In [93]:
```
%matplotlib inline
```

In [94]:
```
import matplotlib as mpl
import matplotlib.pyplot as plt
import numpy as np
import pandas as pd
```

In [95]:
```
data = pd.read_csv('districts.csv')
```

In [96]:
```
# Q.1-describe statistics of all columns
```

In [97]:
```
data.describe()
```

Out[97]:

	districtData/0/active	districtData/0/confirmed	districtData/0/deceased	districtData/0/recovered
count	33.000000	33.000000	33.000000	33.000000
mean	249.818182	317.909091	13.878788	54.212121
std	994.971936	1238.750034	51.887955	193.105016
min	0.000000	1.000000	0.000000	0.000000
25%	2.000000	3.000000	0.000000	1.000000
50%	14.000000	25.000000	1.000000	5.000000
75%	69.000000	79.000000	4.000000	22.000000
max	5679.000000	7061.000000	290.000000	1092.000000

In [98]:
```
data.head(15)
```

Out[98]:

	districtData/0/district	districtData/0/active	districtData/0/confirmed	districtData/0/deceased	districtData/0/recovered
0	Ahmadnagar	17	42	2	23
1	Yavatmal	69	79	0	10
2	Washim	1	2	0	1
3	Solapur	93	99	6	0
4	Sindhudurg	1	2	0	1
5	Satara	21	32	2	9
6	Sangli	3	29	1	25
7	Ratnagiri	2	8	1	5
8	Raigarh	44	71	3	24
9	Parbhani	1	2	0	1
10	Palghar	119	169	4	46
11	Osmanabad	0	3	0	3
12	Nashik	179	197	12	6
13	Nandurbar	10	11	1	0
14	Nanded	3	3	0	0

In [99]:
```
data.tail(10)
```

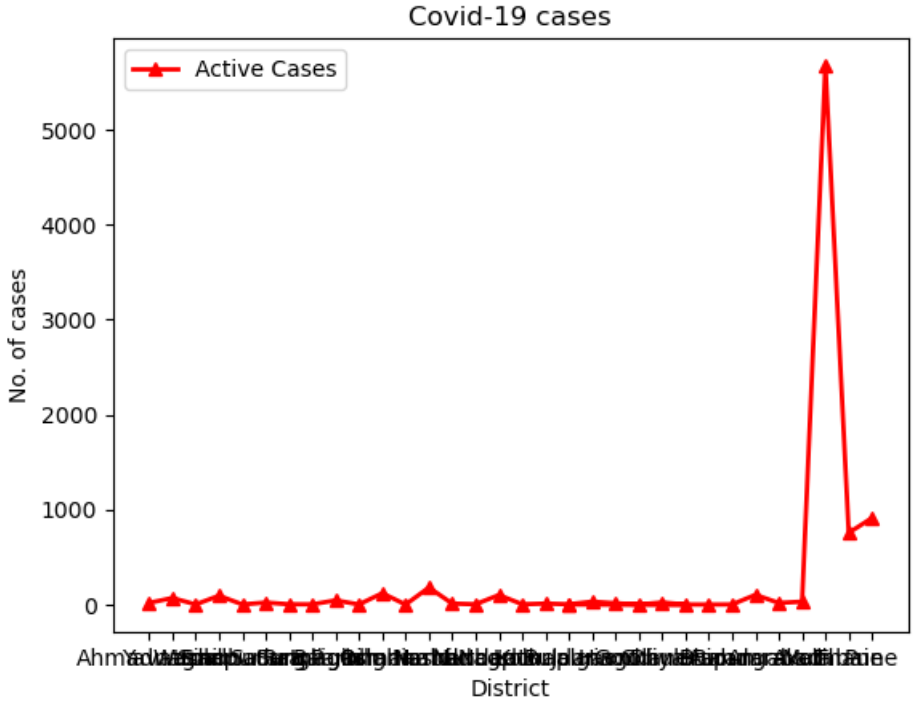
Out[99]:

	districtData/0/district	districtData/0/active	districtData/0/confirmed	districtData/0/deceased	districtData/0/recovered
23	Chandrapur	0	2	0	2
24	Buldana	3	21	1	17
25	Bid	0	1	0	1
26	Bhandara	1	1	0	0
27	Aurangabad	102	131	7	22
28	Amravati	17	28	7	4
29	Akola	30	39	1	8
30	Mumbai	5679	7061	290	1092
31	Thane	755	943	16	172
32	Pune	912	1248	88	248

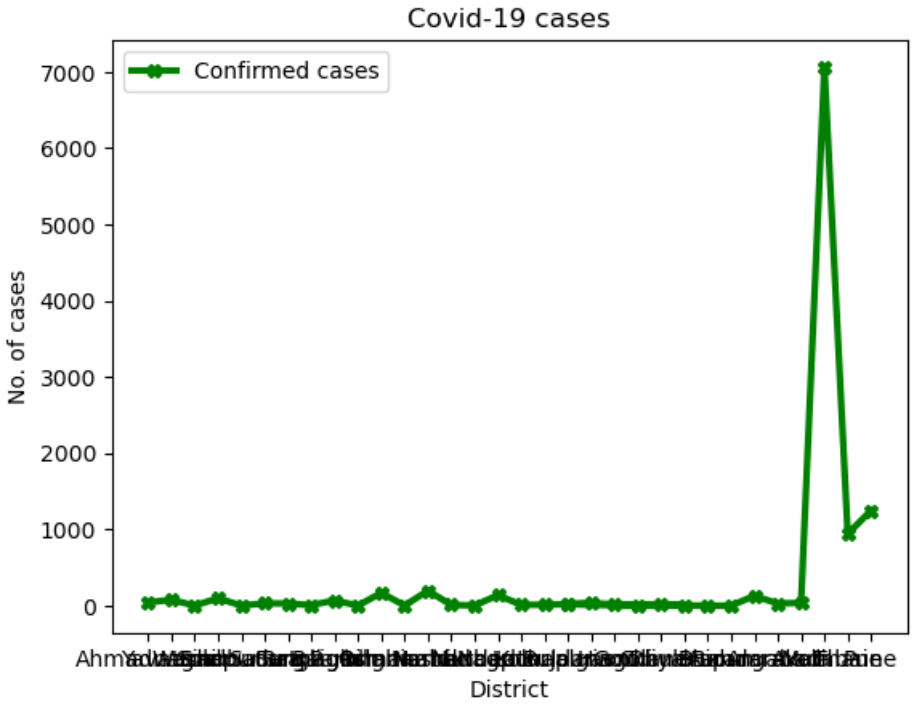
LINE PLOT

In [100]:
```
# Q.2- plot line diagram of active,confirmed,recovered,deceased cases district wise
```

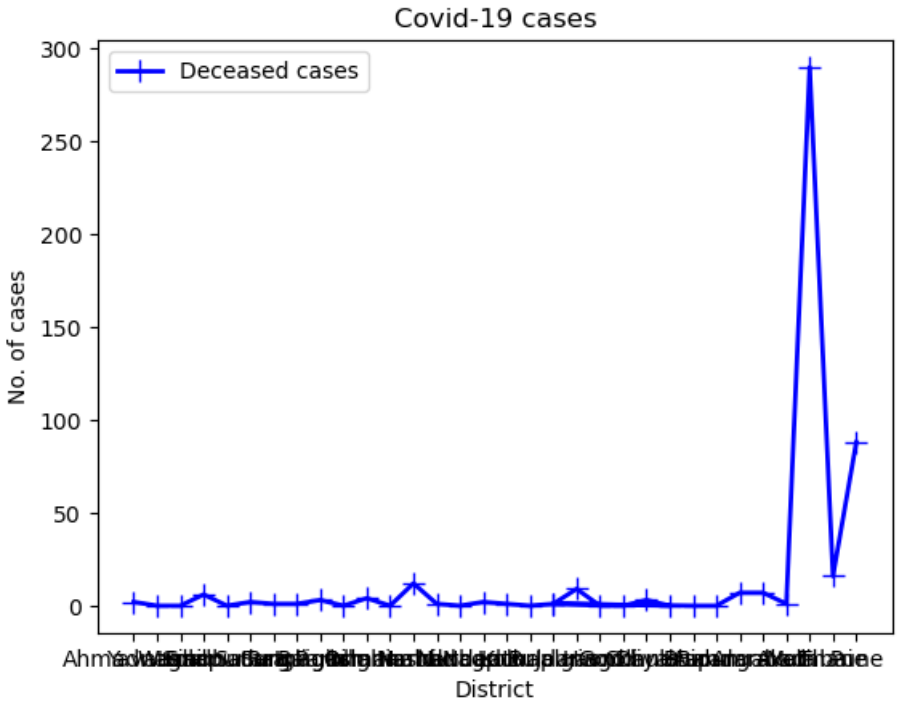
In [101]:
```
A = data.iloc[0:,1].values
C = data.iloc[0:,2].values
D = data.iloc[0:,3].values
R = data.iloc[0:,4].values
Z = data.iloc[0:,0]

plt.plot(Z, A,  label="Active Cases", color= "red", linewidth=2, marker='^')
plt.xlabel('District')
plt.ylabel('No. of cases')
plt.title('Covid-19 cases')
plt.legend()
plt.show()
```



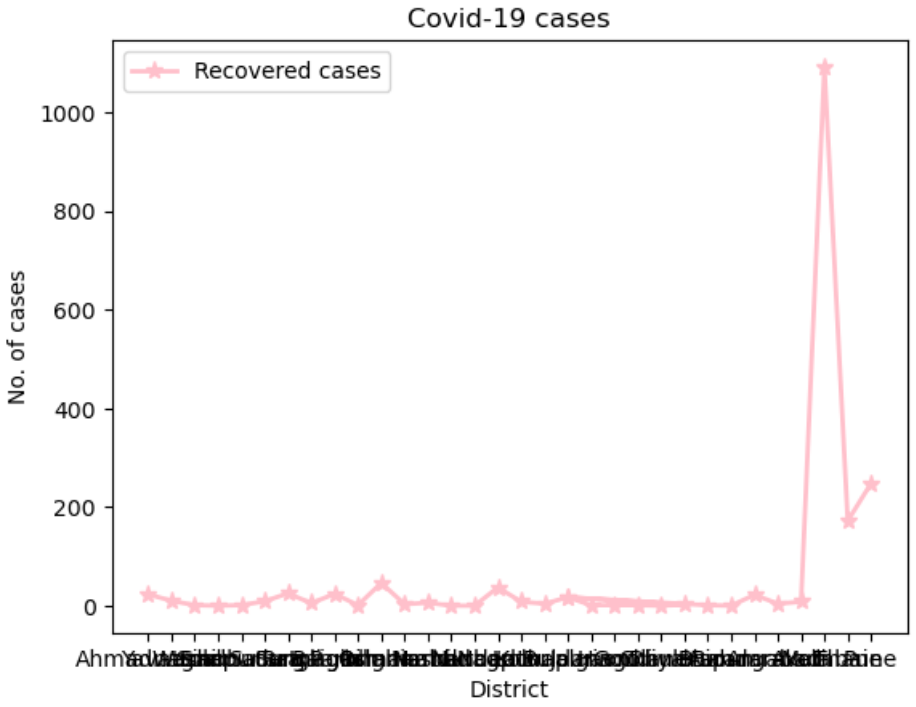
In [102]:
```
A = data.iloc[0:,1].values
C = data.iloc[0:,2].values
D = data.iloc[0:,3].values
R = data.iloc[0:,4].values
Z = data.iloc[0:,0]

plt.plot(Z, C, label="Confirmed cases",color= "green",linewidth=3, marker='X')
plt.xlabel('District')
plt.ylabel('No. of cases')
plt.title('Covid-19 cases')
plt.legend()
plt.show()
```



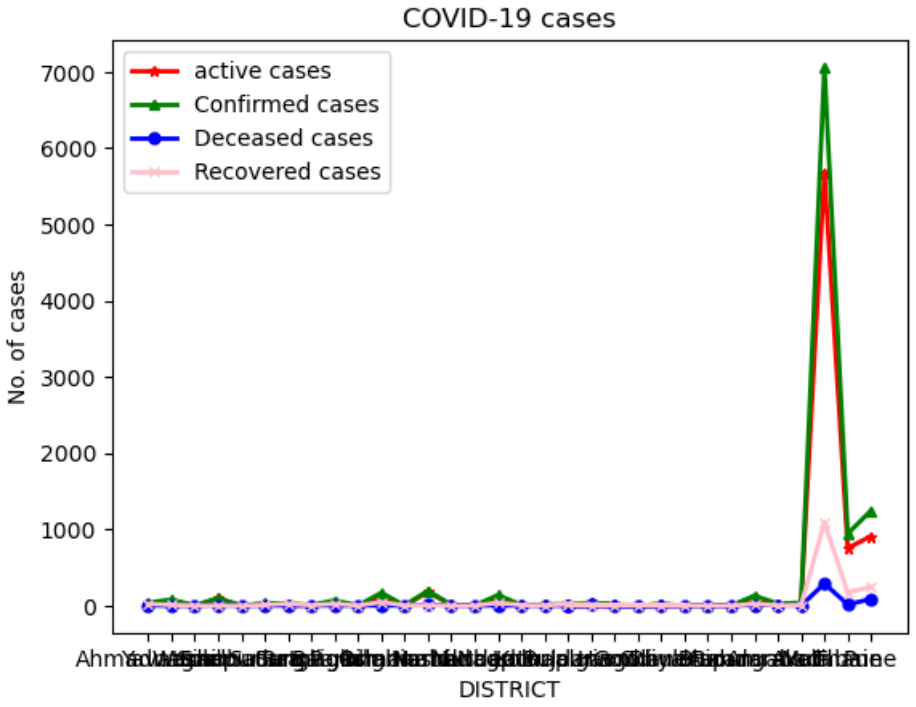
In [103]:
```
A = data.iloc[0:,1].values
C = data.iloc[0:,2].values
D = data.iloc[0:,3].values
R = data.iloc[0:,4].values
Z = data.iloc[0:,0]

plt.plot(Z, D,  label="Deceased cases",color= "blue",linewidth=2, marker='+', markersize= '10')
plt.xlabel('District')
plt.ylabel('No. of cases')
plt.title('Covid-19 cases')
plt.legend()
plt.show()
```



In [104]:
```
A = data.iloc[0:,1].values
C = data.iloc[0:,2].values
D = data.iloc[0:,3].values
R = data.iloc[0:,4].values
Z = data.iloc[0:,0]

plt.plot(Z, R, label="Recovered cases",color= "pink", linewidth=2, marker='*', markersize='8')
plt.xlabel('District')
plt.ylabel('No. of cases')
plt.title('Covid-19 cases')
plt.legend()
plt.show()
```



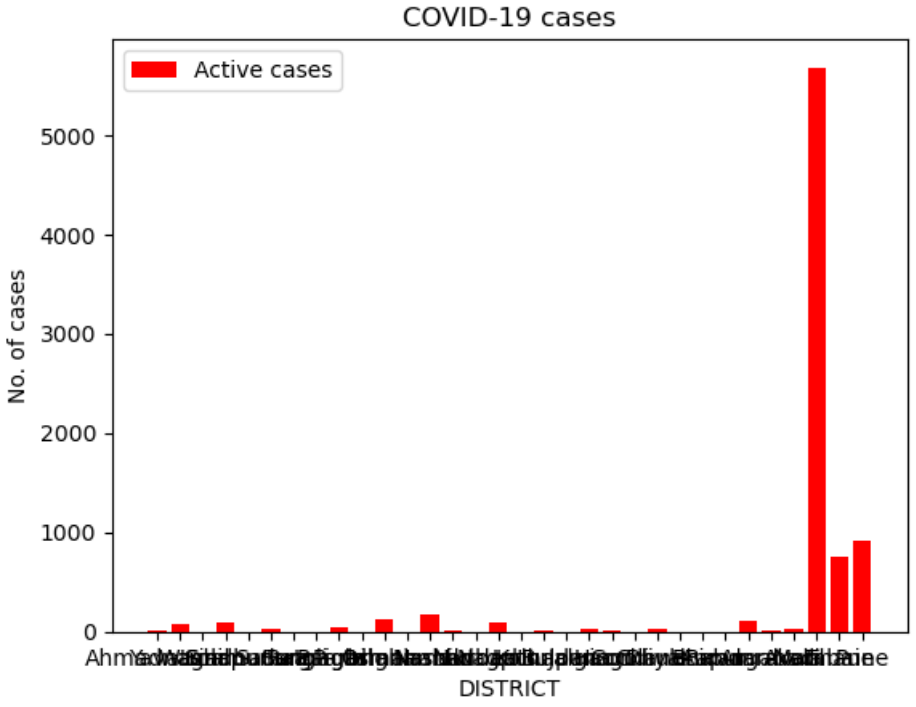
In [105]:
```
plt.plot(Z, A, label="active cases", color= "red",linewidth=2,marker='*', markersize='5')

plt.plot(Z, C, label="Confirmed cases",color= "green",linewidth=2,marker='^',markersize='5')
plt.plot(Z, D, label="Deceased cases",color= "blue",linewidth=2,marker='o',markersize='5')
plt.plot(Z, R, label="Recovered cases",color= "pink",linewidth=2,marker='x',markersize='5')


plt.xlabel('DISTRICT')
plt.ylabel('No. of cases')
plt.title('COVID-19 cases')
plt.legend()
plt.show()
```



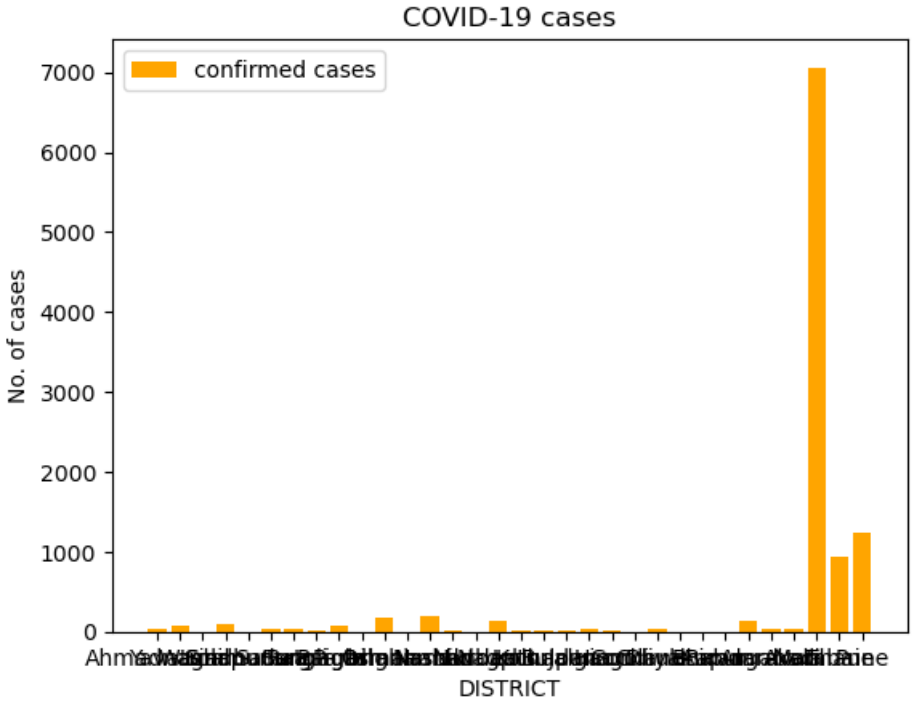
BAR GRAPH

In [106]:
```
#Q.3 - Bar graph-plot a bar diagram including active, confirmed, deceased & recovered cases district wise
```

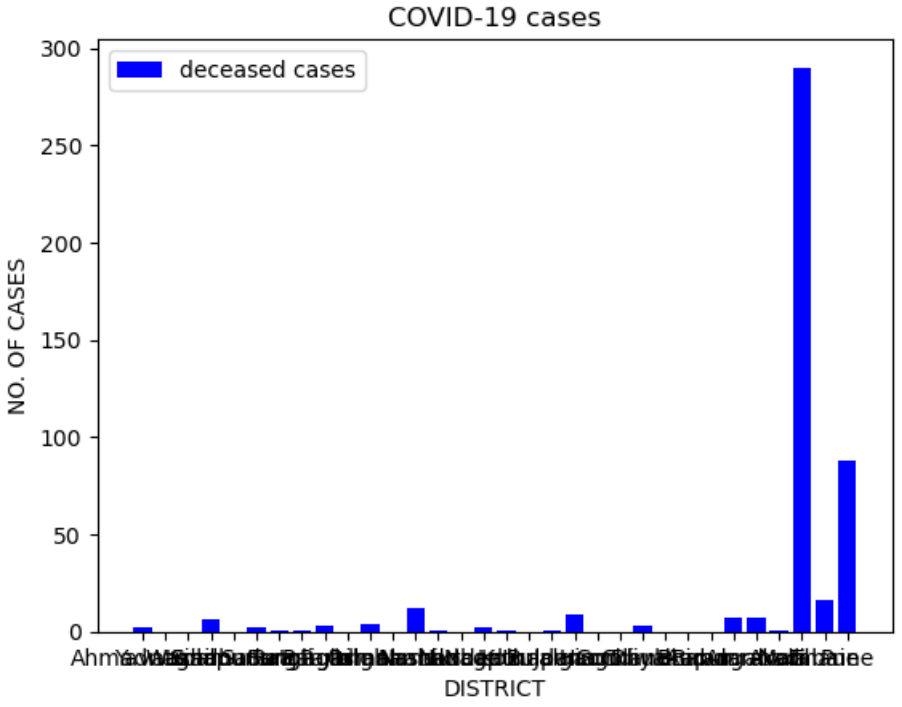
In [107]:
```
plt.bar(Z, A, label="Active cases", color= "red")
plt.xlabel('DISTRICT')
plt.ylabel('No. of cases')
plt.title('COVID-19 cases')
plt.legend()
plt.show()
```



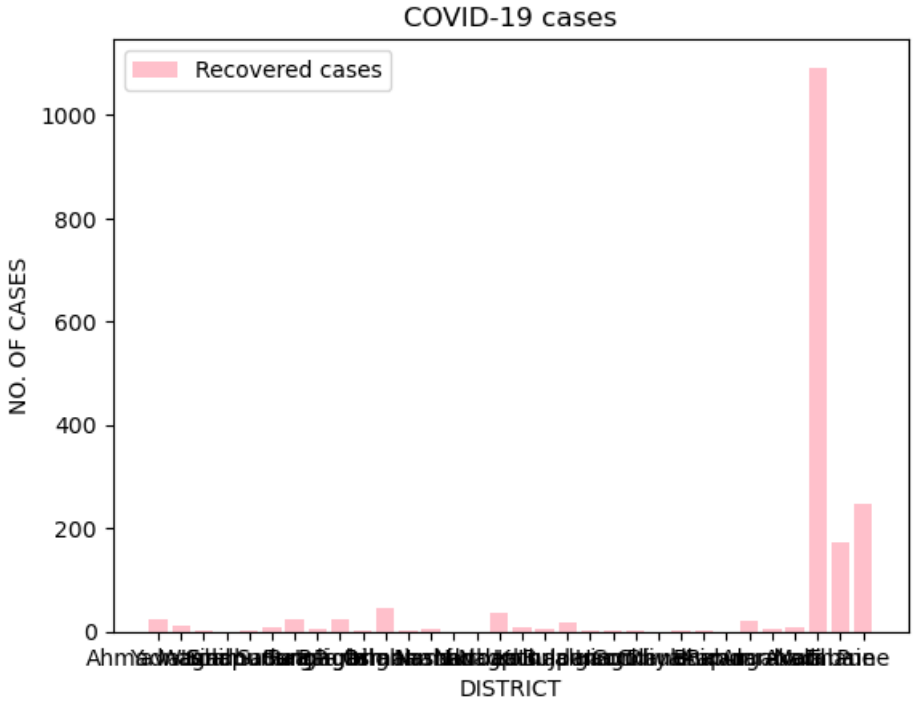
In [108]:
```
plt.bar(Z, C, label="confirmed cases",color="orange")
plt.xlabel('DISTRICT')
plt.ylabel('No. of cases')
plt.title('COVID-19 cases')
plt.legend()
plt.show()
```



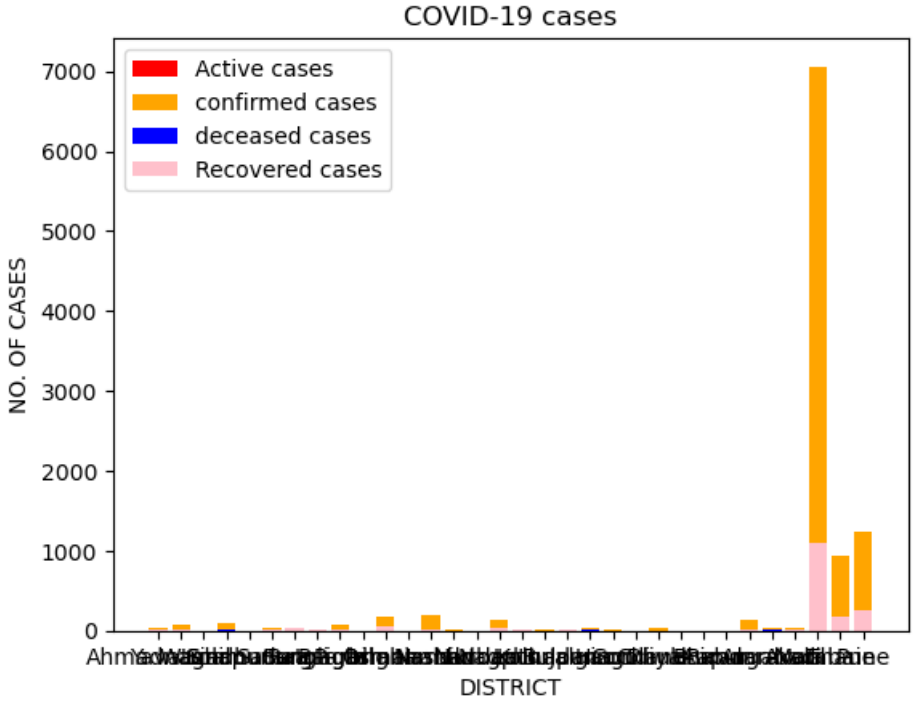
In [109]:
```
plt.bar(Z, D, label="deceased cases",color="blue")
plt.xlabel('DISTRICT')
plt.ylabel('NO. OF CASES')
plt.title('COVID-19 cases')
plt.legend()
plt.show()
```



In [110]:
```
plt.bar(Z, R, label="Recovered cases",color="Pink")
plt.xlabel('DISTRICT')
plt.ylabel('NO. OF CASES')
plt.title('COVID-19 cases')
plt.legend()
plt.show()
```

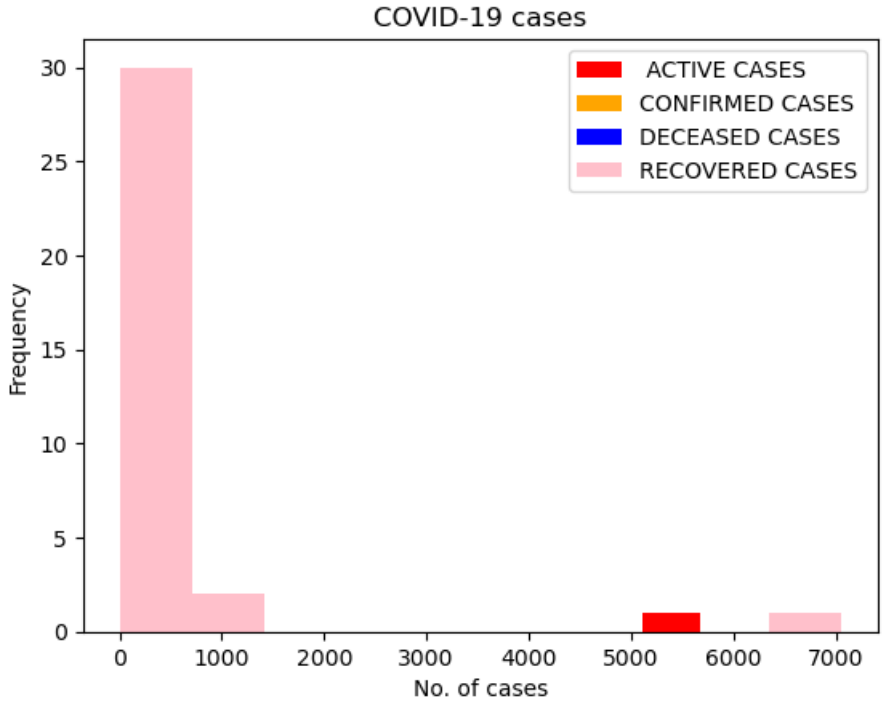


In [111]:
```
plt.bar(Z, A, label="Active cases", color= "red")
plt.bar(Z, C, label="confirmed cases",color="orange")
plt.bar(Z, D, label="deceased cases",color="blue")
plt.bar(Z, R, label="Recovered cases",color="Pink")


plt.xlabel('DISTRICT')
plt.ylabel('NO. OF CASES')
plt.title('COVID-19 cases')
plt.legend()
plt.show()
```



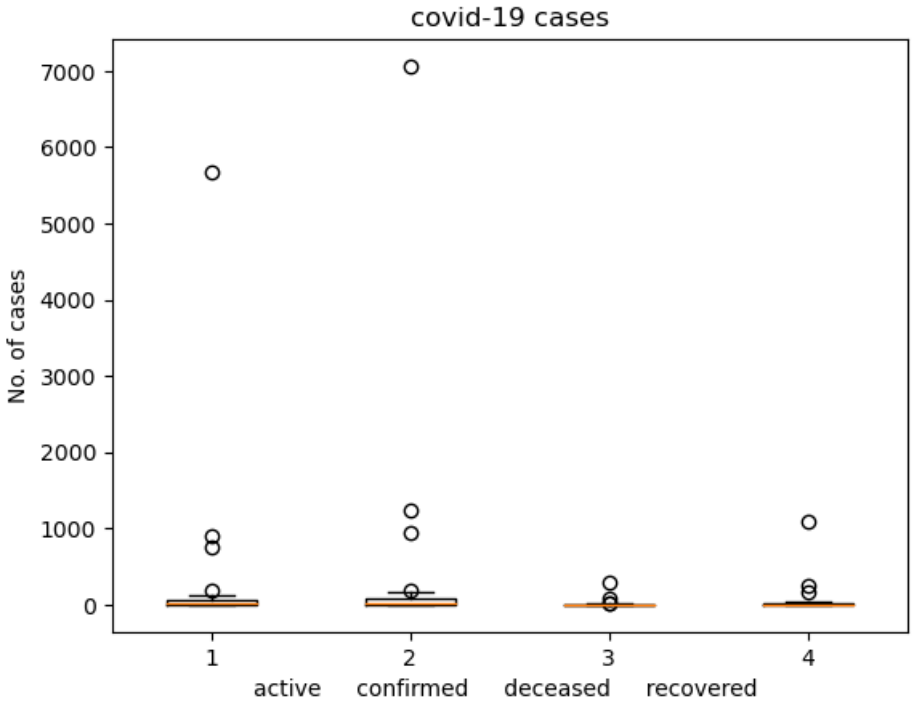
HISTOGRAM

In [112]:
```
plt.hist(A, label=" ACTIVE CASES", color= "red")
plt.hist(C, label="CONFIRMED CASES",color= "orange")
plt.hist(C, label="DECEASED CASES",color= "blue")
plt.hist(C, label="RECOVERED CASES",color= "pink")
plt.xlabel('No. of cases')
plt.ylabel('Frequency')
plt.title('COVID-19 cases')
plt.legend()
plt.show()
```



BOXPLOT

In [113]:
```
covidcases = [A,C,D,R]
plt.boxplot(covidcases)
plt.title('covid-19 cases')
plt.xlabel('active     confirmed     deceased     recovered ')
plt.ylabel('No. of cases')
plt.show()
```



In [114]:
```
# Q.4- plot only active vs recovered cases for top 5 district having highest no. of active cases
```

In [115]:
```
data.sort_values(['districtData/0/active', 'districtData/0/district'], ascending = False)
```

Out[115]:

	districtData/0/district	districtData/0/active	districtData/0/confirmed	districtData/0/deceased	districtData/0/recovered
30	Mumbai	5679	7061	290	1092
32	Pune	912	1248	88	248
31	Thane	755	943	16	172
12	Nashik	179	197	12	6
10	Palghar	119	169	4	46
27	Aurangabad	102	131	7	22
15	Nagpur	100	139	2	37
3	Solapur	93	99	6	0
1	Yavatmal	69	79	0	10
8	Raigarh	44	71	3	24
19	Jalgaon	30	40	9	1
29	Akola	30	39	1	8
22	Dhule	22	25	3	0
5	Satara	21	32	2	9
28	Amravati	17	28	7	4
0	Ahmadnagar	17	42	2	23
20	Hingoli	14	15	0	1
13	Nandurbar	10	11	1	0
17	Kolhapur	10	14	0	4
6	Sangli	3	29	1	25
14	Nanded	3	3	0	0
16	Latur	3	12	1	8
18	Buldana	3	21	1	17
24	Buldana	3	21	1	17
7	Ratnagiri	2	8	1	5
2	Washim	1	2	0	1
4	Sindhudurg	1	2	0	1
9	Parbhani	1	2	0	1
26	Bhandara	1	1	0	0
11	Osmanabad	0	3	0	3
21	Gondiya	0	1	0	1
23	Chandrapur	0	2	0	2
25	Bid	0	1	0	1

In [116]:
```
sortcases = data.sort_values(['districtData/0/active', 'districtData/0/district'], ascending = False)
```

In [117]:
```
sortcases.head(5)
```

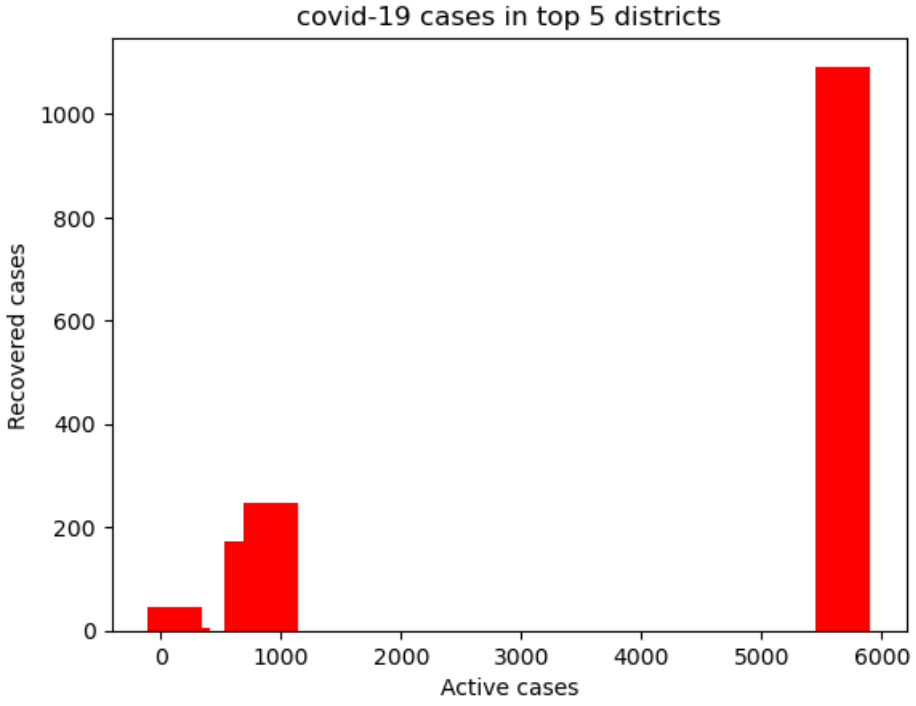
Out[117]:

	districtData/0/district	districtData/0/active	districtData/0/confirmed	districtData/0/deceased	districtData/0/recovered
30	Mumbai	5679	7061	290	1092
32	Pune	912	1248	88	248
31	Thane	755	943	16	172
12	Nashik	179	197	12	6
10	Palghar	119	169	4	46

In [118]:
```
highestcases = sortcases.head(5)
```

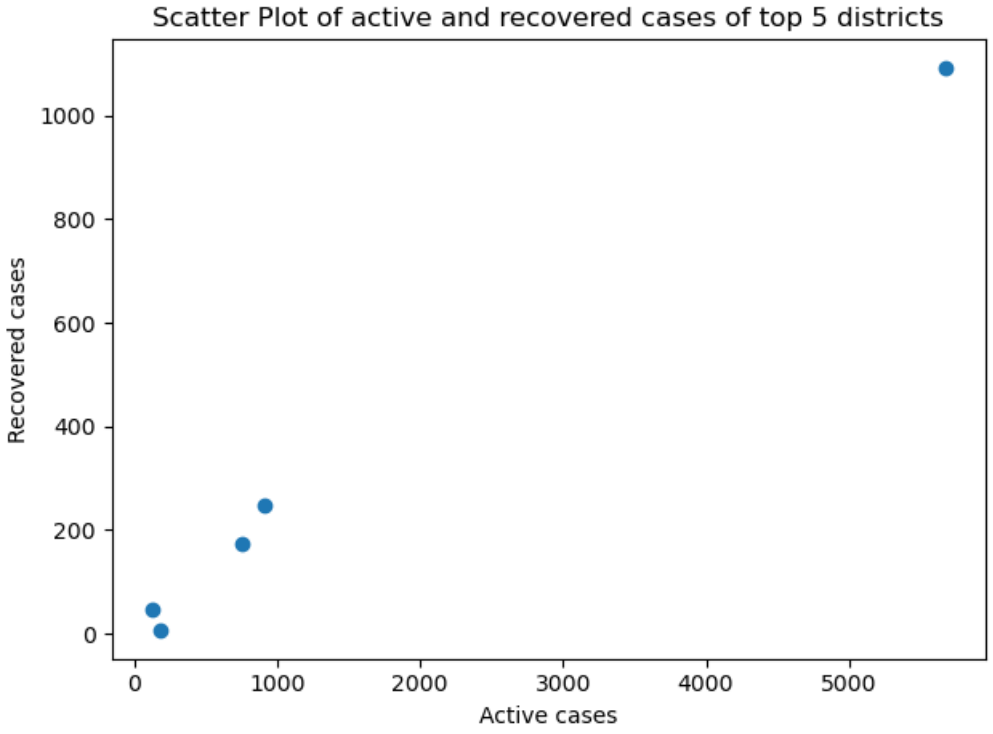
BAR GRAPH

In [119]:
```
a = highestcases.loc[:, "districtData/0/active"]
r = highestcases.loc[:,"districtData/0/recovered"]
plt.bar(a,r, width = 450, color="red")
plt.xlabel("Active cases")
plt.ylabel("Recovered cases")
plt.title("covid-19 cases in top 5 districts")
plt.show()
```



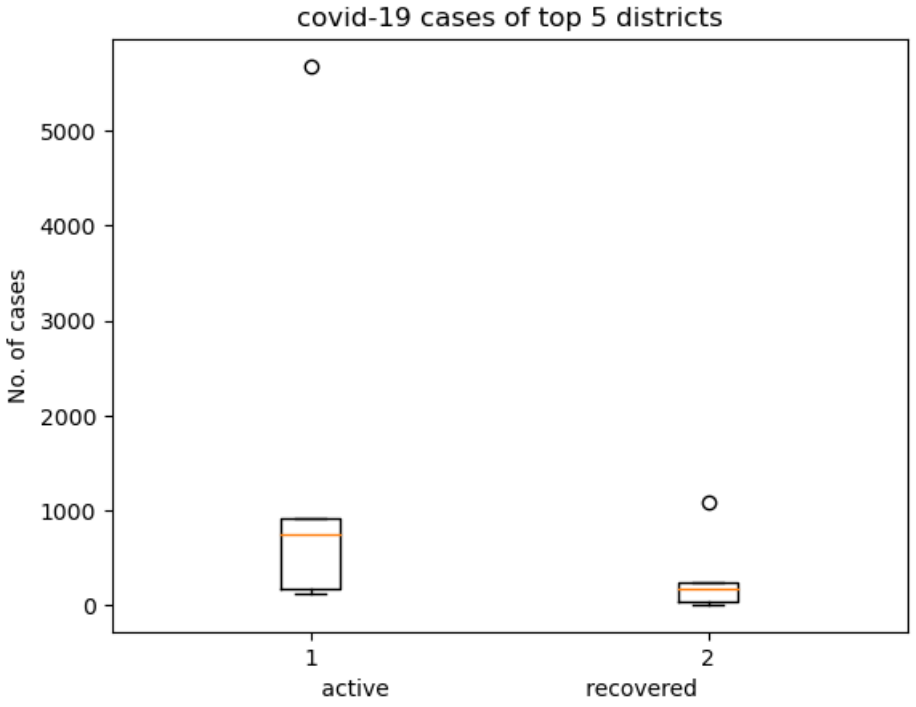
SCATTER PLOT

In [120]:
```
plt.scatter(a, r)
plt.xlabel("Active cases")
plt.ylabel("Recovered cases")
plt.title("Scatter Plot of active and recovered cases of top 5 districts")
plt.tight_layout()
plt.show()
```



BOX PLOT

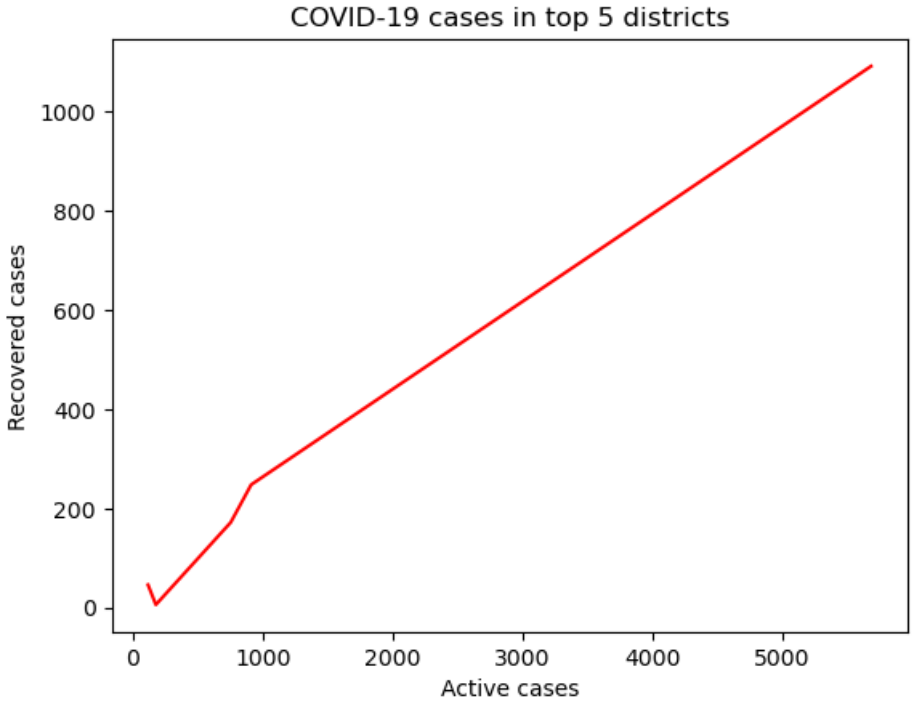
In [121]:
```
Covidcases = [a, r]
plt.boxplot(Covidcases)
plt.title('covid-19 cases of top 5 districts')
plt.xlabel(' active                                recovered ')
plt.ylabel('No. of cases')
plt.show()
```



LINE PLOT

In [122]:

```
plt.plot(a, r,  color= "red")
plt.xlabel('Active cases')
plt.ylabel('Recovered cases')
plt.title('COVID-19 cases in top 5 districts')
plt.show()
```



HISTOGRAM

In [91]:
```
plt.hist(a, label= "Active cases", color = "brown")
plt.hist(r, label= "recovered cases", color = "black")
plt.title('COVID-19 cases in top 5 districts')
plt.xlabel("No. of covid cases")
plt.ylabel(" frequency")
plt.show()
```