

Aphsatou Dia

Ramata Kane

Objectif de notre projet

Notre objectif est d'extraire l'ensemble des tableaux specifiques d'un article wikipedia d'un groupe de rock americain Linkin Park.De ce fait nous allons extraire les types d'albums publiés par Linkin Park dont leur page de discographie sur wikipedia est https://en.wikipedia.org/wiki/Linkin_Park_discography.Nous allons créer un ensemble de données de discographies ou d'albums de musique avec des informations telles que le titre de l'album, la date de sortie, etc...

Module nécessaire:

beautifulsoup4: Beautiful Soup (bs4) est une bibliothèque Python pour extraire des données de fichiers HTML et XML. Ce module n'est pas intégré à Python. Pour installer ce, tapez la commande ci-dessous dans le terminal.

```
pip install beautifulsoup4

Requirement already satisfied: beautifulsoup4 in /usr/local/lib/python3.7/dist-packages (4.6.3)

requests: Request vous permet d'envoyer des requêtes HTTP / 1.1 très facilement. Ce module n'est pas non plus intégré à Python. Pour installer ce, tapez la commande ci-dessous dans le terminal.

pip install requests

Requirement already satisfied: requests in /usr/local/lib/python3.7/dist-packages (2.23.0)
Requirement already satisfied: idna<3.0,>=2.5 in /usr/local/lib/python3.7/dist-packages (from requests) (2.10)
Requirement already satisfied: urllib3<1.25.0,>=1.25.1,!=1.26.0,!=1.21.1 in /usr/local/lib/python3.7/dist-packages (from requests) (1.24.3)
Requirement already satisfied: certifi<=2017.4.17 in /usr/local/lib/python3.7/dist-packages (from requests) (2021.5.30)
Requirement already satisfied: chardet<4,>=3.0.2 in /usr/local/lib/python3.7/dist-packages (from requests) (3.0.4)
```

```
#Importer les packages
#pour effectuer vos requêtes HTTP
import requests
#pour le grattage xml et html
from bs4 import BeautifulSoup
#pour l'analyse des tableaux
import pandas as pd
```

```
!pip install lxml

Requirement already satisfied: lxml in /usr/local/lib/python3.7/dist-packages (4.2.6)
```

```
##Saisir l'URL
url="https://en.wikipedia.org/wiki/Linkin_Park_discography"
```

```
#Importer le code de la page
#La fonction "session()" aide à l'objection ,nous permet de persister certains paramètres à travers les demandes
s= requests.Session()
reponse=s.get(url)
reponse

<Response [200]>
```

D'apres la ligne de code ci-dessus,nous avons obtenu une réponse [200], cela signifie essentiellement que le site a répondu avec un code de réponse HTTP 200 OK et nous a envoyé le contenu HTML de la page.

```
contenu_p = reponse.text
print(contenu_p[:1000])

<!DOCTYPE html>
<html class="client-nojs" lang="en" dir="ltr">
  <head>
    <meta charset="utf-8"/>
    <title>Linkin Park discography - Wikipedia</title>
    <script>document.documentElement.className="client-js";RLCONF={"wgBreakFrames":!1,"wgSeparatorTransformTable":["",""],"wgDigitTransformTable":["",""],"wgDefaultDateFormat":"dmy","wgMonthNames":["", "January", "February", "March", "April", "May", "June", "July", "August", "September", "October", "November", "December"],"wgRequestId":"8388d8fe-f96e-45c3-b3d7-d378fba152bc","wgSPNonce":!1,"wgCanonicalNames
  </script>
```

#Enregistrons le contenu dans un fichier html

```
with open('projet.html', 'w', encoding='utf-8') as file:
    file.write(contenu_p)
```

#Maintenant que nous avons le contenu HTML, nous devons analyser les données.
#Pour cela, nous utiliserons BeautifulSoup avec un parseur html5.
soup=BeautifulSoup(reponse.content,"html.parser")

pour afficher le contenu au format html
pretty_soup = soup.prettify()

```
pretty_soup

<!DOCTYPE html><html class="client-nojs" dir="ltr" lang="en"><head><meta charset="utf-8"/><title>Linkin Park discography - Wikipedia</title><script><!--
  <script>document.documentElement.className="client-js";RLCONF={"wgBreakFrames":!1,"wgSeparatorTransformTable":["",""],"wgDigitTransformTable":["",""],"wgDefaultDateFormat":"dmy","wgMonthNames":["", "January", "February", "Ma
rch", "April", "May", "June", "July", "August", "September", "October", "November", "December"],"wgRequestId":"8388d8fe-f96e-45c3-b3d7-d378fba152bc","wgSPNonce":!1,"wgCanonicalNamespace":"","wgCanonicalSpecialPageName":!1,"wgNamespacesNumber":10,"wgPageName":"Linkin_Park_discography","wgTitle":"Linkin Park discography","wgCurRevisionId":1050704765,"wgRevisionId":1050704765,"wgArticleId":3649517,"wgIs
Article":10,"wgIsRedirect":!1,"wgAction":"view","wgUserName":null,"wgUserGroups":["*"],"wgCategories":["All articles with dead external links","Articles with dead external links from October 2016","CS1 German-langua...
```

Titre du contenu HTML

```
#Titre de la page Wikipédia
title = soup.title.text
print(title)

Linkin Park discography - Wikipedia

#type des albums
spans=soup.findAll("span",class_="toc-text")

len(spans)

21

al=[e.text for e in spans]

print("al, sepa", ")

Albums, Studio albums, Live albums, Compilation albums, Remix albums, Soundtrack albums, Extended plays, LP Underground extended plays, Singles, As lead artist, As featured artist, Promotional singles, Other charted songs, Other appearances, Video albums, Music videos, Traditional videos, Lyric videos, Notes, References, External links
```

Trouver tous les tableaux

```
# trouver tous les tableaux dans le html
all_tables = soup.findAll('table')
```

Trouver les tableaux à extraire

```
#obtenir les bonnes tables à extraire
table=soup.find('table', {"class":"wikitable plainrowheaders"})
#la fonction "find()" nous a permis d'afficher le premier tableau
# la variable "table" represente le premier tableau qui a pour "class="wikitable plainrowheaders"
```

Studio albums

Le nombre de colonnes

```
# Nombre de colonnes dans le tableau
for row in table.findAll('tr'):
    cells = row.findAll('td')

len(cells)

13
```

le nombre de lignes

```
#nombre de lignes dans le tableau, en-tête compris
rows = table.findAll("tr")
len(rows)

9
```

Obtenir des données tabulaires

L'analyse des données

```
lst_data = []
for row in rows[1:]:
    data = [d.text.rstrip() for d in row.find_all('td')]
    lst_data.append(data)

# select also works as find_all
lst_data1 = []
for row in rows[1:]:
    data = [d.text.rstrip() for d in row.select('td')]
    lst_data1.append(data)

## exemples d'enregistrements
lst_data1[1:3]

[['\nReleased: October 24, 2000[28]\nLabel: Warner Bros.\nFormats: CD, cassette, LP, download',
 '2',
 '1',
 '2',
 '5',
 '17',
 '2',
 '4',
 '1',
 '5',
 '4'],
 ['\nmms: 30,000,000[29]\nUS: 10,500,000[30]\nUK: 1,582,943[31]',
 '\nRIAA: 12x Platinum(Diamond)[32]\nARIA: 5x Platinum[33]\nBPI: 6x Platinum[34]\nBVM: 3x Platinum[35]\nIFPI AUT: Platinum[36]\nIFPI SWI: Platinum[37]\nMC: 5x Platinum[38]\nRMWZ: 5x Platinum[39]\nSNEP: Platinum[40]'],
 ['\nReleased: March 25, 2003[41]\nLabel: Warner Bros., Machine Shop\nFormats: CD, cassette, LP, download',
 '1',
 '2',
 '1',
 '2',
 '3',
 '1',
 '1',
 '1',
 '1',
 '1',
 '1',
 '\nmms: 27,000,000[29]\nUS: 6,200,000[30]',
 '\nRIAA: 7x Platinum[32]\nARIA: 4x Platinum[42]\nBPI: 3x Platinum[34]\nBVM: 4x Platinum[35]\nIFPI AUT: Platinum[36]\nIFPI SWI: Platinum[37]\nMC: 4x Platinum[38]\nRMWZ: 3x Platinum[43]\nSNEP: 2x Platinum[40]']]

#longueur de chaque enregistrement
```

```
len(list_data)])

13

list_row = []
for row in table.findall('tr'):
    list_row.append(row)

print('Number of row :',len(list_row))
print('.....')
print(list_row[1])
print('-----')
print('Second Attribute is has link reference')
print('-----')
print(list_row[1].findall('th'))
print('-----')
print(list_row[1].find('a').text)

Number of row : 9
-----
<tr>
<th scope="col" style="width:2.5em;font-size:90%;><a href="/wiki/Billboard_200" title="Billboard 200">US</a><br><sup class="reference" id="cite_ref-US-albums_18-0"><a href="#cite_note-US-albums-18">[18]</a></sup>
</th>
<th scope="col" style="width:2.5em;font-size:90%;><a href="/wiki/ARIA_Charts" title="ARIA Charts">AUS</a><br><sup class="reference" id="cite_ref-AUS_19-0"><a href="#cite_note-AUS-19">[19]</a></sup>
</th>
<th scope="col" style="width:2.5em;font-size:90%;><a href="/wiki/XC3963_Austria_Top_40" title="Ö3 Austria Top 40">AUT</a><br><sup class="reference" id="cite_ref-AUT_20-0"><a href="#cite_note-AUT-20">[20]</a></sup>
</th>
<th scope="col" style="width:2.5em;font-size:90%;><a href="/wiki/Canadian_Albums_Chart" title="Canadian Albums Chart">CAN</a><br><sup class="reference" id="cite_ref-CAN-albums_21-0"><a href="#cite_note-CAN-albums-21">[21]</a></sup>
</th>
<th scope="col" style="width:2.5em;font-size:90%;><a href="/wiki/Syndicat_National_de_13273C389dition_Phonographique" title="Syndicat National de l'Édition Phonographique">FRA</a><br><sup class="reference" id="cite_ref-FRA_22-0"><a href="#cite_note-FRA-22">[22]</a></sup>
</th>
<th scope="col" style="width:2.5em;font-size:90%;><a class="mw-redirect" href="/wiki/6FK_Entertainment_Charts" title="6FK Entertainment Charts">GER</a><br><sup class="reference" id="cite_ref-GER-albums_23-0"><a href="#cite_note-GER-albums-23">[23]</a></sup>
</th>
<th scope="col" style="width:2.5em;font-size:90%;><a href="/wiki/Irish_Albums_Chart" title="Irish Albums Chart">IRL</a><br><sup class="reference" id="cite_ref-IRL_24-0"><a href="#cite_note-IRL-24">[24]</a></sup>
</th>
<th scope="col" style="width:2.5em;font-size:90%;><a href="/wiki/Official_New_Zealand_Music_Chart" title="Official New Zealand Music Chart">NZ</a><br><sup class="reference" id="cite_ref-NZ_25-0"><a href="#cite_note-NZ-25">[25]</a></sup>
</th>
<th scope="col" style="width:2.5em;font-size:90%;><a href="/wiki/Swiss_Hitparade" title="Swiss Hitparade">SWI</a><br><sup class="reference" id="cite_ref-SWI_26-0"><a href="#cite_note-SWI-26">[26]</a></sup>
</th>
<th scope="col" style="width:2.5em;font-size:90%;><a href="/wiki/UK_Albums_Chart" title="UK Albums Chart">UK</a><br><sup class="reference" id="cite_ref-UK-albums_27-0"><a href="#cite_note-UK-albums-27">[27]</a></sup>
</th></tr>
-----
Second Attribute is has link reference
-----
<th scope="col" style="width:2.5em;font-size:90%;><a href="/wiki/Billboard_200" title="Billboard 200">US</a><br><sup class="reference" id="cite_ref-US-albums_18-0"><a href="#cite_note-US-albums-18">[18]</a></sup>
</th>
<th scope="col" style="width:2.5em;font-size:90%;><a href="/wiki/ARIA_Charts" title="ARIA Charts">AUS</a><br><sup class="reference" id="cite_ref-AUS_19-0"><a href="#cite_note-AUS-19">[19]</a></sup>
</th>
<th scope="col" style="width:2.5em;font-size:90%;><a href="/wiki/XC3963_Austria_Top_40" title="Ö3 Austria Top 40">AUT</a><br><sup class="reference" id="cite_ref-AUT_20-0"><a href="#cite_note-AUT-20">[20]</a></sup>
</th>
<th scope="col" style="width:2.5em;font-size:90%;><a href="/wiki/Canadian_Albums_Chart" title="Canadian Albums Chart">CAN</a><br><sup class="reference" id="cite_ref-CAN-albums_21-0"><a href="#cite_note-CAN-albums-21">[21]</a></sup>
</th>
<th scope="col" style="width:2.5em;font-size:90%;><a href="/wiki/Syndicat_National_de_13273C389dition_Phonographique" title="Syndicat National de l'Édition Phonographique">FRA</a><br><sup class="reference" id="cite_ref-FRA_22-0"><a href="#cite_note-FRA-22">[22]</a></sup>
</th>
<th scope="col" style="width:2.5em;font-size:90%;><a class="mw-redirect" href="/wiki/6FK_Entertainment_Charts" title="6FK Entertainment Charts">GER</a><br><sup class="reference" id="cite_ref-GER-albums_23-0"><a href="#cite_note-GER-albums-23">[23]</a></sup>
</th>
<th scope="col" style="width:2.5em;font-size:90%;><a href="/wiki/Irish_Albums_Chart" title="Irish Albums Chart">IRL</a><br><sup class="reference" id="cite_ref-IRL_24-0"><a href="#cite_note-IRL-24">[24]</a></sup>
</th>
<th scope="col" style="width:2.5em;font-size:90%;><a href="/wiki/Official_New_Zealand_Music_Chart" title="Official New Zealand Music Chart">NZ</a><br><sup class="reference" id="cite_ref-NZ_25-0"><a href="#cite_note-NZ-25">[25]</a></sup>
</th>
<th scope="col" style="width:2.5em;font-size:90%;><a href="/wiki/Swiss_Hitparade" title="Swiss Hitparade">SWI</a><br><sup class="reference" id="cite_ref-SWI_26-0"><a href="#cite_note-SWI-26">[26]</a></sup>
</th>
<th scope="col" style="width:2.5em;font-size:90%;><a href="/wiki/UK_Albums_Chart" title="UK Albums Chart">UK</a><br><sup class="reference" id="cite_ref-UK-albums_27-0"><a href="#cite_note-UK-albums-27">[27]</a></sup>
</th></tr>
-----
US
```

Obtenir les attributs d'en-tête de table

```
#attributs d'en-tête du tableau
header = [th.text.rstrip() for th in rows[0].find_all('th')]
header = header[0:2] + [th.text.rstrip()[::-4] for th in rows[1].find_all('th')] + header[3:]
print(header)

['Title', 'Album details', 'US', 'AUS', 'AUT', 'CAN', 'FRA', 'GER', 'IRL', 'NZ', 'SWI', 'UK', 'Sales', 'Certifications']
```

Extraire les données

```
#Extraire les données et ajoutez-les aux listes respectives
lst_data = []
c = [ [], [], [], [], [], [], [], [], [], [], [], [] ]
for row in rows[2:]:
    c[0].append(row.find('th').text.strip())
    for i, x in enumerate(row.find_all('td')):
        c[i+1].append(x.text.strip())
```

Créer un dictionnaire

```
# créer un dictionnaire
d = dict()
for i, h in enumerate(header):
    d[h] = c[i]
```

Creer un DataFrame

```
# convertir dict en DataFrame
df_table=pd.DataFrame(d)
df_table


```

	Title	Album details	US	AUS	AUT	CAN	FRA	GER	IRL	NZ	SWI	UK	Sales	Certifications
0	Hybrid Theory	Released: October 24, 2000[26]nLabel: Warner Br...	2	1	2	5	17	2	4	1	5	4	WW: 30,000,000[29]nUS: 10,500,000[30]nUK: 1,...	RIAA: 12× Platinum(Diamond)[32]nARIA: 5× Plat...
1	Meteora	Released: March 25, 2003[41]nLabel: Warner Br...	1	2	1	2	3	1	1	1	1	1	WW: 27,000,000[29]nUS: 6,200,000[30]	RIAA: 7× Platinum[32]nARIA: 4× Platinum[42]n...
2	Minutes to Midnight	Released: May 15, 2007[44]nLabel: Warner Bros...	1	1	1	1	1	1	1	1	1	1	WW: 20,000,000[45]nUS: 3,300,000[46]nUK: 600...	RIAA: 4× Platinum[32]nARIA: 3× Platinum[48]n...
3	A Thousand Suns	Released: September 14, 2010[50]nLabel: Warne...	1	1	1	1	4	1	3	1	1	2	US: 906,000[46]	RIAA: Platinum[32]nARIA: Platinum[51]nBPI: P...
4	Living Things	Released: June 26, 2012[54]nLabel: Warner Bro...	1	2	1	1	2	1	2	1	1	1	US: 681,000[46]	RIAA: Platinum[32]nARIA: Gold[55]nBPI: Gold[...
5	The Hunting Party	Released: June 17, 2014[57]nLabel: Warner Bro...	3	3	2	3	3	1	6	2	1	2	US: 231,150[58]	RIAA: Platinum[32]nBPI: Gold[34]nBVMi: Plati...
6	One More Light	Released: May 19, 2017[59]nLabel: Warner Bros...	1	3	1	1	14	2	6	4	1	4	US: 270,000[60]	RIAA: Gold[32]nBPI: Gold[34]nBVMi: Gold[35]...

```
# 5 premiers records
df_table.tail(5)


```

	Title	Album details	US	AUS	AUT	CAN	FRA	GER	IRL	NZ	SWI	UK	Sales	Certifications
2	Minutes to Midnight	Released: May 15, 2007[44]nLabel: Warner Bros...	1	1	1	1	1	1	1	1	1	1	WW: 20,000,000[45]nUS: 3,300,000[46]nUK: 600...	RIAA: 4× Platinum[32]nARIA: 3× Platinum[48]n...
3	A Thousand Suns	Released: September 14, 2010[50]nLabel: Warne...	1	1	1	1	4	1	3	1	1	2	US: 906,000[46]	RIAA: Platinum[32]nARIA: Platinum[51]nBPI: P...
4	Living Things	Released: June 26, 2012[54]nLabel: Warner Bro...	1	2	1	1	2	1	2	1	1	1	US: 681,000[46]	RIAA: Platinum[32]nARIA: Gold[55]nBPI: Gold[...
5	The Hunting Party	Released: June 17, 2014[57]nLabel: Warner Bro...	3	3	2	3	3	1	6	2	1	2	US: 231,150[58]	RIAA: Platinum[32]nBPI: Gold[34]nBVMi: Plati...
6	One More Light	Released: May 19, 2017[59]nLabel: Warner Bros...	1	3	1	1	14	2	6	4	1	4	US: 270,000[60]	RIAA: Gold[32]nBPI: Gold[34]nBVMi: Gold[35]...

```
#generer le fichier csv
df_table.to_csv("Albums_studio.csv",index=False,encoding="utf-8")
```

Live albums

```
#recupérons tous les tableaux
table_soup=soup.find_all("table")
table_soup

</div></td></tr><tr><th class="navbox-group" scope="row" style="width:1%;background: #EEEEEE;">Studio albums</th><td class="navbox-list navbox-odd" style="text-align:left;border-left-width:2px;border-left-style:solid;width:100%;padding:0px"><div style="padding:0em 0.25em">
<ul><li><a href="/wiki/Hybrid_Theory" title="Hybrid Theory">Hybrid Theory</a></li></ul>
<li><a href="/wiki/Meteora_(album)" title="Meteora (album)">Meteora</a></li></li>
<li><a href="/wiki/Minutes_to_Midnight_(Linkin_Park_album)" title="Minutes to Midnight (Linkin Park album)">Minutes to Midnight</a></li></li>
<li><a href="/wiki/A_Thousand_Suns" title="A Thousand Suns">A Thousand Suns</a></li></li>
<li><a href="/wiki/Living_Things_(Linkin_Park_album)" title="Living Things (Linkin Park album)">Living Things</a></li></li>
<li><a href="/wiki/The_Hunting_Party_(album)" title="The Hunting Party (album)">The Hunting Party</a></li></li>
<li><a href="/wiki/One_More_Light" title="One More Light">One More Light</a></li></li></ul>
</div></td></tr><tr><th class="navbox-group" scope="row" style="width:1%;background: #EEEEEE;">Remix albums</th><td class="navbox-list navbox-even" style="text-align:left;border-left-width:2px;border-left-style:solid;width:100%;padding:0px"><div style="padding:0em 0.25em">
<ul><li><a href="/wiki/Reanimation_(Linkin_Park_album)" title="Reanimation (Linkin Park album)">Reanimation</a></li></li></ul>
<li><a href="/wiki/Recharged_(album)" title="Recharged (album)">Recharged</a></li></li></ul>
</div></td></tr><tr><th class="navbox-group" scope="row" style="width:1%;background: #EEEEEE;">Live albums</th><td class="navbox-list navbox-odd" style="text-align:left;border-left-width:2px;border-left-style:solid;width:100%;padding:0px"><div style="padding:0em 0.25em">
<ul><li><a href="/wiki/Live_in_Texas_(Linkin_Park_album)" title="Live in Texas (Linkin Park album)">Live in Texas</a></li></li>
<li><a href="/wiki/Road_to_Revolution: Live at Milton Keynes" title="Road to Revolution: Live at Milton Keynes">Road to Revolution: Live at Milton Keynes</a></li></li>
<li><a href="/wiki/One_More_Light_Live" title="One More Light Live">One More Light Live</a></li></li></ul>
</div></td></tr><tr><th class="navbox-group" scope="row" style="width:1%;background: #EEEEEE;">Compilations</th><td class="navbox-list navbox-even" style="text-align:left;border-left-width:2px;border-left-style:solid;width:100%;padding:0px"><div style="padding:0em 0.25em">
<ul><li><a href="/wiki/A_Decade_Underground" title="A Decade Underground">A Decade Underground</a></li></li>
<li><a href="/wiki/Studio_Collection_2000&E28089&E28012" title="Studio Collection 2000–2012">Studio Collection 2000–2012</a></li></li>
<li><a href="/wiki/Hybrid_Theory&E28th_anniversary_edition" title="Hybrid Theory<sup>3</sup>Hybrid Theory</i> (20th Anniversary Edition)"></a></li></li>
</ul></td></tr><tr><td></td><td class="navbox-list navbox-odd" style="text-align:left;border-left-width:2px;border-left-style:solid;width:100%;padding:0px"><div style="padding:0em 0.25em">
<ul><li><a href="/wiki/Collision_Course_(EP)" title="Collision Course (EP)">Collision Course</a></li></ul>
<li><a href="/wiki/Songs_from_the_Underground" title="Songs from the Underground">Songs from the Underground</a></li></li>
</ul></td></tr><tr><th class="navbox-group" scope="row" style="width:1%;background: #EEEEEE;">Video albums</th><td class="navbox-list navbox-even" style="text-align:left;border-left-width:2px;border-left-style:solid;width:100%;padding:0px"><div style="padding:0em 0.25em">
<ul><li><a href="/wiki/From_Party_at_the_Punkake_Festival" title="From Party at the Punkake Festival">From Party at the Punkake Festival</a></li></ul>
<li><a class="mw-redirect" href="/wiki/A_Thousand_Suns_328" title="A Thousand Suns +>A Thousand Suns +</a></li></li>
<li><a href="/wiki/Living_Things_328" title="Living Things +>Living Things +</a></li></li></ul>
</div></td></tr><tr><td></td><td class="navbox-list navbox-odd" style="text-align:left;border-left-width:2px;border-left-style:solid;width:100%;padding:0px"><div style="padding:0em 0.25em">
<ul><li><a href="/wiki/8-Bit_Rebellion!_(soundtrack)" title="8-Bit Rebellion! (soundtrack)">8-Bit Rebellion!</a></li></ul>
<li><a href="/wiki/Mall_(soundtrack)" title="Mall (soundtrack)">Mall</a></li></li></ul>
</div></td></tr><tr><th class="navbox-group" scope="row" style="width:1%;background: #EEEEEE;">Tours</th><td class="navbox-list navbox-even" style="text-align:left;border-left-width:2px;border-left-style:solid;width:100%;padding:0px"><div style="padding:0em 0.25em">
<ul><li><a href="/wiki/Projekt_Revolution" title="Projekt Revolution">Projekt Revolution</a></li>
<li><a href="/wiki/Minutes_to_Midnight_World_Tour" title="Minutes to Midnight World Tour">Minutes to Midnight World Tour</a></li>
<li><a href="/wiki/A_Thousand_Suns_World_Tour" title="A Thousand Suns World Tour">A Thousand Suns World Tour</a></li>
<li><a href="/wiki/11th_Annual_Honda_Civic_Tour" title="11th Annual Honda Civic Tour">11th Annual Honda Civic Tour</a></li>
<li><a href="/wiki/Living_Things_World_Tour" title="Living Things World Tour">Living Things World Tour</a></li>
<li><a href="/wiki/Carnivores_Tour" title="Carnivores Tour">Carnivores Tour</a></li>
<li><a href="/wiki/The_Hunting_Party_Tour" title="The Hunting Party Tour">The Hunting Party Tour</a></li>
<li><a href="/wiki/One_More_Light_World_Tour" title="One More Light World Tour">One More Light World Tour</a></li></ul>
</div></td></tr><tr><td></td><td class="navbox-list navbox-odd" style="text-align:left;border-left-width:2px;border-left-style:solid;width:100%;padding:0px"><div style="padding:0em 0.25em">
<ul><li><a class="mw-selflink selflink">Discography</a></li>
<li><a href="/wiki/List_of_songs_recorded_by_Linkin_Park" title="List of songs recorded by Linkin Park">List of songs</a></li>
<li><a href="/wiki/List_of_wards_and_nominations_received_by_Linkin_Park" title="List of awards and nominations received by Linkin Park">Awards and nominations</a></li>
<li><a href="/wiki/Machine_Shop_Records" title="Machine Shop Records">Machine Shop Records</a></li>
<li><a href="/wiki/Machine_Shop_co." title="Machine Shop co.">Machine Shop co.</a></li>
<li><a href="/wiki/Download_to_Donate" title="Download to Donate">Download to Donate</a></li>
<li><a href="/wiki/Music_for_Relief" title="Music for Relief">Music for Relief</a></li>
<li><a href="/wiki/8-Bit_Rebellion!_EP" title="8-Bit Rebellion!>8-Bit Rebellion!</a></li>
<li><a href="/wiki/IP_Recharge" title="IP Recharge">IP Recharge</a></li>
<li><a href="/wiki/The_Seeds_(film)" title="The Seeds (film)">The Seeds</a></li>
<li><a class="mw-redirect" href="/wiki/Fuse_Present:_Linkin_Park_Live_at_MSG" title="Fuse Present: Linkin Park Live at MSG">Fuse Present: Linkin Park Live at MSG</a></li>
<li><a href="/wiki/Inside_Living_Things" title="Inside Living Things">Inside Living Things</a></li>
<li><a href="/wiki/Mall_(film)" title="Mall (film)">Mall</a></li>
<li><a href="/wiki/Dead_by_Sunrise" title="Dead by Sunrise">Dead by Sunrise</a></li>
<li><a href="/wiki/Fort_Minor" title="Fort Minor">Fort Minor</a></li>
<li><a href="/wiki/Stone_Temple_Pilots" title="Stone Temple Pilots">Stone Temple Pilots</a></li>
<li><a class="mw-redirect" href="/wiki/In_the_Chamber_with_Linkin_Park" title="In the Chamber with Linkin Park">In the Chamber with Linkin Park</a></li>
<li><a href="/wiki/Linkin_Park_and_Friends_Celebrate_Life_in_Honor_of_Chester_Bennington" title="Linkin Park and Friends: Celebrate Life in Honor of Chester Bennington">Linkin Park and Friends: Celebrate Life in Honor of Chester Bennington</a></li></ul>
</div></td></tr></table>
```

```
#recupere tous les tableaux qui ont des attributs <caption>
filtered_table_soup=[table for table in table_soup if table.caption is not None]

required_table=None

#recupere un tableau par l'attribut <caption>
for table in filtered_table_soup:
    if str(table.caption.string.strip())=="list of live albums, with selected chart positions, sales figures and certifications":
        required_table=table
        break

print(required_table)

<table class="wikitable plainrowheaders" style="text-align:center;">
<caption>List of live albums, with selected chart positions, sales figures and certifications
</caption>
<tbody><tr>
<th rowspan="2">
<th scope="col" style="width:10em;">Title
</th>
```

[illegible]

▼ Extraire les données

```

#Extraire les données et ajouter-les aux listes respectives
lst_data0 = []
c0= [ [], [], [], [], [], [], [], [], [], [], [], [] ]
for row in row0[2:-1]:
    c0[0].append(row.find('th').text.strip())
    for i, x in enumerate(row.find_all('td')):
        c0[i+1].append(x.text.strip())
print(c0)

[['Live in Texas', 'Road to Revolution:Live at Milton Keynes', 'One More Light Live'], ['Released: November 18, 2003[62]\nLabel: Warner Bros., Machine Shop\nFormats: CD, CD+DVD, download', 'Released: November 21, 2008[65]\nLabel: Warner Bros., Machine Shop\nFormats: CD, CD+DVD, BD, download', 'Released: December 15, 2017[67]\nLabel: Warner Bros., Machine Shop\nFormats: CD, LP, download'], ['

# créer un dictionnaire
d0 = dict()
for i, h in enumerate(header):
    d0[h] = c0[i]

# convertir dict en DataFrame
df_table0=pd.DataFrame(d0)
df_table0



|   | Title                                    | Album details                                     | US | AUS | AUT | CAN | FRA | GER | IRL | NZ | SWI | UK | Sales             | Certifications                                    |
|---|------------------------------------------|---------------------------------------------------|----|-----|-----|-----|-----|-----|-----|----|-----|----|-------------------|---------------------------------------------------|
| 0 | Live in Texas                            | Released: November 18, 2003[62]\nLabel: Warner... | 23 | 18  | 5   | —   | 8   | 9   | 67  | 17 | 9   | 47 | US: 1,100,000[46] | RIAA: Platinum[32]\nARIA: Gold[63]\nBPI: Gold[... |
| 1 | Road to Revolution:Live at Milton Keynes | Released: November 21, 2008[65]\nLabel: Warner... | 41 | 24  | 14  | 35  | 12  | 11  | 81  | 17 | 16  | 58 | US: 230,000[46]   | BPI: Silver[34]\nBVM: Platinum[35]\nFPI SWI:...   |
| 2 | One More Light Live                      | Released: December 15, 2017[67]\nLabel: Warner... | 28 | 20  | 11  | 29  | 54  | 7   | —   | 32 | 7   | 32 |                   |                                                   |



#generer le fichier csv
df_table0.to_csv("live_albums.csv",index=False,encoding="utf-8")

```

- ▼ Compilation albums

- ▼ Extraire les données

```
#Extraire les données et ajouter-les aux listes respectives
list_dataA = []
cA= [ [], [], [], [] ]
for row in rowsA[2:]:
    cA[0].append(row.find('th').text.strip())
    for i, x in enumerate(row.find_all('td')):
        cA[i+1].append(x.text.strip())

print(cA)

[['A Decade Underground', 'Studio Collection 2000-2012'], ['Released: August 10, 2010[68]\nLabel: Machine Shop\nformats: Download (Only for LP Underground Members)', 'Digital Box set collection\nReleased: January 15, 2013[69]\nLabel: Warner Bros.\nformats: Download'], ['-', '49']]

# créer un dictionnaire
dA = dict()
for i, h in enumerate(headerA):
    dA[h] = cA[i]

# convertir dict en DataFrame
df_tableA=pd.DataFrame(dA)
of_tableA



|   | Title                       | Album details                                     | AUS |
|---|-----------------------------|---------------------------------------------------|-----|
| 0 | A Decade Underground        | Released: August 10, 2010[68]\nLabel: Machine ... | —   |
| 1 | Studio Collection 2000-2012 | Digital Box set collection\nReleased: January ... | 49  |



#generer le fichier csv
df_tableA.to_csv("compilation_albums.csv",index=False,encoding="utf-8")
```

- ▼ Remix albums

1

1

1

[illegible]

```

#attributs d'en-tête du tableau
header3 = [th.txt.rstrip() for th in rows3[0].find_all('th')]
header3 = header3[0:2] + [th.txt.rstrip().split('|')[0] for th in rows3[1].find_all('th')] + header3[3:]
print(header3)

['Title', 'EP details', 'US', 'AUS', 'AUT', 'CAN', 'FRA', 'GER', 'IRL', 'NZ', 'SWI', 'UK', 'Sales', 'Certifications']

```

- ▼ Extraire les données

```
#Extraire les données et ajoutez-les aux listes respectives
```

```
1st_data3 = []
c3= [ [], [], [], [], [], [], [], [], [], [], [] ]
for row in rows3[2:-1]:
    c3[0].append(row.find('th').text.strip())
    for i, x in enumerate(row.find_all('td')):
        c3[i+1].append(x.text.strip())
print(c3)
```

[['Hybrid Theory EP (As Hybrid Theory)', 'In the End: Live & Rare', 'Collision Course (with Jay-Z)', 'iTunes Live from SoHo', 'Songs from the Underground', '2011 North American Tour', 'A Thousand Suns: Puerta de Alcalá', 'iTunes Festival: London 2011', 'Stagelight Demos', 'A Light That Never Comes (Remixes) (with Steve Aoki)', 'Darker Than Blood: Remixes - EP (with Steve Aoki)', '[Released: May

```
# créer un dictionnaire
d3 = dict()
for i, h in enumerate(header):
    d3[h] = c3[i]
```

```
# convertir dict en DataFrame
df_table3=pd.DataFrame(d3)
df_table3
```

[illegible]

```
#generer le fichier csv
df_table3.to_csv("extended_plays.csv",index=False,encoding="utf-8")
```

- LP Underground extended plays

```
#recupere tous les tableaux qui ont des attributs "<caption></caption>"
filtered_table_soup4=[table for table in table_soup if table.caption is not None]
```

```
required_table4=None
```

```
for i in range(len(filtered_table_soup4)):
    if filtered_table_soup4[i].caption.text.strip()=="List of extended plays, with selected chart positions, sales figures and certifications"
        required_table4=filtered_table_soup4[i+1]
        break
print(required_table4)
```

```
attributs d'en-tête du tableau
header4 = [th.text.rstrip() for th in rows4[0].find_all('th')]
header4 = header4[0:2] + [th.text.rstrip().split(',')[0] for th in rows4[1].find_all('th')]
print(header4)

['Title', 'EP details', 'AUT', 'GER', 'SWI']
```

- ▼ Extraire les données

```
#Extraire les données et ajoutez-les aux listes respectives
```

```
1st_data4 = []
c4= [ [], [], [], [], [], [], [], [], [], [], [] ]
for now in rows4[2:-1]:
    c4[0].append(row.find('th').text.strip())
    for i, x in enumerate(row.find_all('td')):
        c4[i+1].append(x.text.strip())
print(c4)
```

[['Underground 2.0[92]', 'Underground 3.0[92]', 'Underground 4.0[92]', 'Underground 5.0[92]', 'Underground 6[92]', 'LP Underground 7[92]', 'mm... Cookies: Sweet Hamster Like Jewels FromAmerica (Underground 8.0[92]', 'Underground 9.0: Demos[92]', 'LP Underground X: Demos[92]', 'Underground 11[92]', 'Underground 12[92]', 'Underground XIII[92]', 'Underground XIV[92]', 'Underground 15[92]',

```
# créer un dictionnaire
d4 = dict()
for i, h in enumerate(header4):
    d4[h] = c4[i]
```

```
# convertir dict en DataFrame
df_table4=pd.DataFrame(d4)
df_table4
```

	Title	EP details	AUT	GER	SWI
0	Underground 2.0[92]	Released: November 18, 2002nType: EPnLabel: ...	—	—	—
1	Underground 3.0[92]	Released: November 17, 2003nType: EPnLabel: ...	—	—	—
2	Underground 4.0[92]	Released: November 22, 2004nType: EPnLabel: ...	—	—	—
3	Underground 5.0[92]	Released: November 21, 2005nType: EPnLiveInLa...	—	—	—
4	Underground 6[92]	Released: December 5, 2006nType: EPnLabel: M...	—	—	—
5	LP Underground 7[92]	Released: December 5, 2007[93]nType: EPnLive...	—	—	—
6	mmm... Cookies: Sweet Hamster Like Jewels From...	Released: December 4, 2008nType: EPnLabel: W...	—	—	—
7	Underground 9.0: Demos[92]	Released: December 3, 2009[94]nType: DemoInLa...	73	66	29
8	LP Underground X: Demos[92]	Released: November 17, 2010[95]nType: DemoInL...	—	—	—
9	Underground 11[92]	Released: November 15, 2011[96]nType: DemoInL...	—	—	—
10	Underground 12[92]	Released: November 16, 2012[97]nType: DemoInL...	—	—	—
11	Underground XIII[92]	Released: November 18, 2013[98]nType: DemoInL...	—	—	—

```
#generer le fichier csv
df_table4.to_csv("lp_underground_extended.csv",index=False,encoding="utf-8")
```

Singles

```
#recupere tous les tableaux qui ont des attributs "<caption></caption>"
filtered_table_soup=[table for table in table_soup if table.caption is not None]

required_table5=None

for table in filtered_table_soup5:
    if str(table.caption.string).strip()!="List of singles as lead artist, with selected chart positions and certifications, showing year released and album name":
        required_table5=table
        break

print(required_table5)

<table class="wikitable plainrowheaders" style="text-align:center">
<caption>List of singles as lead artist, with selected chart positions and certifications, showing year released and album name
</caption>
<tbody><tr>
<th rowspan="2">" scope="col" style="width:18em">Title
</th>
<th rowspan="2">" scope="col">Year
</th>
<th colspan="10">" scope="col">Peak chart positions
</th>
<th rowspan="2">" scope="col" style="width:12em;"><a href="/wiki/List_of_music_recording_certifications" title="List of music recording certifications">Certifications</a>
</th>
<th rowspan="2">" scope="col" style="width:12em;"><a href="/wiki/List_of_music_recording_certifications" title="List of music recording certifications">Album
</th></tr>
<tr>
<th scope="col" style="width:3em;font-size:90%;"><a href="/wiki/Billboard_Hot_100" title="Billboard Hot 100">US</a><br><sup class="reference" id="cite_ref-US-singles_5-1"><a href="#cite_note-US-singles-5">[5]</a></sup>
</th>
<th scope="col" style="width:3em;font-size:90%;"><a href="/wiki/Alternative_Songs" title="Alternative Songs">US</a><br><sup class="reference" id="cite_ref-US-Alternative-singles_99-0"><a href="#cite_note-US-Alternative-singles-99">[99]</a></sup>
</th>
<th scope="col" style="width:3em;font-size:90%;"><a href="/wiki/List_of_number-one_Billboard_Rock_Songs">US</a><br><sup class="reference" id="cite_ref-100"><a href="#cite_note-100">[100]</a></sup>
</th>
<th scope="col" style="width:3em;font-size:90%;"><a href="/wiki/ARIA_Charts" title="ARIA Charts">AUS</a><br><sup class="reference" id="cite_ref-AUS-singles_101-0"><a href="#cite_note-AUS-singles-101">[101]</a></sup>
</th>
<th scope="col" style="width:3em;font-size:90%;"><a href="/wiki/Canadian_Hot_100" title="Canadian Hot 100">CAN</a><br><sup class="reference" id="cite_ref-CAN_102-0"><a href="#cite_note-CAN-102">[102]</a></sup>
</th>
<th scope="col" style="width:3em;font-size:90%;"><a href="/wiki/Syndicat_National_de_l'edition_Phonographique">FRA</a><br><sup class="reference" id="cite_ref-FRA_22-4"><a href="#cite_note-FRA-22">[22]</a></sup>
</th>
<th scope="col" style="width:3em;font-size:90%;"><a href="/wiki/6fK_Entertainment_Charts" title="6fK Entertainment Charts">GER</a><br><sup class="reference" id="cite_ref-GER-singles_103-0"><a href="#cite_note-GER-singles-103">[103]</a></sup>
</th>
<th scope="col" style="width:3em;font-size:90%;"><a href="/wiki/Official_New_Zealand_Music_Chart" title="Official New Zealand Music Chart">NZ</a><br><sup class="reference" id="cite_ref-NZ_25-4"><a href="#cite_note-NZ-25">[25]</a></sup>
</th>
<th scope="col" style="width:3em;font-size:90%;"><a href="/wiki/Swiss_Hitparade" title="Swiss Hitparade">SWI</a><br><sup class="reference" id="cite_ref-SWI_26-5"><a href="#cite_note-SWI-26">[26]</a></sup>
</th>
<th scope="col" style="width:3em;font-size:90%;"><a href="/wiki/UK_Singles_Chart" title="UK Singles Chart">UK</a><br><sup class="reference" id="cite_ref-UK-singles_104-0"><a href="#cite_note-UK-singles-104">[104]</a></sup>
</th></tr>
<tr>
<th scope="row">"<a href="/wiki/One_Step_Closer_(Linkin_Park_song)" title="One Step Closer (Linkin Park song)">One Step Closer</a><sup class="reference" id="cite_ref-105"><a href="#cite_note-105">[105]</a></sup>
</th>
<td rowspan="1">2000
</td>
<td>5</td></td>
<td>5</td></td>
<td>4</td></td>
<td>4</td></td>
<td>32</td></td>
<td>42</td></td>
<td>24</td></td>
<td>RIAA: Platinum<sup class="reference" id="cite_ref-RIAA_32-10"><a href="#cite_note-RIAA-32">[32]</a></sup></li>
<li>ARIA: Gold<sup class="reference" id="cite_ref-ARIA-2001-singles_106-0"><a href="#cite_note-ARIA-2001-singles-106">[106]</a></sup></li>
<li>BPI: Gold<sup class="reference" id="cite_ref-BPI_34-11"><a href="#cite_note-BPI-34">[34]</a></sup></li></ul>
<td rowspan="4"><i>Hybrid Theory</i>
</td></tr>
<tr>
<td>75
</td>
<td>5
</td>
<td>3
</td>
<td>33
</td>
<td>—
</td>
<td>106
</td>
<td>14
</td>
<td>37
</td>
<td>43
</td>
<td>16
</td>
<td>RIAA: Gold<[32]nBPI: Gold<[34]
</td>
<td>Hybrid Theory
</td>
</tr>
<tr>
<td>—
</td>
<td>32
</td>
<td>18
</td>
<td>87
</td>
<td>43
</td>
<td>—
</td>
<td>49
</td>
<td>—
</td>
<td>80
</td>
<td>14
</td>
<td>RIAA: Gold<[32]nBPI: Silver<[34]
</td>
<td>Hybrid Theory
</td>
</tr>
<tr>
<td>"In the End"[109]
</td>
<td>2
</td>
<td>1
</td>
<td>3
</td>
<td>4
</td>
<td>2
</td>
<td>23
</td>
<td>4
</td>
<td>10
</td>
<td>4
</td>
<td>8
</td>
<td>RIAA: 4× Platinum<[32]nARIA: Gold<[110]nBPI: 2...
</td>
<td>Hybrid Theory
</td>
</tr>
<tr>
<td>"Pts.OF.Athrty"(with Jay Gordon)
</td>
<td>2002
</td>
<td>—
</td>
<td>29
</td>
<td>—
</td>
<td>44
</td>
<td>47
</td>
<td>—
</td>
<td>31
</td>
<td>—
</td>
<td>61
</td>
<td>9
</td>
<td>Reanimation
</td>
</tr>
<tr>
<td>"Somewhere I Belong"[111]
</td>
<td>2003
</td>
<td>32
</td>
<td>1
</td>
<td>1
</td>
<td>13
</td>
<td>16
</td>
<td>32
</td>
<td>12
</td>
<td>1
</td>
<td>15
</td>
<td>10
</td>
<td>ARIA: Gold<[112]nBPI: Silver<[34]
</td>
<td>Meteora
</td>
</tr>
<tr>
<td>"Faint"[111]
</td>
<td>2003
</td>
<td>48
</td>
<td>1
</td>
<td>2
</td>
<td>25
</td>
<td>27
</td>
<td>158
</td>
<td>40
</td>
<td>—
</td>
<td>32
</td>
<td>15
</td>
<td>RIAA: Platinum<[32]nBPI: Silver<[34]
</td>
<td>Meteora
</td>
</tr>
<tr>
<td>"Numb"[111]
</td>
<td>2003
</td>
<td>11
</td>
<td>1
</td>
<td>1
</td>
<td>10
</td>
<td>27
</td>
<td>9
</td>
<td>13
</td>
<td>13
</td>
<td>5
</td>
<td>14
</td>
<td>RIAA: 4× Platinum<[32]nARIA: Gold<[112]nBPI: 2...
</td>
<td>Meteora
</td>
</tr>
<tr>
<td>"From the Inside"[111]
</td>
<td>2004
</td>
<td>—
</td>
<td>—
</td>
<td>37
</td>
<td>43
</td>
<td>35
</td>
<td>35
</td>
<td>50
</td>
<td>38
</td>
<td>—
</td>
<td>—
</td>
<td>Meteora
</td>
</tr>
<tr>
<td>"Breaking the Habit"[111]
</td>
<td>2004
</td>
<td>20
</td>
<td>1
</td>
<td>1
</td>
<td>23
</td>
<td>43
</td>
<td>27
</td>
<td>25
</td>
<td>27
</td>
<td>56
</td>
<td>39
</td>
<td>RIAA: Gold<[32]nBPI: Silver<[34]
</td>
<td>Meteora
</td>
</tr>
<tr>
<td>"Numb/Encore"(with Jay-Z)[113]
</td>
<td>2004
</td>
<td>20
</td>
<td>—
</td>
<td>17
</td>
<td>3
</td>
<td>3
</td>
<td>5
</td>
<td>4
</td>
<td>—
</td>
<td>10
</td>
<td>14
</td>
<td>RIAA: 3× Platinum<[32]nARIA: Platinum<[114]nBP...
</td>
<td>Collision Course
</td>
</tr>
<tr>
<td>"What I've Done"[115]
</td>
<td>2007
</td>
<td>7
</td>
<td>1
</td>
<td>7
</td>
<td>13
</td>
<td>3
</td>
<td>107
</td>
<td>3
</td>
<td>9
</td>
<td>6
</td>
<td>6
</td>
<td>RIAA: 5× Platinum<[32]nBPI: Platinum<[34]nBVMi...
</td>
<td>Minutes to Midnight
</td>
</tr>
<tr>
<td>"Bleed It Out"[117]
</td>
<td>2007
</td>
<td>52
</td>
<td>2
</td>
<td>16
</td>
<td>24
</td>
<td>22
</td>
<td>—
</td>
<td>40
</td>
<td>7
</td>
<td>42
</td>
<td>29
</td>
<td>RIAA: 2× Platinum<[32]nBPI: Silver<[34]nRMNZ: ...
</td>
<td>Minutes to Midnight
</td>
</tr>
<tr>
<td>"Shadow of the Day"[118]
</td>
<td>2007
</td>
<td>15
</td>
<td>2
</td>
<td>19
</td>
<td>15
</td>
<td>12
</td>
<td>20
</td>
<td>12
</td>
<td>13
</td>
<td>11
</td>
<td>46
</td>
<td>RIAA: 2× Platinum<[32]nIFPI SWI: Gold<[37]nRMN...
</td>
<td>Minutes to Midnight
</td>
</tr>
<tr>
<td>"Given Up"[119]
</td>
<td>2008
</td>
<td>99
</td>
<td>4
</td>
<td>—
</td>
<td>—
</td>
<td>—
</td>
<td>—
</td>
<td>53
</td>
<td>—
</td>
<td>—
</td>
<td>—
</td>
<td>RIAA: Gold<[32]
</td>
<td>Minutes to Midnight
</td>
</tr>
<tr>
<td>"Leave Out All the Rest"[120]
</td>
<td>2008
</td>
<td>94
</td>
<td>11
</td>
<td>22
</td>
<td>24
</td>
<td>—
</td>
<td>17
</td>
<td>15
</td>
<td>38
</td>
<td>36
</td>
<td>90
</td>
<td>RIAA: Platinum<[32]
</td>
<td>Minutes to Midnight
</td>
</tr>
<tr>
<td>"New Divide"[121]
</td>
<td>2009
</td>
<td>6
</td>
<td>1
</td>
<td>1
</td>
<td>3
</td>
<td>3
</td>
<td>—
</td>
<td>4
</td>
<td>2
</td>
<td>7
</td>
<td>19
</td>
<td>RIAA: 3× Platinum<[32]nARIA: Platinum<[122]nBP...
</td>
<td>Transformers: Revenge of the Fallen – The Album
</td>
</tr>
<tr>
<td>"The Catalyst"[123]
</td>
<td>2010
</td>
<td>27
</td>
<td>1
</td>
<td>1
</td>
<td>33
</td>
<td>28
</td>
<td>—
</td>
<td>11
</td>
<td>27
</td>
<td>29
</td>
<td>40
</td>
<td>RIAA: Gold<[32]
</td>
<td>A Thousand Suns
</td>
</tr>
<tr>
<td>"Waiting for the End"[124]
</td>
<td>2010
</td>
<td>42
</td>
<td>1
</td>
<td>2
</td>
<td>—
</td>
<td>55
</td>
<td>—
</td>
<td>29
</td>
<td>—
</td>
<td>58
</td>
<td>90
</td>
<td>RIAA: Platinum<[32]
</td>
<td>A Thousand Suns
</td>
</tr>
<tr>
<td>"Burning in the Skies"[125]
</td>
<td>2011
</td>
<td>—
</td>
<td>—
</td>
<td>—
</td>
<td>—
</td>
<td>—
</td>
<td>43
</td>
<td>—
</td>
<td>41
</td>
<td>—
</td>
<td>—
</td>
<td>A Thousand Suns
</td>
</tr>
<tr>
<td>"Indescent"[126]
</td>
<td>2011
</td>
<td>81
</td>
<td>19
</td>
<td>29
</td>
<td>39
</td>
<td>—
</td>
<td>—
</td>
<td>46
</td>
<td>—
</td>
<td>69
</td>
<td>93
</td>
<td>RIAA: Gold<[32]
</td>
<td>A Thousand Suns
</td>
</tr>
<tr>
<td>"Not Alone"[127]
</td>
<td>2011
</td>
<td>—
</td>
<td>—
</td>
<td>—
</td>
<td>—
</td>
<td>—
</td>
<td>—
</td>
<td>—
</td>
<td>—
</td>
<td>—
</td>
<td>Download to Donate for Haiti
</td>
</tr>
<tr>
<td>"Burn It Down"[128]
</td>
<td>2012
</td>
<td>30
</td>
<td>1
</td>
<td>1
</td>
<td>41
</td>
<td>33
</td>
<td>47
</td>
<td>2
</td>
<td>13
</td>
<td>12
</td>
<td>27
</td>
<td>RIAA: 2× Platinum<[32]nBPI: Silver<[34]nBVMi: ...
</td>
<td>Living Things
</td>
</tr>
<tr>
<td>"Lost in the Echo"[129]
</td>
<td>2012
</td>
<td>95
</td>
<td>12
</td>
<td>10
</td>
<td>—
</td>
<td>—
</td>
<td>150
</td>
<td>68
</td>
<td>—
</td>
<td>—
</td>
<td>175
</td>
<td>—
</td>
<td>Living Things
</td>
</tr>
<tr>
<td>"Powerless"[130]
</td>
<td>2012
</td>
<td>—
</td>
<td>—
</td>
<td>—
</td>
<td>—
</td>
<td>—
</td>
<td>—
</td>
<td>—
</td>
<td>—
</td>
<td>—
</td>
<td>—
</td>
<td>Living Things
</td>
</tr>
<tr>
<td>"Castle of Glass"[131]
</td>
<td>2013
</td>
<td>—
</td>
<td>16
</td>
<td>32
</td>
<td>—
</td>
<td>—
</td>
<td>166
</td>
<td>10
</td>
<td>—
</td>
<td>17
</td>
<td>—
</td>
<td>BPI: Silver<[34]nBVMi: Platinum<[35]nIFPI AUT: ...
</td>
<td>Living Things
</td>
</tr>
<tr>
<td>"A Light That Never Comes"(with Steve Aoki)[132]
</td>
<td>2013
</td>
<td>65
</td>
<td>7
</td>
<td>11
</td>
<td>56
</td>
<td>51
</td>
<td>81
</td>
<td>8
</td>
<td>—
</td>
<td>15
</td>
<td>34
</td>
<td>—
</td>
<td>Recharged
</td>
</tr>
<tr>
<td>"Guilty All the Same"(featuring Rakim)[133]
</td>
<td>2014
</td>
<td>—[A]
</td>
<td>21
</td>
<td>19
</td>
<td>—
</td>
<td>—
</td>
<td>116
</td>
<td>32
</td>
<td>—
</td>
<td>50
</td>
<td>138
</td>
<td>—
</td>
<td>The Hunting Party
</td>
</tr>
<tr>
<td>"Until It's Gone"
</td>
<td>2014
</td>
<td>—[B]
</td>
<td>19
</td>
<td>17
</td>
<td>58
</td>
<td>—
</td>
<td>104
</td>
<td>24
</td>
<td>—
</td>
<td>23
</td>
<td>78
</td>
<td>—
</td>
<td>The Hunting Party
</td>
</tr>
<tr>
<td>"Wastelands"[136]
</td>
<td>2014
</td>
<td>—
</td>
<td>—
</td>
<td>25
</td>
<td>64
</td>
<td>—
</td>
<td>76
</td>
<td>—
</td>
<td>—
</td>
<td>—
</td>
<td>—
</td>
<td>The Hunting Party
</td>
</tr>
<tr>
<td>"Rebellion"[137](featuring Daron Malakian)
</td>
<td>2014
</td>
<td>—
</td>
<td>—
</td>
<td>21
</td>
<td>—
</td>
<td>—
</td>
<td>83
</td>
<td>—
</td>
<td>—
</td>
<td>—
</td>
<td>—
</td>
<td>The Hunting Party
</td>
</tr>
<tr>
<td>"Final Masquerade"[138]
</td>
<td>2014
</td>
<td>—[C]
</td>
<td>32
</td>
<td>18
</td>
<td>43
</td>
<td>85
</td>
<td>45
</td>
<td>70
</td>
<td>30
</td>
<td>64
</td>
<td>106
</td>
<td>—
</td>
<td>The Hunting Party
</td>
</tr>
<tr>
<td>"Heavy"(featuring Kiara)[139]
</td>
<td>2017
</td>
<td>45
</td>
<td>22
</td>
<td>2
</td>
<td>33
</td>
<td>46
</td>
<td>111
</td>
<td>12
</td>
<td>35
</td>
<td>8
</td>
<td>43
</td>
<td>RIAA: Gold<[32]nARIA: Gold<[140]nBPI: Silver<3...
</td>
<td>One More Light
</td>
</tr>
<tr>
<td>"Talking to Myself"[141][142]
</td>
<td>2017
</td>
<td>—[D]
</td>
<td>—
</td>
<td>13
</td>
<td>—
</td>
<td>66
</td>
<td>180
</td>
<td>73
</td>
<td>—
</td>
<td>35
</td>
<td>—
</td>
<td>—
</td>
<td>One More Light
</td>
</tr>
<tr>
<td>"One More Light"[143]
</td>
<td>2017
</td>
<td>—[E]
</td>
<td>21
</td>
<td>6
</td>
<td>85
</td>
<td>91
</td>
<td>113
</td>
<td>51
</td>
<td>—
</td>
<td>48
</td>
<td>—
</td>
<td>RIAA: Gold<[32]nBPI: Silver<[34]
</td>
<td>One More Light
</td>
</tr>
<tr>
<td>"She Couldn"[144][145]
</td>
<td>2020
</td>
<td>—
</td>
<td>—
</td>
<td>47
</td>
<td>—
</td>
<td>—
</td>
<td>—
</td>
<td>—
</td>
<td>—
</td>
<td>—
</td>
<td>—
</td>
<td>Hybrid Theory (20th Anniversary Edition)
</td>
</tr>
```

Extraire les données

```
#Extraire les données et ajoutez-les aux listes respectives
lst_data5 = []
c5=[ [], [], [], [], [], [], [], [], [], [], [], [], [], [], [] ]
for row in rows5[2:-1]:
    c5[0].append(row.find("th").text.strip())
    tds = row.find_all('td')
    if len(tds) == 11:
        for i, x in enumerate(tds):
            c5[i+1].append(x.text.strip())
    elif len(tds) == 12:
        if len(tds[0].text.strip()) == 4:
            for i, x in enumerate(tds):
                c5[i+1].append(x.text.strip())
            c5[13].append(c5[13][-1])
        else:
            c5[1].append(c5[1][-1])
            c5[1].append(c5[1][-1])
            for i, x in enumerate(tds):
                c5[i+2].append(x.text.strip())
    else:
        c5[1].append(c5[1][-1])
        for i, x in enumerate(tds):
            c5[i+2].append(x.text.strip())
        c5[13].append(c5[13][-1])

# créer un dictionnaire
d5 = dict()
for i, h in enumerate(headers5):
    d5[h] = c5[i]

# convertir dict en DataFrame
df_table5=pd.DataFrame(d5)
df_table5
```

	Title	Year	US	USAlt.	USRock	AUS	CAN	FRA	GER	NZ	SWI	UK	Certifications	Album
0	"One Step Closer"[105]	2000	75	5	4	4	—	—	32	—	42	24	RIAA: Platinum[32]nARIA: Gold[106]nBPI: Gold...	Hybrid Theory
1	"Crawling"[107]	2001	79	5	3	33	—	106	14	37	43	16	RIAA: Gold[32]nBPI: Gold[34]	Hybrid Theory
2	"Papercut"[108]	2001	—	32	18	87	43	—	49	—	80	14	RIAA: Gold[32]nBPI: Silver[34]	Hybrid Theory
3	"In the End"[109]	2001	2	1	3	4	2	23	4	10	4	8	RIAA: 4× Platinum[32]nARIA: Gold[110]nBPI: 2...	Hybrid Theory
4	"Pts.OF.Athrty"(with Jay Gordon)	2002	—	29	—	44	47	—	31	—	61	9		Reanimation
5	"Somewhere I Belong"[111]	2003	32	1	1	13	16	32	12	1	15	10	ARIA: Gold[112]nBPI: Silver[34]	Meteora
6	"Faint"[111]	2003	48	1	2	25	27	158	40	—	32	15	RIAA: Platinum[32]nBPI: Silver[34]	Meteora
7	"Numb"[111]	2003	11	1	1	10	27	9	13	13	5	14	RIAA: 4× Platinum[32]nARIA: Gold[112]nBPI: 2...	Meteora
8	"From the Inside"[111]	2004	—	—	—	37	43	35	35	50	38	—		Meteora
9	"Breaking the Habit"[111]	2004	20	1	1	23	43	27	25	27	56	39	RIAA: Gold[32]nBPI: Silver[34]	Meteora
10	"Numb/Encore"(with Jay-Z)[113]	2004	20	—	17	3	3	5	4	—	10	14	RIAA: 3× Platinum[32]nARIA: Platinum[114]nBP...	Collision Course
11	"What I've Done"[115]	2007	7	1	7	13	3	107	3	9	6	6	RIAA: 5× Platinum[32]nBPI: Platinum[34]nBVMi...	Minutes to Midnight
12	"Bleed It Out"[117]	2007	52	2	16	24	22	—	40	7	42	29	RIAA: 2× Platinum[32]nBPI: Silver[34]nRMNZ: ...	Minutes to Midnight
13	"Shadow of the Day"[118]	2007	15	2	19	15	12	20	12	13	11	46	RIAA: 2× Platinum[32]niFPI SWI: Gold[37]nRMN...	Minutes to Midnight
14	"Given Up"[119]	2008	99	4	—	—	—	—	53	—	—	—	RIAA: Gold[32]	Minutes to Midnight
15	"Leave Out All the Rest"[120]	2008	94	11	22	24	—	17	15	38	36	90	RIAA: Platinum[32]	Minutes to Midnight
16	"New Divide"[121]	2009	6	1	1	3	3	—	4	2	7	19	RIAA: 3× Platinum[32]nARIA: Platinum[122]nBP... Transformers: Revenge of the Fallen – The Album	
17	"The Catalyst"[123]	2010	27	1	1	33	28	—	11	27	29	40	RIAA: Gold[32]	A Thousand Suns
18	"Waiting for the End"[124]	2010	42	1	2	—	55	—	29	—	58	90	RIAA: Platinum[32]	A Thousand Suns
19	"Burning in the Skies"[125]	2011	—	—	—	—	—	—	43	—	41	—		A Thousand Suns
20	"Iridescent"[126]	2011	81	19	29	39	—	—	46	—	69	93	RIAA: Gold[32]	A Thousand Suns
21	"Not Alone"[127]	2011	—	—	—	—	—	—	—	—	—	—		Download to Donate for Haiti
22	"Burn It Down"[128]	2012	30	1	1	41	33	47	2	13	12	27	RIAA: 2× Platinum[32]nBPI: Silver[34]nBVMi: ...	Living Things
23	"Lost in the Echo"[129]	2012	95	12	10	—	—	150	68	—	—	175		Living Things
24	"Powerless"[130]	2012	—	—	—	—	—	—	—	—	—	—		Living Things
25	"Castle of Glass"[131]	2013	—	16	32	—	—	166	10	—	17	—	BPI: Silver[34]nBVMi: Platinum[35]niFPI AUT...	Living Things
26	"A Light That Never Comes"(with Steve Aoki)[132]	2013	65	7	11	56	51	81	8	—	15	34		Recharged
27	"Guilty All the Same"(featuring Rakim)[133]	2014	—[A]	21	19	—	—	116	32	—	50	138		The Hunting Party
28	"Until It's Gone"	2014	—[B]	19	17	58	—	104	24	—	23	78		The Hunting Party
29	"Wastelands"[136]	2014	—	—	25	64	—	76	—	—	—	—		The Hunting Party
30	"Rebellion"[137](featuring Daron Malakian)	2014	—	—	21	—	—	83	—	—	—	—		The Hunting Party
31	"Final Masquerade"[138]	2014	—[C]	32	18	43	85	45	70	30	64	106		The Hunting Party
32	"Heavy"(featuring Kiara)[139]	2017	45	22	2	33	46	111	12	35	8	43	RIAA: Gold[32]nARIA: Gold[140]nBPI: Silver[3...	One More Light
33	"Talking to Myself"[141][142]	2017	—[D]	—	13	—	66	180	73	—	35	—		One More Light
34	"One More Light"[143]	2017	—[E]	21	6	85	91	113	51	—	48	—	RIAA: Gold[32]nBPI: Silver[34]	One More Light
35	"She Couldn't"[144][145]	2020	—	—	47	—	—	—	—	—	—	—	Hybrid Theory (20th Anniversary Edition)	

```
filtered_table_soup6=[table for table in table_soup if table.caption is
```

```
required_table6=None
```

- ▼ Extraire les données

```
# créer un dictionnaire
d6 = dict()
for i, h in enumerate(header6):
    d6[h] = c6[i]
```

[illegible]

- ▼ Promotional singles

```
required table7=None
```

[illegible]

- ▼ Extraire les données


```
#generer le fichier csv
df_table7.to_csv("promotional_singles.csv",index=False,encoding="utf-8")
```

▼ Other charted songs

```
required_table8=None
```

```
print(required_table8)
```

- ▼ Extraire les données

```
# convertir dict en DataFrame
df_table8=pd.DataFrame(d8)
df_table8
```

```
#generer le fichier csv
df_table8.to_csv("others_charted_songs.csv",index=False,encoding="utf-8")
```

- Other appearances

```
filtered_table_soup9=soup.find_all('table')
```



```
required_table9=None

for i in range(len(filtered_table_soup9)):
    if filtered_table_soup9[i].caption and filtered_table_soup9[i].caption.text.strip()=="List of other charted songs, with selected chart positions, showing year released and album name":
        required_table9=filtered_table_soup9[i+1]
        break
print(required_table9)

<table class="wikitable plainrowheaders" style="width:820px;">
<tbody><tr>
<th>Title
</th>
<th style="width:30px;">Year
</th>
<th>Album
</th>
<th>Label
</th>
<th>Ref(s)
</th></tr>
<tr>
<th scope="row">"Closing" <small>(Xero)</small>
</th>
<td rowspan="3">"3" style="text-align:center;">1998
</td>
<td><i>Rapology 12</i>
</td>
<td rowspan="3">"3" style="text-align:center;">The Urban Network
</td>
<td>
</td>
</td></tr>
<tr>
<th scope="row">"Drop" <small>(Kenji &amp; Antofficial of Xero)</small>
</th>
<td><i>Rapology 13</i>
</td>
<td>
</td></tr>
<tr>
<th scope="row">"Fiends CD RAP UP" <small>(Xero featuring 007)</small>
</th>
<td><i>Rapology 14</i>
</td>
<td>
</td></tr>
<tr>
<th scope="row">"ca href="/wiki/One_Step_Closer_(Linkin_Park_song)" title="One Step Closer (Linkin Park song)">One Step Closer</a> <small>(Early Mix)</small>
</th>
<td rowspan="2">"2" style="text-align:center;">2000
</td>
<td><i>ECW Extreme Music Volume 2: Anarchy Rocks</i>
</td>
<td><i>V2
</td>
</td></tr>
<tr>
<td style="text-align:center;"><sup class="reference" id="cite_ref-192"><a href="#cite_note-192">[185]</a></sup>
</td></tr>
<tr>
<th scope="row">"My December"
</th>
<td>
</td>
<td><i>The Real Slim Santac</i>
</td>
<td><a href="/wiki/KROQ-FM" title="KROQ-FM">KROQ</a>
</td>
</td></tr>
<tr>
<td style="text-align:center;"><sup class="reference" id="cite_ref-193"><a href="#cite_note-193">[186]</a></sup>
</td></tr>
<tr>
<td>
</td>
</td>
</td>
</td>
</tr>
</tbody>
</table>

rows9 = required_table9.findAll("tr")

#Attributs d'en-tête du tableau
header9 = [th.text.rstrip() for th in rows9[0].find_all('th')[:-1]]
#header9 = header9[0:2] + [th.text.rstrip().split('|')[0] for th in rows9[1].find_all('th')] + header9[3:]
print(header9)

['Title', 'Year', 'Album', 'Label']
```

↳ Extraire les données

```
lst_data9 = []
c9= [ [], [], [], [] ]
for row in rows9[1:]:
    c9[0].append(row.find('th').text.strip())
    tds9 = row.find_all('td')[:-1]
    if len(tds9) == 3:
        for i, x in enumerate(tds9):
            c9[i+1].append(x.text.strip())
    elif len(tds9) == 2:
        if len(tds9[0].text.strip()) == 3:
            for i, x in enumerate(tds9):
                c9[i+1].append(x.text.strip())
            c9[3].append(c9[3][-1])
        else:
            c9[1].append(c9[1][-1])
            for i, x in enumerate(tds9):
                c9[i+2].append(x.text.strip())
    elif len(tds9) == 1:
        c9[1].append(c9[1][-1])
        c9[2].append(tds9[0].text.strip())
        c9[3].append(c9[3][-1])
    else:
        c9[1].append(c9[1][-1])
        c9[2].append(c9[2][-1])
        c9[3].append(c9[3][-1])

# créer un dictionnaire
d9 = dict()
for i, h in enumerate(header9):
    d9[h] = c9[i]

# convertir dict en DataFrame
df_table9=pd.DataFrame(d9)
df_table9

   Title  Year  Album  Label
0      "Closing" (Xero)  1998  Rapology 12  The Urban Network
1      "Drop" (Kenji & Antofficial of Xero)  1998  Rapology 13  The Urban Network
2      "Fiends CD RAP UP" (Xero featuring 007)  1998  Rapology 14  The Urban Network
3      "One Step Closer" (Early Mix)  2000  ECW Extreme Music Volume 2: Anarchy Rocks  V2
4      "My December"  2000  The Real Slim Santa  KROQ
5      "With You" (Live)  2001  Ozzfest 2001: The Second Millennium  Sony
6      "Runaway" (Live)  2002  The Family Values Tour 2001  Elektra/nWEA
7      "One Step Closer" (Live) (featuring Aaron Lewis)  2002  The Family Values Tour 2001  Elektra/nWEA
8      "It's Goin' Down" (The X-Ecutioners featuring ...  2002  Built from Scratch  Loud Records/nSony BMG
9      "My December" (Live)  2003  The Year They Recalled Santa Claus  KROQ
10     "Nobody's Listening" (Green Lantern Remix)  2005  Fort Minor: We Major  Machine Shop Recordings
11     "Bleed It Out" (Live)  2007  Live Earth: The Concerts for a Climate in Crisis  Warner Bros. Records
12     "Blackout" (Early Mix)  2010  FIFA 11  EA Sports
13     "Issho Ni"  2011  Download to Donate: Tsunami Relief  Machine Shop/Warner Bros.
14     "Indescent" (Radio Edit)  2011  Transformers: Dark of the Moon – The Album  Reprise Records
15     "Blackout" (Renholder Remix)  2012  Underworld: Awakening  Lakeshore
16     "Wretches" (Remix) (Apathy featuring Linkin Pa...  2012  It's the Bootleg, Muthafuckas! Volume 3: Fire ...  Demigodz
17     "All for Nothing" (Radio Edit) (featuring Page...  2014  Pro Evolution Soccer 2015  Konami
18     "Things in My Jeep" (The Lonely Island featur...  2016  Popstar: Never Stop Never Stopping  Universal Republic
19     "Battle Symphony"  2017  Pro Evolution Soccer 2018  Konami

#generer le fichier csv
df_table9.to_csv("others_appearances.csv",index=False,encoding="utf-8")
```

↳ Video albums

```
#Récupere tous les tableaux
filtered_table_soup10=soup.find_all('table')

required_table10=None

for i in range(len(filtered_table_soup10)):
    if filtered_table_soup10[i].caption and filtered_table_soup10[i].caption.text.strip()=="List of other charted songs, with selected chart positions, showing year released and album name":
        required_table10=filtered_table_soup10[i+2]
        break
print(required_table10)

<table class="wikitable" style="width:500px;">
<tbody><tr>
<th style="width:30px;">Year
</th>
<th>Details
</th>
<th>Ref(s)
</th></tr>
<tr>
<td>2001
</td>
<td><i><b><a href="/wiki/Frat_Party_at_the_Pankake_Festival" title="Frat Party at the Pankake Festival">Frat Party at the Pankake Festival</a></b></i>
<ul><li>Release: November 20, 2001</li>
<li>Label: Warner Bros., Machine Shop</li>
<li>Format: DVD, <a href="/wiki/VHS" title="VHS">VHS</a></li>
<li>US certification: Platinum<sup class="reference" id="cite_ref-RIAA_32-30"><a href="#cite_note-RIAA-32">[32]</a></sup></li>
<li>BPI certification: Gold<sup class="reference" id="cite_ref-BPI_34-28"><a href="#cite_note-BPI-34">[34]</a></sup></li></ul>
</td>
<td style="text-align:center;"><sup class="reference" id="cite_ref-286"><a href="#cite_note-286">[199]</a></sup>
</td></tr>
<tr>
<td rowspan="2">"2">2003
</td>
<td><i><b>The Making of Meteorac</b></i>
<ul><li>Release: March 25, 2003</li>
<li>Label: Warner Bros., Machine Shop</li>
<li>Format: DVD</li></ul>
</td>
<td style="text-align:center;"><sup class="reference" id="cite_ref-207"><a href="#cite_note-207">[200]</a></sup>
</td></tr>
<tr>
<td><i><b><a href="/wiki/Live_in_Texas_(Linkin_Park_album)" title="Live in Texas (Linkin Park album)">Live in Texas</a></b></i>
<ul><li>Release: November 18, 2003</li>
<li>Label: Warner Bros., Machine Shop</li>
<li>Format: DVD</li>
<li>US certification: Platinum<sup class="reference" id="cite_ref-RIAA_32-31"><a href="#cite_note-RIAA-32">[32]</a></sup></li></ul>
</td>
<td style="text-align:center;"><sup class="reference" id="cite_ref-208"><a href="#cite_note-208">[201]</a></sup>
</td></tr>
<tr>
<td rowspan="2">"2">2004
</td>
<td><i><b>Breaking the Habit</b></i>
<ul><li>Release: July 27, 2004</li>
<li>Label: <a href="/wiki/Locomotive_Music" title="Locomotive Music">Locomotive Music</a></li>
<li>Format: DVD</li></ul>
</td>
<td style="text-align:center;"><sup class="reference" id="cite_ref-209"><a href="#cite_note-209">[202]</a></sup>
</td></tr>
<tr>
<td><i><b>Collision Course</b></i>
<ul><li>Release: November 30, 2004</li>
<li>Label: Warner Bros., Machine Shop</li>
<li>Format: DVD</li></ul>
</td>
<td style="text-align:center;"><sup class="reference" id="cite_ref-210"><a href="#cite_note-210">[203]</a></sup>
</td></tr>
<tr>
<td>2007
</td>
<td>
</td>
</td>
</tr>
</tbody>
</table>

rows10 = required_table10.findAll("tr")

#Attributs d'en-tête du tableau
header10 = [th.text.rstrip() for th in rows10[0].find_all('th')[:-1]]
print(header10)
```

↕

Extraire les données

```
#Extraire les données et ajoutez-les aux listes respectives
lst_data10 = []
c10= [ [], [] ]
for row in rows10[1:]:
    tds10=row.find_all('td')[:-1]
    if len(tds10)==2:
        c10[0].append(tds10[0].text.strip())
        c10[1].append(tds10[1].text.strip())
    else:
        c10[0].append(c10[0][:-1])
        c10[1].append(tds10[0].text.strip())
```

```
# créer un dictionnaire
d10 = dict()
for i, h in enumerate(header10):
    d10[h] = c10[i]
```

```
# convertir dict en DataFrame
df_table10=pd.DataFrame(d10)
df_table10
```

	Year	Details
0	2001	Frat Party at the Pankake Festival\nRelease: N...
1	2003	The Making of Meteor\nRelease: March 25, 2003...
2	2003	Live in Texas\nRelease: November 18, 2003\nLab...
3	2004	Breaking the Habit\nRelease: July 27, 2004\nLa...
4	2004	Collision Course\nRelease: November 30, 2004\n...
5	2007	The Making of Minutes to Midnight\nRelease: Ma...
6	2008	Road to Revolution: Live at Milton Keynes\nRel...
7	2010	The Meeting of a Thousand Suns\nRelease: Septe...
8	2011	A Thousand Suns*\nRelease: April 1, 2011\nLabe...
9	2012	Inside Living Things\nRelease: June 19, 2012\n...
10	2013	Living Things *\nRelease: March 22, 2013\nLabe...
11	2014	The Hunting Party (DVD)\nRelease: June 13, 201...

```
#generer le fichier csv
df_table10.to_csv("video_albums.csv",index=False,encoding="utf-8")
```

↕

Traditional videos

```
#recupere tous les tableaux
filtered_table_soup1=soup.find_all('table')
```

```
required_table11=None
```

```
for i in range(len(filtered_table_soup1)):
    if filtered_table_soup1[i].caption.text.strip()=="List of other charted songs, with selected chart positions, showing year released and album name":
        break
print(required_table11)
```

<div><div><div><div><div><div></div><div>tbody><tr> <th style="width:30px;">Year </th> <th>Details </th> <th>Director(s) </th> <th>Album </th> <th>Type </th> <th>Link </th></tr> <tr> <td>2000 </td> <td>One Step Closer </td> <td>Gregory Dark<sup class="reference" id="cite_ref-218">[211]</sup> </td> <td rowspan="6"><i>Hybrid Theory</i> </td> <td rowspan="3">Performance </td> <td><sup class="reference" id="cite_ref-219">[212]</sup> </td></tr> <tr> <td rowspan="2">2001 </td> <td>Crawling </td> <td>Brothers Strause<sup class="reference" id="cite_ref-220">[213]</sup> </td> <td><sup class="reference" id="cite_ref-221">[214]</sup> </td></tr> <tr> <td>Papercut </td> <td rowspan="2">Nathan "Karma" Cox and Joe Hahn<sup class="reference" id="cite_ref-222">[215]</sup><sup class="reference" id="cite_ref-223">[216]</sup> </td> <td><sup class="reference" id="cite_ref-224">[217]</sup> </td></tr> <tr> <td>In the End </td> <td>Surrealist </td> <td><sup class="reference" id="cite_ref-225">[218]</sup> </td></tr> <tr> <td>Points of Authority </td> <td>Nathan "Karma" Cox </td> <td>live, studio footage </td> <td><sup class="reference" id="cite_ref-226">[219]</sup> </td></tr> </tbody></div></div></div></div></div></div>
--

```
rows11 = required_table11.findAll("tr")
```

```
#Attributs d'en-tête du tableau
header11 = [th.text.rstrip() for th in rows11[0].find_all('th')[:-1]]
print(header11)
```

```
['Year', 'Details', 'Director(s)', 'Album', 'Type']
```

↕

Extraire les données

```
#Extraire les données et ajoutez-les aux listes respectives
lst_data11 = []
c11= [ [], [], [], [], [] ]
for row in rows11[1:]:
    tds11 = row.find_all('td')[:-1]
    i = 0
    j = 0
    while j < len(c11):
        if len(c11[j]) and c11[j][:-1][1] > 1:
            c11[j].append((c11[j][:-1][0], c11[j][:-1][1]-1))
        else:
            k = 0
            try:
                k = tds11[i]['rowspan']
            except:
                pass
            c11[j].append((tds11[i].text.strip(), int(k)))
            i += 1
            j += 1
```

```
c11 = [[x[0] for x in y] for y in c11]
```

```
# créer un dictionnaire
d11 = dict()
for i, h in enumerate(header11):
    d11[h] = c11[i]
```

```
# convertir dict en DataFrame
df_table11=pd.DataFrame(d11)
df_table11
```

	Year	Details	Director(s)	Album	Type
0	2000	"One Step Closer"	Gregory Dark[211]	Hybrid Theory	Performance
1	2001	"Crawling"	Brothers Strause[213]	Hybrid Theory	Performance
2	2001	"Papercut"	Nathan "Karma" Cox and Joe Hahn[215][216]	Hybrid Theory	Performance
3	2001	"In the End"	Nathan "Karma" Cox and Joe Hahn[215][216]	Hybrid Theory	Surrealist
4	2001	"Points of Authority"	Nathan "Karma" Cox	Hybrid Theory	Live, studio footage
...
64	2017	"Heavy"	Tim Mattia	One More Light	Narrative
65	2017	"Good Goodbye"	Isaac Rentz	One More Light	Narrative
66	2017	"Talking to Myself"	Mark Fiore	One More Light	Live footage
67	2017	"One More Light"	Joe Hahn & Mark Fiore	One More Light	Live footage
68	2021	"One Step Closer (100 geeks Reanimation)"	Weston Allen	N/A	Narrative, visualiser
69	rows × 5 columns				

```
#generer le fichier csv
df_table11.to_csv("music_videos.csv",index=False,encoding="utf-8")
```

↕

Lyric videos

```
#récupere tous les tableaux
filtered_table_soup12=soup.find_all('table')
```

```
required_table12=None
```

```
for i in range(len(filtered_table_soup12)):
    if filtered_table_soup12[i].caption.text.strip()=="List of other charted songs, with selected chart positions, showing year released and album name":
        break
print(required_table12)
```

<div><div><div><div><div><div></div><div>tbody><tr> <th style="width:30px;">Year </th> <th>Song </th> <th>Album </th> <th>Type </th> <th>Link </th></tr> <tr> <td rowspan="4">2010 </td> <td>The Catalyst </td></div></div></div></div></div></div>

```
#attributs d'en-tête du tableau
header12 = [th.text.rstrip() for th in rows12[0].find_all('th')[:-1]]
print(header12)
```

▼ Extraire les données

```
#Extraire les données et ajoutez-les aux listes respectives
```

```

lst_data2 = []
c12= [ [], [], [], [] ]
for row in rows2[1:]:
    tds12 = row.find_all('td')[:-1]
    i = 0
    j = 0
    while j < len(c12):
        if len(c12[j]) and c12[j][-1][1] > 1:
            c12[j].append((c12[j][-1][0], c12[j][-1][1]-1))
        else:
            k = 0
            try:
                k = tds12[i]['rowspan']
            except:
                pass
            c12[j].append((tds12[i].text.strip(), int(k)))
            i += 1
        j += 1

```

```
# créer un dictionnaire
d12 = dict()
for i, h in enumerate(header12):
    d12[h] = c12[i]
```

```
# convertir dict en DataFrame
df_table12=pd.DataFrame(d12)
df_table12
```

	Year	Song	Album	Type
0	2010	"The Catalyst"	A Thousand Suns	Static and chaotic imagery
1	2010	"Waiting for the End"	A Thousand Suns	Static and chaotic imagery
2	2010	"Blackout"	A Thousand Suns	Static and chaotic imagery
3	2010	"Burning in the Skies"	A Thousand Suns	Static and chaotic imagery
4	2012	"Burn It Down"	Living Things	Rotating Living Things art
5	2012	"Lies Greed Misery"	Living Things	Rotating Living Things art
6	2012	"Lost in the Echo"	Living Things	Rotating Living Things art
7	2013	"A Light That Never Comes"	Recharged	Rotating Recharged art
8	2014	"Guilty All the Same"	The Hunting Party	Cloud animation
9	2014	"Until It's Gone"	The Hunting Party	Narrative film clip
10	2014	"Wastelands"	The Hunting Party	White text on black background
11	2014	"Rebellion"	The Hunting Party	White text on black background
12	2014	"Final Masquerade"	The Hunting Party	White text on black background
13	2015	"Darker Than Blood"	Neon Future II	Moving shape animations
14	2016	"In the End"	Hybrid Theory	White text on black background
15	2017	"Heavy"	One More Light	Chester and Kiara animation
16	2017	"Battle Symphony"	One More Light	Narrative film clip
17	2017	"Good Goodbye"	One More Light	Stock footage and animations
18	2017	"Invisible"	One More Light	Photography of various scenery and people
19	2017	"One More Light"	One More Light	Compilation of fan lyric videos
20	2017	"Talking to Myself"	One More Light	Performance stills of the band

```
#generer le fichier csv
df_table12.to_csv("lyrics_videos.csv",index=False,encoding="utf-8")
```