In [1]:
```
# Importing necessary packages
import selenium
```

In [2]:
```
from selenium import webdriver
```

In [44]:
```
driver= webdriver.Chrome()
```

In [4]:
```
from selenium.webdriver.chrome.service import Service
```

In [5]:
```
from selenium.webdriver.chrome.options import Options
```

In [6]:
```
chrome_options= Options()
chrome_options.add_experimental_option("detach", True)
```

In [7]:
```
driver.get("https://www.ycombinator.com/companies?batch=W24")
```

In [16]:
```
# Extracting company name and locations
company_name=[]
company_location=[]
driver.implicitly_wait(25)
for i in range(1,250):
    try:
        company_name_element= driver.find_element('xpath','/html/body/div/div[2]/section[2]/di
        company_name.append(company_name_element.text)
    except:
        company_name.append("NAN")

    try:
        company_location_element= driver.find_element('xpath','/html/body/div/div[2]/section[2
        company_location.append(company_location_element.text)
    except:
        company_location.append("NAN")
```

In [17]:
```
company_name
```

Out[17]:
```
['Alacrity',
 'ParcelBio',
 'K-Scale Labs',
 'NAN',
 'Marr Labs',
 'Forge Rewards',
 'FanCave',
 'RetailReady',
 'Million',
 'NowHouse',
 'Crux',
 'Reprompt',
 'InspectMind AI',
 'Yarn',
 'Blacksmith',
 'CrowdVolt',
 'kater.ai',
 'MathGPTPro',
 'Quivr',
```

In [29]:
```
len(company_location)
```

Out[29]:
```
248
```

In [19]:
```
import pandas as pd
```

In [20]:
```
# Making the dataframe from company name and Comapny location
data = {'Company Name': company_name, 'Company Location': company_location}
df = pd.DataFrame(data)

# Display the DataFrame
print(df)
```

```
            Company Name                Company Location
0              Alacrity           San Francisco, CA, USA
1             ParcelBio           San Francisco, CA, USA
2           K-Scale Labs                New York, NY, USA
3                    NAN                             NAN
4             Marr Labs           San Francisco, CA, USA
..                   ...                             ...
244             Lantern           San Francisco, CA, USA
245             Danswer
246               Yenmo             Bengaluru, KA, India
247           GovernGPT              Toronto, ON, Canada
248  Stitch Technologies  London, England, United Kingdom

[249 rows x 2 columns]
```

In [21]:
```
# replacing '.' to '-' from the company name because it will not coming in the company page li
# from there we have to make links and it will be required there.
# similarly remove '()', make all latters small, and fill space by '-' for generating the link
df['Company Name'] = df['Company Name'].str.replace(r'\b(\w+)\.(\w+)\b', r'\1-\2', regex=True)
```

In [27]:
```
# saving company name list and locations to a csv file
df.to_csv('Companies_list.csv', index= False)
```

In [28]:
```
df.head()
```

Out[28]:

	Company Name	Company Location
0	Alacrity	San Francisco, CA, USA
1	ParcelBio	San Francisco, CA, USA
2	K-Scale Labs	New York, NY, USA
3	NAN	NAN
4	Marr Labs	San Francisco, CA, USA

In [26]:
```
df.head()
```

Out[26]:

	Company Name	Company Location
0	Alacrity	San Francisco, CA, USA
1	ParcelBio	San Francisco, CA, USA
2	K-Scale Labs	New York, NY, USA
3	NAN	NAN
4	Marr Labs	San Francisco, CA, USA

In [29]:
```
df.isnull().sum()
```

Out[29]:
```
Company Name        0
Company Location    0
dtype: int64
```

In [30]:
```
df['Company Name'] = df['Company Name'].str.replace(r'\([^)]*\)', '', regex=True)
```

In [31]:
```
df.head()
```

Out[31]:

	Company Name	Company Location
0	Alacrity	San Francisco, CA, USA
1	ParcelBio	San Francisco, CA, USA
2	K-Scale Labs	New York, NY, USA
3	NAN	NAN
4	Marr Labs	San Francisco, CA, USA

In [32]:
```
df['Company Name'] = df['Company Name'].str.lower()
```

In [33]:
```
df.head()
```

Out[33]:

	Company Name	Company Location
0	alacrity	San Francisco, CA, USA
1	parcelbio	San Francisco, CA, USA
2	k-scale labs	New York, NY, USA
3	nan	NAN
4	marr labs	San Francisco, CA, USA

In [34]:
```
df['Company Name'] = df['Company Name'].str.replace(' ', '-')
```

In [39]:
```
df['Company Name'] = df['Company Name'].str.replace(r'[\(\),]', '', regex=True)
```

In [40]:
```
df.head(100)
```

Out[40]:

	Company Name	Company Location
0	alacrity	San Francisco, CA, USA
1	parcelbio	San Francisco, CA, USA
2	k-scale-labs	New York, NY, USA
3	nan	NAN
4	marr-labs	San Francisco, CA, USA
...	...	...
95	lumina	San Francisco, CA, USA
96	centauri-ai	Alameda, CA, USA
97	prosights	San Francisco, CA, USA
98	manifold-freight	Seattle, WA, USA
99	edgetrace	

100 rows × 2 columns

In [50]:
```
# Extracting all other info which are asked in 2nd point of the assignment.
# like Comapny name, location, founded in, team size, founders
Company_list=[]
Founded_in=[]
Team_size=[]
Location=[]
Group_partner=[]
Link=[]
Active_founder=[]
# driver.implicitly_wait(10)
base_url="https://www.ycombinator.com/companies/{}"
for index, row in df.iterrows():
    # Get the page number from the DataFrame column
    company= row['Company Name']

    # Create the dynamic URL for each row
    url = base_url.format(company)
#     print(url)

    driver.get(url)
    # Company name
    try:
        company_name_element= driver.find_element('xpath','/html/body/div/div[2]/section[1]/di
        Company_list.append(company_name_element.text)
    except:
        Company_list.append("NaN")
    # Company founded year
    try:
        company_founded_element= driver.find_element('xpath','/html/body/div/div[2]/section[1]
        Founded_in.append(company_founded_element.text)
    except:
        Founded_in.append('NaN')
    # Team size of the company
    try:
        company_team_size_element= driver.find_element('xpath','/html/body/div/div[2]/section[
        Team_size.append(company_team_size_element.text)
    except:
        Team_size.append('NaN')

    # Companie's location
    try:
        company_location_element= driver.find_element('xpath','/html/body/div/div[2]/section[1
        Location.append(company_location_element.text)
    except:
        Location.append('NaN')

    # Group Partner
    try:
        company_group_partner_element= driver.find_element('xpath','/html/body/div/div[2]/sect
        Group_partner.append(company_group_partner_element.text)
    except:
        Group_partner.append('NaN')

    # Company link
    try:
        company_link_element= driver.find_element('xpath','/html/body/div/div[2]/section[1]/di
        Link.append(company_link_element.text)
    except:
        Link.append('NaN')

    # Active founders
    for i in range(1,3):
        try:
            active_founders_element= driver.find_element('xpath','/html/body/div/div[2]/div/se
            Active_founder.append(active_founders_element.text)
```

```
        except:
            Active_founder.append('NaN')
```

In [49]:
```
Company_list
```

Out[49]:
```
[]
```

In [52]:
```
# Company_list=[]
# Founded_in=[]
# Team_size=[]
# Location=[]
# Group_partner=[]
# Link=[]
# Active_founder=[]
Final_data = {'Company Name': Company_list, 'Founded_in': Founded_in, 'Team_size': Team_size,
              'Group Partner': Group_partner, 'Website Link': Link}
df_final= pd.DataFrame(Final_data)

# Display the DataFrame
print(df_final)
```

```
     Company Name Founded_in Team_size               Location  \
0        Alacrity       2024         2          San Francisco
1       ParcelBio       2023         2          San Francisco
2    K-Scale Labs       2024         3               New York
3             NaN        NaN       NaN                    NaN
4       Marr Labs       2023         6          San Francisco
..            ...        ...       ...                    ...
244       Lantern       2019         4          San Francisco
245           NaN       2023         2
246         Yenmo                    5       Bengaluru, India
247     GovernGPT       2023         2        Toronto, Canada
248           NaN       2023         0  London, United Kingdom

        Group Partner                 Website Link
0         Pete Koomen  http://www.joinalacrity.com (http://www.joinalacrity.com)
1        Surbhi Sarna       https://parcelbio.com/ (https://parcelbio.com/)
2         Harj Taggar          https://kscale.dev/ (https://kscale.dev/)
3                 NaN                          NaN
4    Gustaf Alstromer    https://www.marrlabs.com/ (https://www.marrlabs.com/)
..                ...                          ...
244    Jared Friedman       https://www.lantern.so (https://www.lantern.so)
245    Jared Friedman      https://www.danswer.ai/ (https://www.danswer.ai/)
246     Tom Blomfield           https://yenmo.in/ (https://yenmo.in/)
247     Tom Blomfield    https://www.governgpt.ai/ (https://www.governgpt.ai/)
248         Garry Tan      https://www.stitch.tech (https://www.stitch.tech)

[249 rows x 6 columns]
```

In [54]:
```
df_final.head()
```

Out[54]:

	Company Name	Founded_in	Team_size	Location	Group Partner	Website Link
0	Alacrity	2024	2	San Francisco	Pete Koomen	http://www.joinalacrity.com
1	ParcelBio	2023	2	San Francisco	Surbhi Sarna	https://parcelbio.com/
2	K-Scale Labs	2024	3	New York	Harj Taggar	https://kscale.dev/
3	NaN	NaN	NaN	NaN	NaN	NaN
4	Marr Labs	2023	6	San Francisco	Gustaf Alstromer	https://www.marrlabs.com/

In [61]:
```
Active_founder
```

```
 'Pawel Budzianowski',
 'NaN',
 'NaN',
 'Dave Grannan',
 'Han Shu',
 'Ethan Chang',
 'Isaac Kan',
 'Luke Bogus',
 'Nick Siscoe',
 'Elle Smyth',
 'Sarah Hamer',
 'Aiden Bai',
 'Nisarg Patel',
 'Dhaval Gajiwala',
 'NaN',
 'Himank Jain',
 'Atharva Padhye',
 'Lukas Martinelli',
 'Rob Balian',
 'Aakash Prasad',
```

In [63]:
```
df_active_founders= pd.DataFrame({'Active_founder': Active_founder})
```

In [64]:
```
df_active_founders.head()
```

Out[64]:

	Active_founder
0	Omar Draz
1	Anderthan Hsieh
2	David Weinberg
3	Chris Carlson
4	Benjamin Bolte

In [65]:
```
# i did this because when we see , all the compnies have 2 founders, and what i did that i appe
# so now i am making a column of active founders and assigning first two elements to the first
# and so o
Active_founders= [', '.join(df_active_founders['Active_founder'][i:i+2]) for i in range(0, len
```

In [68]:
```
Active_founders
```

```
 'Rachit Kataria, Will Wang',
 'Pierre-Louis Biojout, Paul-Louis Venard',
 'Grant Margerum, Garrett Graves',
 'Jonathan Ou, NaN',
 'Aman Gottumukkala, Kevin Tang',
 'Zayne Sagar, Shelby Bons',
 'Paul Lafforgue, Thomas Sohet',
 'Jithin James, Shahul ES',
 'Prady Modukuru, Prajwal K R',
 'Gabriele Venturi, NaN',
 'Sourav Choraria, Sidharth Choraria',
 'Mitch Patin, Eric Ciminelli',
 'Marie Schneegans, Michael Fester',
 'Deniz Kavi, Sherry Liu',
 'Ryan Gallagher, Jeffrey Lamothe',
 'NaN, NaN',
 'Sasha Zhang, Jordan Wick',
 'Rohan Mayya, Saifur Rahman',
 'Michael Rosenfield, Rohan Das',
 'Lina Colucci, Sidney Primas',
```

In [69]:
```
df_final['Active_founders']=Active_founders
```

In [70]:
```
df_final
```

Out[70]:

	Company Name	Founded_in	Team_size	Location	Group Partner	Website Link	Active_founders
0	Alacrity	2024	2	San Francisco	Pete Koomen	http://www.joinalacrity.com	Omar Draz, Anderthan Hsieh
1	ParcelBio	2023	2	San Francisco	Surbhi Sarna	https://parcelbio.com/	David Weinberg, Chris Carlson
2	K-Scale Labs	2024	3	New York	Harj Taggar	https://kscale.dev/	Benjamin Bolte, Pawel Budzianowski
3	NaN	NaN	NaN	NaN	NaN	NaN	NaN, NaN
4	Marr Labs	2023	6	San Francisco	Gustaf Alstromer	https://www.marrlabs.com/	Dave Grannan, Han Shu
...	...	...	...	...	...	...	...
244	Lantern	2019	4	San Francisco	Jared Friedman	https://www.lantern.so	Bastien Beurier, Guillaume Lachaud
245	NaN	2023	2		Jared Friedman	https://www.danswer.ai/	Yuhong Sun, Chris Weaver
246	Yenmo		5	Bengaluru, India	Tom Blomfield	https://yenmo.in/	Ashutosh Purohit, Aryan Agarwal
247	GovernGPT	2023	2	Toronto, Canada	Tom Blomfield	https://www.governgpt.ai/	Mamal Amini, Oliver Walerys
248	NaN	2023	0	London, United Kingdom	Garry Tan	https://www.stitch.tech	Till Kern, Yuriy Oparenko

249 rows × 7 columns

In [71]:
```
# this is the final output
df_final.to_csv('Companies_Detail.csv')
```

In [ ]:

In [46]:
```
# trying to fetch some info before using it to the main code
active_founder=[]
for i in range(1,3):
    try:
        active_founders_element= driver.find_element('xpath','/html/body/div/div[2]/div/sectio
        active_founder.append(active_founders_element.text)
    except:
        active_founder.append('NaN')

print(active_founder)
```

```
['Omar Draz', 'Anderthan Hsieh']
```

In [ ]:

In [44]:
```
# trying to fetch some info before using it to the main code
driver.get('https://www.ycombinator.com/companies/alacrity')
```

In [45]:
```
# trying to fetch some info before using it to the main code
company_name_element= driver.find_element('xpath','/html/body/div/div[2]/section[1]/div[2]/div
print(company_name_element.text)

company_founded_element= driver.find_element('xpath','/html/body/div/div[2]/section[1]/div[2]/
print(company_founded_element.text)

company_team_size_element= driver.find_element('xpath','/html/body/div/div[2]/section[1]/div[2
print(company_team_size_element.text)

company_location_element= driver.find_element('xpath','/html/body/div/div[2]/section[1]/div[2]
print(company_location_element.text)

company_group_partner_element= driver.find_element('xpath','/html/body/div/div[2]/section[1]/d
print(company_group_partner_element.text)

company_link_element= driver.find_element('xpath','/html/body/div/div[2]/section[1]/div[2]/div
print(company_link_element.text)

active_founders_element= driver.find_element('xpath','/html/body/div/div[2]/div/section/div[2]
print(active_founders_element.text)
```

```
Alacrity
2024
2
San Francisco
Pete Koomen
http://www.joinalacrity.com (http://www.joinalacrity.com)
ACTIVE FOUNDERS
```

In [57]:
```
# trying to fetch some info before using it to the main code
driver.get('https://www.ycombinator.com/companies/goldenbasis')
```

In [58]:
```
# trying to fetch some info before using it to the main code
company_name_element= driver.find_element('xpath','/html/body/div/div[2]/section[1]/div[2]/div
company_name_element.text
```

Out[58]: 'GoldenBasis'

In [ ]:

In [43]:
```
# trying to fetch some info before using it to the main code
Company
```

```
 'NaN',
 'NaN',
 'Quivr',
 'Dragoneye',
 'renderlet',
 'Fume',
 'Numo',
 'Forge',
 'Taiki',
 'Konstructly',
 'NaN',
 'Granza Bio',
 'Haplotype Labs',
 'Healia',
 'TrueClaim',
 'CoCrafter',
 'Reform',
 'Piramidal',
 'Junction Bioscience',
```

In [73]:
```
# trying to fetch some info before using it to the main code
base_url="https://www.ycombinator.com/companies/{}"
for index, row in df.iterrows():
    # Get the page number from the DataFrame column
    company= row['Company Name']

    # Create the dynamic URL for each row
    url = base_url.format(company)

    Company=[]
#     Founded_in=[]
#     Team_size=[]
#     Location=[]
#     Group_Partner=[]
#     Active_Founders=[]



    company_name_element= driver.find_elements('xpath','/html/body/div/div[2]/section[1]/div[2
    Company.append(company_name_element.text)


#     try:
#         company_location_element= driver.find_element('xpath',)
#         company_location.append(company_location_element.text)
#     except:
#         company_location.append("NAN")
```

In [58]:
```
# trying to fetch some info before using it to the main code
base_url="https://www.ycombinator.com/companies/{}"
for index, row in df.iterrows():
    # Get the page number from the DataFrame column
    company= row['Company Name']

    # Create the dynamic URL for each row
    url = base_url.format(company)
    print(url)
```

```
derlet)
https://www.ycombinator.com/companies/fume (https://www.ycombinator.com/companies/fume)
https://www.ycombinator.com/companies/numo (https://www.ycombinator.com/companies/numo)
https://www.ycombinator.com/companies/forge (https://www.ycombinator.com/companies/forge)
https://www.ycombinator.com/companies/taiki (https://www.ycombinator.com/companies/taiki)
https://www.ycombinator.com/companies/konstructly (https://www.ycombinator.com/companies/k
onstructly)
https://www.ycombinator.com/companies/pico (https://www.ycombinator.com/companies/pico)
https://www.ycombinator.com/companies/granza-Bio (https://www.ycombinator.com/companies/gr
anza-Bio)
https://www.ycombinator.com/companies/haplotype-Labs (https://www.ycombinator.com/companie
s/haplotype-Labs)
https://www.ycombinator.com/companies/healia (https://www.ycombinator.com/companies/heali
a)
https://www.ycombinator.com/companies/trueClaim (https://www.ycombinator.com/companies/tru
eClaim)
https://www.ycombinator.com/companies/coCrafter (https://www.ycombinator.com/companies/coC
rafter)
https://www.ycombinator.com/companies/reform (https://www.ycombinator.com/companies/refor
```

In [72]:
```
# company_name_element= driver.find_element('xpath','/html/body/div/div[2]/section[1]/div[2]/d
```

In [ ]:
```

```