

```
In [381... # import pandas
import pandas as pd
# import numpy
import numpy as np
# import matplotlib
from matplotlib import pyplot as plt
# make plots appear and be stored within the notebook
%matplotlib inline
```

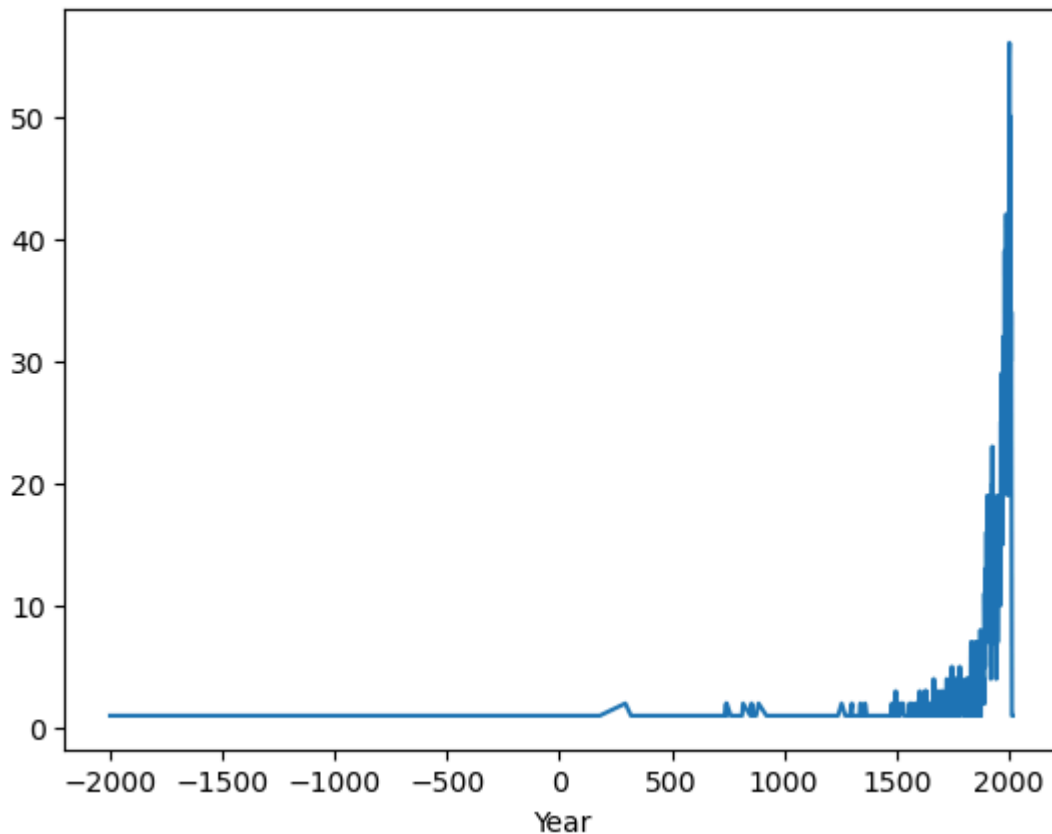
```
In [383... #exercice1
#1.1
Sig_Eqs = pd.read_csv("earthquakes-2024-11-02_17-35-50_+0800.tsv", sep='\t') #读
Total_Deaths = Sig_Eqs.groupby(["Country"])[ 'Deaths' ].sum().sort_values(ascending=
print(Total_Deaths[0:21]) #打印前20
```

	Country	Deaths
0	CHINA	2075947.0
1	TURKEY	1188881.0
2	IRAN	1011453.0
3	ITALY	498418.0
4	SYRIA	439224.0
5	HAITI	323478.0
6	AZERBAIJAN	317219.0
7	JAPAN	279607.0
8	ARMENIA	191890.0
9	PAKISTAN	145083.0
10	IRAQ	136200.0
11	ECUADOR	135496.0
12	TURKMENISTAN	117412.0
13	PERU	102169.0
14	ISRAEL	90388.0
15	PORTUGAL	83572.0
16	GREECE	80378.0
17	CHILE	64277.0
18	INDIA	63507.0
19	TAIWAN	57153.0
20	TUNISIA	48013.0

```
In [385... #1.2
Ms_counts = Sig_Eqs.loc[(Sig_Eqs['Ms']>3.0)].groupby('Year').size() #按照年分组,
Ms_counts.plot()

#发现趋势: 1500年之前基本上是平的, 从1500年到现在显著增加。
#可能是1500年之前大地震的记录很少或者没有被保存下来
```

```
Out[385... <Axes: xlabel='Year'>
```



```
In [387... Sig_Eqs
Sig_Eqs.columns
```

```
Out[387... Index(['Search Parameters', 'Id', 'Year', 'Mo', 'Dy', 'Hr', 'Mn', 'Sec', 'Tsu',
      'Vol', 'Country', 'Area', 'Region', 'Location Name', 'Latitude',
      'Longitude', 'Focal Depth (km)', 'Mag', 'Mw', 'Ms', 'Mb', 'Ml', 'Mfa',
      'Unk', 'MMI Int', 'Deaths', 'Death Description', 'Missing',
      'Missing Description', 'Injuries', 'Injuries Description',
      'Damage ($Mil)', 'Damage Description', 'Houses Destroyed',
      'Houses Destroyed Description', 'Houses Damaged',
      'Houses Damaged Description', 'Total Deaths', 'Total Death Description',
      'Total Missing', 'Total Missing Description', 'Total Injuries',
      'Total Injuries Description', 'Total Damage ($Mil)',
      'Total Damage Description', 'Total Houses Destroyed',
      'Total Houses Destroyed Description', 'Total Houses Damaged',
      'Total Houses Damaged Description'],
      dtype='object')
```

```
In [389... #1.3
def CountEq_LargestEq(country):
    print(Total_Deaths.loc[Total_Deaths['Country']==country]) #打印输入国家的死
    #只筛选输入的国家，按照Ms降序排序，定位在第1行，'Year','Mo','Dy','Location Na
    print(Sig_Eqs.loc[Sig_Eqs['Country']==country].sort_values(['Ms'],ascending

Country=list(Total_Deaths['Country']) #按照死亡总人数降序创建国家的列表

for element in Country: #遍历集合里的每一个国家
    CountEq_LargestEq(element) #打印死亡总人数和最大地震的时间地点
```

	Country	Deaths				Location Name
0	CHINA	2075947.0				
	Year	Mo	Dy			
1920.0	12.0	16.0	CHINA:	GANSU PROVINCE,	SHANXI PROVINCE	
	Country	Deaths				
1	TURKEY	1188881.0				
	Year	Mo	Dy			Location Name
1999.0	8.0	17.0	TURKEY:	ISTANBUL,	KOCAELI, SAKARYA	
	Country	Deaths				
2	IRAN	1011453.0				
	Year	Mo	Dy			Location Name
856.0	12.0	22.0	IRAN:	DAMGHAN,	QUMIS	
	Country	Deaths				
3	ITALY	498418.0				
	Year	Mo	Dy			Location Name
1915.0	1.0	13.0	ITALY:	MARSICA,	AVEZZANO, ABRUZZI	
	Country	Deaths				
4	SYRIA	439224.0				
	Year	Mo	Dy			Location Name
1202.0	5.0	20.0	SYRIA:	SOUTHWESTERN		
	Country	Deaths				
5	HAITI	323478.0				
	Year	Mo	Dy			Location Name
1842.0	5.0	7.0	HAITI:	CAP-HAITIEN		
	Country	Deaths				
6	AZERBAIJAN	317219.0				
	Year	Mo	Dy			Location Name
1667.0	11.0	NaN	AZERBAIJAN:	SHEMAKHA (SAMAXI)		
	Country	Deaths				
7	JAPAN	279607.0				
	Year	Mo	Dy			Location Name
869.0	7.0	13.0	JAPAN:	SANRIKU		
	Country	Deaths				
8	ARMENIA	191890.0				
	Year	Mo	Dy			Location Name
1988.0	12.0	7.0	ARMENIA:	LENINAKAN,	SPITAK, KIROVAKAN	
	Country	Deaths				
9	PAKISTAN	145083.0				
	Year	Mo	Dy			Location Name
1945.0	11.0	27.0	PAKISTAN:	MAKRAN COAST		
	Country	Deaths				
10	IRAQ	136200.0				
	Year	Mo	Dy			Location Name
1864.0	12.0	2.0	IRAQ:	ZURBATIYAH,	BADRAH, TURSAQ, BAGHDAD	
	Country	Deaths				
11	ECUADOR	135496.0				
	Year	Mo	Dy			Location Name
1906.0	1.0	31.0	ECUADOR:	OFF COAST		
	Country	Deaths				
12	TURKMENISTAN	117412.0				
	Year	Mo	Dy			Location Name
1895.0	7.0	8.0	TURKMENISTAN:	UZUN-ADA		
	Country	Deaths				
13	PERU	102169.0				
	Year	Mo	Dy			Location Name
1619.0	2.0	14.0	PERU:	TRUJILLO,	PIURA, SANTA	
	Country	Deaths				
14	ISRAEL	90388.0				
	Year	Mo	Dy			Location Name
1759.0	10.0	30.0	ISRAEL:	ZEFAT (SAFED)		

	Country	Deaths					Location Name
15	PORTUGAL	83572.0					
	Year	Mo	Dy				Location Name
1761.0	3.0	30.0		PORTUGAL:	LISBON,	PORTO	
	Country	Deaths					
16	GREECE	80378.0					
	Year	Mo	Dy				Location Name
1303.0	8.0	8.0		GREECE:	CRETE		
	Country	Deaths					
17	CHILE	64277.0					
	Year	Mo	Dy				Location Name
1730.0	7.0	8.0		CHILE:	VALPARAISO		
	Country	Deaths					
18	INDIA	63507.0					
	Year	Mo	Dy				Location Name
1897.0	6.0	12.0		INDIA:	ASSAM;	BANGLADESH	
	Country	Deaths					
19	TAIWAN	57153.0					
	Year	Mo	Dy				Location Name
1920.0	6.0	5.0			TAIWAN		
	Country	Deaths					
20	TUNISIA	48013.0					
	Year	Mo	Dy				Location Name
1957.0	2.0	20.0		TUNISIA:	SIDI ABID,SIDI TOUIL (LA MEDJA),CAILLOUX		
	Country	Deaths					
21	VENEZUELA	44480.0					
	Year	Mo	Dy				Location Name
1900.0	10.0	29.0		VENEZUELA:	MACUTO		
	Country	Deaths					
22	EGYPT	41765.0					
	Year	Mo	Dy				Location Name
1995.0	11.0	22.0		EGYPT:	NUWAYBI; SAUDI ARABIA; ISRAEL; JORDAN		
	Country	Deaths					
23	INDONESIA	40805.0					
	Year	Mo	Dy				Location Name
2004.0	12.0	26.0		INDONESIA:	SUMATRA: ACEH: OFF WEST COAST		
	Country	Deaths					
24	ALGERIA	39339.0					
	Year	Mo	Dy				Location Name
1980.0	10.0	10.0		ALGERIA:	NORTHERN		
	Country	Deaths					
25	GUATEMALA	36189.0					
	Year	Mo	Dy				Location Name
1942.0	8.0	6.0		GUATEMALA:	NEAR S COAST		
	Country	Deaths					
26	LEBANON	30208.0					
	Year	Mo	Dy				Location Name
-590.0	NaN	NaN		LEBANON:	SUR (TYRE)		
	Country	Deaths					
27	MOROCCO	22775.0					
	Year	Mo	Dy				Location Name
2004.0	2.0	24.0		MOROCCO:	AL HOCEIMA, IMZOURENE, BENI ABDALLAH		
	Country	Deaths					
28	ARGENTINA	22520.0					
	Year	Mo	Dy				Location Name
1894.0	10.0	27.0		ARGENTINA:	LA RIOJA, SAN JUAN, MENDOZA		
	Country	Deaths					
29	NEPAL	21619.0					
	Year	Mo	Dy				Location Name
1505.0	6.0	6.0		NEPAL-INDIA			

	Country	Deaths							
30	UZBEKISTAN	19904.0							
	Year	Mo	Dy		Location Name				
	1976.0	4.0	8.0		UZBEKISTAN: GAZLI				
	Country	Deaths							
31	MEXICO	16740.0							
	Year	Mo	Dy		Location Name				
	1787.0	3.0	28.0		MEXICO: SAN MARCOS, OAXACA				
	Country	Deaths							
32	TAJIKISTAN	16373.0							
	Year	Mo	Dy		Location Name				
	1907.0	10.0	21.0		TAJIKISTAN: KARATAG				
	Country	Deaths							
33	AFGHANISTAN	15651.0							
	Year	Mo	Dy		Location Name				
	1909.0	7.0	7.0		AFGHANISTAN: HINDU-KUSH				
	Country	Deaths							
34	NICARAGUA	13460.0							
	Year	Mo	Dy		Location Name				
	1898.0	4.0	29.0		NICARAGUA: LEON, CHINANDEGA, MANAGUA				
	Country	Deaths							
35	AZORES (PORTUGAL)	6353.0							
	Year	Mo	Dy		Location Name				
	1980.0	1.0	1.0		AZORES: TERCEIRA, ANGRA DO HEROISMO				
	Country	Deaths							
36	PHILIPPINES	6184.0							
	Year	Mo	Dy		Location Name				
	1897.0	9.0	21.0		PHILIPPINES: MINDANAO, ZAMBOANGA, SULU, ISABELA				
	Country	Deaths							
37	EL SALVADOR	6081.0							
	Year	Mo	Dy		Location Name				
	2001.0	1.0	13.0		EL SALVADOR; GUATEMALA				
	Country	Deaths							
38	FRANCE	5854.0							
	Year	Mo	Dy		Location Name				
	1909.0	6.0	11.0		FRANCE: VERNEGUES, CHARLEVAL, LA ROQUE D'ANTHERON				
	Country	Deaths							
39	COLOMBIA	5616.0							
	Year	Mo	Dy		Location Name				
	1904.0	1.0	20.0		PANAMA-COLOMBIA				
	Country	Deaths							
40	SPAIN	5502.0							
	Year	Mo	Dy		Location Name				
	1884.0	12.0	25.0		SPAIN: ARENAS DEL REY, VEGA, ALHAMA, MALAGA				
	Country	Deaths							
41	AUSTRIA	5040.0							
	Year	Mo	Dy		Location Name				
	1998.0	4.0	12.0		AUSTRIA: ARNOLDSTEIN; SLOVENIA: BOVEC, KOBARID				
	Country	Deaths							
42	CROATIA	5017.0							
	Year	Mo	Dy		Location Name				
	1667.0	4.0	6.0		BALKANS NW: CROATIA: DUBROVNIK: RAGUSA				
	Country	Deaths							
43	GUADELOUPE	5007.0							
	Year	Mo	Dy		Location Name				
	1969.0	12.0	25.0		GUADELOUPE: GRAND BOURG				
	Country	Deaths							
44	YEMEN	4192.0							
	Year	Mo	Dy		Location Name				
	1982.0	12.0	13.0		YEMEN: DHAMAR				

	Country	Deaths				Location Name
45	ALBANIA	3132.0				
	Year	Mo	Dy			
	1893.0	6.0	14.0	ALBANIA:	HIMARA, DHERMI, KUC, KUDHESI, VLORE, KANINA	
	Country	Deaths				
46	BARBADOS	3000.0				
	Year	Mo	Dy			Location Name
	1831.0	8.0	11.0	BARBADOS, SAINT VINCENT, DOMINICA, ANTIGUA		
	Country	Deaths				
47	ROMANIA	2701.0				
	Year	Mo	Dy			Location Name
	1986.0	8.0	30.0	ROMANIA: BUCHAREST, KISHINEV-KAGUL		
	Country	Deaths				
48	COSTA RICA	2655.0				
	Year	Mo	Dy			Location Name
	2012.0	9.0	5.0	COSTA RICA: NICOYA		
	Country	Deaths				
49	RUSSIA	2472.0				
	Year	Mo	Dy			Location Name
	1952.0	11.0	4.0	RUSSIA: KAMCHATKA PENINSULA		
	Country	Deaths				
50	JAMAICA	2300.0				
	Year	Mo	Dy			Location Name
	1899.0	6.0	14.0	JAMAICA		
	Country	Deaths				
51	GERMANY	2000.0				
	Year	Mo	Dy			Location Name
	1978.0	9.0	3.0	GERMANY		
	Country	Deaths				
52	UK	1400.0				
	Year	Mo	Dy			Location Name
	1931.0	6.0	7.0	UK: SCARBOROUGH, GRIMSBY		
	Country	Deaths				
53	USA	1371.0				
	Year	Mo	Dy			Location Name
	1957.0	3.0	9.0	ALASKA		
	Country	Deaths				
54	MYANMAR (BURMA)	1169.0				
	Year	Mo	Dy			Location Name
	1912.0	5.0	23.0	MYANMAR (BURMA): MANDALAY, MOGOK, MAYMYO		
	Country	Deaths				
55	MACEDONIA	1083.0				
	Year	Mo	Dy			Location Name
	1979.0	5.0	24.0	BALKANS NW: MACEDONIA: DEBAR		
	Country	Deaths				
56	SAINT LUCIA	900.0				
	Year	Mo	Dy			Location Name
	1788.0	10.0	12.0	SAINT LUCIA		
	Country	Deaths				
57	KYRGYZSTAN	657.0				
	Year	Mo	Dy			Location Name
	1911.0	1.0	3.0	KAZAKHSTAN: ALMA-ATA, TURKESTAN; AFGHANISTAN		
	Country	Deaths				
58	NEW ZEALAND	476.0				
	Year	Mo	Dy			Location Name
	1826.0	NaN	NaN	NEW ZEALAND: FJORDLAND		
	Country	Deaths				
59	GUINEA	443.0				
	Year	Mo	Dy			Location Name
	1983.0	12.0	22.0	GUINEA: GAOUAL-KOUMBIA		

	Country	Deaths			
60	SWITZERLAND	410.0			
	Year	Mo	Dy	Location	Name
1021.0	5.0	12.0		SWITZERLAND	
	Country	Deaths			
61	MARTINIQUE	391.0			
	Year	Mo	Dy	Location	Name
1906.0	12.0	3.0		MARTINIQUE	
	Country	Deaths			
62	BANGLADESH	336.0			
	Year	Mo	Dy	Location	Name
1918.0	7.0	8.0		BANGLADESH: SRIMANGAL	
	Country	Deaths			
63	PAPUA NEW GUINEA	333.0			
	Year	Mo	Dy	Location	Name
2000.0	11.0	16.0		PAPUA NEW GUINEA: NEW IRELAND, DUKE OF YORK	
	Country	Deaths			
64	CANADA	305.0			
	Year	Mo	Dy	Location	Name
1949.0	8.0	22.0		CANADA: QUEEN CHARLOTTE ISLANDS	
	Country	Deaths			
65	LIBYA	300.0			
	Year	Mo	Dy	Location	Name
1963.0	2.0	21.0		LIBYA: BARCE (AL MARJ)	
	Country	Deaths			
66	GEORGIA	285.0			
	Year	Mo	Dy	Location	Name
1905.0	10.0	21.0		GEORGIA: CAUCASUS	
	Country	Deaths			
67	ETHIOPIA	200.0			
	Year	Mo	Dy	Location	Name
1961.0	6.0	1.0		ETHIOPIA: KARAKORE	
	Country	Deaths			
68	UGANDA	171.0			
	Year	Mo	Dy	Location	Name
1966.0	3.0	20.0		UGANDA: KICHWAMBA, BONDIBOGYO; TANZANIA; DR CONGO	
	Country	Deaths			
69	SOUTH KOREA	151.0			
	Year	Mo	Dy	Location	Name
1643.0	7.0	25.0		SEA OF JAPAN	
	Country	Deaths			
70	BULGARIA	138.0			
	Year	Mo	Dy	Location	Name
1904.0	4.0	4.0		BULGARIA: STRUMA	
	Country	Deaths			
71	MONTENEGRO	131.0			
	Year	Mo	Dy	Location	Name
1979.0	4.0	15.0		BALKANS NW: MONTENEGRO	
	Country	Deaths			
72	BOLIVIA	111.0			
	Year	Mo	Dy	Location	Name
1916.0	8.0	25.0		BOLIVIA-NORTHERN CHILE	
	Country	Deaths			
73	IRELAND	100.0			
	Year	Mo	Dy	Location	Name
1490.0	NaN	NaN		IRELAND: SLIGO, MAYO	
	Country	Deaths			
74	HUNGARY	85.0			
	Year	Mo	Dy	Location	Name
1763.0	6.0	28.0		HUNGARY: GYOR; CZECHOSLOVAKIA: KOMAROM	

	Country	Deaths			
75	CONGO	80.0			
	Year	Mo	Dy	Location Name	
	2005.0	12.0	5.0	CONGO: KALEMIE	
				Country	Deaths
76	SOLOMON ISLANDS	56.0			
	Year	Mo	Dy	Location Name	
	1977.0	4.0	21.0	SOLOMON ISLANDS	
				Country	Deaths
77	SOUTH AFRICA	55.0			
	Year	Mo	Dy	Location Name	
	1942.0	11.0	10.0	SOUTH AFRICA: PRINCE EDWARD ISLAND	
				Country	Deaths
78	CUBA	54.0			
	Year	Mo	Dy	Location Name	
	1992.0	5.0	25.0	CUBA: PILON, MANZANILLO	
				Country	Deaths
79	BOSNIA-HERZEGOVINA	48.0			
	Year	Mo	Dy	Location Name	
	1990.0	11.0	27.0	BALKANS NW: BOSNIA-HERZEGOVINA: TITOGRAĐ	
				Country	Deaths
80	CYPRUS	42.0			
	Year	Mo	Dy	Location Name	
	1953.0	9.0	10.0	CYPRUS: PAPHOS	
				Country	Deaths
81	ICELAND	41.0			
	Year	Mo	Dy	Location Name	
	2000.0	6.0	17.0	ICELAND: VESTMANNAEYJAR, HELLA	
				Country	Deaths
82	SOUTH SUDAN	31.0			
	Year	Mo	Dy	Location Name	
	1990.0	5.0	20.0	SOUTH SUDAN: JUBA, MAYA; UGANDA: NAKURA	
				Country	Deaths
83	SLOVAKIA	30.0			
	Year	Mo	Dy	Location Name	
	1443.0	6.0	5.0	SLOVAKIA: BANSKA STIAVNICA	
				Country	Deaths
84	MONGOLIA	30.0			
	Year	Mo	Dy	Location Name	
	1905.0	7.0	23.0	MONGOLIA	
				Country	Deaths
85	TANZANIA	28.0			
	Year	Mo	Dy	Location Name	
	1910.0	12.0	13.0	TANZANIA: RUKWA	
				Country	Deaths
86	POLAND	28.0			
	Year	Mo	Dy	Location Name	
	2019.0	7.0	1.0	POLAND: KATOWICE	
				Country	Deaths
87	GHANA	25.0			
	Year	Mo	Dy	Location Name	
	1939.0	6.0	22.0	GHANA: ACCRA	
				Country	Deaths
88	SLOVENIA	23.0			
	Year	Mo	Dy	Location Name	
	1511.0	3.0	26.0	BALKANS NW: SLOVENIA: IDRIJA, SKOFJA LOKA	
				Country	Deaths
89	MALAYSIA	19.0			
	Year	Mo	Dy	Location Name	
	1976.0	7.0	26.0	MALAYSIA: SABAH: LAHAD, DATU, KANAK	

	Country	Deaths			
90	MALAWI	13.0			
	Year	Mo	Dy	Location Name	
1989.0	3.0	10.0	MALAWI:	SALIMA, DEDZA, MOHINJI	
	Country	Deaths			
91	DOMINICAN REPUBLIC	13.0			
	Year	Mo	Dy	Location Name	
1946.0	8.0	4.0	DOMINICAN REPUBLIC:	NORTHEASTERN COAST	
	Country	Deaths			
92	AUSTRALIA	12.0			
	Year	Mo	Dy	Location Name	
1989.0	5.0	23.0	MACQUARIE ISLAND:	MACQUARIE STATION; NEW ZEALAND	
	Country	Deaths			
93	RWANDA	12.0			
	Year	Mo	Dy	Location Name	
2008.0	2.0	14.0	RWANDA:	GISENYI	
	Country	Deaths			
94	HONDURAS	12.0			
	Year	Mo	Dy	Location Name	
1856.0	8.0	4.0	HONDURAS:	COAST	
	Country	Deaths			
95	BHUTAN	11.0			
	Year	Mo	Dy	Location Name	
2009.0	9.0	21.0	BHUTAN:	TASHIGANG	
	Country	Deaths			
96	UKRAINE	11.0			
	Year	Mo	Dy	Location Name	
1927.0	9.0	11.0	UKRAINE:	CRIMEA: SEBASTOPOL	
	Country	Deaths			
97	SERBIA	7.0			
	Year	Mo	Dy	Location Name	
1980.0	5.0	18.0	BALKANS NW:	SERBIA	
	Country	Deaths			
98	PANAMA	7.0			
	Year	Mo	Dy	Location Name	
1882.0	9.0	7.0	PANAMA:	SAN BLAS ARCHIPELAGO	
	Country	Deaths			
99	VANUATU	6.0			
	Year	Mo	Dy	Location Name	
1910.0	6.0	16.0	VANUATU	ISLANDS	
	Country	Deaths			
100	DJIBOUTI	6.0			
	Year	Mo	Dy	Location Name	
1989.0	8.0	20.0	DJIBOUTI:	GALAFI, YABAKI; ETHIOPIA	
	Country	Deaths			
101	WALLIS AND FUTUNA (FRENCH TERRITORY)	5.0			
	Year	Mo	Dy	Location Name	
1993.0	3.0	12.0	FUTUNA	ISLAND	
	Country	Deaths			
102	MOZAMBIQUE	4.0			
	Year	Mo	Dy	Location Name	
2006.0	2.0	22.0	MOZAMBIQUE		
	Country	Deaths			
103	USA TERRITORY	4.0			
	Year	Mo	Dy	Location Name	
1902.0	9.0	22.0	GUAM:	AGANA	
	Country	Deaths			
104	BURUNDI	3.0			
	Year	Mo	Dy	Location Name	
2004.0	2.0	24.0	BURUNDI:	RUYAGA	

	Country	Deaths					Location Name
105	BRAZIL	3.0					
	Year	Mo	Dy				
1986.0	11.0	30.0	BRAZIL:	JOAO CAMARA,	NATAL		
	Country	Deaths					
106	FIJI	2.0					
	Year	Mo	Dy				
1919.0	1.0	1.0	FIJI ISLANDS				
	Country	Deaths					
107	KAZAKHSTAN	2.0					
	Year	Mo	Dy				
1978.0	3.0	24.0	KAZAKHSTAN:	ALMA-ATA			
	Country	Deaths					
108	TRINIDAD AND TOBAGO	2.0					
	Year	Mo	Dy				
1831.0	12.0	3.0	TRINIDAD & ST. CHRISTOPHER				
	Country	Deaths					
109	MADAGASCAR	2.0					
	Year	Mo	Dy				
2017.0	1.0	11.0	MADAGASCAR:	ANTSIRABE			
	Country	Deaths					
110	BELGIUM	2.0					
	Year	Mo	Dy				
1983.0	11.0	8.0	BELGIUM				
	Country	Deaths					
111	CZECH REPUBLIC	2.0					
	Year	Mo	Dy				
2008.0	11.0	22.0	CZECH REPUBLIC:	KARVINA			
	Country	Deaths					
112	SUDAN	2.0					
	Year	Mo	Dy				
1993.0	8.0	1.0	SUDAN:	KHARTOUM			
	Country	Deaths					
113	THAILAND	1.0					
	Year	Mo	Dy				
1983.0	4.0	22.0	THAILAND:	BANGKOK			
	Country	Deaths					
114	TONGA	1.0					
	Year	Mo	Dy				
1919.0	4.0	30.0	TONGA ISLANDS				
	Country	Deaths					
115	NETHERLANDS	1.0					
	Year	Mo	Dy				Location Name
1992.0	4.0	13.0	THE NETHERLANDS:	ROERMOND;	GERMANY:	BONN,	HEINSBERG
	Country	Deaths					
116	KENYA	1.0					
	Year	Mo	Dy				
1928.0	1.0	6.0	KENYA:	SUBUKIA			
	Country	Deaths					
117	UK TERRITORY	0.0					
	Year	Mo	Dy				Location Name
1983.0	11.0	30.0	INDIAN OCEAN:	CHAGOS ARCHIPELAGO:	DIEGO GARCIA		
	Country	Deaths					
118	SRI LANKA	0.0					
	Year	Mo	Dy				Location Name
1882.0	1.0	NaN	SRI LANKA:	TRINCOMALEE			
	Country	Deaths					
119	VIETNAM	0.0					
	Year	Mo	Dy				Location Name
1983.0	6.0	24.0	VIETNAM				

	Country	Deaths	
120	URUGUAY	0.0	
	Year	Mo	Dy Location Name
1888.0	6.0	5.0	URUGUAY: COLOGNE
	Country	Deaths	
121	SOUTH GEORGIA AND THE SOUTH SANDWICH ISLANDS	0.0	
	Year	Mo	Dy Location Name
1929.0	6.0	27.0	SOUTH SANDWICH ISLANDS
	Country	Deaths	
122	TOGO	0.0	
	Year	Mo	Dy Location Name
1788.0	NaN	NaN	TOGO: AGUNAH, KANA; BENIN: ABOMEY
	Country	Deaths	
123	LAOS	0.0	
	Year	Mo	Dy Location Name
2007.0	5.0	16.0	LAOS: BOKEO; THAILAND: CHIANG RAI, CHIAN SAEN
	Country	Deaths	
124	SOLOMON SEA	0.0	
	Year	Mo	Dy Location Name
1895.0	3.0	6.0	W. SOLOMON SEA
	Country	Deaths	
125	GRENADA	0.0	
	Year	Mo	Dy Location Name
1822.0	12.0	1.0	GRENADA
	Country	Deaths	
126	ANTARCTICA	0.0	
	Year	Mo	Dy Location Name
1998.0	3.0	25.0	BALLENY ISLANDS
	Country	Deaths	
127	ANTIGUA AND BARBUDA	0.0	
	Year	Mo	Dy Location Name
1690.0	4.0	16.0	ANTIGUA; SAINT KITTS AND NEVIS
	Country	Deaths	
128	ATLANTIC OCEAN	0.0	
	Year	Mo	Dy Location Name
1941.0	11.0	25.0	ATLANTIC OCEAN: NORTHERN
	Country	Deaths	
129	BERING SEA	0.0	
	Year	Mo	Dy Location Name
1991.0	2.0	21.0	BERING SEA
	Country	Deaths	
130	BRITISH VIRGIN ISLANDS	0.0	
	Year	Mo	Dy Location Name
1871.0	9.0	NaN	BRITISH VIRGIN ISLANDS: TORTOLA ISLAND
	Country	Deaths	
131	CAMEROON	0.0	
	Year	Mo	Dy Location Name
1909.0	4.0	26.0	CAMEROON: BUEA
	Country	Deaths	
132	CANARY ISLANDS	0.0	
	Year	Mo	Dy Location Name
1800.0	NaN	NaN	CANARY ISLANDS: PENON DE LA GOMERA
	Country	Deaths	
133	CENTRAL AFRICAN REPUBLIC	0.0	
	Year	Mo	Dy Location Name
1921.0	9.0	16.0	CENTRAL AFRICAN REPUBLIC: NOLA
	Country	Deaths	
134	COMOROS	0.0	
	Year	Mo	Dy Location Name
2018.0	5.0	15.0	COMOROS: MAYOTTE

	Country	Deaths	
135	COTE D'IVOIRE	0.0	
	Year Mo Dy		Location Name
1879.0	2.0 11.0		COTE D'IVOIRE: ABIDJAN
	Country	Deaths	
136	ERITREA	0.0	
	Year Mo Dy		Location Name
1915.0	9.0 23.0		ERITREA: ASMARA
	Country	Deaths	
137	FRENCH GUIANA	0.0	
	Year Mo Dy		Location Name
1774.0	NaN NaN		FRENCH GUIANA: CAYENNE
	Country	Deaths	
138	FRENCH POLYNESIA	0.0	
	Year Mo Dy		Location Name
1848.0	7.0 12.0		FRENCH POLYNESIA: TAHITI
	Country	Deaths	
139	GABON	0.0	
	Year Mo Dy		Location Name
1974.0	9.0 23.0		GABON
	Country	Deaths	
140	INDIAN OCEAN	0.0	
	Year Mo Dy		Location Name
1928.0	3.0 9.0		INDIAN OCEAN: S
	Country	Deaths	
141	SIERRA LEONE	0.0	
	Year Mo Dy		Location Name
1795.0	5.0 20.0		SIERRA LEONE
	Country	Deaths	
142	JORDAN	0.0	
	Year Mo Dy		Location Name
362.0	5.0 24.0		JORDAN: AL-KARAK
	Country	Deaths	
143	KERMADEC ISLANDS (NEW ZEALAND)	0.0	
	Year Mo Dy		Location Name
1986.0	10.0 20.0		KERMADEC ISLANDS: RAOUL
	Country	Deaths	
144	KIRIBATI	0.0	
	Year Mo Dy		Location Name
1905.0	6.0 30.0		KIRIBATI: PHOENIX ISLANDS
	Country	Deaths	
145	MICRONESIA, FED. STATES OF	0.0	
	Year Mo Dy		Location Name
1911.0	8.0 16.0		MICRONESIA, FED. STATES OF: CAROLINE ISLANDS
	Country	Deaths	
146	MONTSERRAT	0.0	
	Year Mo Dy		Location Name
1897.0	4.0 25.0		MONTSERRAT
	Country	Deaths	
147	NEW CALEDONIA	0.0	
	Year Mo Dy		Location Name
1875.0	3.0 28.0		NEW CALEDONIA: LOYALTY ISLANDS: LIFOU ISLAND
	Country	Deaths	
148	NORTH KOREA	0.0	
	Year Mo Dy		Location Name
1668.0	7.0 31.0		NORTH KOREA: YELLOW SEA
	Country	Deaths	
149	NORWAY	0.0	
	Year Mo Dy		Location Name
1819.0	8.0 31.0		NORWAY: RANA REGION: LUROY

```

Country Deaths
150 OMAN 0.0
Year Mo Dy Location Name
1497.0 NaN NaN OMAN: QALHAT
Country Deaths
151 PACIFIC OCEAN 0.0
Year Mo Dy Location Name
1958.0 11.0 4.0 EAST PACIFIC RIDGE
Country Deaths
152 PALAU 0.0
Year Mo Dy Location Name
1914.0 10.0 23.0 MICRONESIA, FED. STATES OF: CAROLINE ISLANDS
Country Deaths
153 SAINT VINCENT AND THE GRENADINES 0.0
Year Mo Dy Location Name
1844.0 8.0 30.0 SAINT VINCENT: KINGSTOWN
Country Deaths
154 SAMOA 0.0
Year Mo Dy Location Name
1917.0 6.0 26.0 SAMOA ISLANDS
Country Deaths
155 SAUDI ARABIA 0.0
Year Mo Dy Location Name
2009.0 5.0 19.0 SAUDI ARABIA: WESTERN
Country Deaths
156 ZAMBIA 0.0
Year Mo Dy Location Name
2017.0 2.0 24.0 ZAMBIA: KAPUTA

```

```

In [37]: #Exercise2
Air_data = pd.read_csv('Baoan_Weather_1998_2022.csv') #读取数据

# 使用 str.split 提取数字,将逗号前后数字分别保存为两列,并将温度数据格式转为FLOAT。
Air_data[['Temperature', 'DOM']] = Air_data['TMP'].str.split(',', expand=True)
Air_data['Temperature'] = Air_data['Temperature'].astype(float)/10
#检查dom数字有哪些
DOM_check = Air_data['DOM'].unique()
Temperature_check = Air_data['Temperature'].unique()
print(DOM_check)
print(Temperature_check)
#排除dom=2和温度=999.9的数据
Air_data2=Air_data.loc[(Air_data['DOM']!='2') & (Air_data['Temperature']!='999.9')]
#日期格式
Air_data2['DATE'] = pd.to_datetime(Air_data2['DATE'])
Air_data2
# 提取年份和月份
Air_data2['Year'] = Air_data2['DATE'].dt.year
Air_data2['Month'] = Air_data2['DATE'].dt.month
# 按年和月分组并计算每个月的平均温度
monthly_avg_temperature = Air_data2.groupby(['Year', 'Month'])['Temperature'].me

monthly_avg_temperature['Temperature'].plot() #绘图

#发现月平均气温在25年内具有周期性。

```

```

C:\Users\15484\AppData\Local\Temp\ipykernel_10096\1423402279.py:2: DtypeWarning:
Columns (4,8,9,10,11,14,15,24,25,27,29,31,34,37,38,40,41,45,49,50) have mixed typ
es. Specify dtype option on import or set low_memory=False.

```

```

Air_data = pd.read_csv('Baoan_Weather_1998_2022.csv') #读取数据

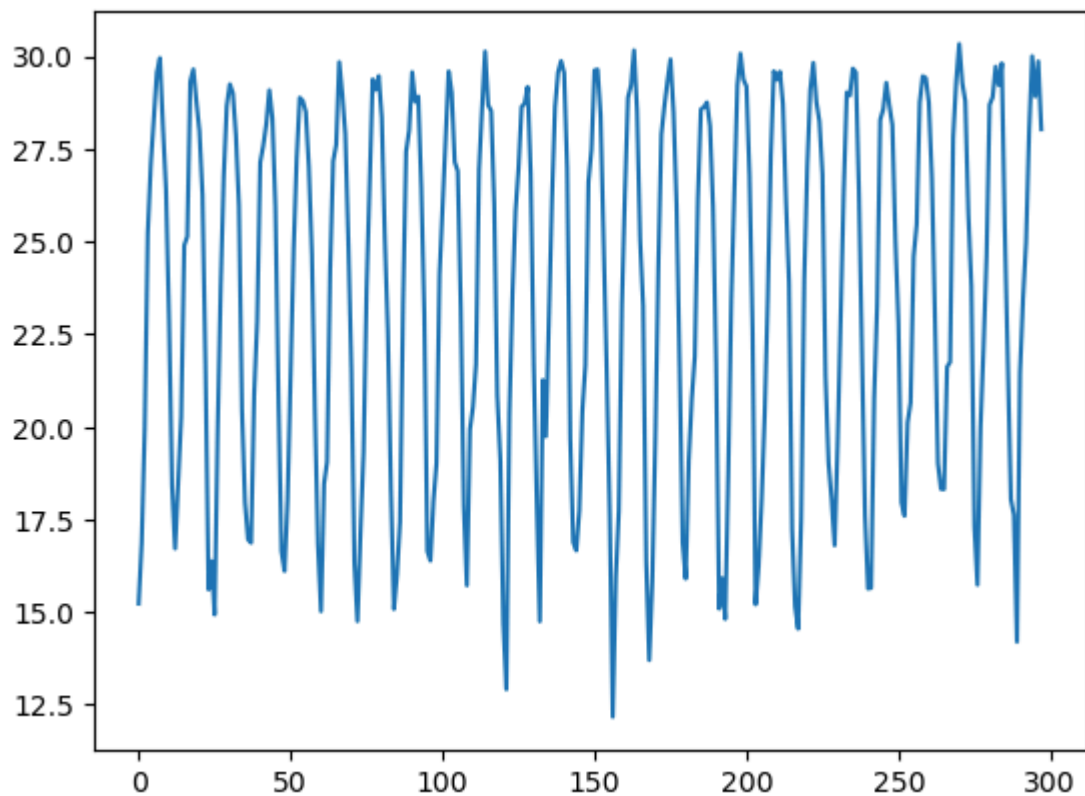
```

```

['1' '9' '2' '5']
[ 18.6  22.  24.  22.1  25.4  25.  25.3  23.  22.2  21.  20.  19.
  18.1  17.5  17.  16.  16.7  20.6  21.2  20.8  19.7  19.5  19.3  19.4
  21.7  22.5  22.3  18.9  18.7  20.1  16.8  15.  14.5  14.  13.  12.9
  12.  12.5  13.7  15.4  16.4  15.7  14.2  18.  19.1  17.4  17.9  18.5
  17.8  26.  29.  24.9  20.3  20.9  20.7  19.8  19.9  25.9  24.1  20.4
  23.6  24.5  21.4  16.2  18.4  21.1  23.7  16.6  13.5  12.1  11.  11.1
  14.9  14.7  12.2  10.  10.3  10.2  10.8  12.6  15.5  15.6  13.8  9.
   8.6  8.  7.9  10.1  8.2  7.8  8.4  9.3  8.3  8.1  9.8  10.7
   8.9  9.4  10.9  11.3  12.3  17.1  11.5  9.5  12.7  8.5  7.6  8.7
   8.8  9.7  9.2  11.7  12.8  13.3  15.1  14.8  13.2  14.6  12.4  14.3
  13.4  13.1  15.9  17.2  23.4  16.1  16.3  17.7  20.5  18.3  18.2  22.7
  11.8  7.  7.4  7.7  7.5  6.8  7.2  10.5  10.4  16.9  18.8  16.5
  21.6  25.5  24.7  23.1  21.9  27.  23.9  22.6  23.3  25.6  20.2  25.7
  25.2  24.8  23.5  22.4  25.8  26.4  19.6  11.9  15.2  999.9  15.8  14.1
  13.6  15.3  17.3  21.3  21.8  23.2  26.1  19.2  28.  27.1  26.8  22.9
  28.7  27.4  24.2  21.5  27.8  24.3  22.8  27.9  28.3  27.6  27.5  24.4
  28.4  26.7  27.7  25.1  29.1  28.8  28.9  27.3  30.2  29.6  29.2  28.1
  26.6  30.  30.5  26.5  26.2  29.5  29.3  32.2  32.  31.1  31.  31.3
  30.4  26.3  26.9  28.2  29.4  29.7  24.6  30.6  29.9  23.8  28.5  30.1
  31.6  31.2  30.8  27.2  29.8  30.9  32.9  28.6  32.4  32.3  32.1  31.4
  31.5  31.7  33.5  31.9  32.5  33.  30.3  31.8  32.6  33.1  33.7  30.7
  33.3  33.6  33.2  33.9  32.7  32.8  34.8  34.  34.2  33.4  34.1  34.6
  34.3  35.5  34.5  35.1  35.8  34.4  33.8  35.  34.7  13.9  11.6  11.4
   9.6  9.9  10.6  17.6  14.4  11.2  34.9  36.8  36.  36.3  35.3  6.
   6.6  5.  6.2  5.4  4.  4.4  3.  3.8  6.4  9.1  6.1  7.1
   5.5  7.3  6.5  6.7  6.9  4.9  4.8  5.6  5.9  35.2  35.4  37.
  37.3  36.9  36.1  6.3  5.1  4.1  3.4  5.7  38.  36.7  36.5  5.8
  35.9  35.6  4.3  3.5  3.6  4.7  3.9  4.6  5.2  35.7  5.3  2.1
   2.2  2.5  1.8  3.7  4.5  3.2  2.8  3.3]

```

Out[37]: <Axes: >



In [391...

```

#Exercise3
df = pd.read_csv('ibtracs.ALL.list.v04r00.csv',
                 usecols=range(15),

```

```

        skiprows=[1],
        parse_dates=['ISO_TIME'],
    )
# 替换空字符串为 NaN, 并转化数据类型为float
df['WMO_WIND'] = df['WMO_WIND'].replace(' ', 0)

df['WMO_WIND'] =df['WMO_WIND'].astype(float)
# 按 SID 分组并找出每个风暴的最大风速和对应的名称
max_wind = df.groupby('SID').agg({'NAME': 'first', 'WMO_WIND': 'max'})
max_wind
# 按最大风速排序, 并获取前十个
top_hurricanes = max_wind.sort_values(by='WMO_WIND', ascending=False).reset_index
top_hurricanes
print(top_hurricanes[['NAME', 'WMO_WIND']])

```

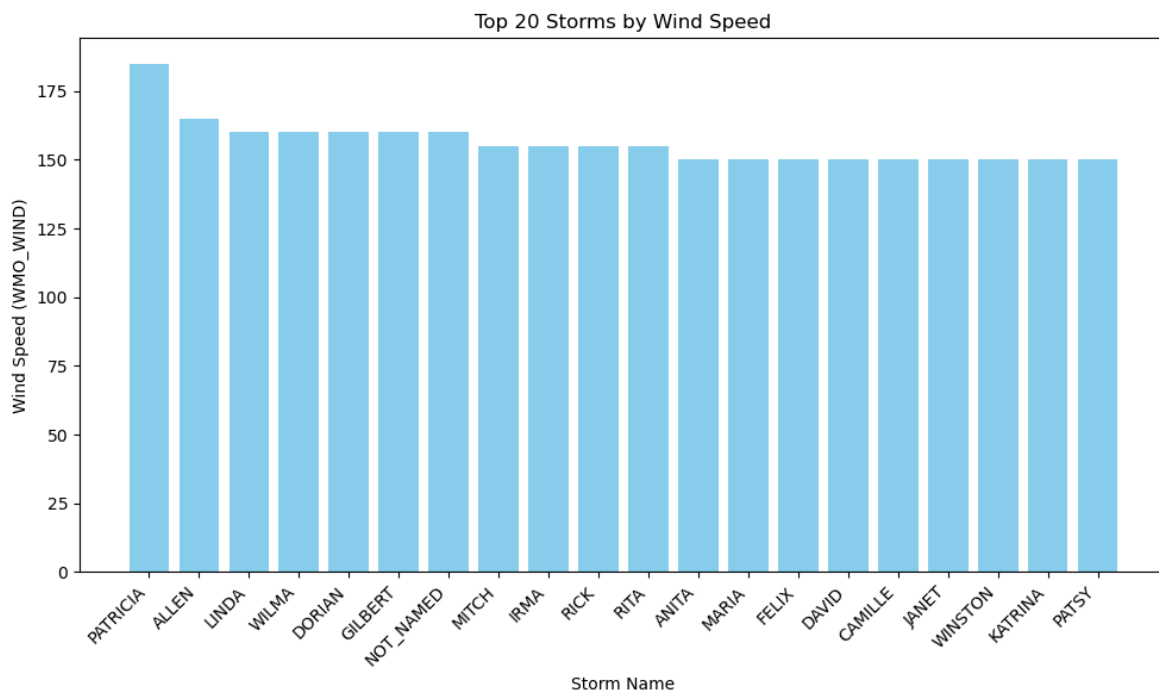
	NAME	WMO_WIND
0	PATRICIA	185.0
1	ALLEN	165.0
2	LINDA	160.0
3	WILMA	160.0
4	DORIAN	160.0
5	GILBERT	160.0
6	NOT_NAMED	160.0
7	MITCH	155.0
8	IRMA	155.0
9	RICK	155.0

In [393...

```

#3.2
strongest_20wind = max_wind.sort_values(by='WMO_WIND', ascending=False).reset_index
# 绘制图形
plt.figure(figsize=(10, 6))
plt.bar(strongest_20wind['NAME'], strongest_20wind['WMO_WIND'], color='skyblue')
plt.xlabel('Storm Name')
plt.ylabel('Wind Speed (WMO_WIND)')
plt.title('Top 20 Storms by Wind Speed')
plt.xticks(rotation=45, ha='right') # 旋转横坐标标签
plt.tight_layout() # 自适应布局
plt.show()

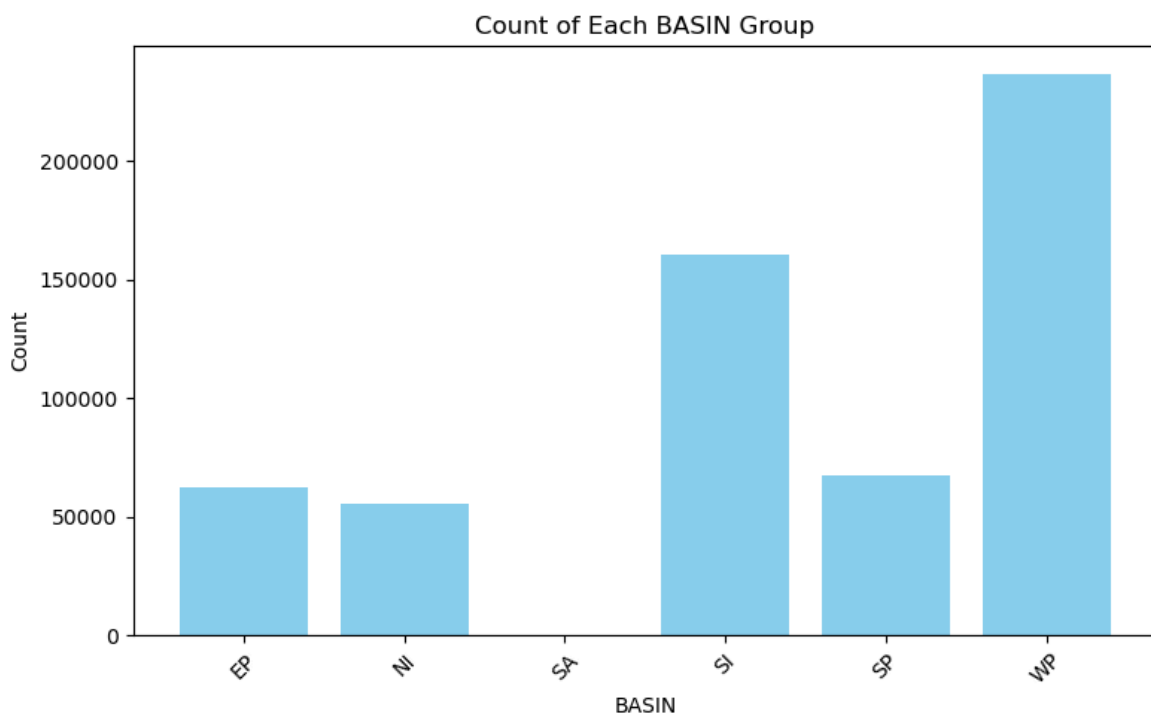
```



In [395...

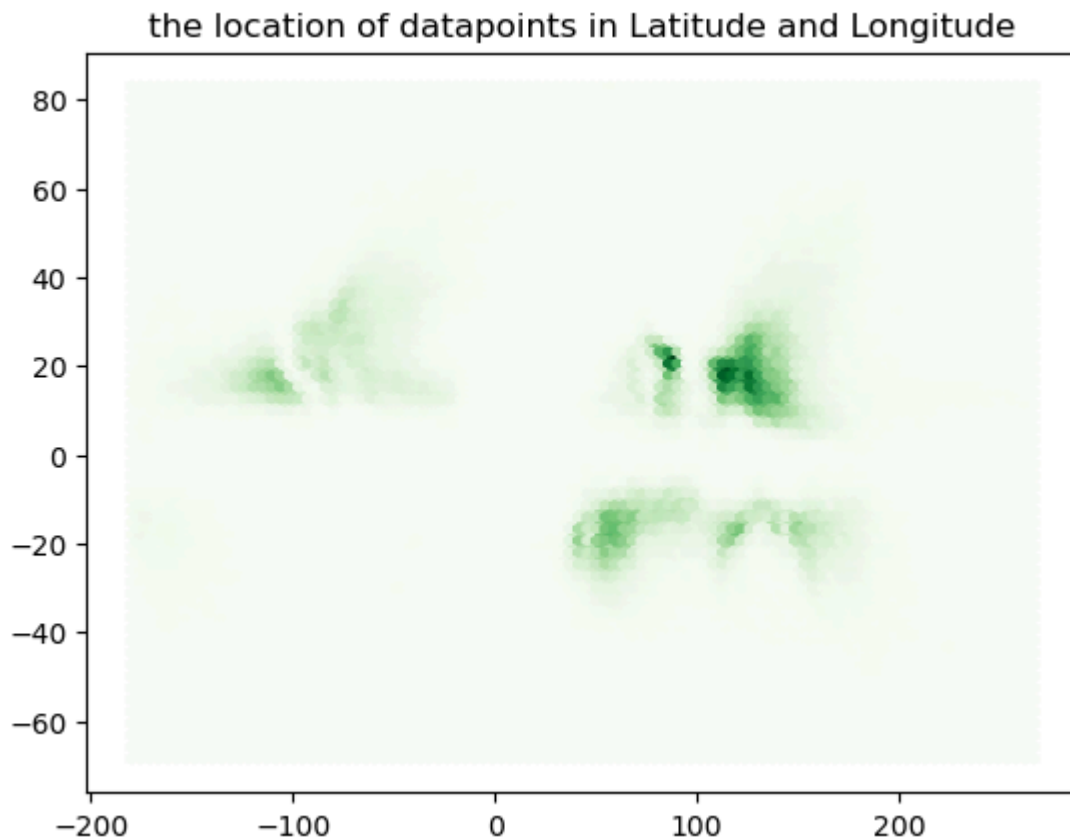
```
#3.3# 按 BASIN 分组并计算每组的个数
BASIN_count=df.groupby('BASIN').size().reset_index()
BASIN_count.columns = ['BASIN', 'Count'] # 重命名列
print(BASIN_count)
# 绘制条形图
plt.figure(figsize=(8, 5))
plt.bar(BASIN_count['BASIN'], BASIN_count['Count'], color='skyblue')
plt.xlabel('BASIN')
plt.ylabel('Count')
plt.title('Count of Each BASIN Group')
plt.xticks(rotation=45) # 旋转横坐标标签
plt.tight_layout() # 自适应布局
plt.show()
```

	BASIN	Count
0	EP	62412
1	NI	55402
2	SA	119
3	SI	160668
4	SP	67119
5	WP	236576



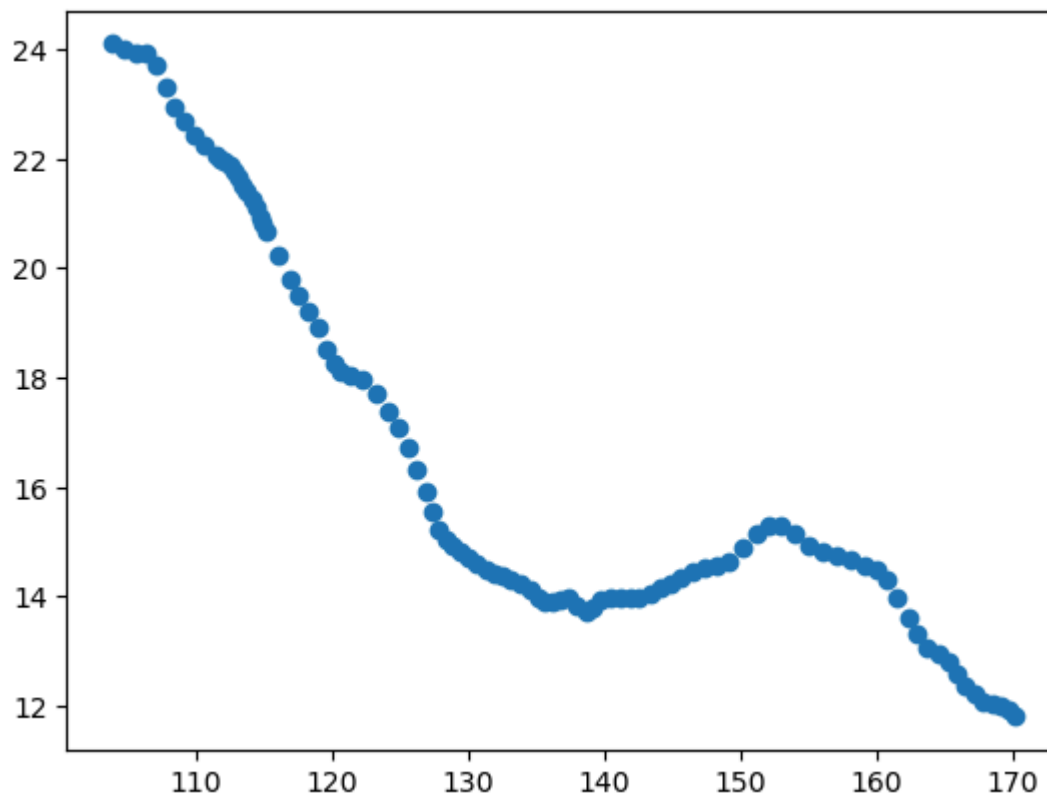
In [397...

```
#3.4
# 按 LAT, LON分组并计算每组的个数
Location_count=df.groupby(['LAT', 'LON']).size().reset_index()
Location_count.columns = ['LAT', 'LON', 'Count'] # 重命名列
Location_count.sort_values('Count', ascending=False).reset_index().head(20) #展示
plt.hexbin(Location_count['LON'], Location_count['LAT'], gridsize = 100, cmap
plt.title('the location of datapoints in Latitude and Longitude')
plt.show()
```

In [399...

```
#3.5
Mangkhutdf_2018=df.loc[(df['NAME']=='MANGKHUT') & (df['SEASON']==2018)][['LON',
x=Mangkhutdf_2018['LON']
y=Mangkhutdf_2018['LAT'] #经纬度坐标
plt.scatter(x, y)         #绘制散点图
plt.show()
```



In [401...

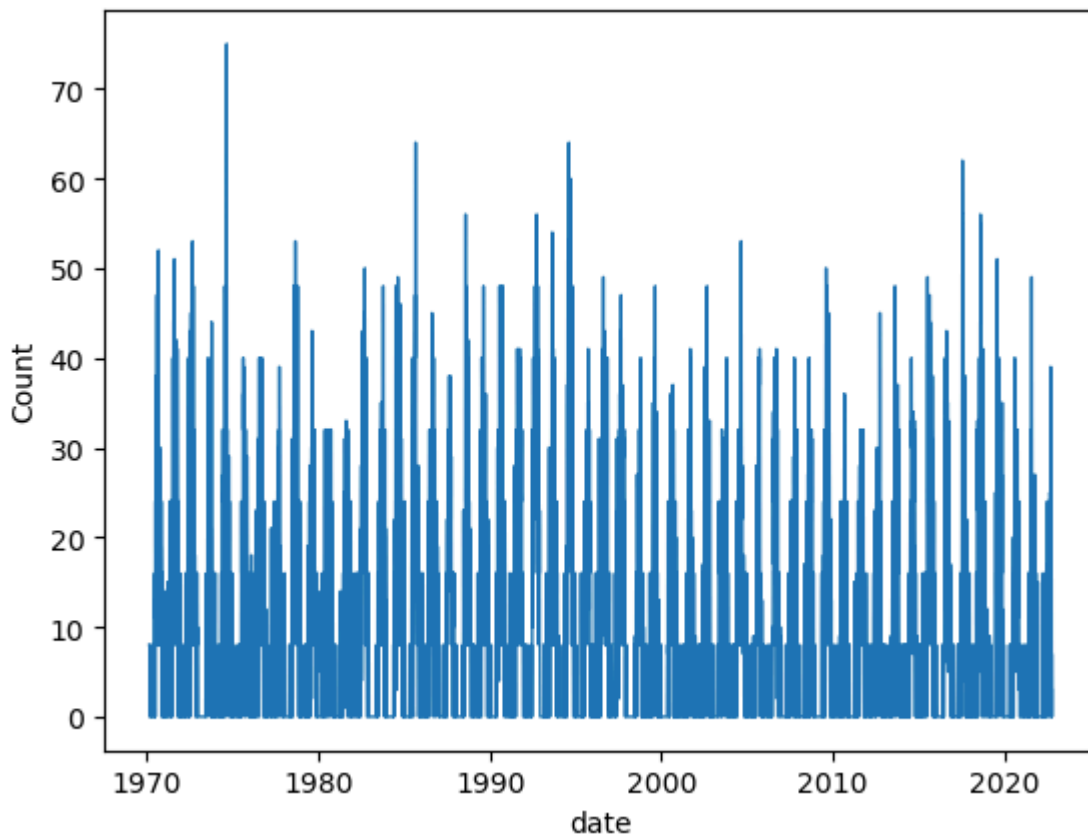
#3.6

```
df_new = df.loc[(df['SEASON']>=1970)].loc[(df['BASIN']=='WP')|(df['BASIN']=='EP
```

In [403...

#3.7

```
datapoint_number=df_new.groupby(pd.Grouper(key='ISO_TIME', axis=0, freq='1D', so
datapoint_number.columns = ['DATE', 'Count'] # 重命名列
plt.plot(datapoint_number['DATE'], datapoint_number['Count'], linewidth=1)#'bo-
plt.xlabel('date')
plt.ylabel('Count')
plt.show()
```



In [405...

#3.8, 不太理解

提取月和日

```
datapoint_number['Day'] = datapoint_number['DATE'].dt.day
```

```
datapoint_number['Month'] = datapoint_number['DATE'].dt.month
```

按月和日分组并计算自从1970年每天的平均数量

```
datapoint_number_clim = datapoint_number.groupby(['Month', 'Day'])['Count'].mean()
```

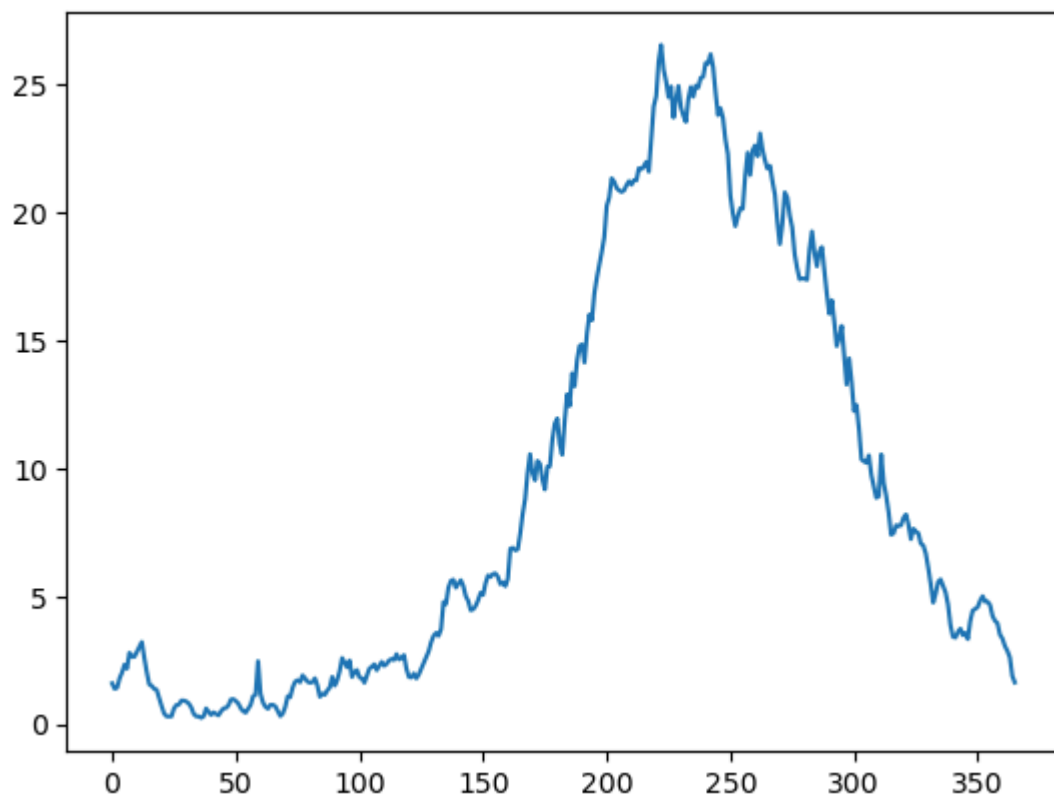
```
datapoint_number_clim['Count'].plot() #绘图
```

```
#不太理解climatology是什么意思。是按照365天每一天进行分组，然后计算这几十年的平均
datapoint_number_clim
```

Out[405...

	Month	Day	Count
0	1	1	1.596154
1	1	2	1.384615
2	1	3	1.423077
3	1	4	1.788462
4	1	5	2.019231
...
361	12	27	3.057692
362	12	28	2.865385
363	12	29	2.615385
364	12	30	1.884615
365	12	31	1.634615

366 rows × 3 columns



In [407...

```
#3.9
#看不懂题目的意思。是用1970年开始每一天减去对应的climatology吗?
group_data = datapoint_number.groupby(['Month', 'Day'])
# Apply mean to grouped data, and then compute the anomaly
datapoint_anom = group_data['Count']-group_data['Count'].mean()
```

```

-----
ValueError                                Traceback (most recent call last)
Cell In[407], line 5
      3 group_data = datapoint_number.groupby(['Month','Day'])
      4 # Apply mean to grouped data, and then compute the anomaly
----> 5 datapoint_anom = group_data['Count']-group_data['Count'].mean()

File D:\Anaconda3\Lib\site-packages\pandas\core\ops\common.py:76, in _unpack_zero
dim_and_defer.<locals>.new_method(self, other)
      72         return NotImplemented
      74 other = item_from_zerodim(other)
--> 76 return method(self, other)

File D:\Anaconda3\Lib\site-packages\pandas\core\arraylike.py:198, in OpsMixin.__r
sub__(self, other)
     196 @unpack_zerodim_and_defer("__rsub__")
     197 def __rsub__(self, other):
--> 198     return self._arith_method(other, roperator.rsub)

File D:\Anaconda3\Lib\site-packages\pandas\core\series.py:6135, in Series._arith
method(self, other, op)
     6133 def _arith_method(self, other, op):
     6134     self, other = self._align_for_op(other)
-> 6135     return base.IndexOpsMixin._arith_method(self, other, op)

File D:\Anaconda3\Lib\site-packages\pandas\core\base.py:1382, in IndexOpsMixin._a
rith_method(self, other, op)
     1379     rvalues = np.arange(rvalues.start, rvalues.stop, rvalues.step)
     1381 with np.errstate(all="ignore"):
-> 1382     result = ops.arithmetic_op(lvalues, rvalues, op)
     1384 return self._construct_result(result, name=res_name)

File D:\Anaconda3\Lib\site-packages\pandas\core\ops\array_ops.py:283, in arithmet
ic_op(left, right, op)
     279     _bool_arith_check(op, left, right) # type: ignore[arg-type]
     281     # error: Argument 1 to "_na_arithmetic_op" has incompatible type
     282     # "Union[ExtensionArray, ndarray[Any, Any]]"; expected "ndarray[Any,
Any]"
--> 283     res_values = _na_arithmetic_op(left, right, op) # type: ignore[arg-t
ype]
     285 return res_values

File D:\Anaconda3\Lib\site-packages\pandas\core\ops\array_ops.py:218, in _na_arit
hmetic_op(left, right, op, is_cmp)
     215     func = partial(expressions.evaluate, op)
     217 try:
--> 218     result = func(left, right)
     219 except TypeError:
     220     if not is_cmp and (
     221         left.dtype == object or getattr(right, "dtype", None) == object
     222     ):
     (...)
```

```

241         # error: "None" not callable
--> 242         return _evaluate(op, op_str, a, b) # type: ignore[misc]
243 return _evaluate_standard(op, op_str, a, b)

File D:\Anaconda3\Lib\site-packages\pandas\core\computation\expressions.py:131, in
_n_evaluate_numexpr(op, op_str, a, b)
    128     _store_test_result(result is not None)
    130 if result is None:
--> 131     result = _evaluate_standard(op, op_str, a, b)
    133 return result

File D:\Anaconda3\Lib\site-packages\pandas\core\computation\expressions.py:73, in
_evaluate_standard(op, op_str, a, b)
    71 if _TEST_MODE:
    72     _store_test_result(False)
---> 73 return op(a, b)

File D:\Anaconda3\Lib\site-packages\pandas\core\roperator.py:15, in rsub(left, ri
ght)
    14 def rsub(left, right):
---> 15     return right - left

ValueError: setting an array element with a sequence. The requested array has an
inhomogeneous shape after 2 dimensions. The detected shape was (366, 2) + inhomog
eneous part.

```

```

In [ ]: #3.10
        #不会做，也看不懂。什么是anomalous hurricane activity呢？跟均值差多少算呢？
        #

```

```

In [409... #Exercise4
#4.1
#获取数据from National Centers for Environmental Information (NCEI) 的UNIVERSITY
data = pd.read_csv('USW00024089.csv') #读取csv文件
#关注PRCP这一列。似乎是降水？
data = data.dropna(subset=['PRCP']) #删除PRCP列中含空值的行
data['DATE'] = pd.to_datetime(data['DATE'], errors='coerce') # 将日期列转换为日
data.set_index('DATE', inplace=True) #将日期设置为索引

data

```

Out[409...

	STATION	LATITUDE	LONGITUDE	ELEVATION	NAME	ADPT	ADP'
DATE							
1948-08-01	USW00024089	42.89778	-106.47361	1621.1	CASPER NATRONA CO AIRPORT, WY US	NaN	
1948-09-01	USW00024089	42.89778	-106.47361	1621.1	CASPER NATRONA CO AIRPORT, WY US	NaN	
1948-10-01	USW00024089	42.89778	-106.47361	1621.1	CASPER NATRONA CO AIRPORT, WY US	NaN	
1948-11-01	USW00024089	42.89778	-106.47361	1621.1	CASPER NATRONA CO AIRPORT, WY US	NaN	
1948-12-01	USW00024089	42.89778	-106.47361	1621.1	CASPER NATRONA CO AIRPORT, WY US	NaN	
...	
2024-05-01	USW00024089	42.89778	-106.47361	1621.1	CASPER NATRONA CO AIRPORT, WY US	0.78	
2024-06-01	USW00024089	42.89778	-106.47361	1621.1	CASPER NATRONA CO AIRPORT, WY US	5.56	
2024-07-01	USW00024089	42.89778	-106.47361	1621.1	CASPER NATRONA CO AIRPORT, WY US	5.02	
2024-08-01	USW00024089	42.89778	-106.47361	1621.1	CASPER NATRONA CO AIRPORT, WY US	NaN	
2024-09-01	USW00024089	42.89778	-106.47361	1621.1	CASPER NATRONA CO AIRPORT, WY US	NaN	

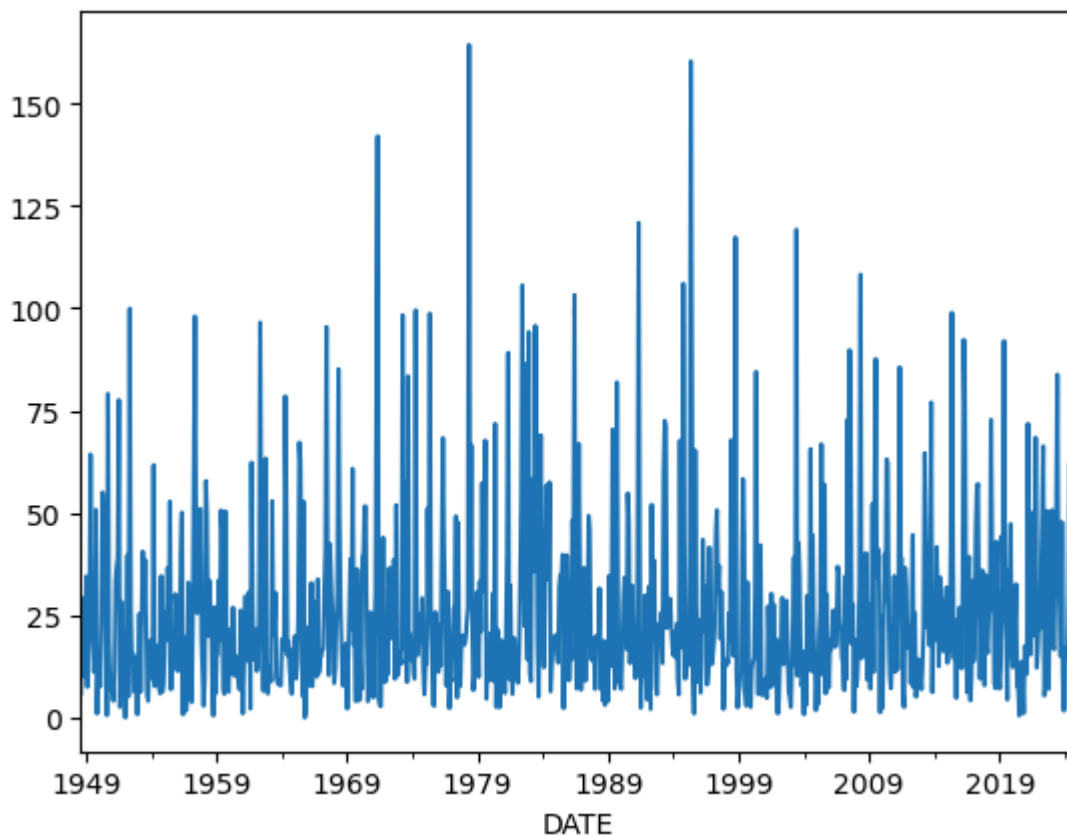
914 rows × 139 columns

In [411...

```
#4.2
data['PRCP'].plot()
```

Out[411...

```
<Axes: xlabel='DATE'>
```



In [413...

```
#4.3
#1.按月分组，计算每个月的这几十年的均值
data['Month'] = data.index.month
month_clim=data.groupby('Month')['PRCP'].mean()
month_clim.columns = ['Month','PRCP'] #修改列名，报错了，不知道为啥
print(month_clim)

month_clim['PRCP'].plot() #画图也画不出来
#发现一年中间的月份降水多，两边的月份降水少
```

Month

```
1    13.781579
2    14.717105
3    24.188158
4    36.693421
5    53.485526
6    34.961842
7    29.593421
8    18.276623
9    23.197403
10   26.826316
11   17.607895
12   15.207895
```

```
Name: PRCP, dtype: float64
```

```
C:\Users\15484\AppData\Local\Temp\ipykernel_15636\3401064726.py:3: PerformanceWarning: DataFrame is highly fragmented.  This is usually the result of calling `frame.insert` many times, which has poor performance.  Consider joining all columns at once using `pd.concat(axis=1)` instead. To get a de-fragmented frame, use `newframe = frame.copy()`  
    data['Month'] = data.index.month
```



```

-----
KeyError                                Traceback (most recent call last)
File D:\Anaconda3\Lib\site-packages\pandas\core\indexes\base.py:3805, in Index.get_loc(self, key)
    3804 try:
-> 3805     return self._engine.get_loc(casted_key)
    3806 except KeyError as err:

File index.pyx:167, in pandas._libs.index.IndexEngine.get_loc()

File index.pyx:175, in pandas._libs.index.IndexEngine.get_loc()

File pandas\_libs\index_class_helper.pxi:100, in pandas._libs.index.Int32Engine._check_type()

```

KeyError: 'PRCP'

The above exception was the direct cause of the following exception:

```

KeyError                                Traceback (most recent call last)
Cell In[413], line 8
      5 month_clim.columns = ['Month', 'PRCP']          #修改列名，报错了，不知道为啥
      6 print(month_clim)
----> 8 month_clim['PRCP'].plot()

File D:\Anaconda3\Lib\site-packages\pandas\core\series.py:1121, in Series._getitem__(self, key)
    1118     return self._values[key]
    1120 elif key_is_scalar:
-> 1121     return self._get_value(key)
    1123 # Convert generator to list before going through hashable part
    1124 # (We will iterate through the generator there to check for slices)
    1125 if is_iterator(key):

File D:\Anaconda3\Lib\site-packages\pandas\core\series.py:1237, in Series._get_value(self, label, takeable)
    1234     return self._values[label]
    1236 # Similar to Index.get_value, but we do not fall back to positional
-> 1237 loc = self.index.get_loc(label)
    1239 if is_integer(loc):
    1240     return self._values[loc]

File D:\Anaconda3\Lib\site-packages\pandas\core\indexes\base.py:3812, in Index.get_loc(self, key)
    3807     if isinstance(casted_key, slice) or (
    3808         isinstance(casted_key, abc.Iterable)
    3809         and any(isinstance(x, slice) for x in casted_key)
    3810     ):
    3811         raise InvalidIndexError(key)
-> 3812     raise KeyError(key) from err
    3813 except TypeError:
    3814     # If we have a listlike key, _check_indexing_error will raise
    3815     # InvalidIndexError. Otherwise we fall through and re-raise
    3816     # the TypeError.
    3817     self._check_indexing_error(key)

```

KeyError: 'PRCP'

```
In [415... #2. 计算每一年的均值
data['Year'] = data.index.year
year_ave=data.groupby('Year')['PRCP'].mean()
year_ave
#发现有的年降水多，有的年少
```

C:\Users\15484\AppData\Local\Temp\ipykernel_15636\2019269790.py:2: PerformanceWarning: DataFrame is highly fragmented. This is usually the result of calling `frame.insert` many times, which has poor performance. Consider joining all columns at once using `pd.concat(axis=1)` instead. To get a de-fragmented frame, use `newframe = frame.copy()`

```
data['Year'] = data.index.year
```

```
Out[415... Year
1948    17.040000
1949    23.075000
1950    27.016667
1951    23.216667
1952    21.208333
...
2020    12.325000
2021    30.791667
2022    28.350000
2023    32.608333
2024    20.455556
Name: PRCP, Length: 77, dtype: float64
```

```
In [417... # 3. 描述性统计
data_describe = data['PRCP'].describe()
print(data_describe)
```

```
count    914.000000
mean      25.700547
std       22.155365
min        0.000000
25%       10.825000
50%       19.200000
75%       33.000000
max       164.300000
Name: PRCP, dtype: float64
```

```
In [419... #4. 每一年的PRCP的标准差和方差
std_dev = data.groupby('Year')['PRCP'].std()
variance = data.groupby('Year')['PRCP'].var()
print(f"标准差: {std_dev}, 方差: {variance}")
```

标准差: Year

1948	9.328076
1949	18.309418
1950	23.809312
1951	21.747386
1952	27.173231

...

2020	10.025887
2021	22.356632
2022	19.446594
2023	23.083306
2024	17.226949

Name: PRCP, Length: 77, dtype: float64, 方差: Year

1948	87.013000
1949	335.234773
1950	566.883333
1951	472.948788
1952	738.384470

...

2020	100.518409
2021	499.819015
2022	378.170000
2023	532.839015
2024	296.767778

Name: PRCP, Length: 77, dtype: float64

In [421...

```
#5 按年分布的偏度和峰度
skewness = data.groupby('Year')['PRCP'].skew()
print(f"偏度: {skewness}")
```

偏度: Year

1948	0.101182
1949	1.330050
1950	1.081320
1951	1.473459
1952	2.537936

...

2020	0.667700
2021	1.009079
2022	0.673240
2023	0.856451
2024	2.100206

Name: PRCP, Length: 77, dtype: float64