og:url"
content="https://www.sephora.com/product/dramatically-different-moisturizing-lotion-P381030"/><meta name="og:image" content="/productimages/sku/s1538354-main-
hero-300.jpg"/><meta name="apple-mobile-web-app-capable" content="yes"/><meta name="format-detection" content="telephone=no"/><script>if (typeof global ===
"undefined") window.global = window.global.Sephora = global.Sephora || {};Sephora.targetersToInclude = "Y";Sephora.template =
"Product/ProductPage";sephora.renderedData = {"rendered": "2018-02-13
04:01:39,850", "template": "Product/ProductPage", "channelProp": "PS", "renderHost": "ph626201.bw140g", "pageRenderTime": 167.293};Sephora.renderQueryParams =
{"hash": "e5809e2a080e4ade592b01316101083fc08", "channel": "PS", "country": "US", "urlPath": "/2product/2Pdramatically-different-moisturizing-lotion-P381030"};
</script><script>Sephora.productPage = { defaultSkuld: 1538354 };</script><script>use strict;try{(var ce=ew
window.CustomEvent("test"));if(ce.preventDefault(),10!==(ce.defaultPrevented?throw new Error("Could not prevent default")):catch(e){var CustomEvent=function(e,t)
{var h;return t=t||{bubbles:!1,cancelable:!1,detail:void
0},n=document.createEvent("CustomEvent"),n.initCustomEvent(e,t,bubbles,t.cancelable,t.detail),r=n.preventDefault,n.preventDefault=function()
{
```

Getting the Reviews - Infinite Scroll Problem

How to solve this?

- Inspect Element
- Go to the Network, reload the page
- Notice that when you scroll to make the reviews appear, a json fetch item appears



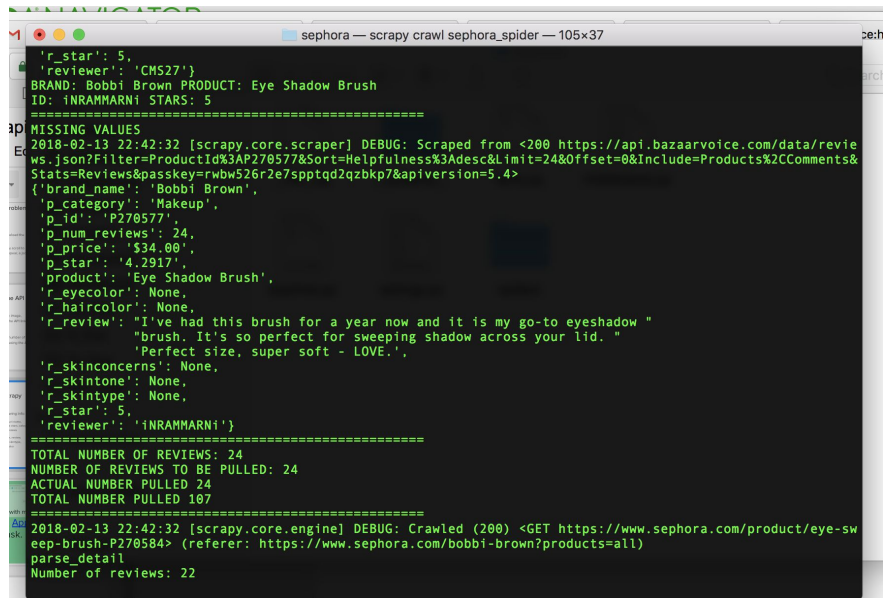
Constructing the API link

- As we can see in the image, there's a pattern to the API link
- Product ID
- And we can set the number of reviews we can call using the API

[https://api.bazaarvoice.com/data/reviews.json?Filter=ProductId%3A**P381030**&Sort=Helpfulness%3Adesc&**Limit=30**&Offset=0&Include=Products%2CComments&Stats=Reviews&passkey=rwbw526r2e7spptqd2qzbkp7&api-version=5.4](https://api.bazaarvoice.com/data/reviews.json?Filter=ProductId%3AP381030&Sort=Helpfulness%3Adesc&Limit=30&Offset=0&Include=Products%2CComments&Stats=Reviews&passkey=rwbw526r2e7spptqd2qzbkp7&api-version=5.4)

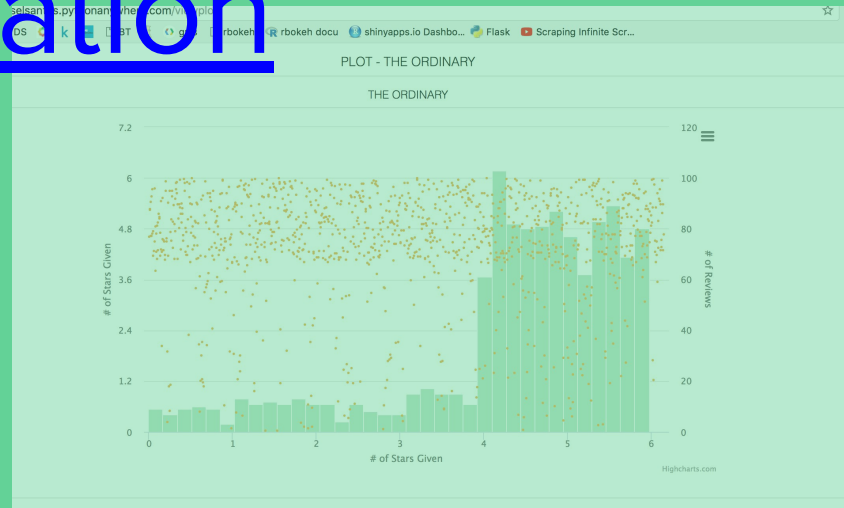
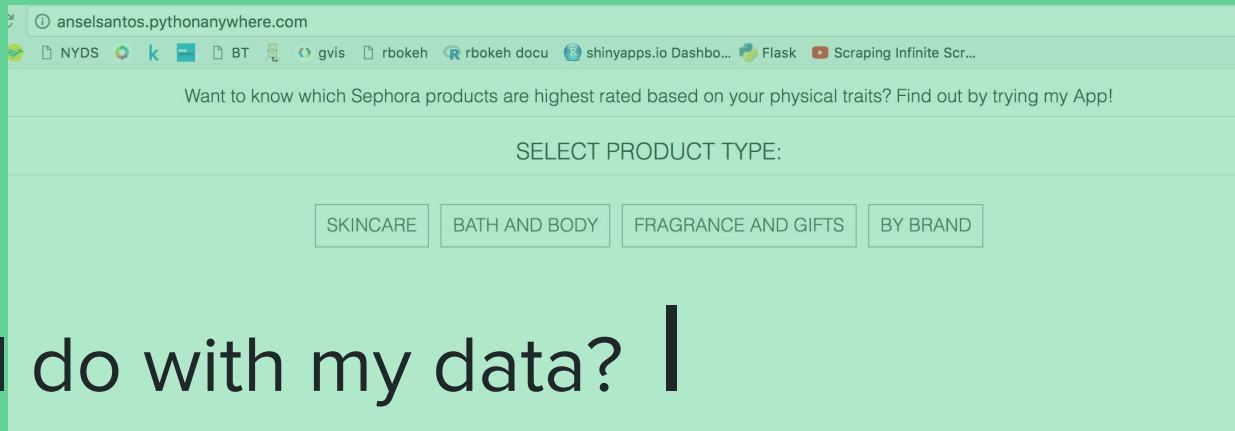
Running the Scrapy program

- I collected the following info:
 - Brand Name, product name, product ID, average stars, category, price, number of reviews
 - Reviewer nickname, review, reviewer skintone, skintype, haircolor and eyecolor
 - I collected around **80k+** rows of data



```
sephora — scrapy crawl sephora_spider — 105x37
{'r_star': 5,
 'reviewer': 'CMS27'}
BRAND: Bobbi Brown PRODUCT: Eye Shadow Brush
ID: INRAMMARNi STARS: 5
=====
MISSING VALUES
2018-02-13 22:42:32 [scrapy.core.scrapy] DEBUG: Scraped from <200 https://api.bazaarvoice.com/data/reviews.json?Filter=ProductId%3AP270577&Sort=Helpfulness%3Adesc&Limit=24&Offset=0&Include=Products%2CComments&Stats=Reviews&passkey=rwbw526r2e7spptqd2qzbp7&apiversion=5.4>
{'brand_name': 'Bobbi Brown',
 'p_category': 'Makeup',
 'p_id': 'P270577',
 'p_num_reviews': 24,
 'p_price': '$34.00',
 'p_star': '4.2917',
 'product': 'Eye Shadow Brush',
 'r_eyecolor': None,
 'r_haircolor': None,
 'r_review': "I've had this brush for a year now and it is my go-to eyeshadow "
              "brush. It's so perfect for sweeping shadow across your lid. "
              "Perfect size, super soft - LOVE.",
 'r_skinconcerns': None,
 'r_skintone': None,
 'r_skintype': None,
 'r_star': 5,
 'reviewer': 'INRAMMARNi'}
=====
TOTAL NUMBER OF REVIEWS: 24
NUMBER OF REVIEWS TO BE PULLED: 24
ACTUAL NUMBER PULLED 24
TOTAL NUMBER PULLED 107
=====
2018-02-13 22:42:32 [scrapy.core.engine] DEBUG: Crawled (200) <GET https://www.sephora.com/product/eye-shadow-brush-P270584> (referer: https://www.sephora.com/bobbi-brown?products=all)
parse_detail
Number of reviews: 22
```


What did I do with my data? I
made an Application
using Flask.



Conclusion

To sum things up, I was able to show that Sephora's website can be scraped relatively easily and I learned how to create an Application using Flask and sqlite3, achieving my goals for the project.