CODE-REVIEW SCRAPPER

from flask import Flask, render\_template, request,jsonify  
from flask\_cors import CORS,cross\_origin  
import requests  
from bs4 import BeautifulSoup as bs  
from urllib.request import urlopen as uReq  
  
app = Flask(\_\_name\_\_)  
  
@app.route('/',methods=['GET']) # route to display the home page  
@cross\_origin()  
def homePage():  
 return render\_template("index.html")  
  
@app.route('/review',methods=['POST','GET']) # route to show the review comments in a web UI  
@cross\_origin()  
def index():  
 if request.method == 'POST':  
 try:  
 searchString = request.form['content'].replace(" ","")  
 flipkart\_url = "https://www.flipkart.com/search?q=" + searchString  
 uClient = uReq(flipkart\_url)  
 flipkartPage = uClient.read()  
 uClient.close()  
 flipkart\_html = bs(flipkartPage, "html.parser")  
 bigboxes = flipkart\_html.findAll("div", {"class": "bhgxx2 col-12-12"})  
 del bigboxes[0:3]  
 box = bigboxes[0]  
 productLink = "https://www.flipkart.com" + box.div.div.div.a['href']  
 prodRes = requests.get(productLink)  
 prodRes.encoding='utf-8'  
 prod\_html = bs(prodRes.text, "html.parser")  
 #print(prod\_html)  
 commentboxes = prod\_html.find\_all('div', {'class': "\_3nrCtb"})  
  
 filename = searchString + ".csv"  
 fw = open(filename, "w")  
 headers = "Product, Customer Name, Rating, Heading, Comment \n"  
 fw.write(headers)  
 reviews = []  
 for commentbox in commentboxes:  
 try:  
 #name.encode(encoding='utf-8')  
 name = commentbox.div.div.find\_all('p', {'class': '\_3LYOAd \_3sxSiS'})[0].text  
  
 except:  
 name = 'No Name'  
  
 try:  
 #rating.encode(encoding='utf-8')  
 rating = commentbox.div.div.div.div.text  
  
  
 except:  
 rating = 'No Rating'  
  
 try:  
 #commentHead.encode(encoding='utf-8')  
 commentHead = commentbox.div.div.div.p.text  
  
 except:  
 commentHead = 'No Comment Heading'  
 try:  
 comtag = commentbox.div.div.find\_all('div', {'class': ''})  
 #custComment.encode(encoding='utf-8')  
 custComment = comtag[0].div.text  
 except Exception as e:  
 print("Exception while creating dictionary: ",e)  
  
 mydict = {"Product": searchString, "Name": name, "Rating": rating, "CommentHead": commentHead,  
 "Comment": custComment}  
  
 reviews.append(mydict)  
 return render\_template('results.html', reviews=reviews[0:(len(reviews)-1)])  
 except Exception as e:  
 print('The Exception message is: ',e)  
 return 'something is wrong'  
 # return render\_template('results.html')  
  
 else:  
 return render\_template('index.html')  
  
if \_\_name\_\_ == "\_\_main\_\_":  
 #app.run(host='127.0.0.1', port=8001, debug=True)  
 app.run(debug=True)

**HEROKU DEPLOYMENT**

<https://demo-review-scrapper.herokuapp.com/>

Heroku URL --https://git.heroku.com/demo-review-scrapper.git





