In [16]:

```
import itertools
import pandas as pd

# The new library!
from thefuzz import fuzz, process
```

In [17]:

```
df1 = pd.read_csv('companies_1.csv')
df2 = pd.read_csv('companies_2.csv')
```

In [18]:

```
print('companies_1:',df1.shape)
print('companies_2:',df2.shape)
```

```
companies_1: (266, 1)
companies_2: (368, 1)
```

In [19]:

```
df1.head()
```

Out[19]:

	CLIENT
0	Adobe Systems, Inc.
1	Adventist Health
2	AECOM
3	Aerojet Rockedyne Holdings (GenCorp)
4	Alameda-Contra Costa Transit District

In [20]:

```
df2.head()
```

Out[20]:

	Firm Name
0	AAA Northern California, Nevada & Utah Auto Ex...
1	ACCO Engineered Systems
2	Adams County Retirement Plan
3	Adidas America, Inc.
4	Adobe Systems, Inc.

Data Preprocessing

1. Create the `df` dataframe containing the product of the two CSVs

In [26]:

```
col1 = list(df1['CLIENT'])
col2 = list(df2['Firm Name'])

combination = list(itertools.product(col1,col2))
df = pd.DataFrame(combination,columns=['CSV 1','CSV 2'])
df.head()
```

Out[26]:

	CSV 1	CSV 2
0	Adobe Systems, Inc.	AAA Northern California, Nevada & Utah Auto Ex...
1	Adobe Systems, Inc.	ACCO Engineered Systems
2	Adobe Systems, Inc.	Adams County Retirement Plan
3	Adobe Systems, Inc.	Adidas America, Inc.
4	Adobe Systems, Inc.	Adobe Systems, Inc.

In [27]:
```
df.shape
```

Out[27]:
```
(97888, 2)
```

Calculating the Levenshtein distance

Now, we will learn how to calculate the Levenshtein distance between two strings. Here we will user `partial_ratio` function from the `fuzz` module to compute the "ratio" between two strings. The result is a number between `0` and `100`, with `100` indicating a "perfect" match. Please note that `partial_ratio` gives ratio of the shortest string length to the longest string length. For example, if the first string is `ABC` and the second string is `ABCD`, then the ratio will be `3/4 = 0.75`.

In [30]:
```
fuzz.partial_ratio("Apple", "Apple Inc.")
```

Out[30]:
```
100
```

In [31]:
```
fuzz.partial_ratio("Microsoft", "Apple Inc.")
```

Out[31]:
```
11
```

In [33]:
```
fuzz.partial_ratio("Microsoft", "MSFT")
```

Out[33]:
```
25
```

If we have list of strings, we can calculate the Levenshtein distance between each pair of strings in the list.

In [34]:
```
A = ["Apple", "Alphabet", "Microsoft"]
B = ["MSFT", "Alphabet/Google", "Apple inc."]
```

Below, we combined the two list `A` and `B` into a list of tuples `companies` using `product` function from `itertools` module.

Then, we calculated the partial ratio for each pair of strings in the list `companies` using `partial_ratio` function from `fuzz`.

In [35]:

```python
companies = list(itertools.product(A, B))
companies
```

Out[35]:

```
[('Apple', 'MSFT'),
 ('Apple', 'Alphabet/Google'),
 ('Apple', 'Apple inc.'),
 ('Alphabet', 'MSFT'),
 ('Alphabet', 'Alphabet/Google'),
 ('Alphabet', 'Apple inc.'),
 ('Microsoft', 'MSFT'),
 ('Microsoft', 'Alphabet/Google'),
 ('Microsoft', 'Apple inc.')]
```

In [36]:

```python
for c1, c2 in companies:
    ratio = fuzz.partial_ratio(c1, c2)
    print(f"{c1} > {c2}: {ratio}")
```

```
Apple > MSFT: 0
Apple > Alphabet/Google: 40
Apple > Apple inc.: 100
Alphabet > MSFT: 0
Alphabet > Alphabet/Google: 100
Alphabet > Apple inc.: 38
Microsoft > MSFT: 25
Microsoft > Alphabet/Google: 22
Microsoft > Apple inc.: 22
```

You will see the greater the ratio, the more similar the strings are.

2. Create a new column `Ratio Score` that contains the distance for all the rows in `df`

In [39]:

```python
fuzz.partial_ratio(df['CSV 1'][0],df['CSV 2'][0])
```

Out[39]:

```
26
```

In [45]:

```python
df['Ratio Score'] = [fuzz.partial_ratio(df['CSV 1'][i],df['CSV 2'][i]) for i in range(len(df))]
```

In [46]:
```
df.head()
```

Out[46]:

	CSV 1	CSV 2	Ratio Score
0	Adobe Systems, Inc.	AAA Northern California, Nevada & Utah Auto Ex...	26
1	Adobe Systems, Inc.	ACCO Engineered Systems	56
2	Adobe Systems, Inc.	Adams County Retirement Plan	32
3	Adobe Systems, Inc.	Adidas America, Inc.	47
4	Adobe Systems, Inc.	Adobe Systems, Inc.	100

3. How many rows have a Ratio score of `90` or more?

In [49]:

```
# Try your code here
df_90 = df.loc[df['Ratio Score'] >=90]
df_90
```

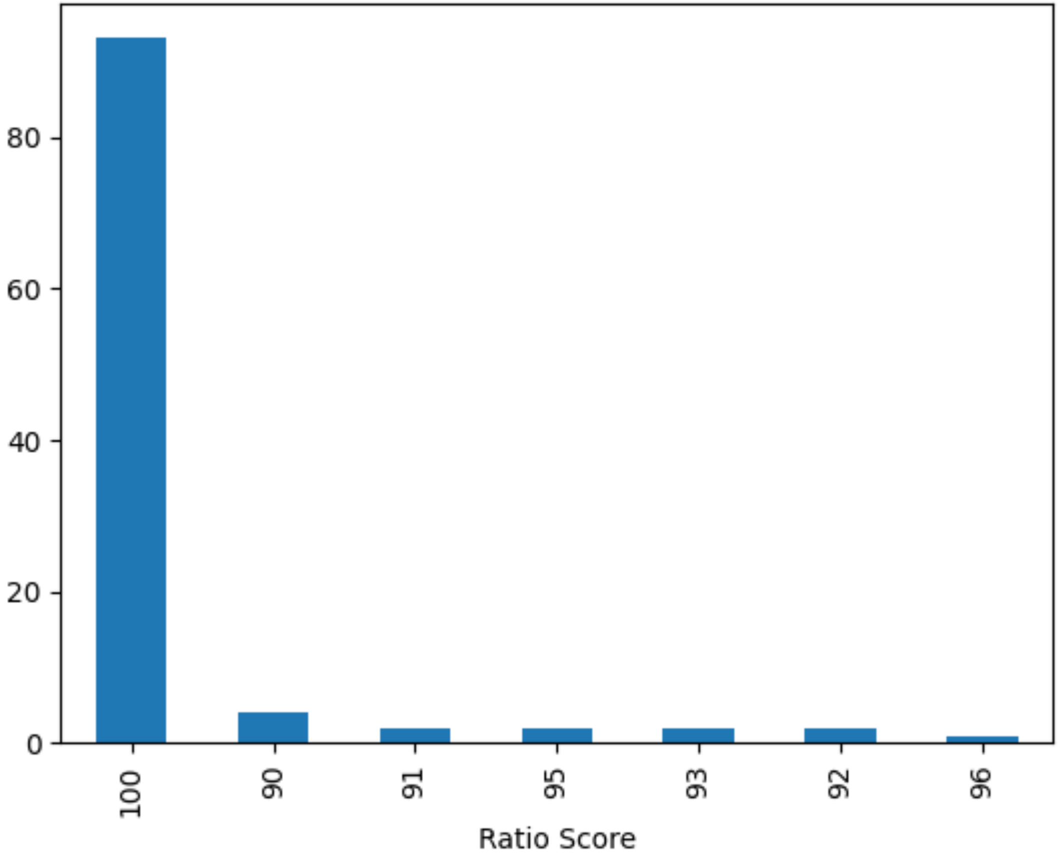
Out[49]:

	CSV 1	CSV 2	Ratio Score
4	Adobe Systems, Inc.	Adobe Systems, Inc.	100
742	AECOM	AECOM Technology Corporation	100
1484	Alameda-Contra Costa Transit District	Alameda-Contra Costa Transit District	100
3697	Amazon	Amazon.com Holdings, Inc.	100
4435	Amgen Inc.	Amgen Inc.	100
...	...	...	...
94923	Virginia Mason Medical Center	Virginia Mason Medical Center	100
96033	Wells Fargo	Wells Fargo & Company	100
96402	Western Digital	Western Digital Corp.	100
96771	Western Union Financial Services, Inc.	Western Union Financial Services, Inc.	100
97141	Weyerhaeuser Company	Weyerhaeuser Company	100

106 rows × 3 columns

```
In [60]: df_90['Ratio Score'].value_counts().plot(kind='bar')
```

```
Out[60]: <Axes: xlabel='Ratio Score'>
```



4. What's the corresponding company in CSV2 to `AECOM` in CSV1?

```
In [52]: # Try your code here
         df_90.loc[df_90['CSV 1']=='AECOM']
```

```
Out[52]:
             CSV 1                      CSV 2   Ratio Score
        742  AECOM  AECOM Technology Corporation        100
```

5. What's the corresponding CSV2 company of Starbucks?

```
In [53]: # Try your code here
         df_90.loc[df_90['CSV 1']=='Starbucks']
```

```
Out[53]:
                   CSV 1                  CSV 2   Ratio Score
        77948  Starbucks  Starbucks Corporation        100
```

6. Is there a matching company for `Pinnacle West Capital Corporation` ?

```
In [54]: # Try your code here
         df_90.loc[df_90['CSV 1']=='Pinnacle West Capital Corporation']
```

```
Out[54]:
           CSV 1   CSV 2   Ratio Score
```

```
In [55]: df_90.loc[df_90['CSV 2']=='Pinnacle West Capital Corporation']
```

```
Out[55]:
           CSV 1   CSV 2   Ratio Score
```

7. How many matching companies are there for `County of Los Angeles Deferred Compensation Program` ?

In [56]:
```
# Try your code here
df_90.loc[df_90['CSV 1']=='County of Los Angeles Deferred Compensation Program']
```

Out[56]:

	CSV 1	CSV 2	Ratio Score
26206	County of Los Angeles Deferred Compensation Pr...	City of Los Angeles Deferred Compensation	95
26227	County of Los Angeles Deferred Compensation Pr...	County of Los Angeles Deferred Compensation Pr...	100

8. Is there a matching company for `The Queens Health Systems` ?

In [58]:
```
# Try your code here
df_90.loc[df_90['CSV 1']=='The Queens Health Systems']
```

Out[58]:

	CSV 1	CSV 2	Ratio Score
84220	The Queens Health Systems	The Queen's Health Systems	96

The End!