In [1]:
```
#Python Imports
import os
import sys
import csv
import json
import time
import itertools
import numpy as np
import pandas as pd
from fuzzywuzzy import fuzz
from fuzzywuzzy import process
from selenium import webdriver
from IPython.display import Image
from selenium.webdriver.common.by import By
from selenium.webdriver.chrome.options import Options
from selenium.webdriver.chrome.service import Service
from selenium.webdriver.common.keys import Keys
from selenium.webdriver.support.ui import WebDriverWait
from selenium.webdriver.common.by import By
from selenium.webdriver.support import expected_conditions as EC


#Selenium Imports
chrome_options = Options()
#chrome_options.add_argument("--headless") # Ensure GUI is off
#chrome_options.add_argument("--no-sandbox")
browser = webdriver.Chrome(options=chrome_options)
browser.implicitly_wait(15) # seconds
```

In [2]:
```
#######
search_term_raw = 'amanda upricard'
brand_url = 'https://amandauprichard.com/pages/store-locator-1'
########

search_term = search_term_raw + " in "
pages = 2
zipcodes = pd.read_json('OnePager/top_500_zipcodes.json')['zip'].apply(lambd
zipcodesSample = zipcodes[0:3]#['94123', '19104', '77494']
```

In [3]:
```
resultsList = []
url = brand_url

browser.get(url)
time.sleep(2)

for i, zipcode in enumerate(zipcodesSample, start=1):
    try:
        browser.get(url)

        query_entry=browser.find_element(By.CLASS_NAME, 'stockist-query-entr
        input_field = query_entry.find_element(By.TAG_NAME, 'input')
        submit = query_entry.find_element(By.CLASS_NAME, 'stockist-search-bu

        input_field.clear()
```

```
        input_field.send_keys(zipcode)
        submit.click()

        time.sleep(2)
        search_results = browser.find_element(By.CLASS_NAME, 'stockist-resul
        res=search_results.find_elements(By.CLASS_NAME,'stockist-result')

        for idx, store in enumerate(res):
            storeInfo = {}
            storeInfo['store_name'] = store.find_element(By.CLASS_NAME, 'sto
            storeInfo['RankNumber'] = idx
            storeInfo['ZipCode'] = zipcode
            address = [line.get_attribute("textContent") for line in
                       store.find_element(By.CLASS_NAME, 'stockist-result-add
            storeInfo['address'] = ", ".join(address)
            storeInfo['distance'] = store.find_element(By.CLASS_NAME, 'stock
            storeInfo['google_maps'] = store.find_element(By.CLASS_NAME, 'st
            storeInfo['starting_url(brand)'] = url
            #If website link in results
            '''
            try:
                storeInfo['website_link'] = store.find_element(By.CLASS_NAME
                print(storeInfo['website_link'])
            except:
                pass
                #print('...no weblink', storeInfo['name'], zipcode)
            '''
            resultsList.append(storeInfo)
        print(zipcode, " results:", len(res), ' -', i)

    except Exception as e:
        print("ERROR", zipcode, i, str(e)[0:75]+"...")
        time.sleep(2)
print("===Done===")
```

```
ERROR 77494 1 Message: element click intercepted: Element <button type="but
ton" class="st...
77449  results: 9  - 2
75034  results: 0  - 3
===Done===
```

In [4]:
```
pd.DataFrame(resultsList).to_csv("upright.csv")
```