

```
import requests
import urllib.request
from bs4 import BeautifulSoup
import csv
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
from matplotlib_venn import venn2
```

➤ Artist 100 chart 2019

ทำการ request ไปที่ url ซึ่งใช้ Python library ที่ชื่อว่า request โดย lib ตัวนี้จะทำการ GET request ไปที่ web server ของ url ที่เราระบุ และจะทำการ download content ของ web page มาให้ และจากนั้น BeautifulSoup ทำการ parse content ที่เรา download ได้มา ให้มันสวยงามเข้าใจได้ง่าย

```
url19 = "https://www.billboard.com/charts/artist-100/2019-09-28?fbclid=IwAR1ln03GaDLponUxEPU42-mv3d5A89Vv1lzMrxKbKhC8kh_mpvqP8Q76lwU"
res19 = requests.get(url19)
soup19 = BeautifulSoup(res19.text, "html.parser")
```

เพื่อที่จะเห็นโครงสร้างเท็กอย่างเป็นระเบียบชัดเจนจะใช้เมธอด .prettyfy() ซึ่งเราจะใช้ tag เพื่อระบุตำแหน่งของข้อมูลที่ต้องการในการดึงข้อมูลออกมา

```
print(soup19.prettyfy())

<!DOCTYPE doctype html>
<html class="" lang="">
<head>
  <meta charset="utf-8"/>
  <meta content="IE=edge" http-equiv="X-UA-Compatible"/>
  <meta content="width=device-width, initial-scale=1, user-scalable=no" name="viewport"/>
  <title>
    Artist 100 Chart | Billboard
  </title>
  <meta content="Artist 100 Chart" name="title" property="title">
  <meta content="@billboard" name="twitter:site"/>
  <meta content="Billboard" property="og:site_name">
  <meta content="article" property="og:type">
  <link href="/manifest.json" rel="manifest"/>
  <link href="/charts/artist-100/2019-09-28" rel="canonical"/>
  <link href="https://www.billboard.com/assets/1603122088/images/favicon.ico?7311d39e6d29085cb52a" rel="shortcut icon" type="image/x-icon"/>
  <link href="https://www.billboard.com/assets/1603122088/images/BB_favicon144.png?7311d39e6d29085cb52a" rel="apple-touch-icon"/>
  <link href="https://www.billboard.com" rel="dns-prefetch"/>
  <link href="https://www.google-analytics.com/" rel="dns-prefetch"/>
  <link href="https://www.googletagservices.com/" rel="dns-prefetch"/>
  <link href="https://www.googletagmanager.com/" rel="dns-prefetch"/>
  <link href="https://www.instagram.com/" rel="dns-prefetch"/>
  <link href="https://www.zergnet.com/" rel="dns-prefetch"/>
  <link href="https://players.brightcove.net/" rel="dns-prefetch"/>
  <link href="https://platform.zig.com/" rel="dns-prefetch"/>
  <link href="https://www.googleadservices.com" rel="dns-prefetch"/>
  <link href="https://c.amazon-adsystem.com/" rel="dns-prefetch"/>
  <link href="https://securepubads.g.doubleclick.net/" rel="dns-prefetch"/>
  <link href="https://cdn.krxd.net/" rel="dns-prefetch"/>
  <link href="https://beacon.krxd.net/" rel="dns-prefetch"/>
  <link href="https://a.quora.com/" rel="dns-prefetch"/>
  <link href="https://connect.facebook.net/" rel="dns-prefetch"/>
  <link href="https://s0.2mdn.net/" rel="dns-prefetch"/>
  <link href="https://c.amazon-adsystem.com" rel="dns-prefetch"/>
  <link href="https://sejs.moatads.com/" rel="dns-prefetch"/>
  <link href="https://mb.moatads.com/" rel="dns-prefetch"/>
  <link href="https://tags.bluekai.com/" rel="dns-prefetch"/>
  <link href="https://platform.twitter.com/" rel="dns-prefetch"/>
  <link href="https://analytics.twitter.com/" rel="dns-prefetch"/>
  <link href="https://lib.pgmcdn.com" rel="dns-prefetch"/>
  <script type="text/javascript">
    var PGM = window.PGM || {};
    PGM.config = PGM.config || {
      assetHost: "https://www.billboard.com/assets/1603122088",
      contentHost: "https://www.billboard.com",
      sailthruCustomerId: "89582e5ca0930701aaa4fbf88e0ff0d4"
    };
    (function(PGM) {
      var debug_queue = [];
      PGM.debug = function() {debug_queue.push(arguments)};
      PGM.debug.dump = function() {return debug_queue;}
      PGM.debug.initialCategories = {
      };
    })(PGM);

    PGM.events = (function() {
      var events_queue = [];
```

```
artist19 = soup19.find_all('div', {'class' : 'chart-list-item'})
print(len(artist19))
```

100

```
print(artist19[0].prettify())
```

```
<div class="chart-list-item " data-artist="" data-has-content="true" data-rank="1" data-title="Post Malone">
<div class="chart-list-item__first-row chart-list-item__cursor-pointer">
<div class="chart-list-item__position chart-list-item__position--centered">
<div class="chart-list-item__rank ">
1
</div>
<div class="chart-list-item__award">
</div>
</div>
<div class="chart-list-item__image-wrapper">
<div class="chart-list-item__trend-icon">
</div>

<div class="chart-list-item__text ">
<div class="chart-list-item__title">
<span class="chart-list-item__title-text">
<a href="/music/post-malone">
Post Malone
</a>
</span>
</div>
<div class="chart-list-item__artist">
</div>
<div class="chart-list-item__ministats ">
<div class="chart-list-item__ministats-cell">
1
<span class="chart-list-item__ministats-cell-heading">
Last
</span>
</div>
<div class="chart-list-item__ministats-cell">
1
<span class="chart-list-item__ministats-cell-heading">
Peak
</span>
</div>
<div class="chart-list-item__ministats-cell">
169
<span class="chart-list-item__ministats-cell-heading">
Weeks
</span>
</div>
</div>
</div>
<div class="chart-list-item__chevron-wrapper">
<i class="fa fa-chevron-down">
</i>
</div>
</div>
<div class="chart-list-item__extra-info">
<div class="chart-list-item__extra-info-shadow">
</div>
<div class="chart-list-item__stats">
<div class="chart-list-item__stats-cell chart-list-item__stats-cell--first-cell">
```

```
artist19[0].find('span', {'class' : 'chart-list-item__title-text'}).text
```

'\n\nPost Malone\n\n'

สร้าง list วางขึ้น จากนั้นเลือกtagและชื่อclassของข้อมูลที่ต้องการเพื่อนำไปเก็บไว้ในตัวแปรและเพิ่มตัวแปรทั้งหมดที่ต้องการไปเก็บใน list และทำการตัดอักขระ \n\n ด้านหน้าและด้านหลัง

```
art19 = []
for i in artist19:
    name = i.find('span', {'class' : 'chart-list-item__title-text'}).text.strip('\n\n')
    last_w = i.find('div', {'class' : 'chart-list-item__last-week'}).text
    two_w = i.find('div', {'class' : 'chart-list-item__stats-cell '}).text.lstrip('\n\n').rstrip('\nTWO WEEKS AGO\n')
    peak_pos = i.find('div', {'class' : 'chart-list-item__weeks-at-one'}).text
    w_on_chart = i.find('div', {'class' : 'chart-list-item__weeks-on-chart'}).text
    imprint = i.find('div', {'class' : 'chart-list-item__people_data'}).text.rstrip('\n\n')
    art19.append([name, last_w, two_w, peak_pos, w_on_chart, imprint[28:]])
```

```
for i in art19:
    if i[0].startswith(' '):
        i[0] = i[0].lstrip(' ')
```

```
pd.options.display.max_colwidth = 0
```

สร้างตารางแสดงข้อมูลจากตัวแปร art19 โดยมีคอลัมน์ชื่อ name, last_week, two_week_ago, peak_position, weeks_on_chart, Imprint/Promotion Label

```
df19 = pd.DataFrame(art19, columns = ['name', 'last_week', 'two_week_ago', 'peak_position', 'weeks_on_chart', 'Imprint/Promotion Label'])
df19
```

	name	last_week	two_week_ago	peak_position	weeks_on_chart	Imprint/Promotion Label
0	Post Malone	1	3	1	169	
1	The Lumineers	-	-	1	47	
2	Taylor Swift	2	2	1	269	
3	Lizzo	4	5	2	22	Nice Life/Atlantic
4	Billie Eilish	3	6	1	56	Darkroom/Interscope
...	
95	Michael Jackson	77	91	20	228	Never Again/At&A
96	YoungBoy Never Broke Again	86	-	26	62	Neon/Atlantic

ทำการแทนที่ข้อมูลในคอลัมน์ last_week, two_week_ago, peak_position, weeks_on_chart จาก - เป็นค่า 0 เพื่อให้ข้อมูลมีความต่อเนื่องและสามารถนำไปคำนวณเชิงคณิตศาสตร์ได้

```
df19['last_week'].replace({'-':'0'}, inplace = True)
df19['two_week_ago'].replace({'-':'0'}, inplace = True)
df19['peak_position'].replace({'-':'0'}, inplace = True)
df19['weeks_on_chart'].replace({'-':'0'}, inplace = True)
```

เปลี่ยนชนิดของข้อมูลที่เป็นตัวหนังสือจากคอลัมน์ last_week, two_week_ago, peak_position, weeks_on_chart เป็นข้อมูลชนิดจำนวนเต็ม

```
df19['last_week'] = pd.to_numeric(df19['last_week'])
df19['two_week_ago'] = pd.to_numeric(df19['two_week_ago'])
df19['peak_position'] = pd.to_numeric(df19['peak_position'])
df19['weeks_on_chart'] = pd.to_numeric(df19['weeks_on_chart'])
```

```
df19[['Imprint/Promotion Label 1', 'Imprint/Promotion Label 2']] = df19['Imprint/Promotion Label'].str.split("|", expand = True)
df19['Imprint/Promotion Label 2'] = df19['Imprint/Promotion Label 2'].fillna('-')
df19.drop('Imprint/Promotion Label', axis = 1, inplace = True)
```

```
df19
```

name last_week two_week_ago peak_position weeks_on_chart

Imprint/Pi

```
df19.to_csv('artist_year_2019.csv', index = False)
```

maione

▶ Artist 100 chart 2020

[] ↳ 21 cells hidden

▶ เปรียบเทียบอันดับ most popular

[] ↳ 7 cells hidden

-- Michael -- -- -- --

▶ ส่วนที่ 2

[] ↳ 6 cells hidden

▼ ศิลปินที่ติด Billboard Chart ทั้งปี 2019 และ 2020

ทำการเชื่อม ทั้งสองปีเข้าด้วยกันโดยให้ปี 2019 เป็นหลักและทำการนำข้อมูลปี 2020 รวมเข้าด้วยกัน

```
yearall = pd.merge(left=data_1,right=data_2, how='left', left_on='name', right_on='name')
yearall
```

	name	year_x	year_y
0	Post Malone	2019	2020.0
1	The Lumineers	2019	NaN
2	Taylor Swift	2019	2020.0
3	Lizzo	2019	2020.0
4	Billie Eilish	2019	2020.0
...
95	Michael Jackson	2019	2020.0
96	YoungBoy Never Broke Again\n	2019	NaN
97	Ozuna	2019	NaN
98	Stereolab	2019	NaN
99	Russell Dickerson	2019	NaN

100 rows × 3 columns

```
yearall.rename(columns={"year_x": "year_2019", "year_y": "year_2020"})
```

	name	year_2019	year_2020
0	Post Malone	2019	2020.0
1	The Lumineers	2019	NaN
2	Taylor Swift	2019	2020.0
3	Lizzo	2019	2020.0
4	Billie Eilish	2019	2020.0
...
95	Michael Jackson	2019	2020.0
96	YoungBoy Never Broke Again\n	2019	NaN
97	Ozuna	2019	NaN
98	Stereolab	2019	NaN
99	Russell Dickerson	2019	NaN

100 rows × 3 columns

ทำการตัดข้อมูลที่มีค่าเป็น NaN ออกเพื่อเอาข้อมูลศิลปินที่ติดอันดับทั้งสองปี และเรียงตามตัวอักษร

```
yearall.dropna(inplace=True)
yearall_s=yearall.sort_values('name')
yearall_s
```

	name	year_x	year_y
84	Adele	2019	2020.0
10	Ariana Grande	2019	2020.0
20	BTS	2019	2020.0
48	Bad Bunny	2019	2020.0
4	Billie Eilish	2019	2020.0

```

yearall_s.index = np.arange(1, len(yearall) + 1)
yearall_s['name']

```

```

1 Adele
2 Ariana Grande
3 BTS
4 Bad Bunny
5 Billie Eilish
6 Blake Shelton
7 Bruno Mars
8 Cardi B
9 Chris Brown
10 Chris Stapleton
11 DaBaby
12 Dan + Shay
13 Drake
14 Ed Sheeran
15 Gunna
16 Halsey
17 Imagine Dragons
18 J Balvin
19 Jason Aldean
20 Jonas Brothers
21 Juice WRLD
22 Justin Bieber
23 Kane Brown
24 Katy Perry
25 Khalid
26 Lady Gaga
27 Lewis Capaldi
28 Lil Baby
29 Lizzo
30 Luke Bryan
31 Luke Combs
32 Maren Morris
33 Maroon 5
34 Marshmello
35 Megan Thee Stallion
36 Michael Jackson
37 Miley Cyrus
38 Morgan Wallen
39 Old Dominion
40 Panic! At The Disco
41 Polo G
42 Post Malone
43 Queen
44 Saweetie
45 Taylor Swift
46 The Beatles
47 Thomas Rhett
48 Travis Scott
49 Young Thug
Name: name, dtype: object

```

ศิลปินที่ติด Billboard Chart เฉพาะปี 2019

```

41 Miley Cyrus 2019 2020.0

```

▼ ศิลปินที่ติด Billboard Chart เฉพาะปี 2019

```

22 Panic! At The Disco 2019 2020.0

```

ทำการรวมข้อมูลทั้งสองเข้าด้วยกันโดยใช้ปี 2019 เป็นหลัก

```

year2019 = pd.merge(left=data_1,right=data_2, how='left', left_on='name', right_on='name')
year2019

```

	name	year_x	year_y
0	Post Malone	2019	2020.0
1	The Lumineers	2019	NaN
2	Taylor Swift	2019	2020.0
3	Lizzo	2019	2020.0
4	Billie Eilish	2019	2020.0
...
95	Michael Jackson	2019	2020.0

ตัดข้อมูลที่มีปี 2020 ออก

37	Uzuna	2019	NaN
----	-------	------	-----

```
new = year2019[year2019['year_y'] == 2020 ].index
year2019.drop(new, inplace=True)
year2019_s=year2019.sort_values('name')
year2019_s
```

	name	year_x	year_y
52	5 Seconds Of Summer	2019	NaN
46	A Boogie Wit da Hoodie	2019	NaN
93	Bazzi	2019	NaN
59	Blanco Brown	2019	NaN
15	Camila Cabello	2019	NaN
61	Carrie Underwood	2019	NaN
73	Chris Lane	2019	NaN
94	Daddy Yankee	2019	NaN
81	Dierks Bentley	2019	NaN
30	Eddie Money	2019	NaN
74	Elton John	2019	NaN
54	Eminem	2019	NaN
71	Goo Goo Dolls	2019	NaN
50	J. Cole	2019	NaN
89	Justin Moore	2019	NaN

```
year2019_s.index = np.arange(1, len(year2019) + 1)
year2019_s['name']
```

```
1    5 Seconds Of Summer
2    A Boogie Wit da Hoodie
3    Bazzi
4    Blanco Brown
5    Camila Cabello
6    Carrie Underwood
7    Chris Lane
8    Daddy Yankee
9    Dierks Bentley
10   Eddie Money
11   Elton John
12   Eminem
13   Goo Goo Dolls
14   J. Cole
15   Justin Moore
16   Korn
17   Lana Del Rey
18   Lauren Daigle
19   Lil Nas X
20   Lil Tecca
21   Lil Tjay
22   Lynyrd Skynyrd
23   Matt Stell
24   Meek Mill
25   Melanie Martinez
26   Mustard
27   NF
28   NLE Choppa
29   Normani
30   OneRepublic
31   Ozuna
32   P!nk
33   Prince
34   Rick Ross
35   Russell Dickerson
36   SHAED
37   Sam Smith
38   Shawn Mendes
39   Starset
40   Stereolab
41   SuperM
42   Swae Lee
43   The Cars
44   The Chainsmokers
45   The Highwomen
46   The Hu
47   The Lumineers
48   Tool
49   XXXTENTACION
50   YoungBoy Never Broke Again\n
51   twenty one pilots
Name: name, dtype: object
4    The Lumineers    2019    NaN
```


▼ ศิลปินที่ติด Billboard Chart เฉพาะปี 2020

ทำการรวมข้อมูลทั้งสองปี โดยเอาข้อมูลปี 2020 เป็นหลัก

```
year2020 = pd.merge(left=data_2,right=data_1, how='left', left_on='name', right_on='name')
year2020
```

	name	year_x	year_y
0	YoungBoy Never Broke Again	2020	NaN
1	BTS	2020	2019.0
2	Taylor Swift	2020	2019.0
3	Juice WRLD	2020	2019.0
4	Marilyn Manson	2020	NaN
...
95	Jon Pardi	2020	NaN
96	Riley Green	2020	NaN
97	Maluma	2020	NaN
98	Adele	2020	2019.0
99	Miranda Lambert	2020	NaN

100 rows × 3 columns

ตัดข้อมูลที่มีค่าเป็น 2019 ออกและทำการเรียงข้อมูลตามชื่อ

```
new2 = year2020[year2020['year_y'] == 2019 ].index
year2020.drop(new2, inplace=True)
year2020_s=year2020.sort_values('name')
year2020_s
```

	name	year_x	year_y
44	24kGoldn	2020	NaN
68	AC/DC	2020	NaN
76	AJR	2020	NaN
37	Andrew Jannakos	2020	NaN
71	Ava Max	2020	NaN
85	BLACKPINK	2020	NaN
20	Big Sean	2020	NaN
75	Creedence Clearwater Revival	2020	NaN
42	DJ Khaled	2020	NaN
52	Doja Cat	2020	NaN
16	Dua Lipa	2020	NaN
90	Eagles	2020	NaN
31	Fleetwood Mac	2020	NaN
59	Florida Georgia Line	2020	NaN
22	Gabby Barrett	2020	NaN
92	HARDY	2020	NaN
5	Harry Styles	2020	NaN
56	Jack Harlow	2020	NaN
49	Jason Derulo	2020	NaN
66	Jawsh 685	2020	NaN
95	Jon Pardi	2020	NaN
64	Kanye West	2020	NaN
63	Keith Urban	2020	NaN
82	Kelsea Ballerini	2020	NaN
67	Kendrick Lamar	2020	NaN
81	Led Zeppelin	2020	NaN
29	Lee Brice	2020	NaN

```
year2020_s.index = np.arange(1, len(year2020) + 1)
year2020_s['name']
```

```
1 24kGoldn
2 AC/DC
3 AJR
4 Andrew Jannakos
5 Ava Max
6 BLACKPINK
7 Big Sean
8 Creedence Clearwater Revival
9 DJ Khaled
10 Doja Cat
11 Dua Lipa
12 Eagles
13 Fleetwood Mac
14 Florida Georgia Line
15 Gabby Barrett
16 HARDY
17 Harry Styles
18 Jack Harlow
19 Jason Derulo
20 Jawsh 685
21 Jon Pardi
22 Kanye West
23 Keith Urban
24 Kelsea Ballerini
25 Kendrick Lamar
26 Led Zeppelin
27 Lee Brice
28 Lil Durk
29 Maddie & Tae
30 Maluma
31 Marilyn Manson
32 Mastodon
33 Metallica
34 Miranda Lambert
```

```
35 Moneybagg Yo
36 Nirvana
37 PJ Harvey
38 Pink Floyd
39 Pop Smoke
40 Riley Green
41 Rod Wave
42 Roddy Ricch
43 SAINT JHN
44 SZA
45 Sam Hunt
46 The Flaming Lips
47 The Rolling Stones
48 The Weeknd
49 Tim McGraw
50 YoungBoy Never Broke Again
51 blackbear
Name: name, dtype: object
```

กราฟแสดงจำนวน Artist ในแต่ละปี

```
venn2(subsets = (51,51,49), set_labels = ('Yaer2019', 'Yaer2020'))
plt.title('Amount Of artist from Billbord 2019-2020')
plt.show()
```

