Problem 1

In [1]:
```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
import os
sns.set()
```

In [2]:
```
with open('acm (1).txt', 'r', newline = '', encoding = 'utf-8') as file:
    data = file.readlines()
```

In [3]:
```
len(data)
```

Out[3]: 25494853

In [4]:
```
data.append('\n')
data = [a.replace('  ', ' ') for a in data]
data = [a.replace(', Jr.', 'Jr.') for a in data]
data = [a.replace(', I', 'I') for a in data]
data = [a.replace(', II', 'II') for a in data]
data = [a.replace(', -', ' ') for a in data]
data = [a.replace(',  -', '') for a in data]

data = [a.replace('  ', ' ') for a in data]
```

In [5]:
```
df = []
i = 0
j = 0

while i < len(data):
  if(data[i] == '\n'):
   df.append(data[j:i])
   j = i + 1
  i = i + 1

textdata = {'Title':[], 'Author': [], 'Year': [], 'Publication Venue': [], 'Index':[], 'References': [], 'Abstract': []}

for x in df:
  title = []
  author = []
  year = []
  publication_venue = []
  index_id = []
  references = []
  abstract = []

  for y in x:
    if(y[:2] == '#*'):
      title.append(y[2:].split('\n')[0])
    if (y[:2] == '#@'):
      author.append(y[2:].split('\n')[0])
    if (y[:2] == '#t'):
      year.append(y[2:].split('\n')[0])
    if (y[:2] == '#c'):
      publication_venue.append(y[2:].split('\n')[0])
    if (y[:6] == '#index'):
      index_id.append(y[6:].split('\n')[0])
    if (y[:2] == '#%'):
      references.append(y[2:].split('\n')[0])
    if (y[:2] == '#!'):
      abstract.append(y[2:].split('\n')[0])

  textdata["Title"].append('; '.join(title))
  textdata["Author"].append('; '.join(author))
  textdata["Year"].append('; '.join(year))
  textdata["Publication Venue"].append('; '.join(publication_venue))
  textdata["Index"].append('; '.join(index_id))
  textdata["References"].append('; '.join(references))
  textdata["Abstract"].append('; '.join(abstract))

textdata = pd.DataFrame(textdata)
```

In [6]:
```
textdata.head()
```

Out[6]:

	Title	Author	Year	Publication Venue	Index	References	Abstract
0	MOSFET table look-up models for circuit simula...		1984	Integration, the VLSI Journal	1		
1	The verification of the protection mechanisms ...	Virgil D. Gligor	1984	International Journal of Parallel Programming	2		
2	Another view of functional and multivalued dep...	M. Gyssens, J. Paredaens	1984	International Journal of Parallel Programming	3		
3	Entity-relationship diagrams which are in BCNF	Sushil Jajodia, Peter A. Ng, Frederick N. Spri...	1984	International Journal of Parallel Programming	4		
4	The computer comes of age	Rene Moreau	1984	The computer comes of age	5		

In [7]:
```
textdata.tail()
```

Out[7]:

	Title	Author	Year	Publication Venue	Index	References	Abstract
2385062	Linear-time computation of prefix table for we...	-	2016	Theoretical Computer Science	2385063	2381731	The prefix table of a string is one of the mos...
2385063	A space-efficient alphabet-independent Four-Ru...	-	2016	Theoretical Computer Science	2385064	2381731	Given two strings X (| X | = m) and Y (| Y ...
2385064	Computers in Entertainment (CIE) - Special Iss...		2016	Computers in Entertainment (CIE)	2385065		
2385065	Computers in Entertainment (CIE) - Special Iss...		2016	Computers in Entertainment (CIE)	2385066		
2385066							

Part A

In [8]:
```
len(textdata['Author'].explode().unique())
```

Out[8]: 1670103

In [9]:
```
len(textdata['Publication Venue'].unique())
```

Out[9]: 273330

In [10]:
```
len(textdata['Title'].unique())
```

Out[10]: 2183552

In [11]:
```
len(textdata['References'].unique())
```

Out[11]: 884933

Part B

In [12]:
```
textdata[textdata['Publication Venue'].str.contains('Principles and Practice of Knowledge Discovery in Databases')]
```

Out[12]:

	Title	Author	Year	Publication Venue	Index	References	Abstract
799595	Summarization of dynamic content in web collec...	Adam Jatowt, Mitsuru Ishizuka	2004	PKDD '04 Proceedings of the 8th European Confe...	799596	168250; 207271; 217577; 272248; 287615; 357907...	This paper describes a new research proposal o...
799732	Proceedings of the 8th European Conference on ...	Jean-François Boulicaut, Floriana Esposito, Fo...	2004	PKDD '04 Proceedings of the 8th European Confe...	799733		
799733	Random matrices in data analysis	Dimitris Achlioptas	2004	PKDD '04 Proceedings of the 8th European Confe...	799734		We show how carefully crafted random matrices ...
799734	Data privacy	Rakesh Agrawal	2004	PKDD '04 Proceedings of the 8th European Confe...	799735		There is increasing need to build information ...
799735	Breaking through the syntax barrier: searching...	Soumen Chakrabarti	2004	PKDD '04 Proceedings of the 8th European Confe...	799736		The next wave in search technology will be dri...
...	...	...	...	...	...	...	...
1673617	Speeding up logistic model tree induction	Marc Sumner, Eibe Frank, Mark Hall	2005	PKDD'05 Proceedings of the 9th European confer...	1673618	136349; 290481; 810934; 2135000	Logistic Model Trees have been shown to be ver...
1673618	A random method for quantifying changing distr...	Haixun Wang, Jian Pei	2005	PKDD'05 Proceedings of the 9th European confer...	1673619	115607; 342599; 400846; 424996; 443615; 481459...	In applications such as fraud and intrusion de...
1673619	Deriving class association rules based on leve...	Takashi Washio, Koutarou Nakanishi, Hiroshi Mo...	2005	PKDD'05 Proceedings of the 9th European confer...	1673620	210159; 248791; 397383; 466482; 481289; 546046...	Most approaches of Class Association Rule (CAR...
1673620	An incremental algorithm for mining generators...	Lijun Xu, Kanglin Xie	2005	PKDD'05 Proceedings of the 9th European confer...	1673621	280466; 464203; 466663; 481289; 511332; 546697...	This paper presents an efficient algorithm for...
1673621	Hybrid technique for artificial neural network...	Cleber Zanchettin, Teresa Bernarda Ludermir	2005	PKDD'05 Proceedings of the 9th European confer...	1673622	11719; 36407; 369235; 386198; 388153; 465881; ...	This work presents a technique that integrates...

212 rows × 7 columns

In [13]:
```
len(textdata[textdata['Publication Venue'].str.contains('Principles and Practice of Knowledge Discovery in Databases')])
```

Out[13]: 212

Observation:
The Count numbers don't seem accurate. For the same conference, we can see that the venues have different names. While a different publication don't have the same names. This will increase our count of venues than the true count as it is not consistent.

Part C

In [14]:
```
textdata['Author'] = textdata['Author'].str.split(', ')
```

In [15]:
```
textdata = textdata.explode('Author')
```

In [16]:
```
pubs_per_author = textdata.groupby(['Author'],as_index=False)['Title'].count()
```

In [17]:
```
pubs_per_author = pubs_per_author[pubs_per_author["Author"]!= ""]
```

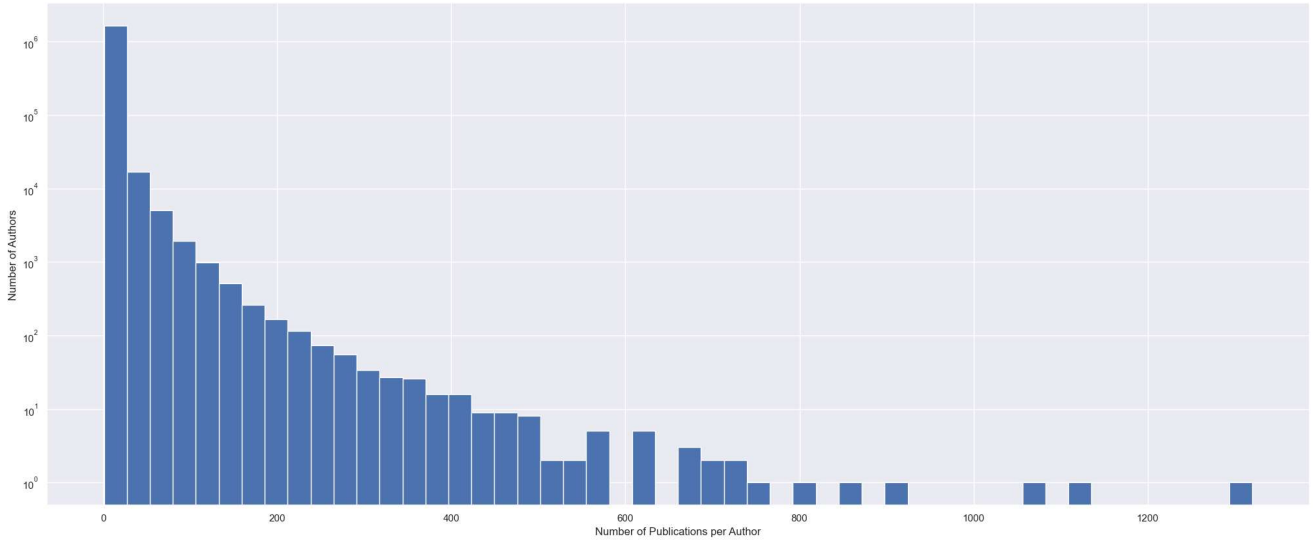
In [18]:
```
pubs_per_author.sort_values(["Title"],ascending = False)
```

Out[18]:

	Author	Title
1539675	Wei Wang	1320
275763	Computer Staff	1111
878237	Linux Journal Staff	1082
860548	Lei Zhang	906
1539831	Wei Zhang	859
...	...	...
692385	Jia-Shiuan Tsai	1
692383	Jia-Shing Sheu	1
692382	Jia-Shing Ma	1
692381	Jia-Shing Chen	1
1668882	脓bero Camilo Kreps Ben挪tez	1

1668882 rows × 2 columns

In [19]:
```
pubs_per_author['Title'].plot(kind = "hist", logy = True, bins = 50, figsize = (25,10))

plt.xlabel("Number of Publications per Author")
plt.ylabel("Number of Authors")
plt.show()
```



Part D

In [20]:
```
mean = np.mean(pubs_per_author['Title'])
std_dev = np.std(pubs_per_author['Title'])

q1 = np.percentile(pubs_per_author['Title'], 25)
q2 = np.percentile(pubs_per_author['Title'], 50)
q3 = np.percentile(pubs_per_author['Title'], 75)

print("Mean:", mean)
print("Standard Deviation:", std_dev)
print("Q1 (First Quartile):", q1)
print("Q2 (Second quartile or median):", q2)
print("Q3 (Third quartile):", q3)
```

```
Mean: 3.388953203402038
Standard Deviation: 9.826593915411662
Q1 (First Quartile): 1.0
Q2 (Second quartile or median): 1.0
Q3 (Third quartile): 2.0
```

Majority of authors have one publications as both the first quartile & median values are 1. The distribution is also skewed to the right.

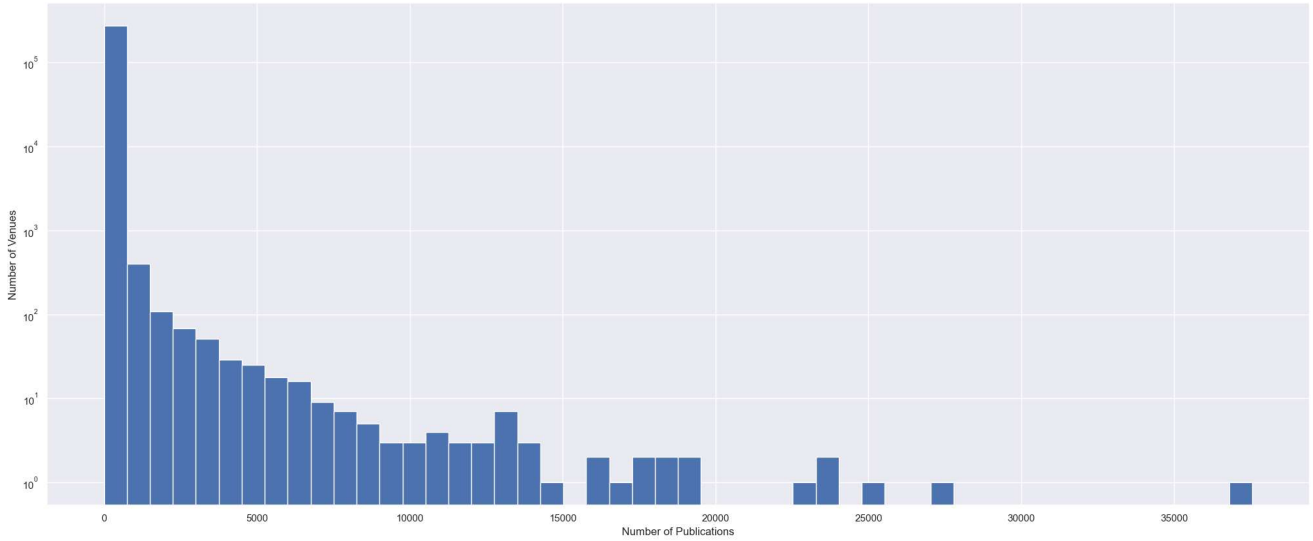
Part E

In [21]:
```
pubs_per_venue = textdata.groupby('Publication Venue')['Title'].count()
```

In [22]:
```
pubs_per_venue.sort_values(ascending = False)
```

Out[22]:
```
Publication Venue
Microelectronic Engineering                                                                    37546
Bioinformatics                                                                                 27385
IEEE Transactions on Information Theory                                                        25174
Expert Systems with Applications: An International Journal                                     23465
IEEE Transactions on Signal Processing                                                         23419
                                                                                               ...
A hop by hop architecture for multicast transport in ad hoc wireless networks                      1
A history-based semantics for algebraic methods in object-oriented software engineering           1
A history of modern computing                                                                      1
A history of general purpose computer uses in the united states 1954 to 1977 and likely future trends.    1
“Virtual fixtures”: perceptual overlays enhance operator performance in telepresence tasks         1
Name: Title, Length: 273330, dtype: int64
```

In [23]:
```
pubs_per_venue.plot(kind="hist", logy = True, bins = 50, figsize = (25,10))

plt.xlabel("Number of Publications")
plt.ylabel("Number of Venues")
plt.show()
```



In [24]:
```
most_pubs_venue = pubs_per_venue.idxmax()

print("Venue with the largest number of publications:", most_pubs_venue)
```

```
Venue with the largest number of publications: Microelectronic Engineering
```

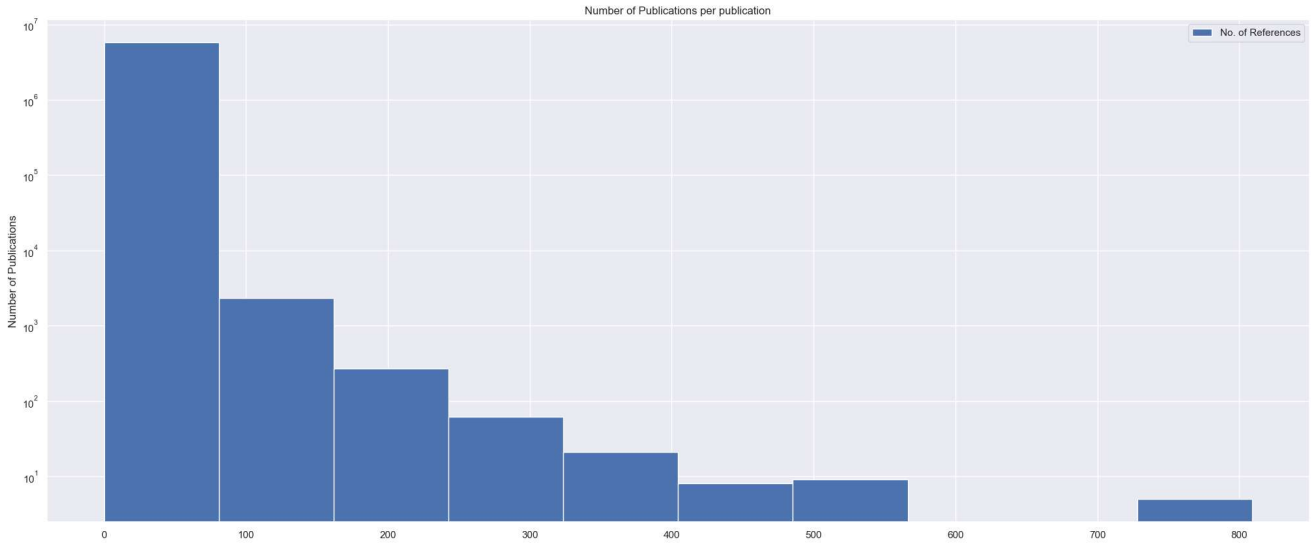
Part F

In [25]:
```
textdata_1 = textdata.copy()
textdata_1['References'] = textdata_1['References'].apply(lambda x: x.split('; '))
textdata_1['No. of References'] = textdata_1['References'].apply(lambda x: 0 if x==[''] else len(x))
textdata_1.tail()
```

Out[25]:

	Title	Author	Year	Publication Venue	Index	References	Abstract	No. of References
2385062	Linear-time computation of prefix table for we...	-	2016	Theoretical Computer Science	2385063	[2381731]	The prefix table of a string is one of the mos...	1
2385063	A space-efficient alphabet-independent Four-Ru...	-	2016	Theoretical Computer Science	2385064	[2381731]	Given two strings X (| X | = m) and Y (| Y ...	1
2385064	Computers in Entertainment (CIE) - Special Iss...		2016	Computers in Entertainment (CIE)	2385065	[]		0
2385065	Computers in Entertainment (CIE) - Special Iss...		2016	Computers in Entertainment (CIE)	2385066	[]		0
2385066						[]		0

In [26]:
```
textdata_1.plot(kind="hist", logy = True, figsize = (25,10), title = 'Number of Publications per publication')

plt.ylabel("Number of Publications")
plt.show()
```



In [27]:
```
reference_high = textdata_1.sort_values('No. of References', ascending = False)['Title'].values[0]
reference_high
```

Out[27]: 'Proceedings of the Twenty-Fourth ACM Symposium on Operating Systems Principles'

In [28]:
```
textdata_1.tail()
```

Out[28]:

	Title	Author	Year	Publication Venue	Index	References	Abstract	No. of References
2385062	Linear-time computation of prefix table for we...	-	2016	Theoretical Computer Science	2385063	[2381731]	The prefix table of a string is one of the mos...	1
2385063	A space-efficient alphabet-independent Four-Ru...	-	2016	Theoretical Computer Science	2385064	[2381731]	Given two strings X (| X | = m) and Y (| Y ...	1
2385064	Computers in Entertainment (CIE) - Special Iss...		2016	Computers in Entertainment (CIE)	2385065	[]		0
2385065	Computers in Entertainment (CIE) - Special Iss...		2016	Computers in Entertainment (CIE)	2385066	[]		0
2385066						[]		0

In [29]:
```
textdata_1 = textdata_1.explode('References').groupby('References',as_index=False).agg(
                                                    No_of_Citations = ('Index',np.count_nonzero))
textdata_1.head()
```

Out[29]:

	References	No_of_Citations
0		2875506
1	10	2
2	1000	6
3	10000	1
4	100000	6

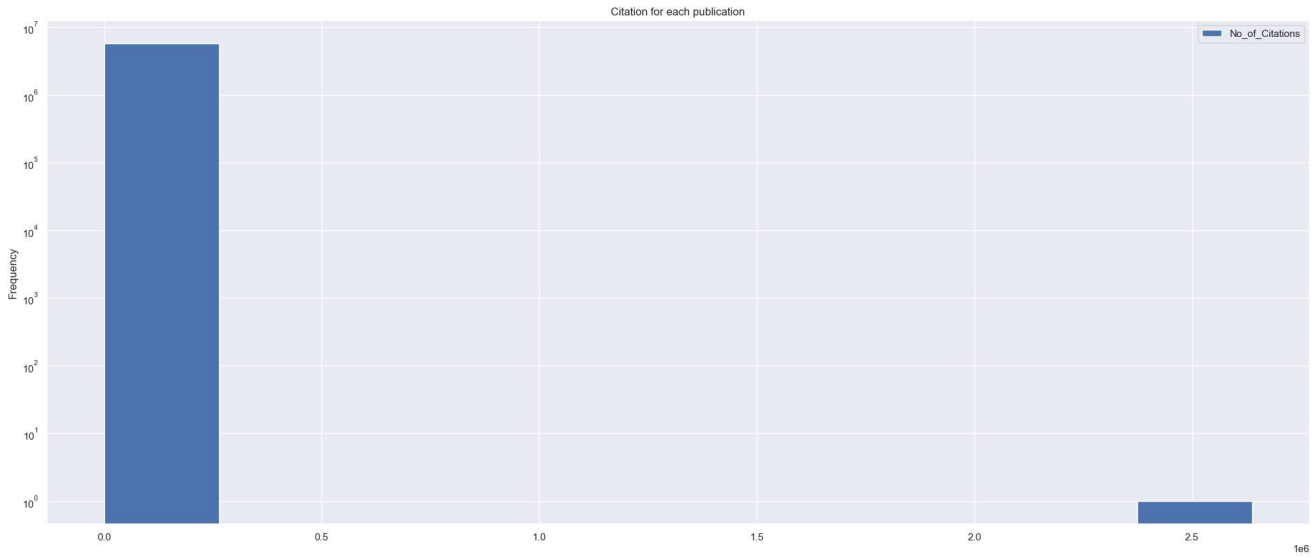
In [30]:
```
textdata = textdata.merge(textdata_1[1:],how='left',left_on='Index',right_on='References').fillna(0)
textdata.head()
```

Out[30]:

	Title	Author	Year	Publication Venue	Index	References_x	Abstract	References_y	No_of_Citations
0	MOSFET table look-up models for circuit simula...		1984	Integration, the VLSI Journal	1			0	0.0
1	The verification of the protection mechanisms ...	Virgil D. Gligor	1984	International Journal of Parallel Programming	2			0	0.0
2	Another view of functional and multivalued dep...	M. Gyssens	1984	International Journal of Parallel Programming	3			3	6.0
3	Another view of functional and multivalued dep...	J. Paredaens	1984	International Journal of Parallel Programming	3			3	6.0
4	Entity-relationship diagrams which are in BCNF	Sushil Jajodia	1984	International Journal of Parallel Programming	4			4	8.0

In [31]:
```
textdata[1:].plot(y = 'No_of_Citations', kind = 'hist', logy = True, figsize = (25,10))
plt.title('Citation for each publication')
```

Out[31]: Text(0.5, 1.0, 'Citation for each publication')



In [32]:
```
textdata[textdata['Index'] == '2135000']['Title'].values[0]
```

Out[32]: 'INFORMS Journal on Computing'

In [33]:
```
textdata_1[1:].sort_values('No_of_Citations', ascending = False)['No_of_Citations'].values[0]
```

Out[33]: 2638303

In [34]:
```
textdata
```

Out[34]:

	Title	Author	Year	Publication Venue	Index	References_x	Abstract	References_y	No_of_Citations
0	MOSFET table look-up models for circuit simula...		1984	Integration, the VLSI Journal	1			0	0.0
1	The verification of the protection mechanisms ...	Virgil D. Gligor	1984	International Journal of Parallel Programming	2			0	0.0
2	Another view of functional and multivalued dep...	M. Gyssens	1984	International Journal of Parallel Programming	3			3	6.0
3	Another view of functional and multivalued dep...	J. Paredaens	1984	International Journal of Parallel Programming	3			3	6.0
4	Entity-relationship diagrams which are in BCNF	Sushil Jajodia	1984	International Journal of Parallel Programming	4			4	8.0
...	...	...	...	...	...	...	...	...	...
5806864	Linear-time computation of prefix table for we...	-	2016	Theoretical Computer Science	2385063	2381731	The prefix table of a string is one of the mos...	0	0.0
5806865	A space-efficient alphabet-independent Four-Ru...	-	2016	Theoretical Computer Science	2385064	2381731	Given two strings X (| X | = m) and Y (| Y ...	0	0.0
5806866	Computers in Entertainment (CIE) - Special Iss...		2016	Computers in Entertainment (CIE)	2385065			0	0.0
5806867	Computers in Entertainment (CIE) - Special Iss...		2016	Computers in Entertainment (CIE)	2385066			0	0.0
5806868								0	0.0

5806869 rows × 9 columns

In [35]:
```
textdata.drop(labels=['References_y'], axis=1, inplace=True)
```

In [36]:
```
textdata.rename(columns={'References_x' :'References'},inplace=True)
```

In [37]:
```
textdata
```

Out[37]:

	Title	Author	Year	Publication Venue	Index	References	Abstract	No_of_Citations
0	MOSFET table look-up models for circuit simula...		1984	Integration, the VLSI Journal	1			0.0
1	The verification of the protection mechanisms ...	Virgil D. Gligor	1984	International Journal of Parallel Programming	2			0.0
2	Another view of functional and multivalued dep...	M. Gyssens	1984	International Journal of Parallel Programming	3			6.0
3	Another view of functional and multivalued dep...	J. Paredaens	1984	International Journal of Parallel Programming	3			6.0
4	Entity-relationship diagrams which are in BCNF	Sushil Jajodia	1984	International Journal of Parallel Programming	4			8.0
...	...	...	...	...	...	...	...	...
5806864	Linear-time computation of prefix table for we...	-	2016	Theoretical Computer Science	2385063	2381731	The prefix table of a string is one of the mos...	0.0
5806865	A space-efficient alphabet-independent Four-Ru...	-	2016	Theoretical Computer Science	2385064	2381731	Given two strings X (| X | = m) and Y (| Y ...	0.0
5806866	Computers in Entertainment (CIE) - Special Iss...		2016	Computers in Entertainment (CIE)	2385065			0.0
5806867	Computers in Entertainment (CIE) - Special Iss...		2016	Computers in Entertainment (CIE)	2385066			0.0
5806868								0.0

5806869 rows × 8 columns

Observation:
The journal 'INFORMS Journal on Computing' has the highest citation number, so I believe that this number is ture.

Part G

In [38]:
```
textdata_2 = textdata.groupby('Publication Venue').agg(Number_of_publications = ('Index',np.count_nonzero),
                                                       Total_Citations = ('No_of_Citations',np.sum))
```

In [39]:
```
textdata_2.head()
```

Out[39]:

Publication Venue	Number_of_publications	Total_Citations
	435	7224.0
!%@ (4th ed.): a directory of electronic mail addressing & networks	2	0.0
!%@:: a directory of electronic mail addressing & networks	2	0.0
!%@:: a directory of electronic mail addressing and networks: second edition	2	2.0
"... but will RISC run LISP??" (a feasibility study)	1	1.0

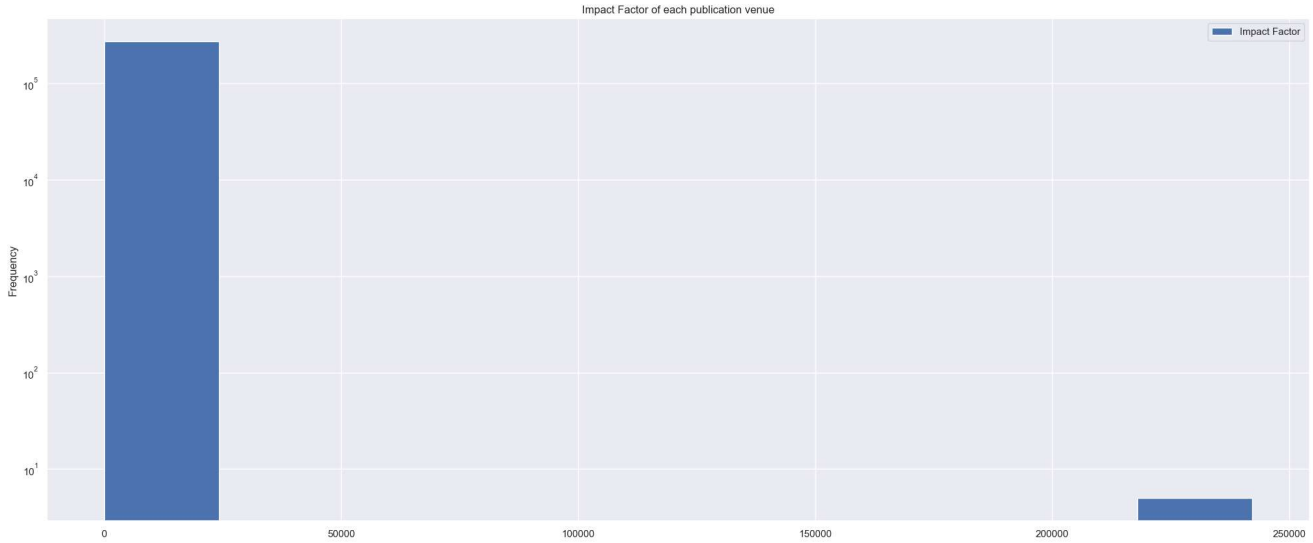
In [40]:
```
textdata_2['Impact Factor'] = textdata_2['Total_Citations'].div(textdata_2['Number_of_publications'])
textdata_2.tail()
```

Out[40]:

Publication Venue	Number_of_publications	Total_Citations	Impact Factor
“High-tech” materials: challenges and opportunities for chemical engineers	1	0.0	0.0
“Meeting the free dreamer”	1	0.0	0.0
“Post-game analysis”: a heuristic resource management framework for concurrent systems	1	10.0	10.0
“Ten trajectories of dawn and a babble of musics”, for computer-generated sounds and video	1	0.0	0.0
“Virtual fixtures”: perceptual overlays enhance operator performance in telepresence tasks	1	8.0	8.0

In [41]:
```
textdata_2.plot(y = 'Impact Factor', kind = 'hist', logy = True, figsize = (25,10))
plt.title('Impact Factor of each publication venue')
```

Out[41]: Text(0.5, 1.0, 'Impact Factor of each publication venue')



Part H

In [42]:
```
textdata_2.sort_values('Impact Factor', ascending = False).head(5)
```

Out[42]:

Publication Venue	Number_of_publications	Total_Citations	Impact Factor
IJIRR: International Journal of Information Retrieval Research.	1	242132.0	242132.0
PVLDB	5	1210660.0	242132.0
AI EDAM	3	726396.0	242132.0
Graphics Interface 1990	2	484264.0	242132.0
Graz	3	726396.0	242132.0

Observation:
We observe that the number of journals which is not equivalent to the highest impact factor. I don't believe this number as these journals only have a single publication.

Part I

In [43]:
```
textdata['Number_of_publications'] = textdata.groupby('Publication Venue')['Index'].transform('count')
textdata.tail()
```

Out[43]:

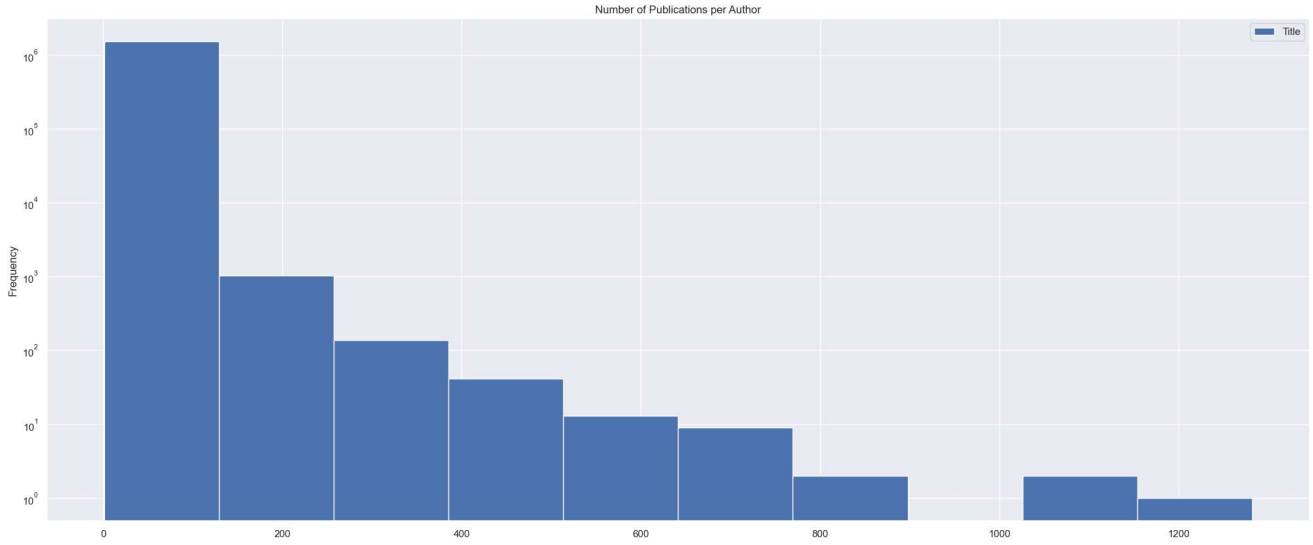
	Title	Author	Year	Publication Venue	Index	References	Abstract	No_of_Citations	Number_of_publications
5806864	Linear-time computation of prefix table for we...	-	2016	Theoretical Computer Science	2385063	2381731	The prefix table of a string is one of the mos...	0.0	17259
5806865	A space-efficient alphabet-independent Four-Ru...	-	2016	Theoretical Computer Science	2385064	2381731	Given two strings X (| X | = m) and Y (| Y ...	0.0	17259
5806866	Computers in Entertainment (CIE) - Special Iss...		2016	Computers in Entertainment (CIE)	2385065			0.0	2
5806867	Computers in Entertainment (CIE) - Special Iss...		2016	Computers in Entertainment (CIE)	2385066			0.0	2
5806868								0.0	436

In [44]:
```
textdata = textdata[textdata['Number_of_publications'] >= 10]
textdata['Author'] = textdata['Author'].apply(lambda x: x.split('; '))
```

```
C:\Users\ayush\AppData\Local\Temp\ipykernel_8996\403841256.py:2: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy (https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy)
  textdata['Author'] = textdata['Author'].apply(lambda x: x.split('; '))
```

In [45]:
```
textdata.explode('Author').groupby('Author').count()[1:].plot(y = 'Title', kind = 'hist', logy = True, figsize = (25,10))
plt.title('Number of Publications per Author')
```

Out[45]: Text(0.5, 1.0, 'Number of Publications per Author')

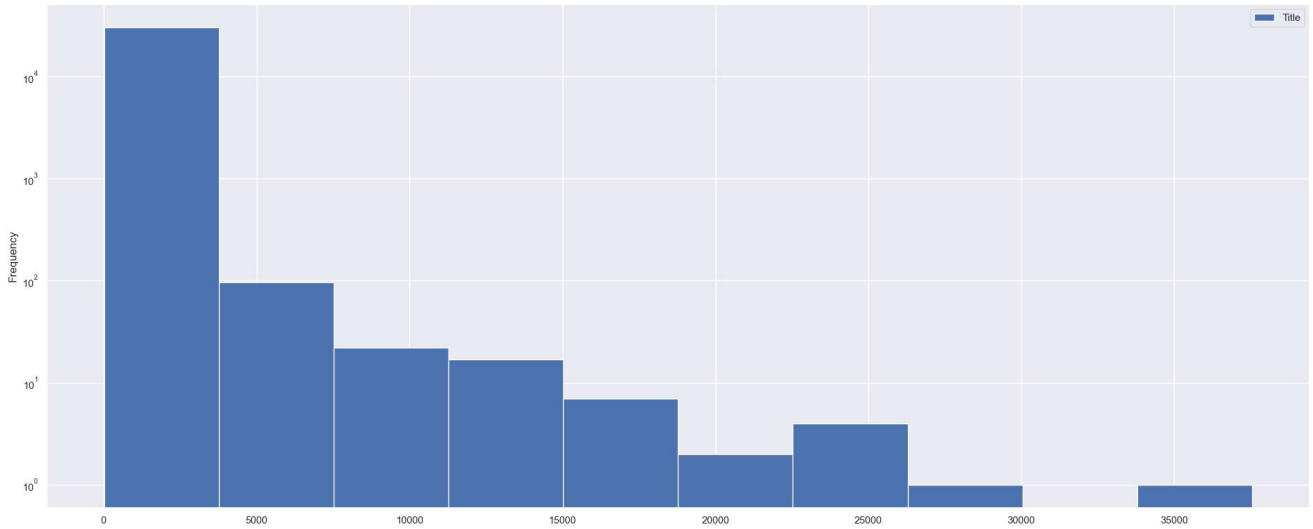


In [46]:
```
textdata_1 = textdata.explode('Author').groupby('Author').count()['Title']
mean = np.mean(textdata_1)
std_dev = np.std(textdata_1)

q1 = np.percentile(textdata_1, 25)
q2 = np.percentile(textdata_1, 50)
q3 = np.percentile(textdata_1, 75)

print("Mean:", mean)
print("Standard Deviation:", std_dev)
print("Q1 (First Quartile):", q1)
print("Q2 (Second quartile or median):", q2)
print("Q3 (Third quartile):", q3)

del textdata_1
```

```
Mean: 3.500984691486151
Standard Deviation: 118.00874824451012
Q1 (First Quartile): 1.0
Q2 (Second quartile or median): 1.0
Q3 (Third quartile): 3.0
```

In [47]:
```
textdata.groupby('Publication Venue').count().plot(y = 'Title', kind = 'hist', logy = True, figsize = (25,10))
```

Out[47]:
```
<AxesSubplot:ylabel='Frequency'>
```



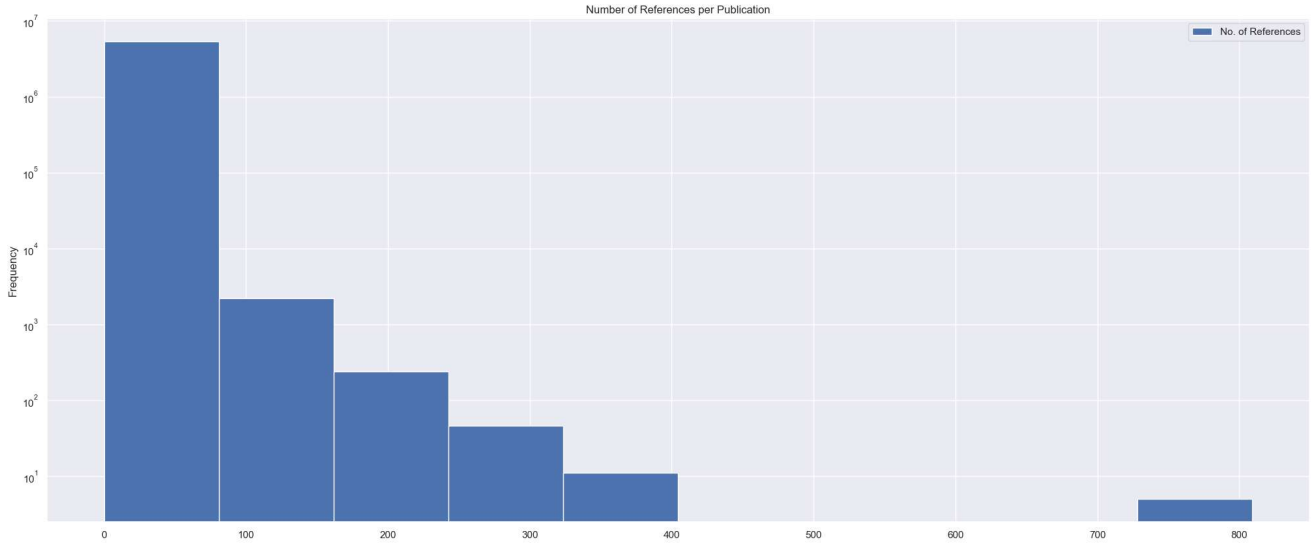
In [48]:
```
textdata_1 = textdata.groupby('Publication Venue').count()
mean = np.mean(textdata_1['Title'])
std_dev = np.std(textdata_1['Title'])

q1 = np.percentile(textdata_1['Title'], 25)
q2 = np.percentile(textdata_1['Title'], 50)
q3 = np.percentile(textdata_1['Title'], 75)

print("Mean:", mean)
print("Standard Deviation:", std_dev)
print("Q1 (First Quartile):", q1)
print("Q2 (Second quartile or median):", q2)
print("Q3 (Third quartile):", q3)
```

```
Mean: 176.2768660145736
Standard Deviation: 733.7731970404155
Q1 (First Quartile): 30.0
Q2 (Second quartile or median): 64.0
Q3 (Third quartile): 142.0
```

In [49]:
```
textdata['References'] = textdata['References'].apply(lambda x: x.split('; '))
textdata['No. of References'] = textdata['References'].apply(lambda x: 0 if x== [''] else len(x))
textdata.plot(y = 'No. of References', kind = 'hist', logy = True, figsize = (25,10))
plt.title('Number of References per Publication')
```

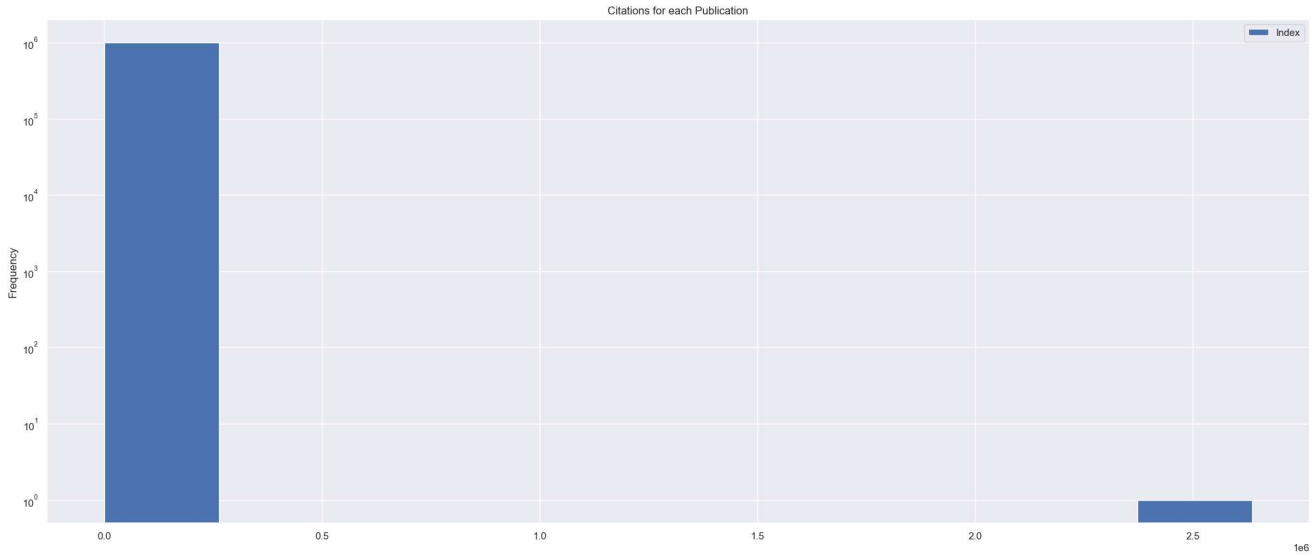
```
C:\Users\ayush\AppData\Local\Temp\ipykernel_8996\3058429958.py:1: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy (https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy)
  textdata['References'] = textdata['References'].apply(lambda x: x.split('; '))
C:\Users\ayush\AppData\Local\Temp\ipykernel_8996\3058429958.py:2: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy (https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy)
  textdata['No. of References'] = textdata['References'].apply(lambda x: 0 if x== [''] else len(x))
```

Out[49]: Text(0.5, 1.0, 'Number of References per Publication')



In [50]:
```
textdata.explode('References').groupby('References').count()[1:].plot(y = 'No_of_Citations', kind = 'hist',
                                                                      logy = True, figsize = (25,10))
plt.title('Citations for each Publication')
```

Out[50]: Text(0.5, 1.0, 'Citations for each Publication')



In [51]:
```
textdata
```

Out[51]:

	Title	Author	Year	Publication Venue	Index	References	Abstract	No_of_Citations	Number_of_publications	No. of References
0	MOSFET table look-up models for circuit simula...	[]	1984	Integration, the VLSI Journal	1	[]		0.0	2335	0
1	The verification of the protection mechanisms ...	[Virgil D. Gligor]	1984	International Journal of Parallel Programming	2	[]		0.0	1749	0
2	Another view of functional and multivalued dep...	[M. Gyssens]	1984	International Journal of Parallel Programming	3	[]		6.0	1749	0
3	Another view of functional and multivalued dep...	[J. Paredaens]	1984	International Journal of Parallel Programming	3	[]		6.0	1749	0
4	Entity-relationship diagrams which are in BCNF	[Sushil Jajodia]	1984	International Journal of Parallel Programming	4	[]		8.0	1749	0
...	...	...	...	...	...	...	...	...	...	...
5806862	Foreword	[-]	2016	Theoretical Computer Science	2385061	[]		0.0	17259	0
5806863	Editorial Board	[]	2016	Theoretical Computer Science	2385062	[]		0.0	17259	0
5806864	Linear-time computation of prefix table for we...	[-]	2016	Theoretical Computer Science	2385063	[2381731]	The prefix table of a string is one of the mos...	0.0	17259	1
5806865	A space-efficient alphabet-independent Four-Ru...	[-]	2016	Theoretical Computer Science	2385064	[2381731]	Given two strings X (| X | = m) and Y (| Y ...	0.0	17259	1
5806868		[]				[]		0.0	436	0

5370451 rows × 10 columns

Observations:
The histogram doesn't change. The mean also changes in this case.

Part J

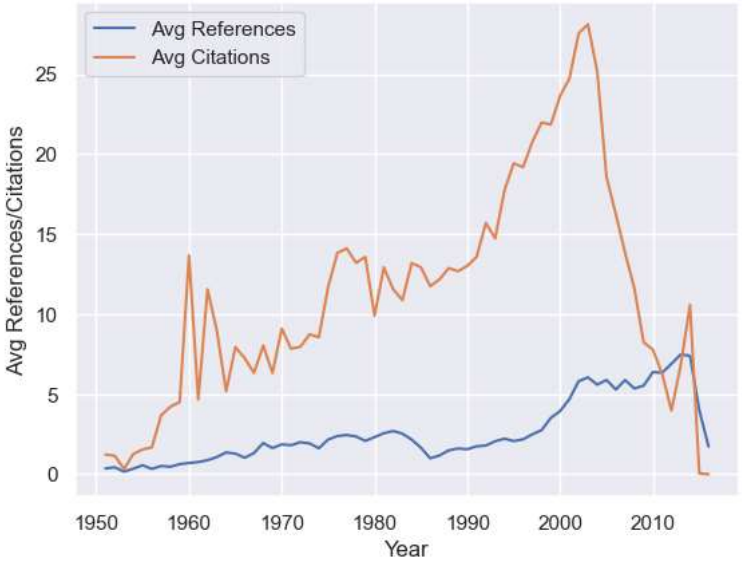
In [52]:
```
textdata['Year'] = pd.to_numeric(textdata['Year'], errors='coerce')
textdata = textdata[pd.notna(textdata['Year'])]
textdata = textdata.groupby('Year').agg(Avg_References = ('No. of References',np.mean),
                                        Avg_Citations = ('No_of_Citations', np.mean))
textdata.head()
```

```
C:\Users\ayush\AppData\Local\Temp\ipykernel_8996\2232921417.py:1: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy (https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy)
  textdata['Year'] = pd.to_numeric(textdata['Year'], errors='coerce')
```

Out[52]:

Year	Avg_References	Avg_Citations
1951.0	0.360656	1.229508
1952.0	0.428571	1.152381
1953.0	0.162162	0.324324
1954.0	0.345133	1.265487
1955.0	0.558824	1.539216

In [53]:
```
import seaborn as sns
sns.lineplot(x = textdata.index, y = "Avg_References", data = textdata, label = 'Avg References')
sns.lineplot(x = textdata.index, y = "Avg_Citations", data = textdata, label = 'Avg Citations')
plt.xlabel('Year')
plt.ylabel('Avg References/Citations')
plt.legend()
plt.show()
```



Observation:
We can see that with increasing number of references, the number of citation also increase tremendously. There is a sudden spike and dip from the year 1990 to 2010.

In [ ]: