Data Gathering

Fetch data from API's

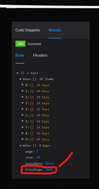
video

github

Fetching data from api from rapidapi

step1 Subscribe api you want to fetch data from

step2 Text run on rapidapi



step3 You get the total pages

step4 Get the code to fetch the data in python

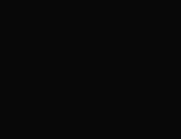
step5 note pad code

```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
import requests

df=pd.DataFrame()

for i in range(1,3):
    url = "https://anime-db.p.rapidapi.com/anime"
    querystring = {"page":i,"size":"10","sortOrder":"asc"}
    headers = {
        "X-RapidAPI-Key": "3c8868eaedmsh242635b63e210f2p1924ccjsn343d2a57bfe3",
        "X-RapidAPI-Host": "anime-db.p.rapidapi.com"
    }
    response = requests.get(url, headers=headers, params=querystring)
    temp_df = pd.DataFrame(response.json()['data'])[['_id','title','alternativeTitles','ranking','genres','episodes','status','synopsis','type']]
    df = pd.concat([df,temp_df],ignore_index=True)
```

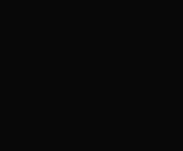
Handling the fetch data from api into csv file

Go to web site rapidapi.com For apis

open that response in the json.view



In json Viewer you can see
1-> Total page
2-> Total results
3-> All the important fields you want to import into csv columns among all of the json parameters



Start making the csv file in Notebook

Basic steps to get data into browse from api

1 Go to the api you want to try
2 get the API request url
3 then create account on main site
4 then in main site go to setting -> api ->api key
5 put api key at the api url and you get the data

Example

Step 1-> search in google Top rated api -> go to api you want to use -> go to the request url -> past that request url on browser
Example url -> For api = top rated = https://api.themoviedb.org/3/movie/top_rated?language=en-US&page=1

step 2-> search in google Top rated -> crate a account -> go to setting -> api -> get api key -> let ->abc
Example -> Final working request url= https://api.themoviedb.org/3/movie/top_rated?**api_key=abc**&language=en-US&page=1

Web Scrapring

video

github

fetch data from web into csv file

Go to web site Example -> ambition box , list of companies in indai

change the page number to the end to get to know how many page have data ,For our example max page number = 500

Deeside what thing you want to extract
For example i want to get these highlighted data

Start making the csv file in Notebook

Step 5 -> How to extract data using beautifulSoup

Basics of BeautifulSoup

Step1 -> import libraries

```
import pandas as pd
import requests
from bs4 import BeautifulSoup
import numpy as np
```

Step2 -> MAke a empty dataframe `df = pd.DataFrame()`

Step 3 ->How to get complete web page content/code from a web page to the python

Example
```
headers={'User-Agent':'Mozilla/5.0 (Windows NT 6.3; Win 64 ; x64) Apple WeKit /537.36(KHTML , like Gecko) Chrome/80.0.3987.162 Safari/537.36'}
#we are using header because if you try to get data without header, web site think you are a bot and give 403 error
#Lets store web page in a variable
webpage=requests.get('url_of_the_page_you_want_to_see',headers=headers).text
```

Step 4 -> Store the code of web page into a BeautifulSoup object `soup=BeautifulSoup(webpage,'lxml')`

Q1 extract all data that is present in h1 tag

step1 Find and Store all h1 tag data in a list like format `soup.find_all('h1')`

step2 this page contain only one h1 tag, so get that h1 tag by using simple indexing `soup.find_all('h1')[0]`

step3 Now to get the text use text `soup.find_all('h1')[0].text`

Q2 print all data that is present in h2 tag Code

```
for i in soup.find_all('h2'):
    print(i.text.strip())

#using text.strip because the person who created the html code have add a lot of space and other special characters , but we need only text
```

Q3 extract all data of reivew

step1 Find the tag and Class which is containing the required data `tag='p' , class='rating'`

```
for i in soup.find_all('p' , class='rating'):
    print(i.text.strip())
```

step2 `#using class also because p is being used for other things also`

Q4 print all data that is present in a single tag and tag as different fields

Example tag='p' and class name ='infoEntity'

```
soup.find_all('div',class_='infoEntity')[0].text.strip()
soup.find_all('div',class_='infoEntity')[1].text.strip()
soup.find_all('div',class_='infoEntity')[2].text.strip()
soup.find_all('div',class_='infoEntity')[3].text.strip()
```

How to get desired data for a single page

Step1 create a soup for complete card of the company `company=soup.find_all('div',class_='company-content-wrapper')`

```
name=[]
rating=[]
reviews=[]
ctype=[]
hq=[]
how_old=[]
no_of_employee=[]
```

step 2 code

Creating separate list for each columns

```
for i in company:

    name.append(i.find('h2').text.strip())
    rating.append(i.find('p',class_='rating').text.strip())
    #we are using find here because we are expecting only one tag h2 or 'p' with class name 'rating' here
    reviews.append(i.find('a' , class_='review-count').text.strip())
    ctype.append(i.find_all('p',class_='infoEntity')[0].text.strip())
    hq.append(i.find_all('p',class_='infoEntity')[1].text.strip())
    how_old.append(i.find_all('p',class_='infoEntity')[2].text.strip())
    no_of_employee.append(i.find_all('p',class_='infoEntity')[3].text.strip())
```

Going from one company to another on the same page

Now creating a data frame

```
df=pd.DataFrame({'name':name,
  'rating':rating,
  'reviews':reviews,
  'company_type':ctype,
  'Head_Quarters':hq,
  'Company_Age':how_old,
  'No_of_Employee':no_of_employee,
  })
```

```
final=pd.DataFrame()
for j in range(1,1001):
  webpage=requests.get('https://www.ambitionbox.com/list-of-companies?page={}'.format(j)).text
  soup=BeautifulSoup(webpage,'lxml')
  company=soup.find_all('div',class_='company-content-wrapper')
  name=[]
  rating=[]
  reviews=[]
  ctype=[]
  hq=[]
  how_old=[]
  no_of_employee=[]

  for i in company:

    try:
      name.append(i.find('h2').text.strip())
    except:
      name.append(np.nan)

    try:
      rating.append(i.find('p',class_='rating').text.strip())
    except:
      rating.append(np.nan)

    try:

      reviews.append(i.find('a' , class_='review-count').text.strip())
    except:
      reviews.append(np.nan)

    try:

      ctype.append(i.find_all('p',class_='infoEntity')[0].text.strip())
    except:
      ctype.append(np.nan)
    try:

      hq.append(i.find_all('p',class_='infoEntity')[1].text.strip())
    except:
      hq.append(np.nan)

    try:

      how_old.append(i.find_all('p',class_='infoEntity')[2].text.strip())
    except:
      how_old.append(np.nan)
    try:
      no_of_employee.append(i.find_all('p',class_='infoEntity')[3].text.strip())
    except:
      no_of_employee.append(np.nan)


  df=pd.DataFrame({'name':name,
    'rating':rating,
    'reviews':reviews,
    'company_type':ctype,
    'Head_Quarters':hq,
    'Company_Age':how_old,
    'No_of_Employee':no_of_employee,
    })

  final=final.append(df,ignore_index=True)
```

How to get desired data for multiple pages code