GDP ANALYSIS

Note-

1. Analysis is done in the form of Q & A format ,questions asked in assignment are answered in this document
2. Questions are in black and solutions are in Green
3. For Jupyter notebook to run successfully seaborn=0.9.0 version is required because catplot demands this version

PartI-A :

- For the analysis below, use the Data I-A.
  df=pd.read_csv('Data1a/DataIA.csv') , Data I-A has been used

- Remove the rows: '(% Growth over the previous year)' and 'GSDP - CURRENT PRICES (` in Crore)' for the year 2016-17.
  df1=df.drop(5)
  df1=df1.drop(10)
  mentioned rows are removed

- Calculate the average growth of states over the duration 2013-14, 2014-15 and 2015-16 by taking the mean of the row '(% Growth over previous year)'. Compare the calculated value and plot it for the states. Make appropriate transformations if necessary to plot the data. Report the average growth rates of the various states:
  Average % Growth over previous year State wise:

```
In [67]: df_gopv=df1[6:]
         df_gopv
```

Out[67]:

	Items Description	Duration	Andhra Pradesh	Arunachal Pradesh	Assam	Bihar	Chhattisgarh	Goa	Gujarat	Haryana	...	Tamil Nadu	Telangana	Tripura	Uttar Pradesh	Uttarakhand	Anda Nic Isla
7	(% Growth over previous year)	2013-14	12.85	16.38	13.31	12.30	16.44	-5.77	11.47	15.45	...	13.51	12.63	18.14	14.73	13.64	1
8	(% Growth over previous year)	2014-15	13.40	14.79	11.45	17.92	13.69	13.12	10.82	9.18	...	12.51	13.05	15.92	10.51	8.12	1
9	(% Growth over previous year)	2015-16	15.85	12.07	13.19	10.59	10.98	10.75	11.09	10.91	...	10.99	12.61	NaN	10.58	13.65	

3 rows × 35 columns

```
In [68]: df_gopv_mean=df_gopv.describe().loc['mean',:]
         df_gopv_mean
```

Out[68]:

```
Andhra Pradesh              14.033333
Arunachal Pradesh           14.413333
Assam                       12.650000
Bihar                       13.603333
Chhattisgarh                13.703333
Goa                          6.033333
Gujarat                     11.126667
Haryana                     11.846667
Himachal Pradesh            12.280000
Jammu & Kashmir             10.900000
```

Comparing the calculated values: Comparing by sorting values in descending order

```
In [69]: df_gopv_mean.sort_values(ascending=False,inplace=True)

In [70]: df_gopv_mean1=pd.DataFrame(df_gopv_mean)
         df_gopv_mean1
```

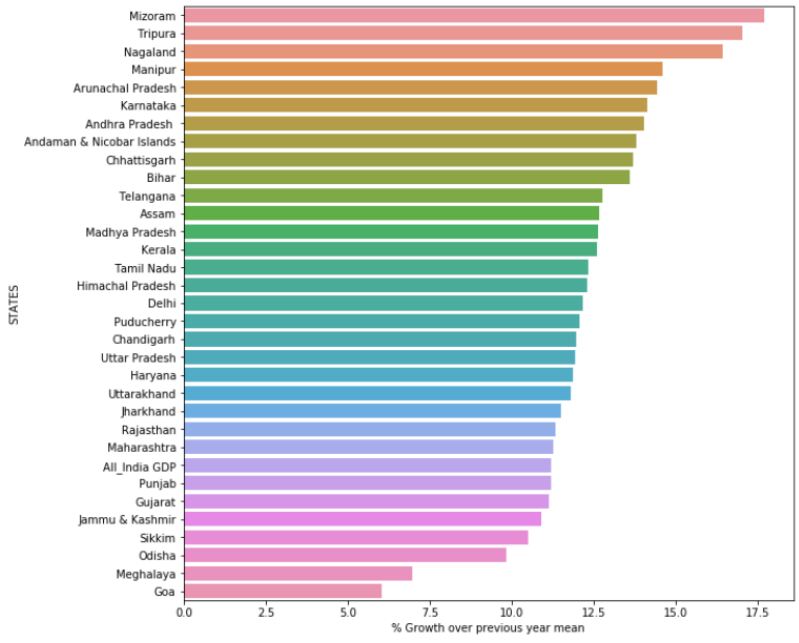
Out[70]:

	mean
Mizoram	17.700000
Tripura	17.030000
Nagaland	16.415000
Manipur	14.610000
Arunachal Pradesh	14.413333
Karnataka	14.120000
Andhra Pradesh	14.033333
Andaman & Nicobar Islands	13.785000
Chhattisgarh	13.703333
Bihar	13.603333
Telangana	12.763333
Assam	12.650000
Madhya Pradesh	12.626667
Kerala	12.583333
Tamil Nadu	12.336667
Himachal Pradesh	12.280000
Delhi	12.160000
Puducherry	12.053333
Chandigarh	11.960000

Plotting the Data:

```
In [377]: plt.figure(figsize=(10,10))
          fig=sns.barplot(y=df_gopv_mean.index.values,x='mean',data=df_gopv_mean1)
          fig.set(xlabel='% Growth over previous year mean',ylabel='STATES')
```

Out[377]: [Text(0,0.5,'STATES'), Text(0.5,0,'% Growth over previous year mean')]



- Which states have been growing consistently fast, and which ones have been struggling?

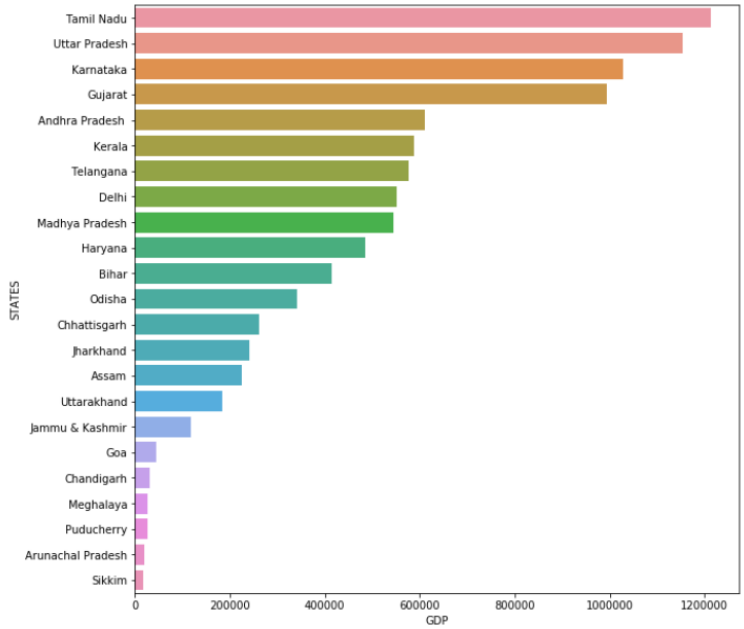
  Mizoram is growing consistently fast as compared to Goa which is lagging behind

- Curiosity exercise - what has been the average growth rate of your home state, and how does it compare to the national average over this duration?
  My Home State is Karnataka which is doing well its in 6th position,average growth rate of karnataka is 1.2 times greater than All India GDP,Karnataka's current growth rate is 14.12%.

- Plot the total GDP of the states for the year 2015-16:

```
In [76]: plt.figure(figsize=[10,10])
         fig=sns.barplot(y=df_gdp_1516.index,x=4,data=df_gdp_1516)
         fig.set(xlabel='GDP',ylabel='STATES')

Out[76]: [Text(0,0.5,'STATES'), Text(0.5,0,'GDP')]
```



- Identify the top-5 and the bottom-5 states based on total GDP

  Top 5 States are:

  1. Tamil Nadu
  2. Uttarpradesh
  3. Karnataka
  4. Gujarat
  5. Andrapradesh

  Bottom 5 states are:

  1. Chandigarh
  2. Meghalaya
  3. puducherry
  4. Arunachal Pradesh

5. Sikkim

Part I-B:

- For the analysis below, use Data I-B. You can also use Data I-B along with Data I-A if required. Also, perform the analysis only for the duration : 2014-15.

  Gsva_df_list is a list of all Dataframes

```
In [77]: #Reading all files in Data1b and storing the corresponding dataframe in list
import os
arr=os.listdir('Data1b/')
States=('Manipur','Nagaland','Mizoram','Maharashtra','Tripura','Punjab','Rajasthan','Himachal_Pradesh')
gsva_df_list=[]
for state in arr:
    df=pd.read_csv('Data1b/'+state,encoding = "ISO-8859-1")
    state=state.split('-')[1] #getting only state name in filename
    df=df.loc[:,['S.No.','Item','2014-15']]#choosing only requried columns
    df['State']=state # adding extra column of state so that we can get to know the dataframe belongs to which state
    gsva_df_list.append(df)
gsva_df_list[2]
```

Out[77]:

	S.No.	Item	2014-15	State
0	1	Agriculture, forestry and fishing	3855548	Assam
1	1.1	Crops	2890544	Assam
2	1.2	Livestock	173478	Assam
3	1.3	Forestry and logging	261987	Assam
4	1.4	Fishing and aquaculture	529539	Assam
5	2	Mining and quarrying	1471149	Assam
6	Total	Primary	5326697	Assam
7	3	Manufacturing	2002936	Assam
8	4	Electricity, gas, water supply & other utility...	296587	Assam
9	5	Construction	1733568	Assam
10	Total	Secondary	4033091	Assam
11	6	Trade, repair, hotels and restaurants	2987155	Assam
12	6.1	Trade & repair services	2876251	Assam

Concatenating all dataframes in to df_merge , so df_merged is final dataframe which has information of all states

```
In [78]: df_merged=pd.DataFrame({'Item':[],'2014-15':[],'State':[]})
for df in gsva_df_list:
    df_merged=pd.concat([df_merged,df],ignore_index=True) #Concatinating all states data frames in to 1
df_merged
```

	2014-15	Item	S.No.	State
0	14819416.0	Agriculture, forestry and fishing	1	Andhra_Pradesh
1	7893514.0	Crops	1.1	Andhra_Pradesh
2	4309078.0	Livestock	1.2	Andhra_Pradesh
3	346160.0	Forestry and logging	1.3	Andhra_Pradesh
4	2270664.0	Fishing and aquaculture	1.4	Andhra_Pradesh
5	1484300.0	Mining and quarrying	2	Andhra_Pradesh
6	16303716.0	Primary	Total	Andhra_Pradesh
7	4672266.0	Manufacturing	3	Andhra_Pradesh
8	1151729.0	Electricity, gas, water supply & other utility...	4	Andhra_Pradesh
9	4664889.0	Construction	5	Andhra_Pradesh
10	10488884.0	Secondary	Total	Andhra_Pradesh
11	4233400.0	Trade, repair, hotels and restaurants	6	Andhra_Pradesh

- Filter out the Union Territories (Delhi, Chandigarh, Andaman and Nicobar Islands etc.) for further analysis since they are governed directly by the centre, not state governments.

  Filtering out the Union Teritories

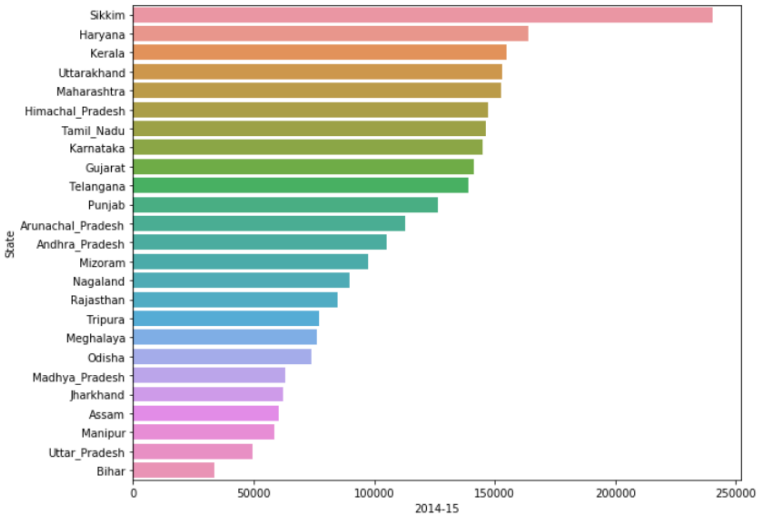
```
In [82]: list_UnionTeritories=['Delhi','Goa','Chhattisgarh']
         df_merged_filtered=df_merged[~df_merged['State'].isin(list_UnionTeritories)] #removing the union territories from datafr
         df_merged_filtered['State'].unique()

Out[82]: array(['Andhra_Pradesh', 'Arunachal_Pradesh', 'Assam', 'Bihar', 'Gujarat',
                'Haryana', 'Himachal_Pradesh', 'Jharkhand', 'Karnataka', 'Kerala',
                'Madhya_Pradesh', 'Maharashtra', 'Manipur', 'Meghalaya', 'Mizoram',
                'Nagaland', 'Odisha', 'Punjab', 'Rajasthan', 'Sikkim',
                'Tamil_Nadu', 'Telangana', 'Tripura', 'Uttarakhand',
                'Uttar_Pradesh'], dtype=object)
```

- Plot the GDP per capita for all the states.

```
In [108]: plt.figure(figsize=[10,8])
          sns.barplot(x='2014-15',y='State',data=df_merged_filtered[df_merged_filtered['Item']=='Per Capita GSDP (Rs.)'].sort_valu

Out[108]: <matplotlib.axes._subplots.AxesSubplot at 0x172a8ea9240>
```



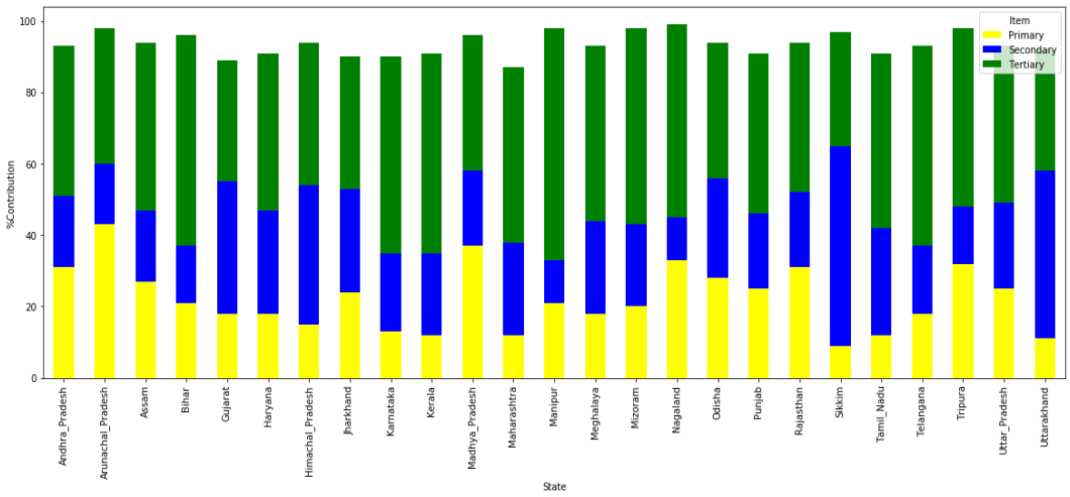
  - Identify the top-5 and the bottom-5 states based on GDP per capita.
    - Top 5 states are:
      - Sikkim
      - Haryana
      - Kerala
      - Uttarakhanda
      - Maharashtra
    - Bottom 5 States are:
      - Jharkhand
      - Assam
      - Manipur
      - Uttar_Pradesh
      - Bihar

  - Find the ratio of highest per capita GDP to the lowest per capita GDP.
    The ratio is 7.08

- Plot the percentage contribution of primary, secondary and tertiary sectors as a percentage of total GDP for all the states.

In [115]:
```
colors = ["yellow", "blue","green"]
pt=pivot_df_merged_filtered_pst.plot.bar(stacked=True, color=colors, figsize=(20,7))
pt.set_ylabel('%Contribution')
```

Out[115]:
```
Text(0,0.5,'%Contribution')
```



- Categorise the states into four categories based on GDP per capita (C1, C2, C3, C4 - C1 would have the highest per capita GDP, C4 the lowest). The quantile values are (0.20,0.5, 0.85, 1), i.e. the states lying between the 85th and the 100th percentile are in C1, those between 50th and 85th percentile are in C2 and so on.

  This can be done in 4 steps:

  1. Fetching only GSDP per capita from dataframe

In [117]:
```
df_merged_filtered_category=df_merged_filtered[df_merged_filtered['Item']=='Per Capita GSDP (Rs.)']
df_merged_filtered_category
```

Out[117]:

	2014-15	Item	S.No.	State
32	104977.0	Per Capita GSDP (Rs.)	17	Andhra_Pradesh
65	112718.0	Per Capita GSDP (Rs.)	17	Arunachal_Pradesh
98	60621.0	Per Capita GSDP (Rs.)	17	Assam
131	33954.0	Per Capita GSDP (Rs.)	17	Bihar
230	141263.0	Per Capita GSDP (Rs.)	17	Gujarat
263	164077.0	Per Capita GSDP (Rs.)	17	Haryana
296	147330.0	Per Capita GSDP (Rs.)	17	Himachal_Pradesh
329	62091.0	Per Capita GSDP (Rs.)	17	Jharkhand
362	145141.0	Per Capita GSDP (Rs.)	17	Karnataka

  2. Creating Percentile column:

In [118]:
```
df_merged_filtered_category['Percentile']=round(100*df_merged_filtered_category['2014-15'].rank(pct=True))
df_merged_filtered_category
```

Out[118]:

	2014-15	Item	S.No.	State	Percentile
32	104977.0	Per Capita GSDP (Rs.)	17	Andhra_Pradesh	52.0
65	112718.0	Per Capita GSDP (Rs.)	17	Arunachal_Pradesh	56.0
98	60621.0	Per Capita GSDP (Rs.)	17	Assam	16.0
131	33954.0	Per Capita GSDP (Rs.)	17	Bihar	4.0
230	141263.0	Per Capita GSDP (Rs.)	17	Gujarat	68.0
263	164077.0	Per Capita GSDP (Rs.)	17	Haryana	96.0
296	147330.0	Per Capita GSDP (Rs.)	17	Himachal_Pradesh	80.0
329	62091.0	Per Capita GSDP (Rs.)	17	Jharkhand	20.0
362	145141.0	Per Capita GSDP (Rs.)	17	Karnataka	72.0
395	154778.0	Per Capita GSDP (Rs.)	17	Kerala	92.0
428	62989.0	Per Capita GSDP (Rs.)	17	Madhya_Pradesh	24.0
461	152853.0	Per Capita GSDP (Rs.)	17	Maharashtra	84.0
494	58442.0	Per Capita GSDP (Rs.)	17	Manipur	12.0

3. Creating categories based on percentile :

In [119]:
```
#Categorizinng states in to c1 c2 c3 c4
def func(num):
    if num<=100 and num>=85:
        return 'C1'
    if num<85 and num>=50:
        return 'C2'
    if num<50 and num>=20:
        return 'C3'
    if num<20:
        return 'C4'

df_merged_filtered_category['Category']=df_merged_filtered_category['Percentile'].apply(func)
df_merged_filtered_category
```

Out[119]:

	2014-15	Item	S.No.	State	Percentile	Category
32	104977.0	Per Capita GSDP (Rs.)	17	Andhra_Pradesh	52.0	C2
65	112718.0	Per Capita GSDP (Rs.)	17	Arunachal_Pradesh	56.0	C2
98	60621.0	Per Capita GSDP (Rs.)	17	Assam	16.0	C4
131	33954.0	Per Capita GSDP (Rs.)	17	Bihar	4.0	C4
230	141263.0	Per Capita GSDP (Rs.)	17	Gujarat	68.0	C2
263	164077.0	Per Capita GSDP (Rs.)	17	Haryana	96.0	C1
296	147330.0	Per Capita GSDP (Rs.)	17	Himachal_Pradesh	80.0	C2
329	62091.0	Per Capita GSDP (Rs.)	17	Jharkhand	20.0	C3
362	145141.0	Per Capita GSDP (Rs.)	17	Karnataka	72.0	C2
395	154778.0	Per Capita GSDP (Rs.)	17	Kerala	92.0	C1
428	62989.0	Per Capita GSDP (Rs.)	17	Madhya_Pradesh	24.0	C3

4. Merging with original dataframe containing all Items:

In [337]:
```
#the newly created categories based on 'per capita GSDP(RS)' is joined with original dataset
#so that coresponding categories created are assigned to original dataframe
df_merged_filtered_subsectors=pd.merge(df_merged_filtered,df_merged_filtered_category_temp,how='inner',on=['State'])
df_merged_filtered_subsectors.head(33)
```

Out[337]:

	2014-15	Item	S.No.	State	Category
0	14819416.0	Agriculture, forestry and fishing	1	Andhra_Pradesh	C2
1	7893514.0	Crops	1.1	Andhra_Pradesh	C2
2	4309078.0	Livestock	1.2	Andhra_Pradesh	C2
3	346160.0	Forestry and logging	1.3	Andhra_Pradesh	C2
4	2270664.0	Fishing and aquaculture	1.4	Andhra_Pradesh	C2
5	1484300.0	Mining and quarrying	2	Andhra_Pradesh	C2
6	16303716.0	Primary	Total	Andhra_Pradesh	C2
7	4672266.0	Manufacturing	3	Andhra_Pradesh	C2
8	1151729.0	Electricity, gas, water supply & other utility...	4	Andhra_Pradesh	C2
9	4664889.0	Construction	5	Andhra_Pradesh	C2
10	10488884.0	Secondary	Total	Andhra_Pradesh	C2
11	4233400.0	Trade, repair, hotels and restaurants	6	Andhra_Pradesh	C2
12	3716000.0	Trade & repair services	6.1	Andhra_Pradesh	C2
13	517400.0	Hotels & restaurants	6.2	Andhra_Pradesh	C2
14	5076984.0	Transport, storage, communication & services r...	7	Andhra_Pradesh	C2
15	424228.0	Railways	7.1	Andhra_Pradesh	C2
16	2816000.0	Road transport	7.2	Andhra_Pradesh	C2

- For each category C1, C2, C3, C4:
  Find the top 3/4/5 sub-sectors (such as agriculture, forestry and fishing, crops, manufacturing etc.) [not primary, secondary and tertiary] which contribute to approx. 80% of the GSDP of each category
  To find top subsectors which contribute to approx. 80% of the GSDP of each category these steps are followed:
  1. Pivoting the Data:

In [143]:
```
pd.set_option('display.max_columns', 33)
df_merged_filtered_subsectors_pvt=df_merged_filtered_subsectors.pivot_table(index='Category',columns='Item',values='2014
df_merged_filtered_subsectors_pvt
```

Out[143]:

Item	Agriculture, forestry and fishing	Air transport	Communication & services related to broadcasting	Construction	Crops	Electricity, gas, water supply & other utility services	Financial services	Fishing and aquaculture	Forestry and logging	Gross State Domestic Product	Hotels & restaurants	Lives
Category												
C1	15684725.0	-1542568.0	2110267.0	12441365.0	8688239.0	2230768.0	4087901.0	819799.0	1195884.0	114065899.0	1177011.0	49808
C2	88427015.0	670852.0	9603839.0	43975718.0	54577463.0	14541227.0	37812475.0	4957690.0	5610080.0	622828765.0	6294893.0	232817
C3	42905337.0	68025.0	3055493.0	14566162.0	28479911.0	3865001.0	5666541.0	942958.0	4512687.0	171730252.0	1108207.0	89697
C4	37288332.0	82692.0	3233800.0	16582963.0	24020576.0	2699870.0	5144263.0	1598736.0	2262886.0	163343179.0	1448303.0	94061

  2. Finding the contribution of Items to the GSDP by dividing all columns by their corresponding GSDP

In [124]:
```
pd.set_option('display.max_columns', 33)
df_merged_filtered_subsectors_pvt_per=round(100*df_merged_filtered_subsectors_pvt.loc[:,'Agriculture, forestry and fishi
df_merged_filtered_subsectors_pvt_per
```

Out[124]:

Item	Agriculture, forestry and fishing	Air transport	Communication & services related to broadcasting	Construction	Crops	Electricity, gas, water supply & other utility services	Financial services	Fishing and aquaculture	Forestry and logging	Gross State Domestic Product	Hotels & restaurants	Livestock	Manufa
Category													
C1	14.0	-1.0	2.0	11.0	8.0	2.0	4.0	1.0	1.0	100.0	1.0	4.0	
C2	14.0	0.0	2.0	7.0	9.0	2.0	6.0	1.0	1.0	100.0	1.0	4.0	
C3	25.0	0.0	2.0	8.0	17.0	2.0	3.0	1.0	3.0	100.0	1.0	5.0	
C4	23.0	0.0	2.0	10.0	15.0	2.0	3.0	1.0	1.0	100.0	1.0	6.0	

3. Selecting only subsectors from the pivoted dataframe:

In [127]:
```
#Selecting only sub sectors as they are required for analysis,sub sub sectors are not considering for analysis
subsectorsLists=['Agriculture, forestry and fishing','Manufacturing','Trade, repair, hotels and restaurants','Real estat
df_merged_filtered_subsectors_pvt_per_subSectors=df_merged_filtered_subsectors_pvt_per[subsectorsLists]
```

In [128]:
```
#sorting by c1
df_merged_filtered_subsectors_pvt_per_subSectors.sort_values(by='C1',axis=1,ascending=False,inplace=True)
df_merged_filtered_subsectors_pvt_per_subSectors
```

```
C:\Users\mahes\Anaconda3\lib\site-packages\ipykernel_launcher.py:2: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame

See the caveats in the documentation: http://pandas.pydata.org/pandas-docs/stable/indexing.html#indexing-view-versus-copy
```

Out[128]:

Item	Manufacturing	Agriculture, forestry and fishing	Trade, repair, hotels and restaurants	Real estate, ownership of dwelling & professional services	Construction	Other services	Transport, storage, communication & services related to broadcasting	Financial services	Public administration	Electricity, gas, water supply & other utility services	Mining and quarrying
Category											
C1	16.0	14.0	13.0	13.0	11.0	8.0	7.0	4.0	3.0	2.0	1.0
C2	17.0	14.0	10.0	15.0	7.0	6.0	6.0	6.0	3.0	2.0	2.0
C3	13.0	25.0	11.0	8.0	8.0	7.0	6.0	3.0	5.0	2.0	6.0
C4	10.0	23.0	12.0	12.0	10.0	7.0	7.0	3.0	6.0	2.0	1.0

4. Finding the top subsectors which contribute approx 80% of GSDP of Categories:

In [129]:
```
#Finding the top subsectors which contribute approx 80% of GSDP of Categories
df_merged_filtered_subsectors_pvt_per_subSectors.loc[:,'Manufacturing':'Transport, storage, communication & services rel
```

Out[129]:
```
Category
C1    82.0
C2    75.0
C3    78.0
C4    81.0
dtype: float64
```

In [130]:
```
#Finding the top subsectors which contribute approx 80% of GSDP of Categories
df_merged_filtered_subsectors_pvt_per_subSectors.loc[:,'Manufacturing':'Financial services'].sum(axis=1)
```

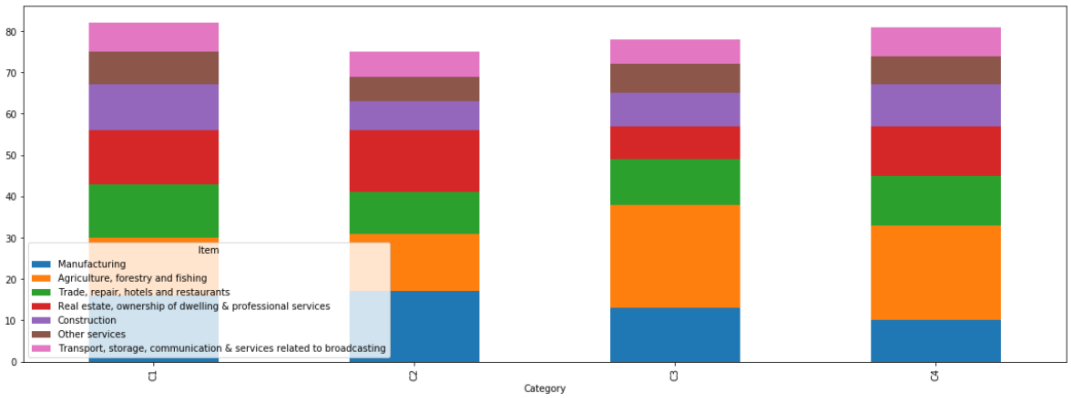
Out[130]:
```
Category
C1    86.0
C2    81.0
C3    81.0
C4    84.0
dtype: float64
```

5. By Observing Step 4 ,Sub Sectors from Manufacturing to Transport, storage, communication & services related to broadcasting (Please refer step 3 for DataFrame contribute approx 80% of GSDP of each category

- Plot the contribution of the sub-sectors as a percentage of the GSDP of each category.

```
In [131]: df_merged_filtered_subsectors_pvt_per_subSectors.loc[:,'Manufacturing':'Transport, storage, communication & services rel
```

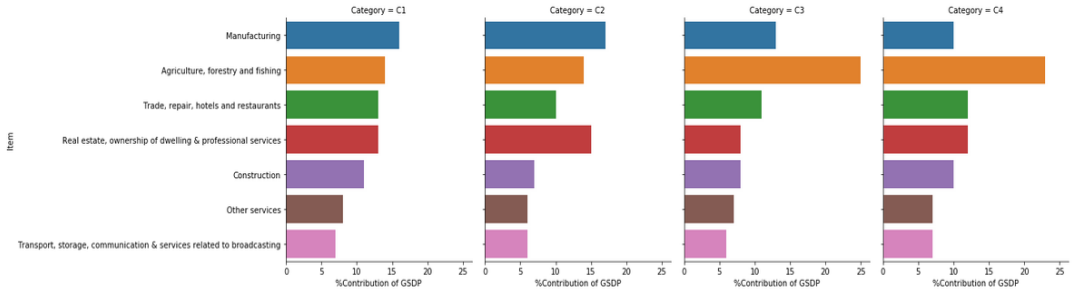
```
Out[131]: <matplotlib.axes._subplots.AxesSubplot at 0x172a974b390>
```



- Now that you have summarised the data in the form of plots, tables etc., try to observe non-obvious insights from it
  - Which sub-sectors do the various categories need to focus on?

```
In [141]: plt.figure(figsize=[20,10])
g=sns.catplot(y='Item',x='%Contribution of GSDP',data=df_merged_filtered_subsectors_per_subSectors,col='Category',kind='
```

```
<Figure size 1440x720 with 0 Axes>
```



Some insights from the above graph

- Transport, storage, communication & services related to broadcasting should be improved in all sectors since that is the least Contributor in all categories.
- C1 and C2 should continue their focus on Manufacturing and Real estate, ownership of dwelling & profession service since they are the most contributing to their economy (Trade, repair, hotels and restaurants is also contributing well to economy but it is almost contributing same for all 4 categories so only deciding factors of top performing states that we can conclude from graph are Manufacturing and Real estate, ownership of dwelling & profession service because they are contributing very less in C3,C4 but contributing high in C1 and C2 to GSDP).
- States GDP depending most on Agriculture,Forestry and fishing are poor performers, these states should concentrate more on manufacturing since it has proved successful for good performing states.

- How GSDP and population are releated for each states?

  The following steps are followed:

  1. Pivot table from dataframe with index as Category and States , columns as GSDP and Population and value as '2014-15' column has to be created

```
In [229]: df_gsdp_pop_pvt=df_merged_filtered_subsectors.loc[(df_merged_filtered_subsectors['Item']=='Gross State Domestic Product'
          df_gsdp_pop_pvt
```

Out[229]:

Category	State	Item	Gross State Domestic Product	Population ('00)
C1	Haryana		43746207.0	266620.0
	Kerala		52600230.0	339843.0
	Sikkim		1520933.0	6330.0
	Uttarakhand		16198529.0	105820.0
C2	Andhra_Pradesh		52646842.0	501510.0
	Arunachal_Pradesh		1676119.0	14870.0
	Gujarat		89502727.0	633590.0
	Himachal_Pradesh		10436879.0	70840.0
	Karnataka		92178806.0	635100.0
	Maharashtra		179212165.0	1172450.0
	Punjab		36801089.0	290673.0
	Tamil_Nadu		109256373.0	745760.0
	Telangana		51117765.0	367660.0
C3	Jharkhand		21710718.0	349660.0
	Madhya_Pradesh		48198169.0	765180.0
	Meghalaya		2440807.0	32020.0
	Mizoram		1155933.0	11833.0

  2. Now index has to be reset in order to get dataframe as shown below:

```
In [230]: df_gsdp_pop_pvt=df_gsdp_pop_pvt.reset_index()
          df_gsdp_pop_pvt.index.name='index'

          df_gsdp_pop_pvt
```

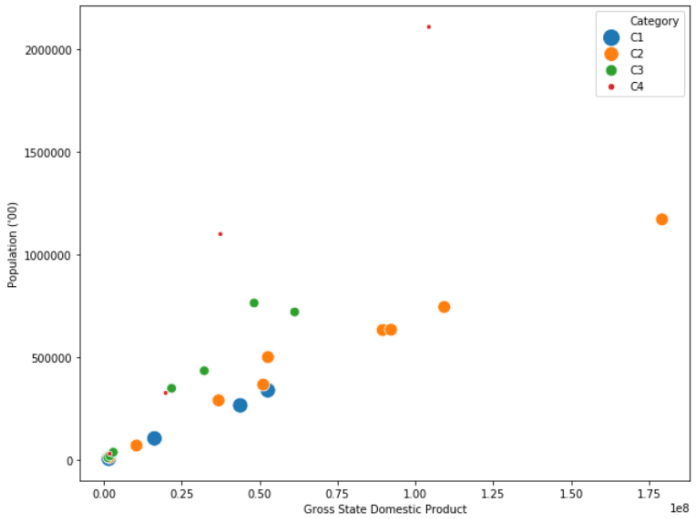
Out[230]:

index	Item	Category	State	Gross State Domestic Product	Population ('00)
0		C1	Haryana	43746207.0	266620.0
1		C1	Kerala	52600230.0	339843.0
2		C1	Sikkim	1520933.0	6330.0
3		C1	Uttarakhand	16198529.0	105820.0
4		C2	Andhra_Pradesh	52646842.0	501510.0
5		C2	Arunachal_Pradesh	1676119.0	14870.0
6		C2	Gujarat	89502727.0	633590.0
7		C2	Himachal_Pradesh	10436879.0	70840.0
8		C2	Karnataka	92178806.0	635100.0
9		C2	Maharashtra	179212165.0	1172450.0
10		C2	Punjab	36801089.0	290673.0
11		C2	Tamil_Nadu	109256373.0	745760.0
12		C2	Telangana	51117765.0	367660.0
13		C3	Jharkhand	21710718.0	349660.0
14		C3	Madhya_Pradesh	48198169.0	765180.0
15		C3	Meghalaya	2440807.0	32020.0
16		C3	Mizoram	1155933.0	11833.0
17		C3	Nagaland	1841424.0	20550.0

  3. Plotting the Scatter plot(since 2 quantitative variables has to be compared therefore choosing scatter plot) :

```
In [378]: plt.figure(figsize=[10,8])
          sns.scatterplot(x='Gross State Domestic Product',y="Population ('00)",data=df_gsdp_pop_pvt,hue='Category',sizes=(20, 200
```

```
Out[378]: <matplotlib.axes._subplots.AxesSubplot at 0x172c8819ef0>
```



4. Insights from above graph
    1. For C1 states GSDP is much lesser than C2 states but since population of C1 states is less ,GDP per capita is more, it means people in C1 states are much richer than C2 states
    2. For most of C3 states population is very less so their GSDP is less
    3. For C4 states their population is high GDP is less so people in those states are very poor
    4. if we observe C1 and C2 we can almost see straight line, so inorder to be top performing state as population increases GSDP should increase linearly (This becomes even more clear by taking correlation between GSDP and Population)

5. Finding the correlation between GSDP, Population for C1 and C2:

```
In [247]: #finding the correleation between GSDP,Population for C1 and C2
          curr_C1_C2.corr()
```

```
Out[247]:
```

Item	Gross State Domestic Product	Population ('00)
Item		
Gross State Domestic Product	1.000000	0.991053
Population ('00)	0.991053	1.000000

6. Finding the correlation between GSDP, Population for C2 and C3:

```
In [256]: #finding the correleation between GSDP,Population for C3 and C4
curr_C3_C4=df_gsdp_pop_pvt.loc[(df_gsdp_pop_pvt['Category']=='C3') | (df_gsdp_pop_pvt['Category']=='C4'),['Gross State I
curr_C3_C4.corr()
```

Out[256]:

Item	Gross State Domestic Product	Population ('00)
Item		
Gross State Domestic Product	1.000000	0.950605
Population ('00)	0.950605	1.000000

6. As we can observe for C1 and C2 correlation between GSDP and Population is 0.99 where as for C2 and C3 correlation between GSDP and Population is 0.95 so in order for states to perform better population should grow linearly with GSDP.

- How are the subsectors correlated with each other and also with GSDP?

Following Steps are followed:

1. Necessary columns has to be selected and pivoted as shown below:

```
In [287]: df_merged_filtered_subsectors_pvt=df_merged_filtered_subsectors_pvt[ListsOfColumns]
df_merged_filtered_subsectors_pvt.drop('TOTAL GSVA at basic prices',axis=1,inplace=True)
df_merged_filtered_subsectors_pvt
```

Out[287]:

Item		Agriculture, forestry and fishing	Manufacturing	Trade, repair, hotels and restaurants	Real estate, ownership of dwelling & professional services	Construction	Other services	Transport, storage, communication & services related to broadcasting	Financial services	Public administration	Ele gas s
Category	**State**										s
C1	**Haryana**	8015238.0	7756921.0	4986319.0	6970183.0	3702571.0	2001581.0	2560623.0	1671486.0	1036377.0	11
	Kerala	5930617.0	4273567.0	8557345.0	7287633.0	7314003.0	5728645.0	4020934.0	2010306.0	2068915.0	4
	Sikkim	137447.0	550697.0	70568.0	75330.0	82058.0	149265.0	47347.0	21079.0	119514.0	2
	Uttarakhand	1601423.0	5866252.0	1743106.0	831307.0	1342733.0	982430.0	1066693.0	385030.0	579409.0	4
C2	**Andhra_Pradesh**	14819416.0	4672266.0	4233400.0	4405409.0	4664889.0	4215389.0	5076984.0	1900863.0	2200897.0	11
	Arunachal_Pradesh	686117.0	26120.0	60421.0	48418.0	147842.0	218728.0	35203.0	25207.0	243867.0	1
	Gujarat	13769969.0	24087538.0	10178713.0	5179502.0	5526017.0	3123413.0	4555910.0	4606644.0	2576195.0	34
	Himachal_Pradesh	1514981.0	2543637.0	615496.0	1125937.0	808256.0	923164.0	552234.0	362521.0	553974.0	7
	Karnataka	11219422.0	12953843.0	8991658.0	24766393.0	6104799.0	5308174.0	5097652.0	4094169.0	2232584.0	14
	Maharashtra	16475655.0	33660294.0	15839100.0	30718051.0	9450211.0	10806363.0	9697246.0	16143324.0	5426991.0	43
	Punjab	9285716.0	4790341.0	4419919.0	3142786.0	2202962.0	3303041.0	1951809.0	2057520.0	1842730.0	9
	Tamil_Nadu	13064238.0	18914794.0	12895842.0	16830213.0	12216718.0	7430115.0	7188320.0	5598498.0	3400800.0	17
	Telangana	7591501.0	6353711.0	6494607.0	9478839.0	2854024.0	4158229.0	3604741.0	3023729.0	1711265.0	7
C3	**Jharkhand**	3211065.0	4114148.0	1991359.0	1656784.0	1789834.0	1375410.0	1470022.0	551441.0	1088325.0	3

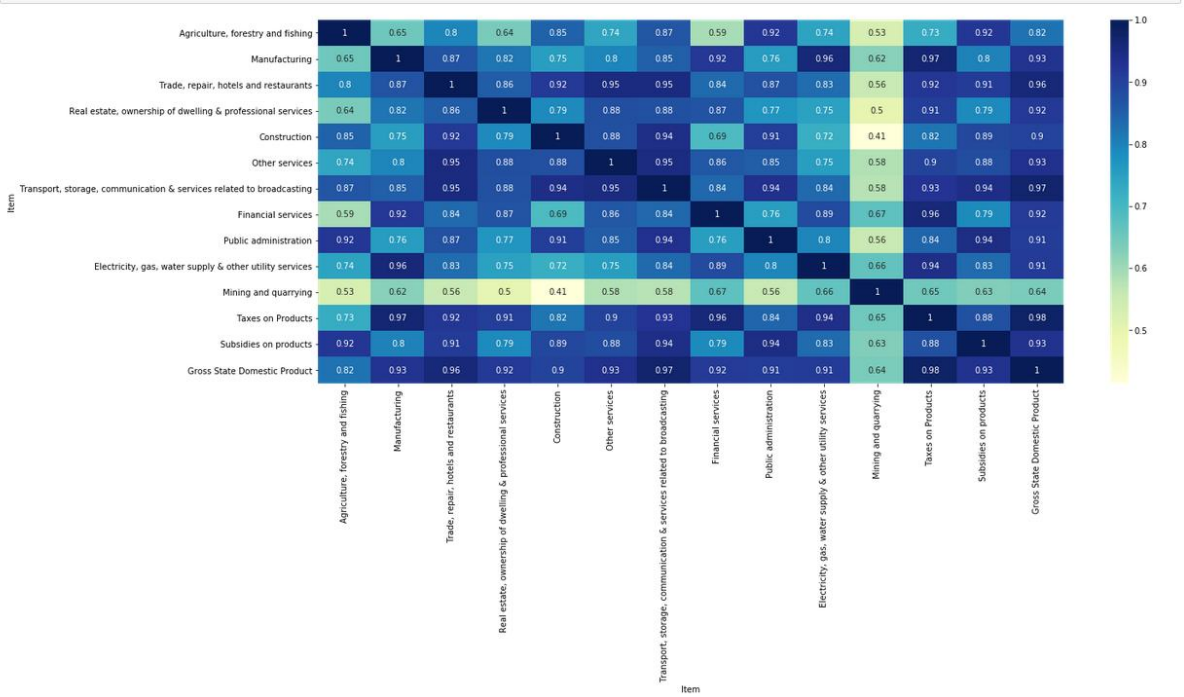
2. Creating the Correlation table :

```
In [290]: #taking correleation of all columns
curr=df_merged_filtered_subsectors_pvt.loc[:,'Agriculture, forestry and fishing':"Gross State Domestic Product"].corr()
curr
```

Out[290]:

Item	Agriculture, forestry and fishing	Manufacturing	Trade, repair, hotels and restaurants	Real estate, ownership of dwelling & professional services	Construction	Other services	Transport, storage, communication & services related to broadcasting	Financial services	Public administration	Electricity, gas, water supply & other utility services	Mining and quarrying
Agriculture, forestry and fishing	1.000000	0.652841	0.795076	0.640536	0.846131	0.740825	0.867156	0.591329	0.919996	0.744057	0.529115
Manufacturing	0.652841	1.000000	0.871131	0.819985	0.754282	0.795766	0.849458	0.917780	0.758989	0.961024	0.623623
Trade, repair, hotels and restaurants	0.795076	0.871131	1.000000	0.855815	0.917197	0.951361	0.951138	0.836858	0.872829	0.831767	0.558506
Real estate, ownership of dwelling & professional services	0.640536	0.819985	0.855815	1.000000	0.790158	0.879256	0.876446	0.871263	0.770606	0.749675	0.503337
Construction	0.846131	0.754282	0.917197	0.790158	1.000000	0.878083	0.942457	0.691477	0.907869	0.721756	0.414746
Other services	0.740825	0.795766	0.951361	0.879256	0.878083	1.000000	0.948624	0.863244	0.854044	0.753682	0.582149
Transport, storage, communication & services related to broadcasting	0.867156	0.849458	0.951138	0.876446	0.942457	0.948624	1.000000	0.839667	0.935274	0.836455	0.578986
Financial services	0.591329	0.917780	0.836858	0.871263	0.691477	0.863244	0.839667	1.000000	0.755256	0.887971	0.672303

3. Plotting the heat map for correlation between columns(heat maps are very efficient in visualising correlation therefore choosing heat maps):



4. Insights from above Heat Map
    1. Trade, repair, hotels and restaurants and Transport, storage, communication & services related to broadcasting are highly correlated with GSDP other subsectors are also at par with the latter except mining and quarrying which is the least correlated with GSDP.
    2. All subsectors are correlated with each other except mining and quarrying. Agriculture, forestry and fishing is correlated with all subsectors except for

Financial services , mining and quarrying, Real estate, ownership of dwelling & professional services, Manufacturing, other Services

3. Agriculture, forestry and fishing is highly correlated with Subsides on Products which is quite logical because Government provides more subsidies for Agriculture, fishing.(Transport, storage, communication & services related to broadcasting and Public administration are also highly correlated to subsides , from previous analysis we came to know that these do not contribute much to the GSDP so considering only Agriculture)
    - In order to give subsides GSDP must be strong, GSDP is highly correlated with Taxes on Products and also from previous analysis we found that C1 and C2 top performing category states has Manufacturing and Real estate, ownership of dwelling & professional services as top contributors for GSDP

    - so in order to grow GSDP main concentration should be on Manufacturing and Real estate, ownership of dwelling & professional services which helps in collecting Taxes(since manufacturing and real estate are both highly correlated with Taxes on Products)which in turn help Agriculture, forestry and fishing to grow because Stronger the GSDP greater the Subsidies(GSDP and Subsides are highly correlated), Subsidies help Agriculture, forestry and fishing to grow(Subsides and help Agriculture, forestry and fishing are highly correlated)

- Finally, provide at least two recommendations for each category to improve the per capita GDP.
  Recommendations for Each Categories

1. C1 and C2 should continue their focus on Manufacturing and Real estate, ownership of dwelling & profession service since they are the most contributing to their economy and helping them to remain top performing states
2. States GDP depending most on Agriculture,Forestry and fishing are poor performers, these states should concentrate more on manufacturing since it has proved successful for good performing states
3. Agriculture, forestry and fishing is highly correlated with Subsides on Products which is quite logical because Government provides more subsidies for Agriculture, fishing.(Transport, storage, communication & services related to broadcasting and Public administration are also highly correlated to subsides , from previous analysis we came to know that these do not contribute much to the GSDP so considering only Agriculture,Forestry and fishing)
    - In order to give subsides GSDP must be strong, GSDP is highly correlated with Taxes on Products and also from previous analysis we found that C1 and C2 top performing category states has Manufacturing and Real estate, ownership of dwelling & professional services as top contributors for GSDP

- so in order to grow GSDP main concentration should be on Manufacturing and Real estate, ownership of dwelling & professional services which helps in collecting Taxes(since manufacturing and real estate are both highly correlated with Taxes on Products)which in turn help Agriculture, forestry and fishing to grow because Stronger the GSDP greater the Subsidies(GSDP and Subsides are highly correlated)

4. C3 and C4 must concentrate on Manufacturing and Real estate, ownership of dwelling & professional services since top performing states C1 and C2 have these as their highest contributors where as in C3 and C4 they are not, so by following footsteps of top performing states C3 and C4 can also reach top
5. C1 and C2 must give subsides from taxes collected from Manufacturing, FinacialServices(highly correlated with taxes) to Public administration, Transport, storage, communication & services related to broadcasting, Agriculture, forestry and fishing(highly correlated with subsidies) so that these subsectors also grow and contribute greater to GSDP (these subsectors are also highly correlated to GSDP)

PART 2: GDP and Education Drop-out Rates

- Analyse if there is any correlation of GDP per capita with dropout rates in education (primary, upper primary and secondary) for the year 2014-2015 for the states. Choose an appropriate plot to conduct this analysis.

  These steps are followed:

  1. Data has to be read from csv file in to DataFrame
  2. Data is cleansed by selecting only necessary columns (Please note : Considering Primary - 2014-2015 just after Primary - 2012-2013 for analysis not Primary - 2014-2015.1).Now the data frame looks as shown below (ignoring null values in DataFrame):

```
In [344]: #Selecting only necessary Columns
          listcolumns=['Level of Education - State','Primary - 2014-2015','Upper Primary - 2014-2015','Secondary - 2014-2015']
          df_dropout_filtered=df_dropout[listcolumns]
          df_dropout_filtered.head()
```

Out[344]:

	Level of Education - State	Primary - 2014-2015	Upper Primary - 2014-2015	Secondary - 2014-2015
0	A & N Islands	1.21	1.69	9.87
1	Andhra Pradesh	4.35	5.20	15.71
2	Arunachal Pradesh	10.89	6.71	17.11
3	Assam	7.44	10.51	27.06
4	Bihar	2.09	4.08	25.90

  3. Renaming Columns appropriately
  4. Using ‘df_merged_filtered_subsectors’ dataframe from previous analysis (Part1-b) to join with part2 dataframe (‘df_merged_filtered_subsectors’ dataframe ie Part1b dataframe contains GSDP per capita which required for analysis therefore joining both dataframes)

     Procedure followed for joining dataframe:

- States names in ‘df_merged_filtered_subsectors’ should match with part2 dataframe, so changing the ‘df_merged_filtered_subsectors’ dataframe names as shown below:

In [340]:
```
#matching the states of both dataframe so that they can be joined latter
def func(val):
    if val=='Arunachal_Pradesh':
        return 'Arunachal Pradesh'
    if val=='Andhra_Pradesh':
        return 'Andhra Pradesh'
    if val=='Himachal_Pradesh':
        return 'Himachal Pradesh'
    if val=='Tamil_Nadu':
        return 'Tamil Nadu'
    if val=='Uttar_Pradesh':
        return 'Uttar Pradesh'
    return val

df_merged_filtered_subsectors['State']=df_merged_filtered_subsectors['State'].apply(func)
df_merged_filtered_subsectors.head()
```

Out[340]:

	2014-15	Item	S.No.	State	Category
0	14819416.0	Agriculture, forestry and fishing	1	Andhra Pradesh	C2
1	7893514.0	Crops	1.1	Andhra Pradesh	C2
2	4309078.0	Livestock	1.2	Andhra Pradesh	C2
3	346160.0	Forestry and logging	1.3	Andhra Pradesh	C2
4	2270664.0	Fishing and aquaculture	1.4	Andhra Pradesh	C2

- Now filtering only GSDP per capita from ‘df_merged_filtered_subsectors’ , removing unnecessary columns from dataframe then it can be merged with part 2 Dataframe, after merging the final dataframe is shown below :

In [357]:
```
#Renaming Column
df_gdpPerCapita_merged.rename(columns={'2014-15':'GSDP per Capita'},inplace=True)
df_gdpPerCapita_merged
```

Out[357]:

	State	Primary - 2014-2015	Upper Primary - 2014-2015	Secondary - 2014-2015	GSDP per Capita	Category
0	Andhra Pradesh	4.35	5.20	15.71	104977.0	C2
1	Arunachal Pradesh	10.89	6.71	17.11	112718.0	C2
2	Assam	7.44	10.51	27.06	60621.0	C4
3	Bihar	2.09	4.08	25.90	33954.0	C4
4	Gujarat	0.76	6.41	25.04	141263.0	C2
5	Haryana	0.41	5.81	15.89	164077.0	C1
6	Himachal Pradesh	0.46	0.87	6.07	147330.0	C2
7	Jharkhand	6.41	8.99	24.00	62091.0	C3
8	Karnataka	2.32	3.85	26.18	145141.0	C2
9	Kerala	NaN	NaN	12.32	154778.0	C1
10	Maharashtra	0.55	1.79	12.87	152853.0	C2
11	Manipur	18.00	4.20	14.38	58442.0	C4
12	Meghalaya	10.34	6.52	20.52	76228.0	C3
13	Mizoram	12.96	4.78	21.88	97687.0	C3
14	Nagaland	19.41	7.92	18.23	89607.0	C3
15	Odisha	2.94	3.81	29.56	73979.0	C3
16	Punjab	1.29	3.22	8.86	126606.0	C2
17	Rajasthan	8.39	3.07	13.48	84837.0	C3
18	Sikkim	4.57	1.57	15.89	240274.0	C1
19	Tamil Nadu	0.46	NaN	8.10	146503.0	C2

5. Finding the correlation between levels of Education:

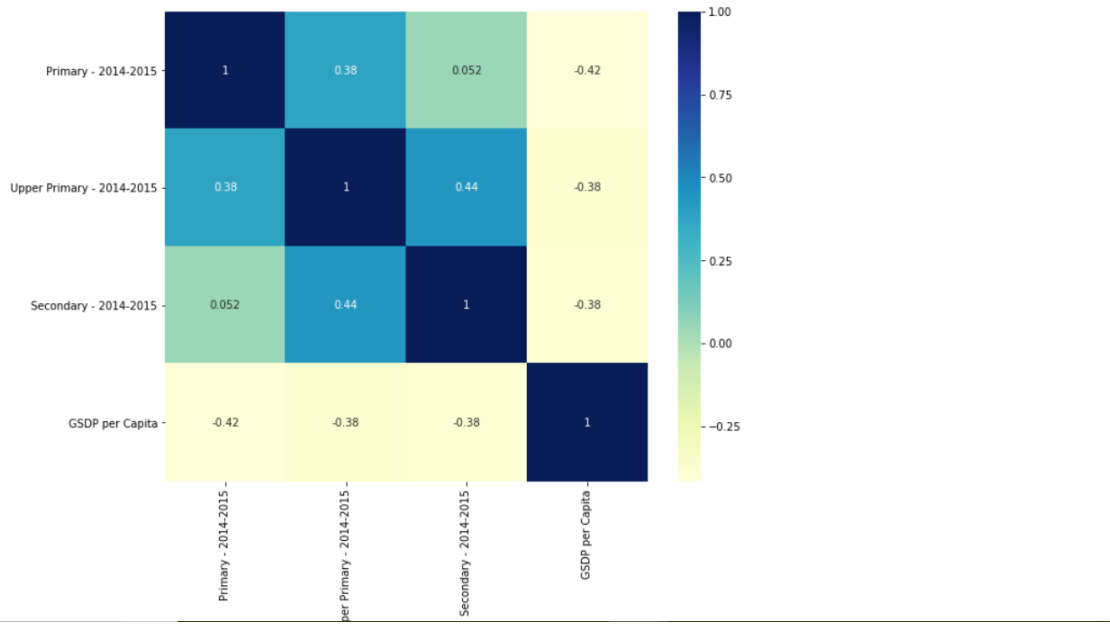
In [366]:
```
df_gdpPerCapita_merged_corr=df_gdpPerCapita_merged[['Primary - 2014-2015','Upper Primary - 2014-2015','Secondary - 2014-
df_gdpPerCapita_merged_corr
```

Out[366]:

	Primary - 2014-2015	Upper Primary - 2014-2015	Secondary - 2014-2015	GSDP per Capita
Primary - 2014-2015	1.000000	0.380254	0.052158	-0.415490
Upper Primary - 2014-2015	0.380254	1.000000	0.442700	-0.382104
Secondary - 2014-2015	0.052158	0.442700	1.000000	-0.384483
GSDP per Capita	-0.415490	-0.382104	-0.384483	1.000000

6. Plotting the heat map for calculated correlation

```
In [368]: plt.figure(figsize=(10,8))
          g=sns.heatmap(df_gdpPerCapita_merged_corr,cmap='YlGnBu',annot=True)
          g=g.set_yticklabels(g.get_xticklabels(),rotation=360)#rotating the y axis labels
```



- Write the key insights you observe from this data:

  Observations from the graph:

  - Primary, upper primary, Secondary dropout rates are negatively correlated with GSDP per Capita which means higher GSDP per capita lower the dropout rates

- Form at least one reasonable hypothesis for the observations from the data

  Hypothesis - Form the above graph we observed that higher GSDP per capita lower the dropout rates, So top performing states C1,C2 must have lower drop rates compared to C3,C4

  Steps to perform this analysis are shown in figure below:

```
In [373]: df_gdpPerCapita_merged_grp=df_gdpPerCapita_merged.groupby('Category')

In [374]: #Taking mean of all columns
          df_gdpPerCapita_merged_grp.mean()
```

Out[374]:

Category	Primary - 2014-2015	Upper Primary - 2014-2015	Secondary - 2014-2015	GSDP per Capita
C1	2.490000	3.690000	14.700000	186376.333333
C2	2.987778	3.793750	15.052222	135158.444444
C3	9.147143	5.297143	22.298571	80255.285714
C4	8.652500	5.372500	19.390000	50616.750000

Hypothesis Testing -

- C1,C2 top performing states which has higher GSDP per Capita has less drop rates in all levels of education compared to poor performing states C3,C4 ,so our hypothesis succeeded in this case.
- When C1 and C2 are compared, C1 has more GSDP than C2 and also C1 has lower drop rates than C2 for all levels of education, so our hypothesis succeeded in this case also.
- When C3 and C4 are compared, C3 has more GSDP than C4 but also C3 has higher drop rates than C4 for primary and Secondary levels of education, so our hypothesis Fails in this case.