

```
In [7]: import pandas as pd
import numpy as np
import seaborn as sns
import matplotlib.pyplot as plt
```

```
In [8]: data=pd.read_csv("country_wise_latest.csv")
data
```

Out[8]:

	Country/Region	Confirmed	Deaths	Recovered	Active	New cases	New deaths	New recovered	Deaths / 100 Cases	Recovered / 100 Cases	Deaths / 100 Recovered	Confirmed last week	1 week change	1 week % increase	WHO Region
0	Afghanistan	36263	1269	25198	9796	106	10	18	3.50	69.49	5.04	35526	737	2.07	Eastern Mediterranean
1	Albania	4880	144	2745	1991	117	6	63	2.95	56.25	5.25	4171	709	17.00	Europe
2	Algeria	27973	1163	18837	7973	616	8	749	4.16	67.34	6.17	23691	4282	18.07	Africa
3	Andorra	907	52	803	52	10	0	0	5.73	88.53	6.48	884	23	2.60	Europe
4	Angola	950	41	242	667	18	1	0	4.32	25.47	16.94	749	201	26.84	Africa
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
182	West Bank and Gaza	10621	78	3752	6791	152	2	0	0.73	35.33	2.08	8916	1705	19.12	Eastern Mediterranean
183	Western Sahara	10	1	8	1	0	0	0	10.00	80.00	12.50	10	0	0.00	Africa
184	Yemen	1691	483	833	375	10	4	36	28.56	49.26	57.98	1619	72	4.45	Eastern Mediterranean
185	Zambia	4552	140	2815	1597	71	1	465	3.08	61.84	4.97	3326	1226	36.86	Africa
186	Zimbabwe	2704	36	542	2126	192	2	24	1.33	20.04	6.64	1713	991	57.85	Africa

187 rows × 15 columns

```
In [9]: data.head()
```

Out[9]:

	Country/Region	Confirmed	Deaths	Recovered	Active	New cases	New deaths	New recovered	Deaths / 100 Cases	Recovered / 100 Cases	Deaths / 100 Recovered	Confirmed last week	1 week change	1 week % increase	WHO Region
0	Afghanistan	36263	1269	25198	9796	106	10	18	3.50	69.49	5.04	35526	737	2.07	Eastern Mediterranean
1	Albania	4880	144	2745	1991	117	6	63	2.95	56.25	5.25	4171	709	17.00	Europe
2	Algeria	27973	1163	18837	7973	616	8	749	4.16	67.34	6.17	23691	4282	18.07	Africa
3	Andorra	907	52	803	52	10	0	0	5.73	88.53	6.48	884	23	2.60	Europe
4	Angola	950	41	242	667	18	1	0	4.32	25.47	16.94	749	201	26.84	Africa

```
In [10]: data.shape
```

Out[10]: (187, 15)

```
In [11]: data.columns
```

Out[11]: Index(['Country/Region', 'Confirmed', 'Deaths', 'Recovered', 'Active', 'New cases', 'New deaths', 'New recovered', 'Deaths / 100 Cases', 'Recovered / 100 Cases', 'Deaths / 100 Recovered', 'Confirmed last week', '1 week change', '1 week % increase', 'WHO Region'], dtype='object')

```
In [12]: data.describe
```

Out[12]: <bound method NDFrame.describe of Country/Region Confirmed Deaths Recovered Active New cases \
0 Afghanistan 36263 1269 25198 9796 106
1 Albania 4880 144 2745 1991 117
2 Algeria 27973 1163 18837 7973 616
3 Andorra 907 52 803 52 10
4 Angola 950 41 242 667 18
... ... ... ... ...
182 West Bank and Gaza 10621 78 3752 6791 152
183 Western Sahara 10 1 8 1 0
184 Yemen 1691 483 833 375 10
185 Zambia 4552 140 2815 1597 71
186 Zimbabwe 2704 36 542 2126 192

New deaths New recovered Deaths / 100 Cases Recovered / 100 Cases \
0 10 18 3.50 69.49
1 6 63 2.95 56.25
2 8 749 4.16 67.34
3 0 0 5.73 88.53
4 1 0 4.32 25.47
.. ... ... ...
182 2 0 0.73 35.33
183 0 0 10.00 80.00
184 4 36 28.56 49.26
185 1 465 3.08 61.84
186 2 24 1.33 20.04

Deaths / 100 Recovered Confirmed last week 1 week change \
0 5.04 35526 737
1 5.25 4171 709
2 6.17 23691 4282
3 6.48 884 23
4 16.94 749 201
.. ... ...
182 2.08 8916 1705
183 12.50 10 0
184 57.98 1619 72
185 4.97 3326 1226
186 6.64 1713 991

1 week % increase WHO Region
0 2.07 Eastern Mediterranean
1 17.00 Europe
2 18.07 Africa
3 2.60 Europe
4 26.84 Africa
.. ...
182 19.12 Eastern Mediterranean
183 0.00 Africa
184 4.45 Eastern Mediterranean
185 36.86 Africa
186 57.85 Africa

[187 rows x 15 columns]>

```
In [13]: cols_to_drop = ["1 week change", "1 week % increase"]
existing_cols = [col for col in cols_to_drop if col in data.columns]
data.drop(existing_cols, axis=1, inplace=True)
data
```

Out[13]:

	Country/Region	Confirmed	Deaths	Recovered	Active	New cases	New deaths	New recovered	Deaths / 100 Cases	Recovered / 100 Cases	Deaths / 100 Recovered	Confirmed last week	WHO Region
0	Afghanistan	36263	1269	25198	9796	106	10	18	3.50	69.49	5.04	35526	Eastern Mediterranean
1	Albania	4880	144	2745	1991	117	6	63	2.95	56.25	5.25	4171	Europe
2	Algeria	27973	1163	18837	7973	616	8	749	4.16	67.34	6.17	23691	Africa
3	Andorra	907	52	803	52	10	0	0	5.73	88.53	6.48	884	Europe
4	Angola	950	41	242	667	18	1	0	4.32	25.47	16.94	749	Africa
...	...	...	...	...	...	...	...	...	...	...	...	...	...
182	West Bank and Gaza	10621	78	3752	6791	152	2	0	0.73	35.33	2.08	8916	Eastern Mediterranean
183	Western Sahara	10	1	8	1	0	0	0	10.00	80.00	12.50	10	Africa
184	Yemen	1691	483	833	375	10	4	36	28.56	49.26	57.98	1619	Eastern Mediterranean
185	Zambia	4552	140	2815	1597	71	1	465	3.08	61.84	4.97	3326	Africa
186	Zimbabwe	2704	36	542	2126	192	2	24	1.33	20.04	6.64	1713	Africa

187 rows × 13 columns

In [14]:

```
corona = data.groupby("Country/Region").sum()
corona.head()
```

Out[14]:

	Confirmed	Deaths	Recovered	Active	New cases	New deaths	New recovered	Deaths / 100 Cases	Recovered / 100 Cases	Deaths / 100 Recovered	Confirmed last week	WHO Region
Country/Region												
Afghanistan	36263	1269	25198	9796	106	10	18	3.50	69.49	5.04	35526	Eastern Mediterranean
Albania	4880	144	2745	1991	117	6	63	2.95	56.25	5.25	4171	Europe
Algeria	27973	1163	18837	7973	616	8	749	4.16	67.34	6.17	23691	Africa
Andorra	907	52	803	52	10	0	0	5.73	88.53	6.48	884	Europe
Angola	950	41	242	667	18	1	0	4.32	25.47	16.94	749	Africa

corona.loc["India"] corona

In [15]:

```
corona.index
```

Out[15]:

```
Index(['Afghanistan', 'Albania', 'Algeria', 'Andorra', 'Angola',
      'Antigua and Barbuda', 'Argentina', 'Armenia', 'Australia', 'Austria',
      ...,
      'United Kingdom', 'Uruguay', 'Uzbekistan', 'Venezuela', 'Vietnam',
      'West Bank and Gaza', 'Western Sahara', 'Yemen', 'Zambia', 'Zimbabwe'],
      dtype='object', name='Country/Region', length=187)
```

In [16]:

```
corona2 = corona.apply(pd.to_numeric, errors='coerce')

corona2
```

Out[16]:

	Confirmed	Deaths	Recovered	Active	New cases	New deaths	New recovered	Deaths / 100 Cases	Recovered / 100 Cases	Deaths / 100 Recovered	Confirmed last week	WHO Region
Country/Region												
Afghanistan	36263	1269	25198	9796	106	10	18	3.50	69.49	5.04	35526	NaN
Albania	4880	144	2745	1991	117	6	63	2.95	56.25	5.25	4171	NaN
Algeria	27973	1163	18837	7973	616	8	749	4.16	67.34	6.17	23691	NaN
Andorra	907	52	803	52	10	0	0	5.73	88.53	6.48	884	NaN
Angola	950	41	242	667	18	1	0	4.32	25.47	16.94	749	NaN
...	...	...	...	...	...	...	...	...	...	...	...	...
West Bank and Gaza	10621	78	3752	6791	152	2	0	0.73	35.33	2.08	8916	NaN
Western Sahara	10	1	8	1	0	0	0	10.00	80.00	12.50	10	NaN
Yemen	1691	483	833	375	10	4	36	28.56	49.26	57.98	1619	NaN
Zambia	4552	140	2815	1597	71	1	465	3.08	61.84	4.97	3326	NaN
Zimbabwe	2704	36	542	2126	192	2	24	1.33	20.04	6.64	1713	NaN

187 rows × 12 columns

In [17]:

```
corona.loc["India"].dtype
```

Out[17]:

```
dtype('O')
```

In [18]:

```
corona.columns
```

Out[18]:

```
Index(['Confirmed', 'Deaths', 'Recovered', 'Active', 'New cases', 'New deaths',
      'New recovered', 'Deaths / 100 Cases', 'Recovered / 100 Cases',
      'Deaths / 100 Recovered', 'Confirmed last week', 'WHO Region'],
      dtype='object')
```

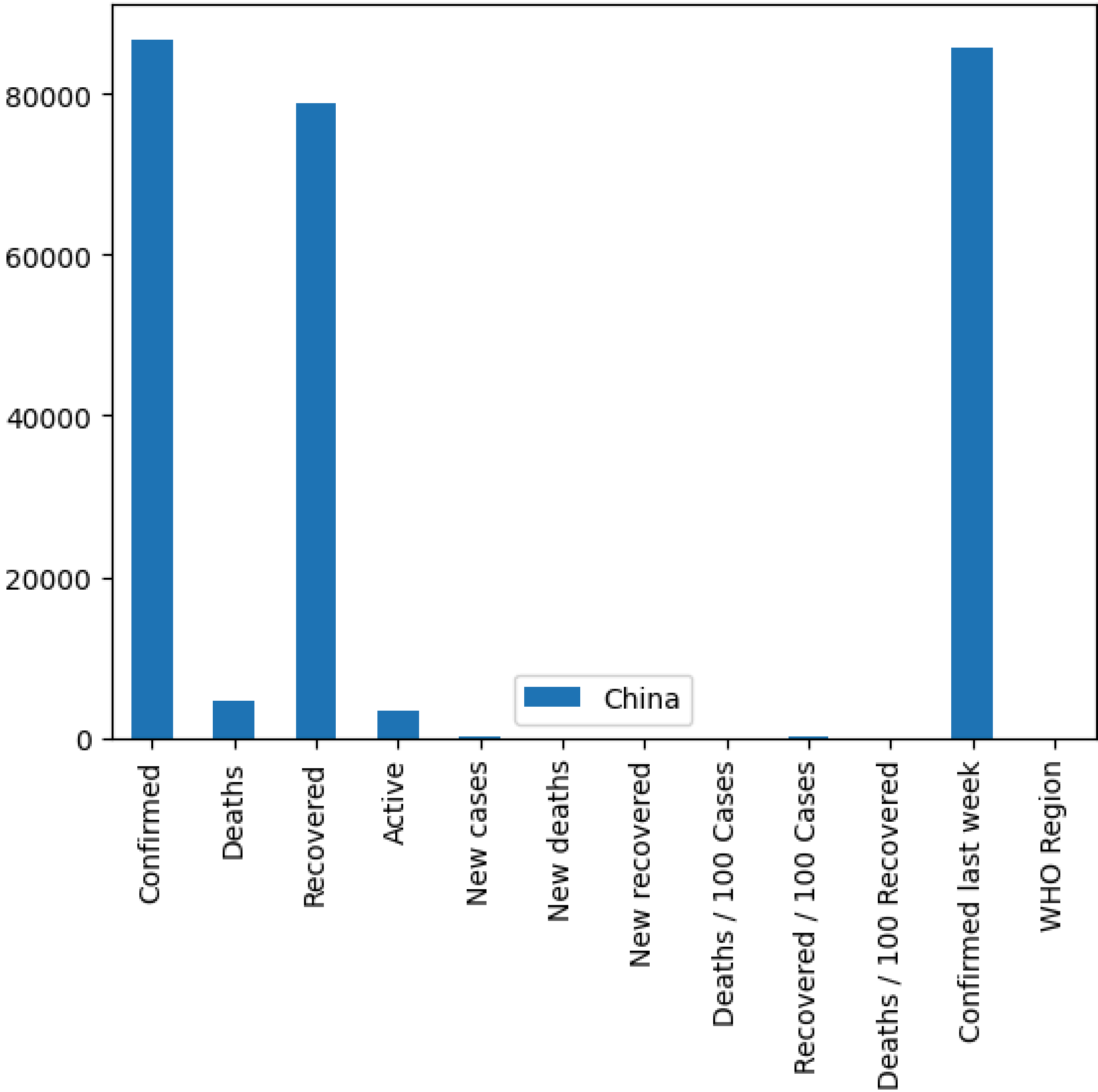
In [19]:

```
corona2.loc['China'].plot(kind="bar")
plt.xticks(rotation=90)
plt.legend()
```

Out[19]:

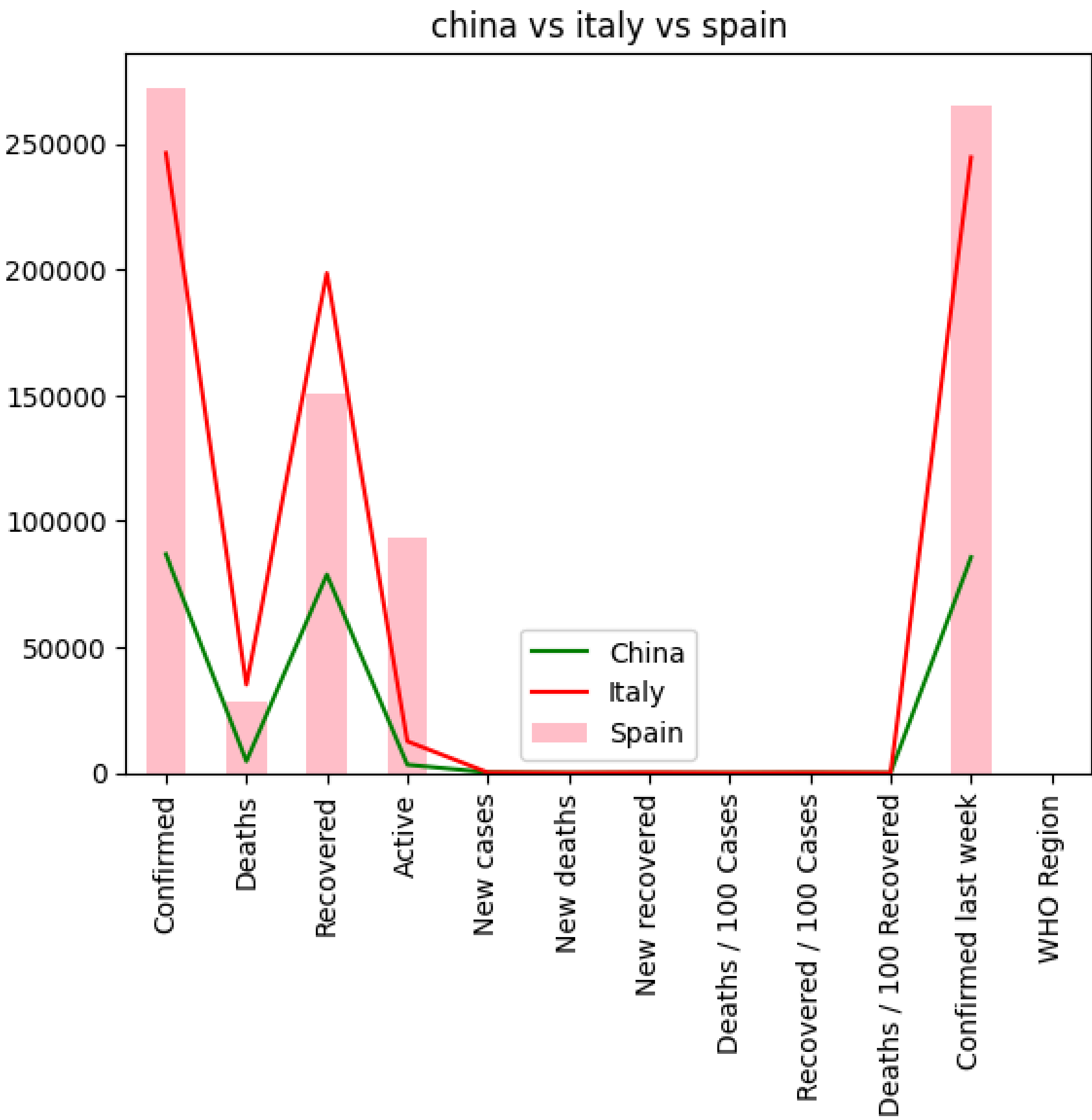
```
<matplotlib.legend.Legend at 0x1a565ed7a50>
```





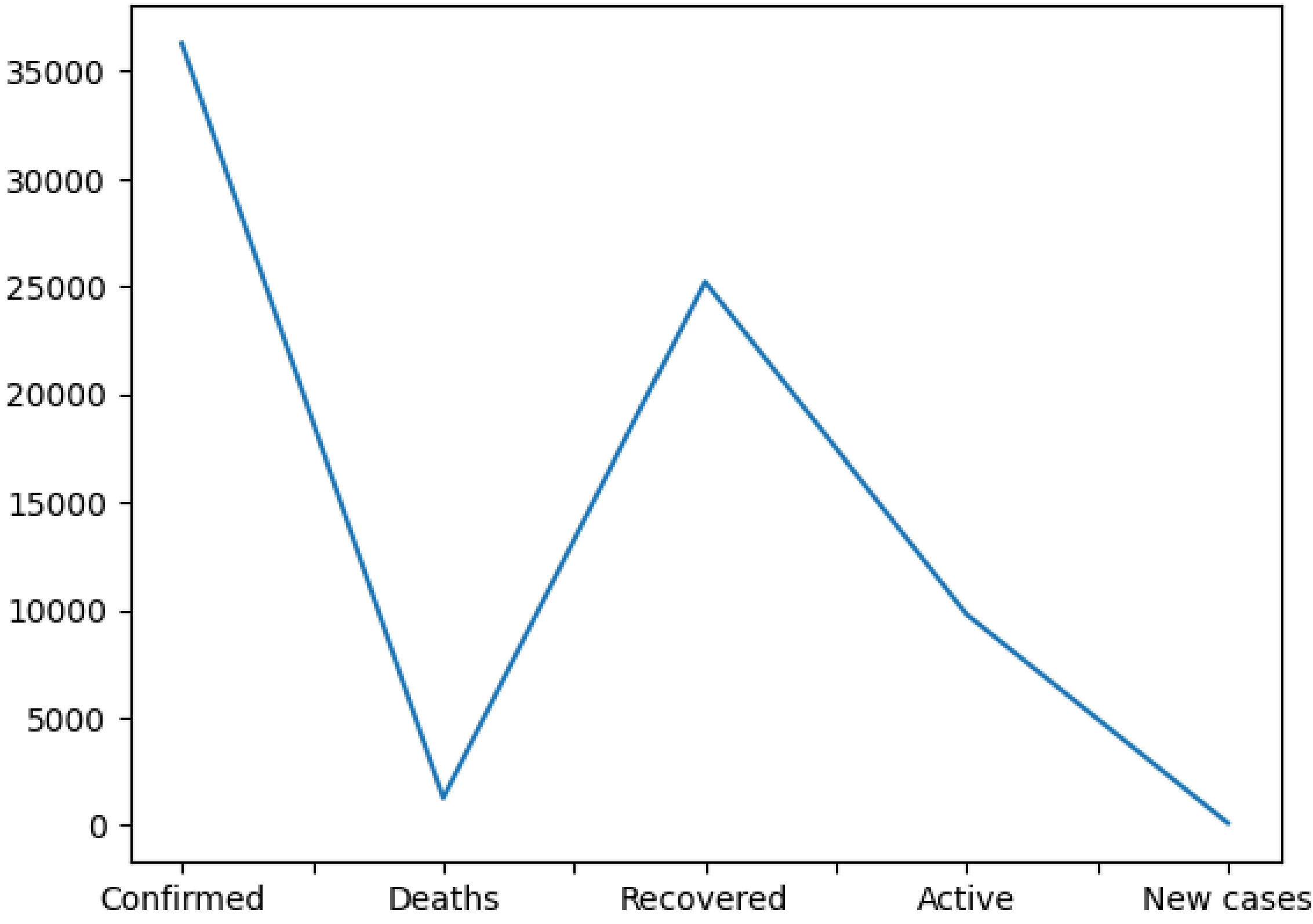
```
In [20]: corona2.loc['China'].plot(c='green')
corona2.loc['Italy'].plot(c='red')
corona2.loc['Spain'].plot(kind='bar',color='pink')
plt.legend()
plt.title("china vs italy vs spain")
```

Out[20]: Text(0.5, 1.0, 'china vs italy vs spain')

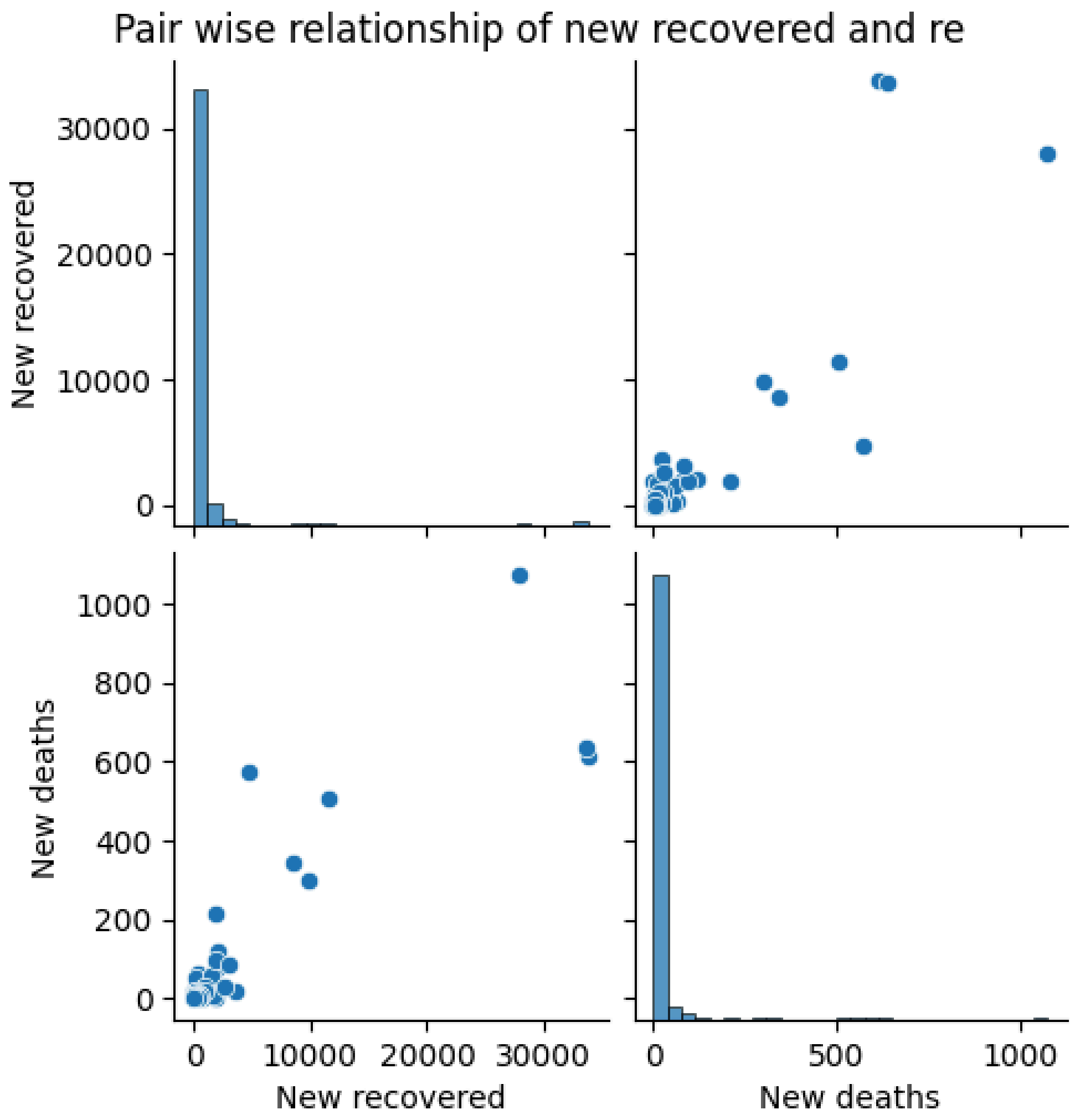


```
In [21]: #Spread of the virus in China for the first 3 dates only
corona2.loc['Afghanistan'][:5].plot()
```

Out[21]: <Axes: >



```
In [22]: sns.pairplot(data[['New recovered','New deaths']])
plt.suptitle("Pair wise relationship of new recovered and re",y=1.02)
plt.show()
```



```
In [23]: corona2.columns.tolist()
```

```
Out[23]: ['Confirmed',
'Deaths',
'Recovered',
'Active',
'New cases',
'New deaths',
'New recovered',
'Deaths / 100 Cases',
'Recovered / 100 Cases',
'Deaths / 100 Recovered',
'Confirmed last week',
'WHO Region']
```

```
In [24]: corona2.columns = corona2.columns.str.strip() # removes spaces around names
```

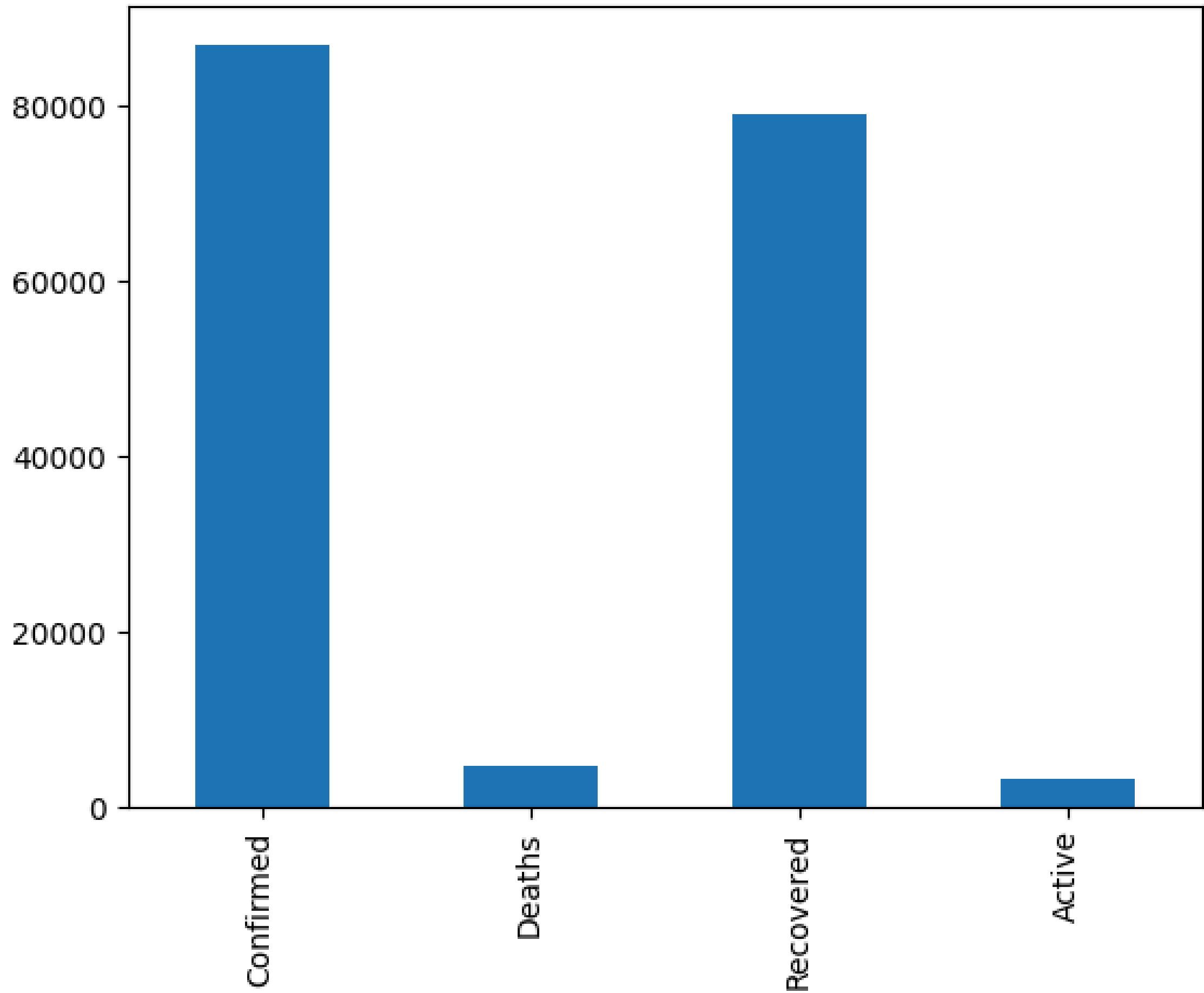
```
In [27]: # Select China row, then plot the Recovered trend
print(corona2.head())
print(corona2.index)
print(corona2.columns)
corona2.loc['China', ['Confirmed', 'Deaths', 'Recovered', 'Active']].plot(kind='bar')
```

	Confirmed	Deaths	Recovered	Active	New cases	New deaths	\
Country/Region							
Afghanistan	36263	1269	25198	9796	106	10	
Albania	4880	144	2745	1991	117	6	
Algeria	27973	1163	18837	7973	616	8	
Andorra	907	52	803	52	10	0	
Angola	950	41	242	667	18	1	

	New recovered	Deaths / 100 Cases	Recovered / 100 Cases	\
Country/Region				
Afghanistan	18	3.50	69.49	
Albania	63	2.95	56.25	
Algeria	749	4.16	67.34	
Andorra	0	5.73	88.53	
Angola	0	4.32	25.47	

	Deaths / 100 Recovered	Confirmed last week	WHO Region
Country/Region			
Afghanistan	5.04	35526	NaN
Albania	5.25	4171	NaN
Algeria	6.17	23691	NaN
Andorra	6.48	884	NaN
Angola	16.94	749	NaN
Index(['Afghanistan', 'Albania', 'Algeria', 'Andorra', 'Angola', 'Antigua and Barbuda', 'Argentina', 'Armenia', 'Australia', 'Austria', ... 'United Kingdom', 'Uruguay', 'Uzbekistan', 'Venezuela', 'Vietnam', 'West Bank and Gaza', 'Western Sahara', 'Yemen', 'Zambia', 'Zimbabwe'], dtype='object', name='Country/Region', length=187)			
Index(['Confirmed', 'Deaths', 'Recovered', 'Active', 'New cases', 'New deaths', 'New recovered', 'Deaths / 100 Cases', 'Recovered / 100 Cases', 'Deaths / 100 Recovered', 'Confirmed last week', 'WHO Region'], dtype='object')			

```
Out[27]: <Axes: >
```



```
In [31]: corona2['Recovered'].sort_values(ascending=False).head(10).plot(
kind='bar',
title='Top 10 Countries by Recovered Cases'
)
```

```
Out[31]: <Axes: title={'center': 'Top 10 Countries by Recovered Cases'}, xlabel='Country/Region'>
```

