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

In [2]:
```
df=pd.read_csv("covid_19_india.csv")
```

In [3]:
```
df.head()
```

Out[3]:

	Sno	Date	Time	State/UnionTerritory	ConfirmedIndianNational	ConfirmedForeignNational
0	1	2020-01-30	6:00 PM	Kerala	1	0
1	2	2020-01-31	6:00 PM	Kerala	1	0
2	3	2020-02-01	6:00 PM	Kerala	2	0
3	4	2020-02-02	6:00 PM	Kerala	3	0
4	5	2020-02-03	6:00 PM	Kerala	3	0

Query 1

In [4]:
```
cdf=df.copy()
```

In [5]:
```
cdf=cdf.groupby(['State/UnionTerritory',])[['Cured','Date']].max()
cdf
```

Out[5]:

State/UnionTerritory	Cured	Date
Andaman and Nicobar Islands	7412	2021-08-11
Andhra Pradesh	1952736	2021-08-11
Arunachal Pradesh	47821	2021-08-11
Assam	559684	2021-08-11
Bihar	715352	2021-08-11
Bihar****	701234	2021-06-11
Cases being reassigned to states	0	2020-07-18
Chandigarh	61150	2021-08-11
Chhattisgarh	988189	2021-08-11
Dadra and Nagar Haveli	10261	2021-06-05
Dadra and Nagar Haveli and Daman and Diu	10646	2021-08-11
Daman & Diu	0	2020-06-11
Delhi	1411280	2021-08-11
Goa	167978	2021-08-11
Gujarat	814802	2021-08-11
Haryana	759790	2021-08-11
Himachal Pradesh	202761	2021-08-11
Himanchal Pradesh	200040	2021-07-20
Jammu and Kashmir	317081	2021-08-11
Jharkhand	342102	2021-08-11
Karanataka	2821491	2021-07-20
Karnataka	2861499	2021-08-11
Kerala	3396184	2021-08-11
Ladakh	20130	2021-08-11
Lakshadweep	10165	2021-08-11
Madhya Pradesh	781330	2021-08-11
Madhya Pradesh***	780735	2021-07-13
Maharashtra	6159676	2021-08-11
Maharashtra***	6000911	2021-07-21
Manipur	96776	2021-08-11
Meghalaya	64157	2021-08-11
Mizoram	33722	2021-08-11
Nagaland	26852	2021-08-11
Odisha	972710	2021-08-11
Puducherry	119115	2021-08-11

State/UnionTerritory	Cured	Date
Punjab	582791	2021-08-11
Rajasthan	944700	2021-08-11
Sikkim	25095	2021-08-11
Tamil Nadu	2524400	2021-08-11
Telangana	638410	2021-08-11
Telengana	362160	2021-05-01
Tripura	77811	2021-08-11
Unassigned	0	2020-04-03
Uttar Pradesh	1685492	2021-08-11
Uttarakhand	334650	2021-08-11
West Bengal	1506532	2021-08-11

Query 2

In [6]:
```
df['Date']=pd.to_datetime(df['Date'])
df['Month']=df['Date'].dt.month_name()
df['Year']=df['Date'].dt.year
```

In [7]:
```
df[df['Date'].dt.day==6].groupby('Date')['Confirmed'].sum()
```

Out[7]:
```
Date
2020-02-06           3
2020-03-06          31
2020-04-06        4281
2020-05-06       49391
2020-06-06      236657
2020-07-06      697413
2020-08-06     1964536
2020-09-06     4113811
2020-10-06     6685082
2020-11-06     8411724
2020-12-06     9644222
2021-01-06    10374932
2021-02-06    10814304
2021-03-06    11192088
2021-04-06    12686049
2021-05-06    21077410
2021-06-06    28809339
2021-07-06    30619932
2021-08-06    31856757
Name: Confirmed, dtype: int64
```

Query 3

In [8]:
```
lst = ['Karnataka', 'Gujarat', 'Haryana', 'Uttar Pradesh']
```

In [9]:
```
kar=df[df['State/UnionTerritory'].isin(lst)].groupby(['State/UnionTerritory', 'Mon
```

In [10]:
```
kar
```

Out[10]:

	State/UnionTerritory	Month	Cured
0	Gujarat	April	9998019
1	Gujarat	August	10838152
2	Gujarat	December	6572851
3	Gujarat	February	7242705
4	Gujarat	January	7537895
5	Gujarat	July	26213645
6	Gujarat	June	24428338
7	Gujarat	March	8433988
8	Gujarat	May	18934056
9	Gujarat	November	5155629
10	Gujarat	October	4244333
11	Gujarat	September	2887516
12	Haryana	April	9297316
13	Haryana	August	9573916
14	Haryana	December	7441429
15	Haryana	February	7423157
16	Haryana	January	8074769
17	Haryana	July	24083927
18	Haryana	June	22626852
19	Haryana	March	8372425
20	Haryana	May	17817747
21	Haryana	November	5374420
22	Haryana	October	4156750
23	Haryana	September	2342634
24	Karnataka	April	30268181
25	Karnataka	August	35666764
26	Karnataka	December	27068337
27	Karnataka	February	25951947
28	Karnataka	January	28192578
29	Karnataka	July	84815054
30	Karnataka	June	76373384
31	Karnataka	March	29173959
32	Karnataka	May	49787540
33	Karnataka	November	24477198
34	Karnataka	October	19151225
35	Karnataka	September	10918193

	State/UnionTerritory	Month	Cured
36	Uttar Pradesh	April	20054528
37	Uttar Pradesh	August	21675662
38	Uttar Pradesh	December	16732432
39	Uttar Pradesh	February	16520422
40	Uttar Pradesh	January	17870678
41	Uttar Pradesh	July	53053765
42	Uttar Pradesh	June	50315909
43	Uttar Pradesh	March	18443966
44	Uttar Pradesh	May	42417838
45	Uttar Pradesh	November	14404679
46	Uttar Pradesh	October	12456635
47	Uttar Pradesh	September	7532837

In [11]:
```
kar=kar.pivot( 'Month','State/UnionTerritory', 'Cured').fillna(0)
kar
```

Out[11]:

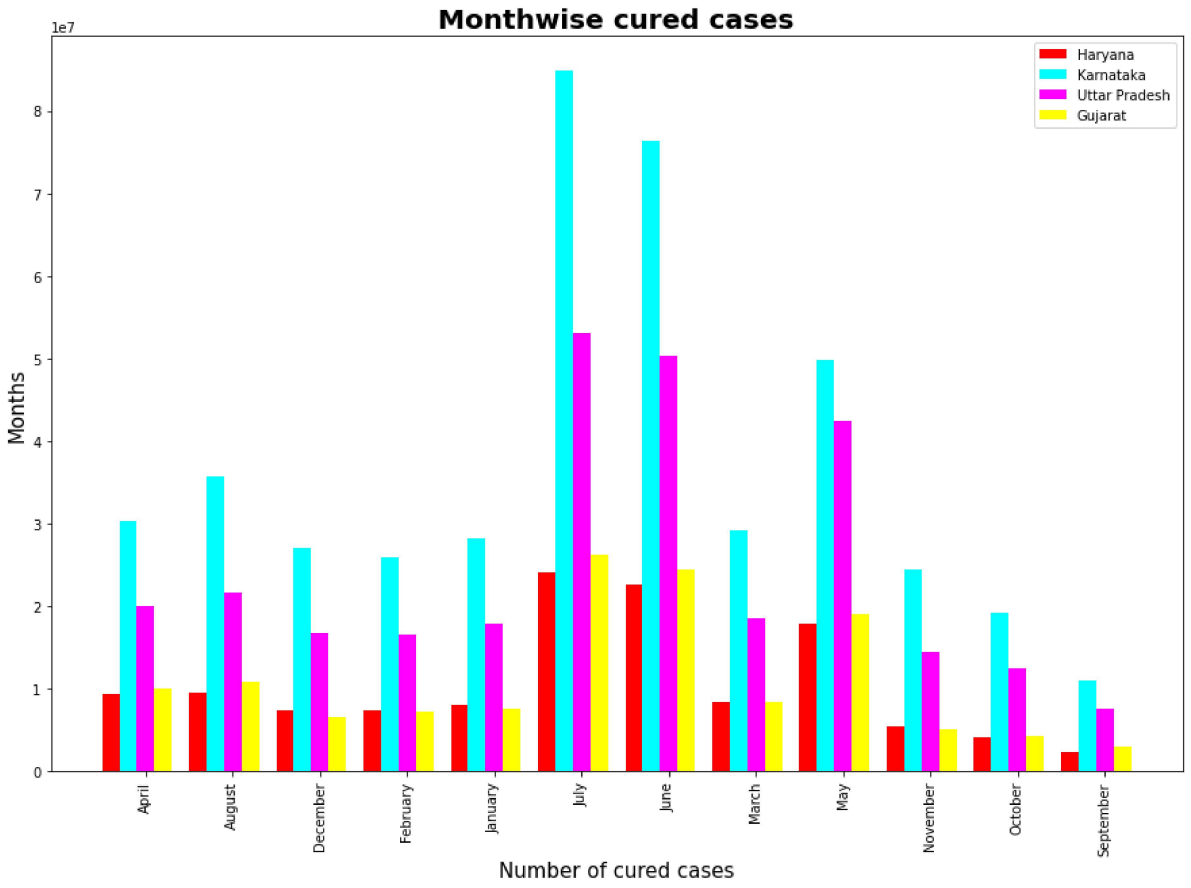
State/UnionTerritory	Gujarat	Haryana	Karnataka	Uttar Pradesh
Month				
April	9998019	9297316	30268181	20054528
August	10838152	9573916	35666764	21675662
December	6572851	7441429	27068337	16732432
February	7242705	7423157	25951947	16520422
January	7537895	8074769	28192578	17870678
July	26213645	24083927	84815054	53053765
June	24428338	22626852	76373384	50315909
March	8433988	8372425	29173959	18443966
May	18934056	17817747	49787540	42417838
November	5155629	5374420	24477198	14404679
October	4244333	4156750	19151225	12456635
September	2887516	2342634	10918193	7532837

In [12]:
```
X_axis = np.arange(len(kar.index))
plt.figure(figsize = (15,10))
plt.bar(X_axis-0.4, kar['Haryana'], 0.2, label = 'Haryana', color='red')
plt.bar(X_axis-0.2, kar['Karnataka'], 0.2, label = 'Karnataka', color='aqua')
plt.bar(X_axis, kar['Uttar Pradesh'], 0.2, label = 'Uttar Pradesh', color='magenta
plt.bar(X_axis+0.2, kar['Gujarat'], 0.2, label = 'Gujarat', color='yellow')
plt.title("Monthwise cured cases", fontsize = 20, fontweight = 'bold')
plt.xlabel("Number of cured cases", fontsize = 15)
plt.ylabel("Months",fontsize=15)
plt.xticks(X_axis, kar.index,rotation=90, size = 10)
```

```
plt.yticks(size = 10)
plt.legend()
```

Out[12]:

```
<matplotlib.legend.Legend at 0x168e30f7130>
```



Query 4

In [13]:

```
lst=['Karnataka','Delhi','Madhya Pradesh']
```

In [14]:

```
may=df[df['Month']=='May'].groupby(['State/UnionTerritory', 'Year'])['Deaths'].sum
```

In [15]:

```
may=may[may['State/UnionTerritory'].isin(lst)].pivot('State/UnionTerritory', 'Year
```

In [16]:

```
may
```

Out[16]:

State/UnionTerritory	Year	2020	2021
Delhi		4916	647295
Karnataka		1103	671523
Madhya Pradesh		7414	213504

In [17]:

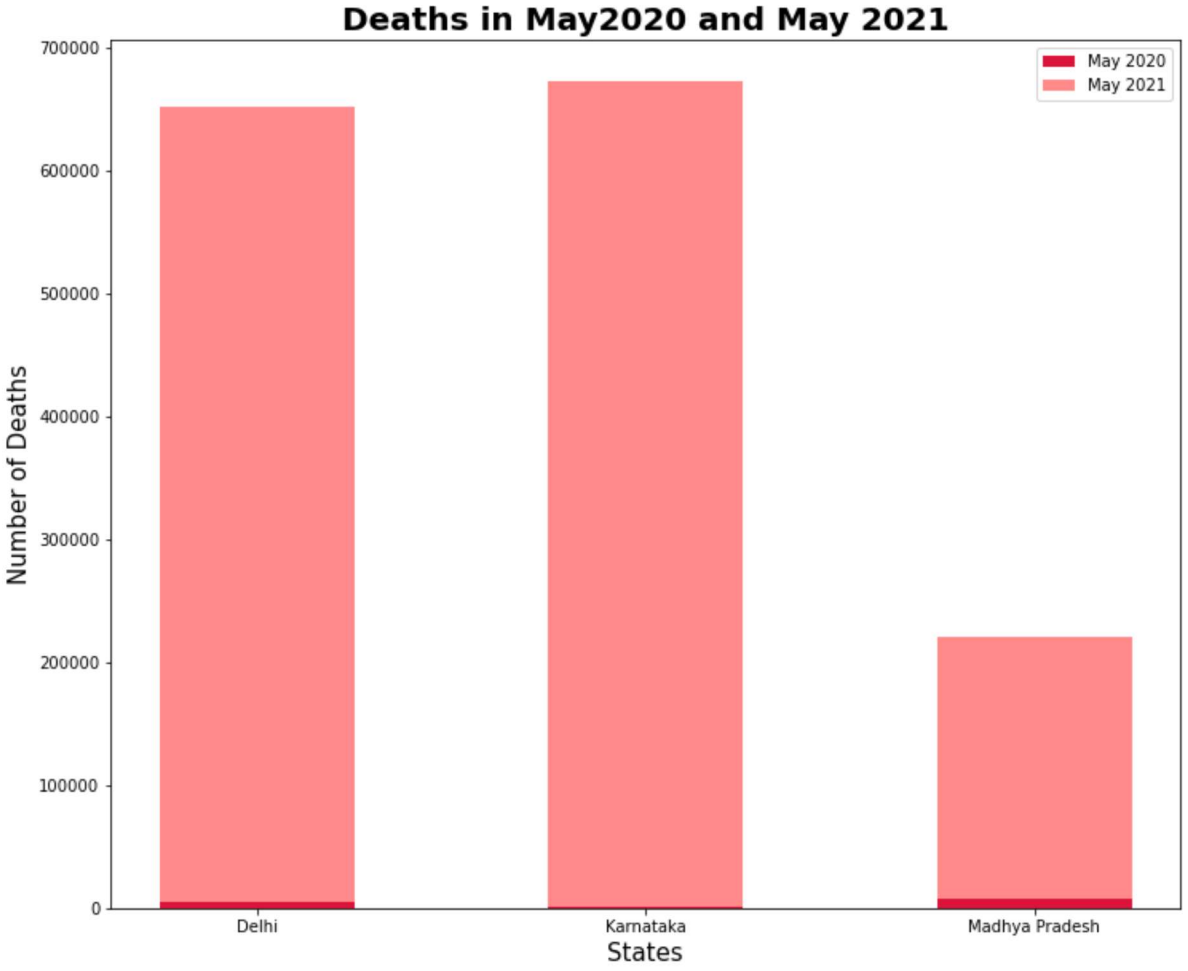
```
plt.figure(figsize = (15,5))
fig = plt.subplots(figsize =(12, 10))
p1 = plt.bar(may.index, may[2020], 0.5, color='#DC143C')
p2 = plt.bar(may.index, may[2021], 0.5,
             bottom = may[2020], color='#FF8A8A')
plt.ylabel('Number of Deaths',fontsize = 15)
plt.xlabel('States',fontsize = 15)
plt.title('Deaths in May2020 and May 2021',fontsize = 20, fontweight = 'bold')
```

```
plt.legend((p1[0], p2[0]), ('May 2020', 'May 2021'))
plt.show
```

Out[17]:

```
<function matplotlib.pyplot.show(close=None, block=None)>

<Figure size 1080x360 with 0 Axes>
```



Query 5

In [18]:

```
up=df[df['State/UnionTerritory']=='Uttar Pradesh']
up
```

Out[18]:

	Sno	Date	Time	State/UnionTerritory	ConfirmedIndianNational	ConfirmedForeignNati
39	40	2020-03-04	6:00 PM	Uttar Pradesh	6	
50	51	2020-03-05	6:00 PM	Uttar Pradesh	7	
55	56	2020-03-06	6:00 PM	Uttar Pradesh	7	
58	59	2020-03-07	6:00 PM	Uttar Pradesh	7	
72	73	2020-03-08	6:00 PM	Uttar Pradesh	7	
...	...	...	...	...	...	
17964	17965	2021-08-07	8:00 AM	Uttar Pradesh	-	
18000	18001	2021-08-08	8:00 AM	Uttar Pradesh	-	
18036	18037	2021-08-09	8:00 AM	Uttar Pradesh	-	
18072	18073	2021-08-10	8:00 AM	Uttar Pradesh	-	
18108	18109	2021-08-11	8:00 AM	Uttar Pradesh	-	

526 rows × 11 columns

In [19]:
```
up=up.groupby('Month').agg({'Confirmed':sum,'Deaths':sum})
```

In [20]:
```
up
```

Out[20]:

Month	Confirmed	Deaths
April	24690552	295351
August	23480883	326464
December	17564562	250784
February	16856451	243563
January	18433015	264524
July	54328963	736790
June	51474534	665720
March	18814701	271304
May	48338036	528765
November	15327991	221126
October	13782962	201433
September	9533193	137626

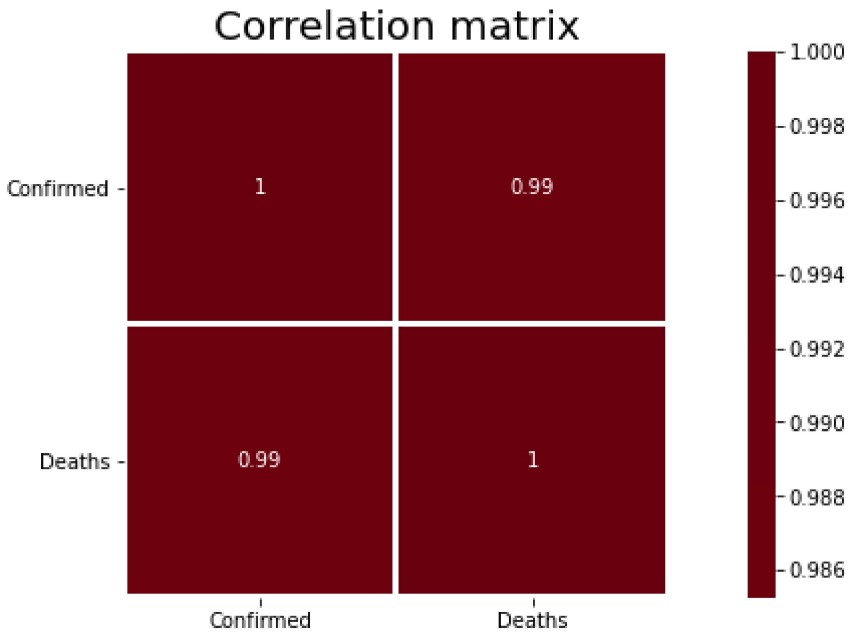
In [21]:
```
corr = up.loc[:,['Confirmed','Deaths']].corr()
corr
```

Out[21]:

	Confirmed	Deaths
Confirmed	1.000000	0.985265
Deaths	0.985265	1.000000

In [22]:
```
fig = plt.figure(figsize=(14, 5))
ax1 = plt.subplot()
ax1 = sns.heatmap(corr, annot=True , cmap='Reds', vmax=1, center=0.5, square=True,
ax1.set_title('Correlation matrix', fontsize=20)
plt.yticks(rotation=0)
```

Out[22]:
```
(array([0.5, 1.5]), [Text(0, 0.5, 'Confirmed'), Text(0, 1.5, 'Deaths')])
```



In [ ]: