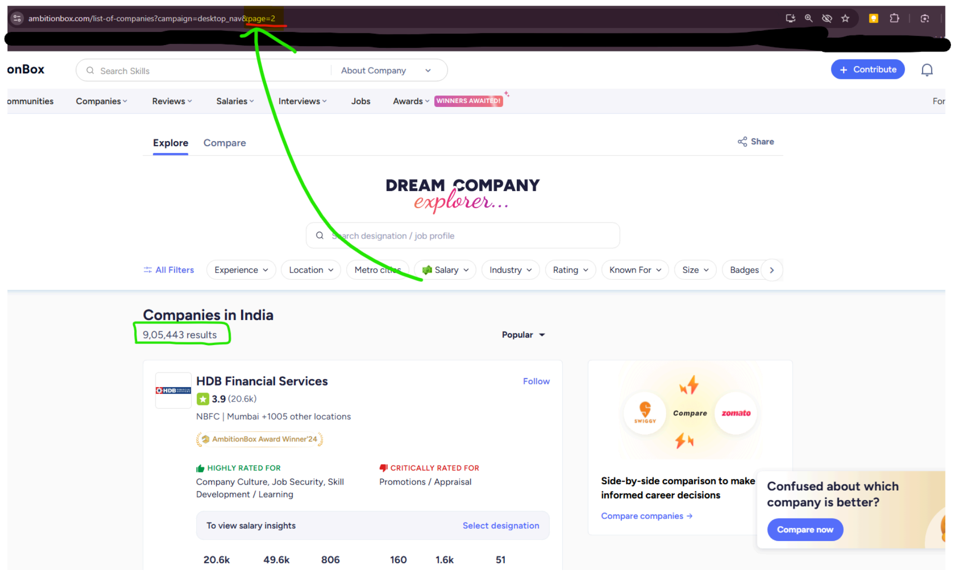
Web Scraping

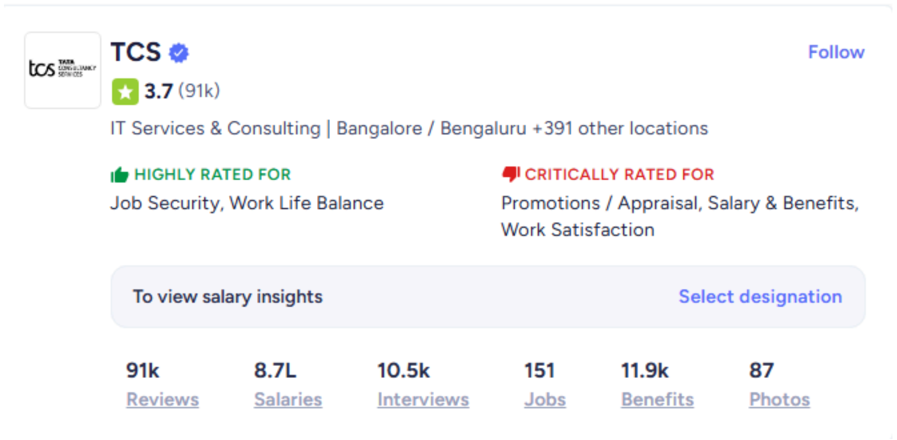
Website to scrape → AmbitionBox

Data to scrape → **Companies in India**



Link → https://www.ambitionbox.com/list-of-companies?page=1

No. of pages = 500


TCS
Follow
3.7 (91k)
IT Services & Consulting | Bangalore / Bengaluru +391 other locations
HIGHLY RATED FOR
Job Security, Work Life Balance
CRITICALLY RATED FOR
Promotions / Appraisal, Salary & Benefits, Work Satisfaction
To view salary insights
Select designation
91k
Reviews
8.7L
Salaries
10.5k
Interviews
151
Jobs
11.9k
Benefits
87
Photos

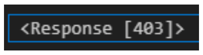
Import Libraries

```
import pandas as pd
import requests
```

```
from bs4 import BeautifulSoup
import numpy as np
```

Create an HTTP request

```
requests.get('https://www.ambitionbox.com/list-of-companies?page=1')
```



```
<Response [403]>
```

- If you add `.text`, you'll get the content 👇

```
requests.get('https://www.ambitionbox.com/list-of-companies?page=1').text
```

'<HTML><HEAD>\n<TITLE>**Access Denied**</TITLE>\n</HEAD><BODY>\n<H1>Access Denied</H1>\n \n**You don\'t have permission to access "http://www.ambitionbox.com/list-of-companies?" on this server.**<P>\nReference #18.69bdef75.1742242146.1791e53e\n<P>https://errors.edgesuite.net/18.69bdef75.1742242146.1791e53e</P>\n</BODY>\n<

- **Disguise yourself as browser**

```
headers={'User-Agent':'Mozilla/5.0 (Windows NT 6.3; Win 64 ; x64) Apple WeKit /537.36(KHTML , like Gecko) Chrome/80.0.3987.162 Safari/537.36'}

requests.get('https://www.ambitionbox.com/list-of-companies?page=1',headers=headers).text
```

💡 👆**This won't work for ambitionbox.❌ The now strict checking for headers.**

Alternative header:

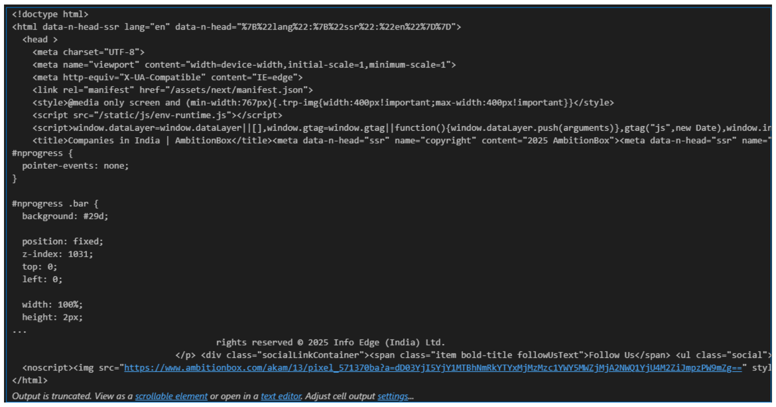
```
headers={'User-Agent':'Mozilla/5.0 (Windows NT 6.3; Win 64 ; x64) Apple WeKit /537.36(KHTML , like Gecko) Chrome/80.0.3987.162 Safari/537.36',"Accept": 'text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,*/*;q=0.8,application/signed-exchange;v=b3;q=0.7', 'sec-ch-ua': 'Chromium";v="130", "Google Chrome";v="130", "Not?A_Brand";v="99', 'sec-ch-ua-mobile':'?0', 'sec-ch-ua-platform': 'Windows', 'sec-fetch-dest': 'document', 'sec-fetch-mode': 'navigate', 'sec-fetch-site': 'same-origin', 'sec-fetch-user': '?1', 'upgrade-insecure-requests': '1'}
```

Now run:

```

webpage= requests.get('https://www.ambitionbox.com/list-of-companies?page=1',headers=headers).text
```

- This will give you the HTML code for the page

```
print(webpage)
```



```
<!doctype html>
<html data-n-head-ssr lang="en" data-n-head="%7B%22lang%22:%7B%22ssr%22:%22en%22%7D%7D">
  <head >
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width,initial-scale=1,minimum-scale=1">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <link rel="manifest" href="/assets/next/manifest.json">
    <style>@media only screen and (min-width:767px){.trp-img{width:400px!important;max-width:400px!important}}</style>
    <script src="/static/js/env-runtime.js"></script>
    <script>window.dataLayer=window.dataLayer||[],window.gtag=window.gtag||function(){window.dataLayer.push(arguments)},gtag("js",new Date),window.ir
    <title>Companies in India | AmbitionBox</title><meta data-n-head="ssr" name="copyright" content="2025 AmbitionBox"><meta data-n-head="ssr" name="
#nprogress {
  pointer-events: none;
}

#nprogress .bar {
  background: #29d;

  position: fixed;
  z-index: 1031;
  top: 0;
  left: 0;

  width: 100%;
  height: 2px;
...
                                        rights reserved © 2025 Info Edge (India) Ltd.
                                </p> <div class="socialLinkContainer"><span class="item bold-title followUsText">Follow Us</span> <ul class="social">
  <noscript><img src="https://www.ambitionbox.com/akam/13/pixel_571370ba?a=dD03YjI5YjY1MTBhNmRkYTYxMjMzMzc1YWY5MWZjMjA2NWQ1YjU4M2ZiJmpzPW9mZg==" styl
</html>
```

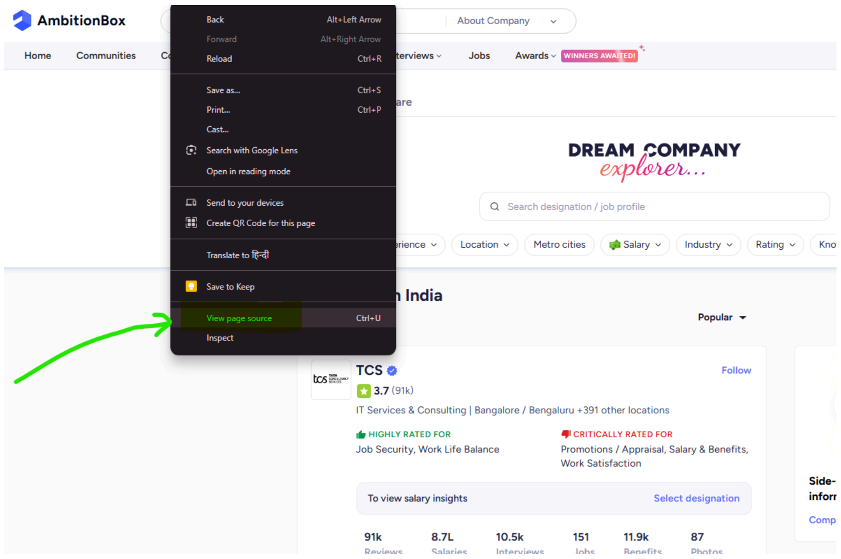
Output is truncated. View as a scrollable element or open in a text editor. Adjust cell output settings...

Make the HTML file readable with:

`prettify()`

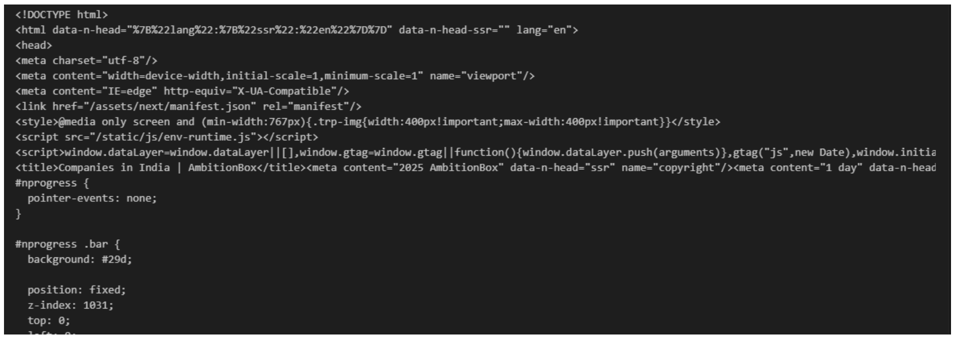
```
soup=BeautifulSoup(webpage,'lxml')
formatted_html = soup.prettify()
```

- You will get same code by right click on page → View page source



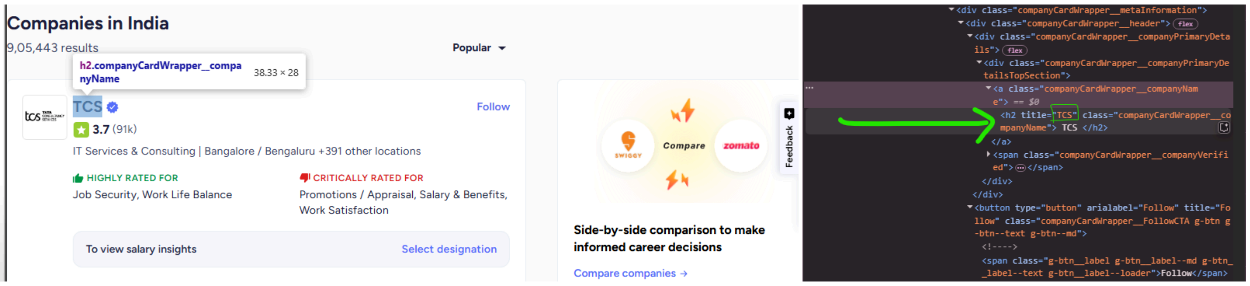
Now, create an object for `BeautifulSoup` & provide it the webpage & **html parser (** `'lxml'` **)**

```
soup=BeautifulSoup(webpage,'lxml')
soup
```



```
<!DOCTYPE html>
<html data-n-head="%7B%22lang%22:%7B%22ssr%22:%22en%22%7D%7D" data-n-head-ssr="" lang="en">
<head>
<meta charset="utf-8"/>
<meta content="width=device-width,initial-scale=1,minimum-scale=1" name="viewport"/>
<meta content="IE=edge" http-equiv="X-UA-Compatible"/>
<link href="/assets/next/manifest.json" rel="manifest"/>
<style>@media only screen and (min-width:767px){.trp-img{width:400px!important;max-width:400px!important}}</style>
<script src="/static/js/env-runtime.js"></script>
<script>window.dataLayer=window.dataLayer||[],window.gtag=window.gtag||function(){window.dataLayer.push(arguments)},gtag("js",new Date),window.initia
<title>Companies in India | AmbitionBox</title><meta content="2025 AmbitionBox" data-n-head="ssr" name="copyright"/><meta content="1 day" data-n-head
#nprogress {
  pointer-events: none;
}

#nprogress .bar {
  background: #29d;

  position: fixed;
  z-index: 1031;
  top: 0;
```

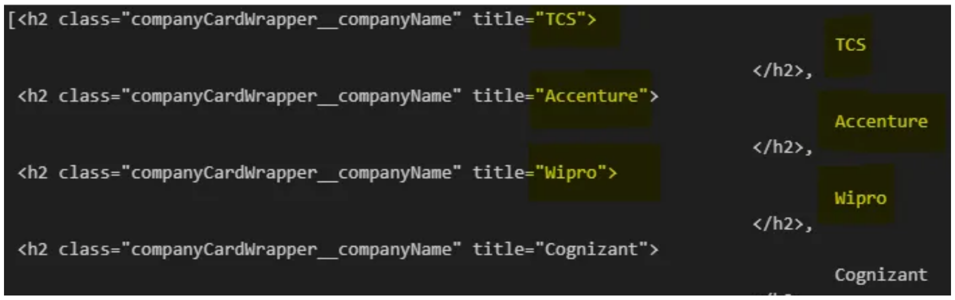
- Go to the webpage → press F12
- Reach the container with company name


Companies in India
9,05,443 results
Popular
h2.companyCardWrapper__companyName 38.33 × 28
TCS
3.7 (91k)
Follow
IT Services & Consulting | Bangalore / Bengaluru +391 other locations
HIGHLY RATED FOR
Job Security, Work Life Balance
CRITICALLY RATED FOR
Promotions / Appraisal, Salary & Benefits, Work Satisfaction
To view salary insights
Select designation
Compare
zomato
Side-by-side comparison to make informed career decisions
Compare companies →
Feedback

```
<div class="companyCardWrapper__metaInformation">
  <div class="companyCardWrapper__header"> flex
    <div class="companyCardWrapper__companyPrimaryDetails"> flex
      <div class="companyCardWrapper__companyPrimaryDetailsTopSection">
        <a class="companyCardWrapper__companyName"> == $0
          <h2 title="TCS" class="companyCardWrapper__companyName"> TCS </h2>
        </a>
        <span class="companyCardWrapper__companyVerified"> ··· </span>
      </div>
    </div>
    <button type="button" arialabel="Follow" title="Follow" class="companyCardWrapper__FollowCTA g-btn g-btn--text g-btn--md">
      <!---->
      <span class="g-btn__label g-btn__label--md g-btn__label--text g-btn__label--loader">Follow</span>
```

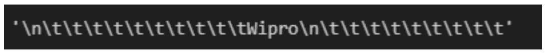
- You can find the tags with 👇

```
soup.find_all('h2')
```



```
[<h2 class="companyCardWrapper__companyName" title="TCS">
                                                              TCS
                                                    </h2>,
 <h2 class="companyCardWrapper__companyName" title="Accenture">
                                                              Accenture
                                                    </h2>,
 <h2 class="companyCardWrapper__companyName" title="Wipro">
                                                              Wipro
                                                    </h2>,
 <h2 class="companyCardWrapper__companyName" title="Cognizant">
                                                              Cognizant
```

```
soup.find_all('h2')[2].text
```

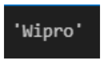


```
'\n\t\t\t\t\t\t\t\t\t\tWipro\n\t\t\t\t\t\t\t\t\t'
```

- Apply `strip` to get the name

```
a= soup.find_all('h2')[2].text.strip()
```

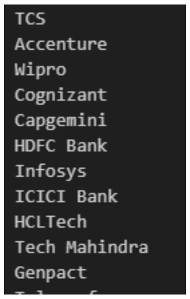
```
a
```



```
'Wipro'
```

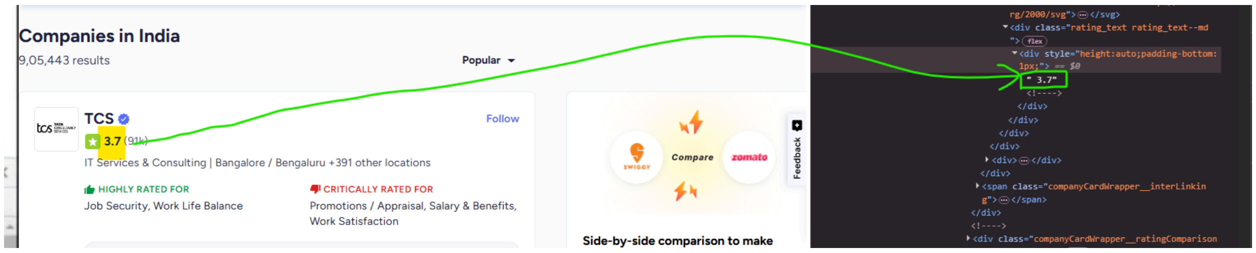
Find out names of all companies:

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



```
TCS
Accenture
Wipro
Cognizant
Capgemini
HDFC Bank
Infosys
ICICI Bank
HCLTech
Tech Mahindra
Genpact
```

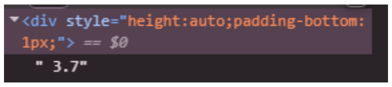
Similarly, find out the rating:



Companies in India
9,05,443 results
Popular
TCS
3.7 (91k)
Follow
IT Services & Consulting | Bangalore / Bengaluru +391 other locations
HIGHLY RATED FOR
Job Security, Work Life Balance
CRITICALLY RATED FOR
Promotions / Appraisal, Salary & Benefits, Work Satisfaction
Compare
zomato
Side-by-side comparison to make
Feedback

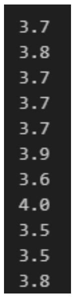
```
rg/2000/svg">⋯</svg>
▾<div class="rating_text rating_text--md
">  flex
  ▾<div style="height:auto;padding-bottom:
   1px;"> == $0
     " 3.7"
     <!---->
   </div>
  </div>
 </div>
</div>
▸<div>⋯</div>
</div>
▸<span class="companyCardWrapper__interLinkin
g">⋯</span>
</div>
<!---->
▸<div class="companyCardWrapper__ratingComparison
```

`style="height:auto;padding-bottom:1px;"`



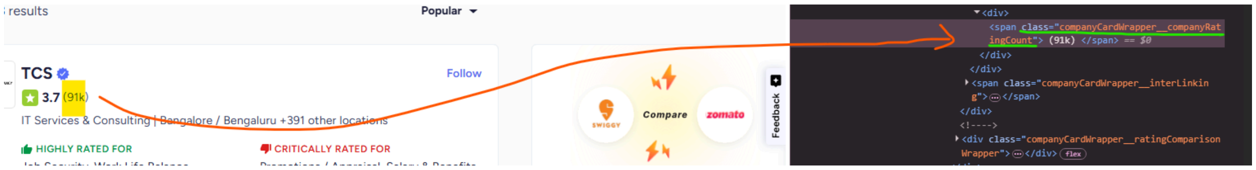
```
▾<div style="height:auto;padding-bottom:
 1px;"> == $0
   " 3.7"
```

```
for i in soup.find_all(style="height:auto;padding-bottom:1px;"):
    print(i.text.strip())
```



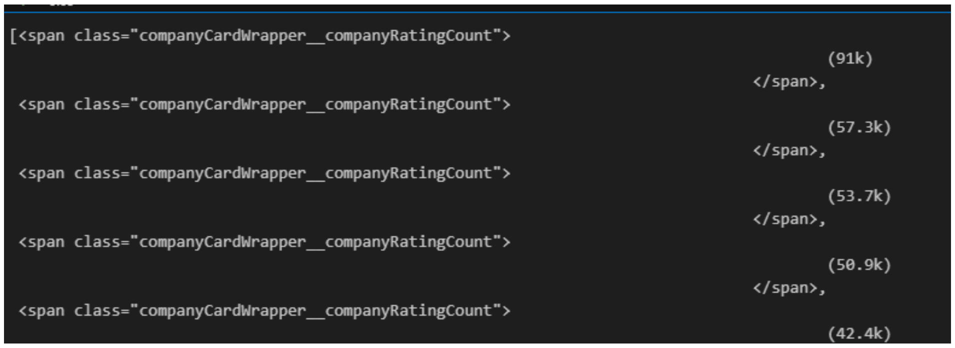
```
3.7
3.8
3.7
3.7
3.7
3.9
3.6
4.0
3.5
3.5
3.8
```

FIND OUT THE NUMBER OF REVIEWS



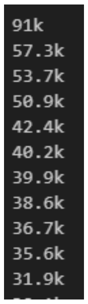
```
▾<div>
   <span class="companyCardWrapper__companyRatingCount"> (91k) </span> == $0
  </div>
 </div>
 ▸<span class="companyCardWrapper__interLinking">…</span>
</div>
<!---->
▸<div class="companyCardWrapper__ratingComparisonWrapper">…</div> flex
```

```
soup.find_all(class_="companyCardWrapper__companyRatingCount")
```



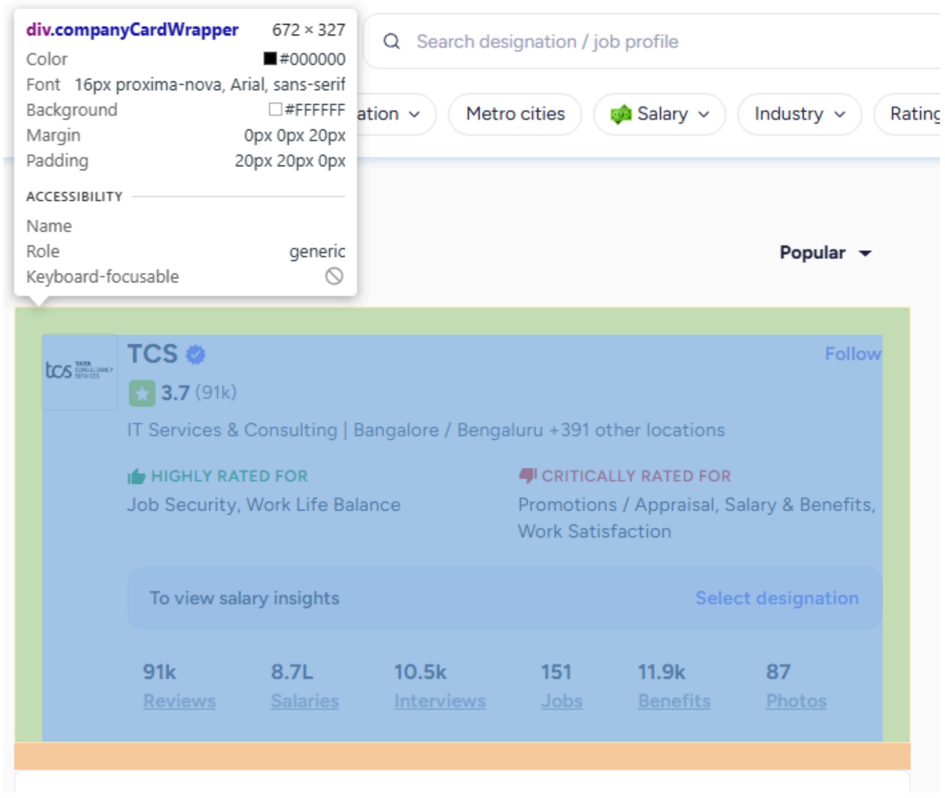
```
[<span class="companyCardWrapper__companyRatingCount">
                                                                  (91k)
                                                          </span>,
 <span class="companyCardWrapper__companyRatingCount">
                                                                  (57.3k)
                                                          </span>,
 <span class="companyCardWrapper__companyRatingCount">
                                                                  (53.7k)
                                                          </span>,
 <span class="companyCardWrapper__companyRatingCount">
                                                                  (50.9k)
                                                          </span>,
 <span class="companyCardWrapper__companyRatingCount">
                                                                  (42.4k)
```

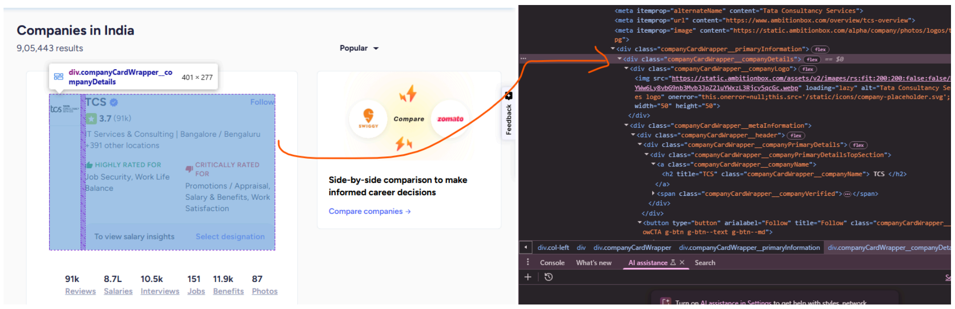
```
for i in soup.find_all(class_="companyCardWrapper__companyRatingCount"):
    print(i.text.strip().strip('()'))
```



```
91k
57.3k
53.7k
50.9k
42.4k
40.2k
39.9k
38.6k
36.7k
35.6k
31.9k
```

- We used `.strip()` twice as we were getting the review number like → **(91k)**
- You can do this 👆 for everything you want.
- But we have everything under this `div` tag 👇 `class="companyCardWrapper__companyDetails"`

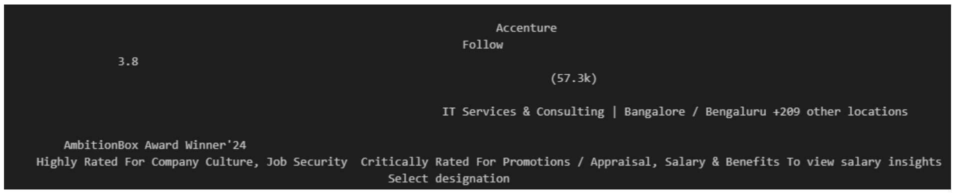



Companies in India
9,05,443 results
Popular
div.companyCardWrapper__companyDetails 401 × 277
Side-by-side comparison to make informed career decisions
Compare companies →
Feedback

```
<meta itemprop="alternateName" content="Tata Consultancy Services">
<meta itemprop="url" content="https://www.ambitionbox.com/overview/tcs-overview">
<meta itemprop="image" content="https://static.ambitionbox.com/alpha/company/photos/logos/t
pg">
▾<div class="companyCardWrapper__primaryInformation"> flex
 ▾<div class="companyCardWrapper__companyDetails"> flex == $0
  ▾<div class="companyCardWrapper__companyLogo"> flex
     <img src="https://static.ambitionbox.com/assets/v2/images/rs:fit:200:200:false:false/
     YWw6Ly8vbG9nb3Mvb3JpZ2luYWxzL3Rjcy5qcGc.webp" loading="lazy" alt="Tata Consultancy Se
     es logo" onerror="this.onerror=null;this.src='/static/icons/company-placeholder.svg';
     width="50" height="50">
   </div>
  ▾<div class="companyCardWrapper__metaInformation">
   ▾<div class="companyCardWrapper__header"> flex
    ▾<div class="companyCardWrapper__companyPrimaryDetails"> flex
     ▾<div class="companyCardWrapper__companyPrimaryDetailsTopSection">
      ▾<a class="companyCardWrapper__companyName">
         <h2 title="TCS" class="companyCardWrapper__companyName"> TCS </h2>
       </a>
      ▸<span class="companyCardWrapper__companyVerified"> … </span>
     </div>
    </div>
    ▾<button type="button" arialabel="Follow" title="Follow" class="companyCardWrapper__
     owCTA g-btn g-btn--text g-btn--md">
```

div.col-left div div.companyCardWrapper div.companyCardWrapper__primaryInformation div.companyCardWrapper__companyDeta
Console What's new AI assistance Search
Turn on AI assistance in Settings to get help with styles, network

```
company=soup.find_all(class_="companyCardWrapper__companyDetails")
print(company[1].text)
```

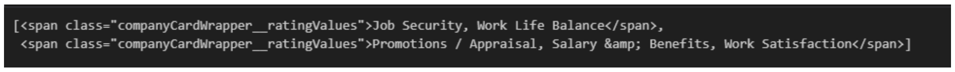


```
                                        Accenture
                                     Follow
         3.8
                                                (57.3k)

                                    IT Services & Consulting | Bangalore / Bengaluru +209 other locations

     AmbitionBox Award Winner'24
  Highly Rated For Company Culture, Job Security  Critically Rated For Promotions / Appraisal, Salary & Benefits To view salary insights
                                 Select designation
```

- This accesses the company at index 1.

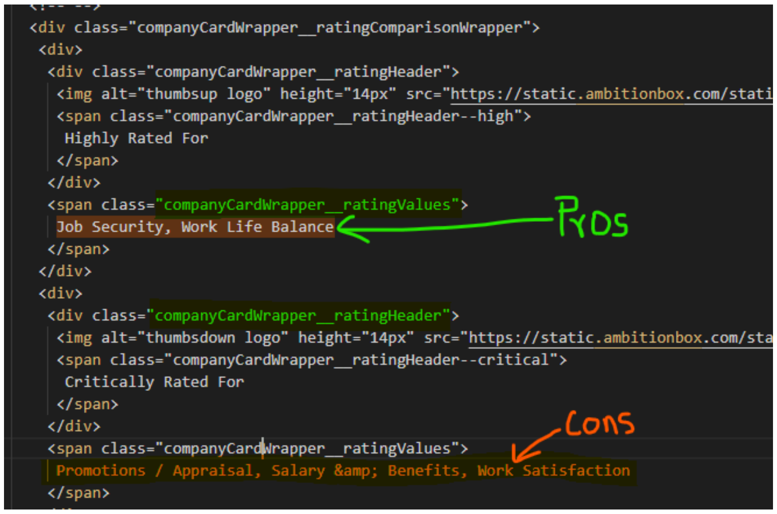
Pros & Cons:

```
company[0](class_="companyCardWrapper__ratingValues")
```



```
[<span class="companyCardWrapper__ratingValues">Job Security, Work Life Balance</span>,
 <span class="companyCardWrapper__ratingValues">Promotions / Appraisal, Salary &amp; Benefits, Work Satisfaction</span>]
```

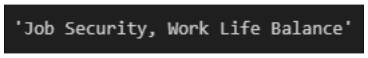
- `class_="companyCardWrapper__ratingValues"` : Gives you both pros and cons as they are under same class



```
<div class="companyCardWrapper__ratingComparisonWrapper">
 <div>
  <div class="companyCardWrapper__ratingHeader">
   <img alt="thumbsup logo" height="14px" src="https://static.ambitionbox.com/stati
   <span class="companyCardWrapper__ratingHeader--high">
    Highly Rated For
   </span>
  </div>
  <span class="companyCardWrapper__ratingValues">
   Job Security, Work Life Balance
  </span>
 </div>
 <div>
  <div class="companyCardWrapper__ratingHeader">
   <img alt="thumbsdown logo" height="14px" src="https://static.ambitionbox.com/sta
   <span class="companyCardWrapper__ratingHeader--critical">
    Critically Rated For
   </span>
  </div>
  <span class="companyCardWrapper__ratingValues">
   Promotions / Appraisal, Salary &amp; Benefits, Work Satisfaction
  </span>
```

- 👆Each container for a company has 2 values for `class_="companyCardWrapper__ratingValues"`

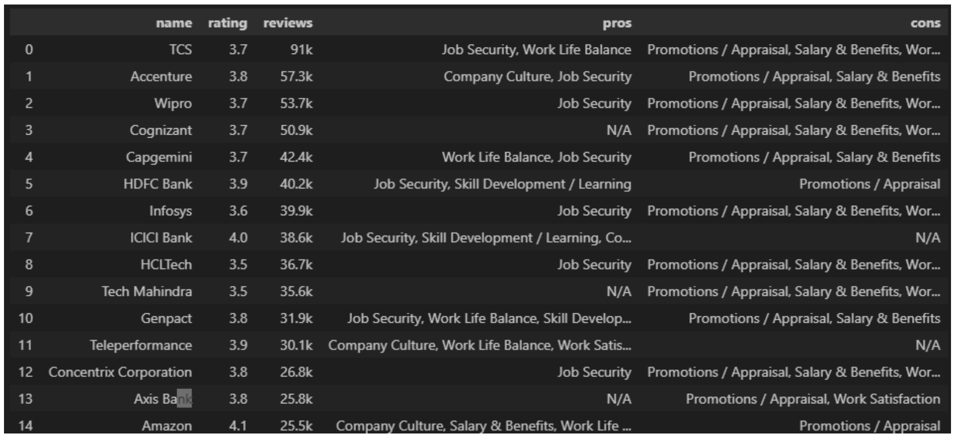
```
#Pros
company[0](class_="companyCardWrapper__ratingValues")[0].text
```



```
'Job Security, Work Life Balance'
```

- But if pro or con is not there, the above code won't work.

```
name = []
rating = []
reviews = []
pros = []
cons = []

for i in company:
    name.append(i.find('h2').text.strip())
    rating.append(i.find(style="height:auto;padding-bottom:1px;").text.strip() if i.find(style="height:auto;padding-bottom:1px;") else "N/A")
    reviews.append(i.find(class_="companyCardWrapper__companyRatingCount").text.strip().strip('()') if i.find(class_="companyCardWrapper__companyRatingCount") else "N/A")

    pros_value = "N/A"
    cons_value = "N/A"
    rating_values = i.find_all(class_="companyCardWrapper__ratingValues")

    for rv in rating_values:
        prev_span = rv.find_previous('span')
        if prev_span:
            if "Highly Rated For" in prev_span.text:
                pros_value = rv.text.strip()
            elif "Critically Rated For" in prev_span.text:
                cons_value = rv.text.strip()

    pros.append(pros_value)
    cons.append(cons_value)

df = pd.DataFrame({
    'name': name,
    'rating': rating,
    'reviews': reviews,
    'pros': pros,
    'cons': cons
})
df
```



	name	rating	reviews	pros	cons
0	TCS	3.7	91k	Job Security, Work Life Balance	Promotions / Appraisal, Salary & Benefits, Wor...
1	Accenture	3.8	57.3k	Company Culture, Job Security	Promotions / Appraisal, Salary & Benefits
2	Wipro	3.7	53.7k	Job Security	Promotions / Appraisal, Salary & Benefits, Wor...
3	Cognizant	3.7	50.9k	N/A	Promotions / Appraisal, Salary & Benefits, Wor...
4	Capgemini	3.7	42.4k	Work Life Balance, Job Security	Promotions / Appraisal, Salary & Benefits
5	HDFC Bank	3.9	40.2k	Job Security, Skill Development / Learning	Promotions / Appraisal
6	Infosys	3.6	39.9k	Job Security	Promotions / Appraisal, Salary & Benefits, Wor...
7	ICICI Bank	4.0	38.6k	Job Security, Skill Development / Learning, Co...	N/A
8	HCLTech	3.5	36.7k	Job Security	Promotions / Appraisal, Salary & Benefits, Wor...
9	Tech Mahindra	3.5	35.6k	N/A	Promotions / Appraisal, Salary & Benefits, Wor...
10	Genpact	3.8	31.9k	Job Security, Work Life Balance, Skill Develop...	Promotions / Appraisal, Salary & Benefits
11	Teleperformance	3.9	30.1k	Company Culture, Work Life Balance, Work Satis...	N/A
12	Concentrix Corporation	3.8	26.8k	Job Security	Promotions / Appraisal, Salary & Benefits, Wor...
13	Axis Bank	3.8	25.8k	N/A	Promotions / Appraisal, Work Satisfaction
14	Amazon	4.1	25.5k	Company Culture, Salary & Benefits, Work Life ...	Promotions / Appraisal

`rating_values = i.find_all(class_="companyCardWrapper__ratingValues")`

- ***What it means:*** Here, it is like one company's card from the webpage. We're asking the computer to look at that card and grab every piece tagged with `class_="companyCardWrapper__ratingValues".`
    - It's like telling a friend, "Find all the lines on this card that list stuff people rate the company for—good or bad."
    - ***You get 2 values:***
        - **Line 1: Pros**
        - **Line 2 Cons**

`for rv in rating_values:`

- This accesses the above pro and con values

Example: If `rating_values = ["Job Security, Work Life Balance", "Promotions / Appraisal"]` , the loop runs twice:

- First time: `rv = "Job Security, Work Life Balance"` (pro)
- Second time: `rv = "Promotions / Appraisal"` (con)

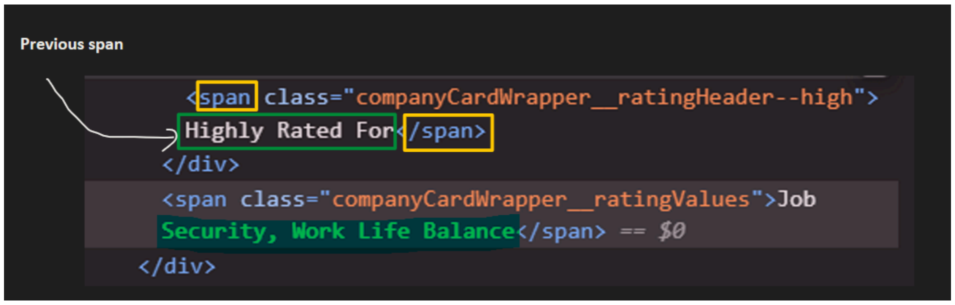
✨ `prev_span = rv.find_previous('span')`

- For each rv (a line like "Job Security, Work Life Balance"), we look backward in the webpage's code to **find the nearest `<span>` tag before it.**
- What's `find_previous`?: Another BeautifulSoup tool. It's like saying, "Look at the stuff before this line and grab the first `<span>` you see."

Example: If the webpage looks like:

```
<span>Highly Rated For</span>
<div class="companyCardWrapper__ratingValues">Job Security, Work Life Balance</div>
```

Then prev_span for rv = "Job Security, Work Life Balance" will be the `<span>` with **"Highly Rated For"**.



```
<span class="companyCardWrapper__ratingHeader--high">
Highly Rated For</span>
</div>
<span class="companyCardWrapper__ratingValues">Job
Security, Work Life Balance</span> == $0
</div>
```

- 👆 The prev span is "**Highly Rated For**"

`if prev_span:`

- ***What it means:*** This checks if we found a `<span>` at all.
  - If prev_span is **empty** (called None in Python), we **skip the next steps**.
  - It's like saying, "If we didn't find a label, don't bother checking further."

`if "Highly Rated For" in prev_span.text:`

- What it means: If we found a `<span>`, we look at its text (what's written inside it) and check if the phrase "Highly Rated For" is there.
- It's like asking, "Does this label say 'Highly Rated For'?"

`pros_value = rv.text.strip()`

- What it means: If the label says "Highly Rated For", we take the text from rv (e.g., "Job Security, Work Life Balance"), clean it up with `strip()` (removes extra spaces), and put it in our pros_value sticky note, replacing "N/A".

`elif "Critically Rated For" in prev_span.text:`

- If "Highly Rated For" wasn't in the label, we check if "Critically Rated For" is there instead. —it's like saying, ***"Okay, if it's not a pro, is it a con?"***

`cons_value = rv.text.strip()`

- If the label says "Critically Rated For", we take rv's text (e.g., "Promotions / Appraisal"), clean it with strip(), and put it in cons_value, replacing "N/A".

`pros.append(pros_value)`
`cons.append(cons_value)`

- Append both the lists

How It All Fits Together?

Imagine we're reading a company card for TCS:

1. Start with `pros_value` = **"N/A"**, `cons_value` = **"N/A".**
2. Find all rating lines: `rating_values` = ["Job Security, Work Life Balance", "Promotions / Appraisal"].
3. Loop through them:
    - First `rv` = "Job Security, Work Life Balance":
        - Look back: `prev_span = <span>Highly Rated For</span>`
        - Check: "Highly Rated For" is there, so `pros_value = "Job Security, Work Life Balance"`.
    - Second `rv = "Promotions / Appraisal"`:
        - Look back: `prev_span = <span>Critically Rated For</span>`
        - Check: "Critically Rated For" is there, so `cons_value = "Promotions / Appraisal"`.
4. Add to lists: pros = ["Job Security, Work Life Balance"], cons = ["Promotions / Appraisal"].

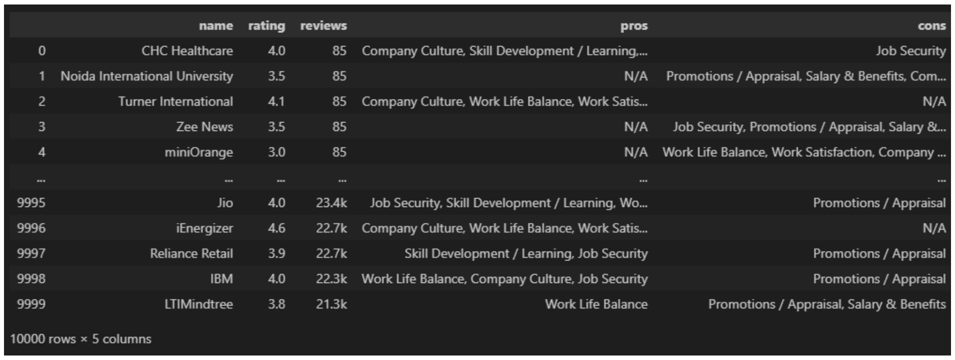
Now, for a company with nothing:

1. Start with `pros_value` = "N/A", `cons_value` = "N/A".
2. Find rating lines: `rating_values = []` (empty).
3. Loop doesn't run (nothing to check).
4. Add to lists: `pros = ["N/A"], cons = ["N/A"].`

Create dataframe for all the pages

```
final=pd.DataFrame()

for j in range(1,51):
    webpage=requests.get(f'https://www.ambitionbox.com/list-of-companies?page={j}', headers=headers).text

    soup=BeautifulSoup(webpage,'lxml')

    company=soup.find_all('div',class_='companyCardWrapper')

    name = []
    rating = []
    reviews = []
    pros = []
    cons = []

    for i in company:
        name.append(i.find('h2').text.strip())
        rating.append(i.find(style="height:auto;padding-bottom:1px;").text.strip() if i.find(style="height:auto;padding-bottom:
        reviews.append(i.find(class_="companyCardWrapper__companyRatingCount").text.strip().strip('()') if i.find(class_="c
        pros_value = "N/A"
        cons_value = "N/A"
        rating_values = i.find_all(class_="companyCardWrapper__ratingValues")

        for rv in rating_values:
            prev_span = rv.find_previous('span')
            if prev_span:
                if "Highly Rated For" in prev_span.text:
                    pros_value = rv.text.strip()
                elif "Critically Rated For" in prev_span.text:
                    cons_value = rv.text.strip()

        pros.append(pros_value)
        cons.append(cons_value)

    df = pd.DataFrame({
        'name': name,
        'rating': rating,
        'reviews': reviews,
        'pros': pros,
        'cons': cons
    })
    final=pd.concat([df, final],ignore_index=True)
```

```
final
```



	name	rating	reviews	pros	cons
0	CHC Healthcare	4.0	85	Company Culture, Skill Development / Learning,...	Job Security
1	Noida International University	3.5	85	N/A	Promotions / Appraisal, Salary & Benefits, Com...
2	Turner International	4.1	85	Company Culture, Work Life Balance, Work Satis...	N/A
3	Zee News	3.5	85	N/A	Job Security, Promotions / Appraisal, Salary &...
4	miniOrange	3.0	85	N/A	Work Life Balance, Work Satisfaction, Company ...
...	...	...	...	...	...
9995	Jio	4.0	23.4k	Job Security, Skill Development / Learning, Wo...	Promotions / Appraisal
9996	iEnergizer	4.6	22.7k	Company Culture, Work Life Balance, Work Satis...	N/A
9997	Reliance Retail	3.9	22.7k	Skill Development / Learning, Job Security	Promotions / Appraisal
9998	IBM	4.0	22.3k	Work Life Balance, Company Culture, Job Security	Promotions / Appraisal
9999	LTIMindtree	3.8	21.3k	Work Life Balance	Promotions / Appraisal, Salary & Benefits

10000 rows × 5 columns