Part I-A: GDP Analysis of the Indian States

Read csv file in Python

In [13]:

```
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
Data1A = pd.read_csv (r'./Data/Data IA.csv')
print(Data1A)
```

	Items	Description	Duration	Andhra Pradesh	Arunachal Pradesh	Assam	Bihar	Chhattisgarh	Goa	Gujarat	Haryana	...	Telangana	Tripura	Uttar Pradesh	Uttarakhand	West Bengal1	Andaman & Nicobar Islands	Chandigarh
0	GSDP - CURRENT PRICES (` in Crore)		2011-12	379402.00	11063.00	143175.00	247144.00	158074.00	42367.00	615606.00	297539.00	...	359433.00	19208.00	724049.00	115523.00	NaN	3979.00	18768.00
1	GSDP - CURRENT PRICES (` in Crore)		2012-13	411404.00	12547.00	156864.00	282368.00	177511.00	38120.00	724495.00	347032.00	...	401493.00	21663.00	822903.00	131835.00	NaN	4421.00	21609.00
2	GSDP - CURRENT PRICES (` in Crore)		2013-14	464272.00	14602.00	177745.00	317101.00	206690.00	35921.00	807623.00	400662.00	...	452186.00	25593.00	944146.00	149817.00	NaN	5159.00	24787.00
3	GSDP - CURRENT PRICES (` in Crore)		2014-15	526468.00	16761.00	198098.00	373920.00	234982.00	40633.00	895027.00	437462.00	...	511178.00	29667.00	1043371.00	161985.00	NaN	5721.00	27844.00
4	GSDP - CURRENT PRICES (` in Crore)		2015-16	609934.00	18784.00	224234.00	413503.00	260776.00	45002.00	994316.00	485184.00	...	575631.00	NaN	1153795.00	184091.00	NaN	NaN	30304.00
5	GSDP - CURRENT PRICES (` in Crore)		2016-17	699307.00	NaN	NaN	NaN	290140.00	NaN	NaN	547396.00	...	654294.00	NaN	NaN	NaN	NaN	NaN	NaN
6		(% Growth over previous year)	2012-13	8.43	13.41	9.56	14.25	12.30	-10.02	17.69	16.63	...	11.70	12.78	13.65	14.12	NaN	11.13	15.14
7		(% Growth over previous year)	2013-14	12.85	16.38	13.31	12.30	16.44	-5.77	11.47	15.45	...	12.63	18.14	14.73	13.64	NaN	16.68	14.71
8		(% Growth over previous year)	2014-15	13.40	14.79	11.45	17.92	13.69	13.12	10.82	9.18	...	13.05	15.92	10.51	8.12	NaN	10.89	12.33
9		(% Growth over previous year)	2015-16	15.85	12.07	13.19	10.59	10.98	10.75	11.09	10.91	...	12.61	NaN	10.58				
10		(% Growth over previous year)	2016-17	14.65	NaN	NaN	NaN	11.26	NaN	NaN	12.82	...	13.67	NaN	NaN				

```
9         13.65           NaN                    NaN        8.84
10          NaN           NaN                    NaN         NaN

        Delhi  Puducherry  All_India GDP
0   343767.00    16818.00     8736039.00
1   391238.00    18875.00     9946636.00
2   443783.00    21870.00    11236635.00
3   492424.00    24089.00    12433749.00
4   551963.00    26533.00    13675331.00
5   622385.00    29557.00    15251028.00
6       13.81       12.23          13.86
7       13.43       15.87          12.97
8       10.96       10.14          10.65
9       12.09       10.15           9.99
10      12.76       11.40          11.52

[11 rows x 36 columns]
```

In [14]:

```
#Remove trailing spaces from column headers like 'Andhra Pradesh ' to 'Andhra Pradesh'
Data1A.rename(columns=lambda x: x.strip(), inplace=True)
Data1A.head()
```

Out[14]:

	Items Description	Duration	Andhra Pradesh	Arunachal Pradesh	Assam	Bihar	Chhattisgarh	Goa	G
0	GSDP - CURRENT PRICES (` in Crore)	2011-12	379402.0	11063.0	143175.0	247144.0	158074.0	42367.0	61
1	GSDP - CURRENT PRICES (` in Crore)	2012-13	411404.0	12547.0	156864.0	282368.0	177511.0	38120.0	72
2	GSDP - CURRENT PRICES (` in Crore)	2013-14	464272.0	14602.0	177745.0	317101.0	206690.0	35921.0	80
3	GSDP - CURRENT PRICES (` in Crore)	2014-15	526468.0	16761.0	198098.0	373920.0	234982.0	40633.0	89
4	GSDP - CURRENT PRICES (` in Crore)	2015-16	609934.0	18784.0	224234.0	413503.0	260776.0	45002.0	99

5 rows × 36 columns

In [15]:

```
# removing WestBengal data is insufficient,it can be ignored
Data1A.drop(['West Bengal1'], axis = 1, inplace = True)
Data1A.head()
```

Out[15]:

	Items Description	Duration	Andhra Pradesh	Arunachal Pradesh	Assam	Bihar	Chhattisgarh	Goa	G
0	GSDP - CURRENT PRICES (` in Crore)	2011-12	379402.0	11063.0	143175.0	247144.0	158074.0	42367.0	61
1	GSDP - CURRENT PRICES (` in Crore)	2012-13	411404.0	12547.0	156864.0	282368.0	177511.0	38120.0	72
2	GSDP - CURRENT PRICES (` in Crore)	2013-14	464272.0	14602.0	177745.0	317101.0	206690.0	35921.0	80
3	GSDP - CURRENT PRICES (` in Crore)	2014-15	526468.0	16761.0	198098.0	373920.0	234982.0	40633.0	89
4	GSDP - CURRENT PRICES (` in Crore)	2015-16	609934.0	18784.0	224234.0	413503.0	260776.0	45002.0	99

5 rows × 35 columns

In [16]:

```
#Set index as 'Duration' column
Data1A.set_index(['Duration'], inplace = True)

# removing 2016-17 rows as data is insufficient can be ignored
Data1A.drop(['2016-17'], inplace = True)

# replacing Nan with 0
Data1A.fillna(0, inplace=True)

Data1A.head()
```

Out[16]:

Duration	Items Description	Andhra Pradesh	Arunachal Pradesh	Assam	Bihar	Chhattisgarh	Goa	Guja
2011-12	GSDP - CURRENT PRICES (` in Crore)	379402.0	11063.0	143175.0	247144.0	158074.0	42367.0	61560
2012-13	GSDP - CURRENT PRICES (` in Crore)	411404.0	12547.0	156864.0	282368.0	177511.0	38120.0	72449
2013-14	GSDP - CURRENT PRICES (` in Crore)	464272.0	14602.0	177745.0	317101.0	206690.0	35921.0	80762
2014-15	GSDP - CURRENT PRICES (` in Crore)	526468.0	16761.0	198098.0	373920.0	234982.0	40633.0	89502
2015-16	GSDP - CURRENT PRICES (` in Crore)	609934.0	18784.0	224234.0	413503.0	260776.0	45002.0	99431

5 rows × 34 columns

In [17]:

```
#Create DataFrame Growth_Data1A with data of Growth Percentage
Growth_Data1A = Data1A[Data1A["Items  Description"] == '(% Growth over previous year)']
Growth_Data1A.head()
```

Out[17]:

Duration	Items Description	Andhra Pradesh	Arunachal Pradesh	Assam	Bihar	Chhattisgarh	Goa	Gujarat	Har
2012-13	(% Growth over previous year)	8.43	13.41	9.56	14.25	12.30	-10.02	17.69	
2013-14	(% Growth over previous year)	12.85	16.38	13.31	12.30	16.44	-5.77	11.47	
2014-15	(% Growth over previous year)	13.40	14.79	11.45	17.92	13.69	13.12	10.82	
2015-16	(% Growth over previous year)	15.85	12.07	13.19	10.59	10.98	10.75	11.09	

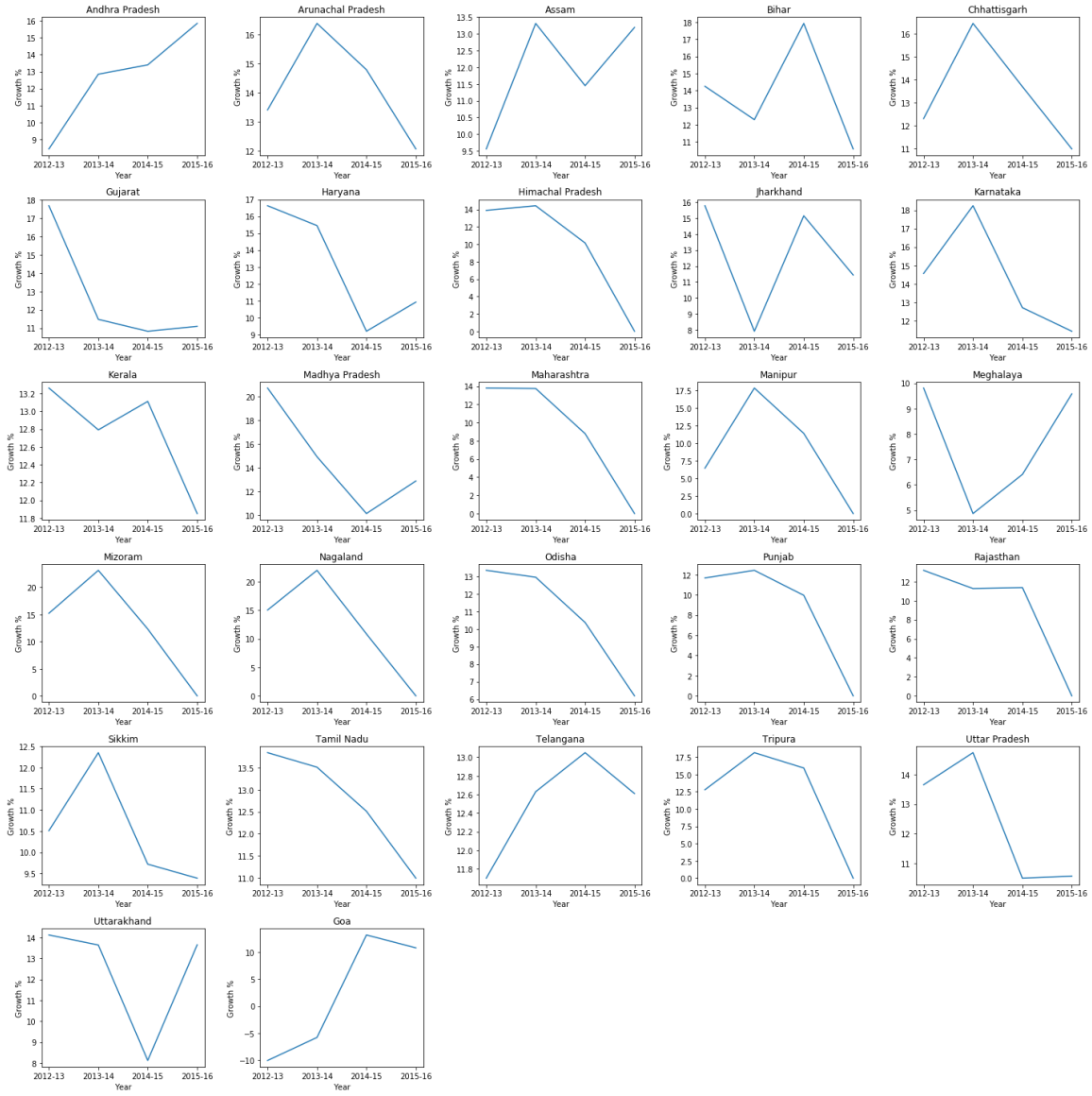
4 rows × 34 columns

In [18]:

```
#Plotting Subplots to visualize GSDP

#Assumption - Ignoring J&K as its a U.T. now

plt.figure(figsize=(20, 20))

plt.subplot(6, 5,1)
plt.title('Andhra Pradesh')
plt.xlabel('Year')
plt.ylabel('Growth %')
sns.lineplot(data = Growth_Data1A['Andhra Pradesh'])

plt.subplot(6, 5,2)
plt.title('Arunachal Pradesh')
plt.xlabel('Year')
plt.ylabel('Growth %')
sns.lineplot(data = Growth_Data1A['Arunachal Pradesh'])

plt.subplot(6, 5,3)
plt.title('Assam')
plt.xlabel('Year')
plt.ylabel('Growth %')
sns.lineplot(data = Growth_Data1A['Assam'])

plt.subplot(6, 5,4)
plt.title('Bihar')
plt.xlabel('Year')
plt.ylabel('Growth %')
sns.lineplot(data = Growth_Data1A['Bihar'])

plt.subplot(6, 5,5)
plt.title('Chhattisgarh')
plt.xlabel('Year')
plt.ylabel('Growth %')
sns.lineplot(data = Growth_Data1A['Chhattisgarh'])

plt.subplot(6, 5,6)
plt.title('Gujarat')
plt.xlabel('Year')
plt.ylabel('Growth %')
sns.lineplot(data = Growth_Data1A['Gujarat'])

plt.subplot(6, 5,7)
plt.title('Haryana')
plt.xlabel('Year')
plt.ylabel('Growth %')
sns.lineplot(data = Growth_Data1A['Haryana'])

plt.subplot(6, 5,8)
plt.title('Himachal Pradesh')
plt.xlabel('Year')
plt.ylabel('Growth %')
sns.lineplot(data = Growth_Data1A['Himachal Pradesh'])

#Ignoring J&K as its a U.T. now
# plt.subplot(6, 5,9)
# plt.title('Jammu & Kashmir')
# plt.xlabel('Year')
# plt.ylabel('Growth %')
```

```
# sns.lineplot(data = Growth_Data1A['Jammu & Kashmir'])

plt.subplot(6, 5,9)
plt.title('Jharkhand')
plt.xlabel('Year')
plt.ylabel('Growth %')
sns.lineplot(data = Growth_Data1A['Jharkhand'])

plt.subplot(6, 5,10)
plt.title('Karnataka')
plt.xlabel('Year')
plt.ylabel('Growth %')
sns.lineplot(data = Growth_Data1A['Karnataka'])

plt.subplot(6, 5,11)
plt.title('Kerala')
plt.xlabel('Year')
plt.ylabel('Growth %')
sns.lineplot(data = Growth_Data1A['Kerala'])

plt.subplot(6, 5,12)
plt.title('Madhya Pradesh')
plt.xlabel('Year')
plt.ylabel('Growth %')
sns.lineplot(data = Growth_Data1A['Madhya Pradesh'])

plt.subplot(6, 5,13)
plt.title('Maharashtra')
plt.xlabel('Year')
plt.ylabel('Growth %')
sns.lineplot(data = Growth_Data1A['Maharashtra'])

plt.subplot(6, 5,14)
plt.title('Manipur')
plt.xlabel('Year')
plt.ylabel('Growth %')
sns.lineplot(data = Growth_Data1A['Manipur'])

plt.subplot(6, 5,15)
plt.title('Meghalaya')
plt.xlabel('Year')
plt.ylabel('Growth %')
sns.lineplot(data = Growth_Data1A['Meghalaya'])

plt.subplot(6, 5,16)
plt.title('Mizoram')
plt.xlabel('Year')
plt.ylabel('Growth %')
sns.lineplot(data = Growth_Data1A['Mizoram'])

plt.subplot(6, 5,17)
plt.title('Nagaland')
plt.xlabel('Year')
plt.ylabel('Growth %')
sns.lineplot(data = Growth_Data1A['Nagaland'])

plt.subplot(6, 5,18)
plt.title('Odisha')
plt.xlabel('Year')
plt.ylabel('Growth %')
sns.lineplot(data = Growth_Data1A['Odisha'])
```

```
plt.subplot(6, 5,19)
plt.title('Punjab')
plt.xlabel('Year')
plt.ylabel('Growth %')
sns.lineplot(data = Growth_Data1A['Punjab'])

plt.subplot(6, 5,20)
plt.title('Rajasthan')
plt.xlabel('Year')
plt.ylabel('Growth %')
sns.lineplot(data = Growth_Data1A['Rajasthan'])

plt.subplot(6, 5,21)
plt.title('Sikkim')
plt.xlabel('Year')
plt.ylabel('Growth %')
sns.lineplot(data = Growth_Data1A['Sikkim'])

plt.subplot(6, 5,22)
plt.title('Tamil Nadu')
plt.xlabel('Year')
plt.ylabel('Growth %')
sns.lineplot(data = Growth_Data1A['Tamil Nadu'])

plt.subplot(6, 5,23)
plt.title('Telangana')
plt.xlabel('Year')
plt.ylabel('Growth %')
sns.lineplot(data = Growth_Data1A['Telangana'])

plt.subplot(6, 5,24)
plt.title('Tripura')
plt.xlabel('Year')
plt.ylabel('Growth %')
sns.lineplot(data = Growth_Data1A['Tripura'])

plt.subplot(6, 5,25)
plt.title('Uttar Pradesh')
plt.xlabel('Year')
plt.ylabel('Growth %')
sns.lineplot(data = Growth_Data1A['Uttar Pradesh'])

plt.subplot(6, 5,26)
plt.title('Uttarakhand')
plt.xlabel('Year')
plt.ylabel('Growth %')
sns.lineplot(data = Growth_Data1A['Uttarakhand'])

plt.subplot(6, 5,27)
plt.title('Goa')
plt.xlabel('Year')
plt.ylabel('Growth %')
sns.lineplot(data = Growth_Data1A['Goa'])

plt.tight_layout()
```

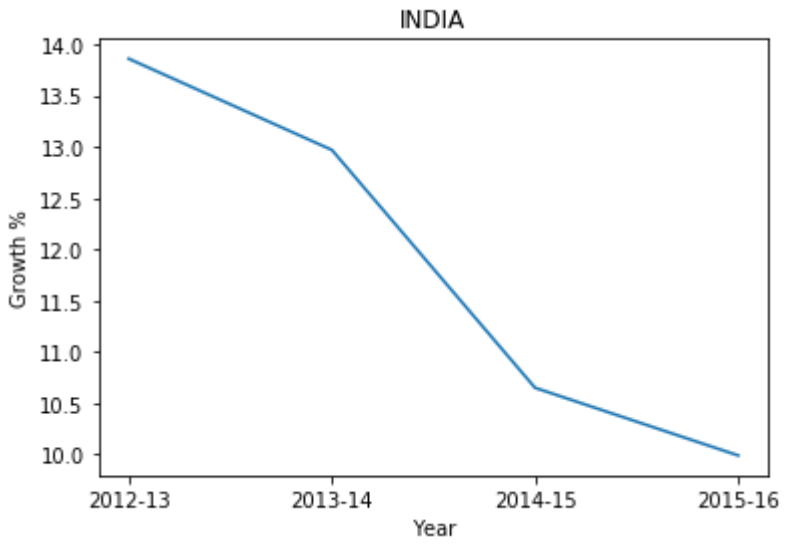

Andhra Pradesh
Arunachal Pradesh
Assam
Bihar
Chhattisgarh
Gujarat
Haryana
Himachal Pradesh
Jharkhand
Karnataka
Kerala
Madhya Pradesh
Maharashtra
Manipur
Meghalaya
Mizoram
Nagaland
Odisha
Punjab
Rajasthan
Sikkim
Tamil Nadu
Telangana
Tripura
Uttar Pradesh
Uttarakhand
Goa
Growth %
Year
2012-13
2013-14
2014-15
2015-16

In [19]:

```
#Line graph for the nation
plt.title('INDIA')
plt.xlabel('Year')
plt.ylabel('Growth %')
sns.lineplot(data = Growth_Data1A['All_India GDP'])
```

Out[19]:

```
<matplotlib.axes._subplots.AxesSubplot at 0x295f3e76668>
```



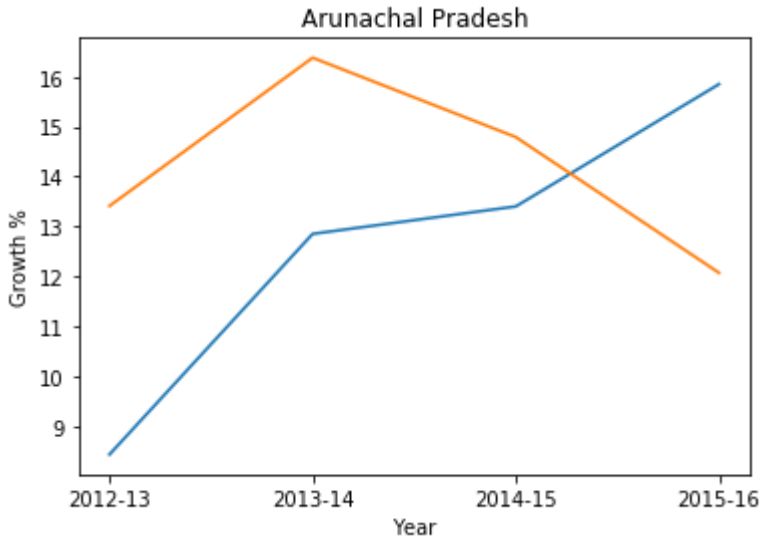
How will you compare the growth rates of any two states?

Ans - By plotting lineplots between 2 states

Example Plot is as below-

In [20]:

```
plt.title('Andhra Pradesh')
plt.xlabel('Year')
plt.ylabel('Growth %')
sns.lineplot(data = Growth_Data1A['Andhra Pradesh'])

plt.title('Arunachal Pradesh')
sns.lineplot(data = Growth_Data1A['Arunachal Pradesh'])
plt.show()
```



Which states have been growing consistently fast, and which ones have been struggling? Rank top 3 fastest and 3 slowest-growing states.

Top 3 Fastest growing states - Andhra Pradesh, Assam, Goa

Top 3 Slowest growing states - Gujarat, Tamil Nadu, Odisha

What is the Nation's growth rate?

India's growth rate during 2015-16 is 9.99%

India's growth rate during 2011-17 is 74.57%

What has been the growth rate of your home state, and how does it compare to the national growth rate?

Growth rate of Karnataka is 11.42% during 2015-16

Growth rate of Karnataka and the Nation are Positively correlated as 0.74

Please look at the Graph below

In [9]:

```
KA_India = Growth_Data1A[['Karnataka', 'All_India GDP']]
cor = KA_India.corr()
sns.heatmap(cor, annot=True)
```

Out[9]:

```
<matplotlib.axes._subplots.AxesSubplot at 0x295f51b2da0>
```

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

In [21]:

```
Data1A = pd.read_csv (r'C:\Users\kzrk386\Desktop\upgrad\Data\Data IA.csv')
pd.options.mode.chained_assignment = None
GDP_Data = Data1A[(Data1A["Items  Description"] == 'GSDP - CURRENT PRICES (` in Crore)'
) & (Data1A["Duration"] == '2015-16')]
#Dropping unwanted columns
GDP_Data.drop(['All_India GDP'],axis = 1, inplace = True)
GDP_Data.drop(['Duration'],axis = 1, inplace = True)
GDP_Data.drop(['Items  Description'],axis = 1, inplace = True)
#Removing Union Territories
GDP_Data.drop(['Chandigarh'],axis = 1, inplace = True)
GDP_Data.drop(['Puducherry'],axis = 1, inplace = True)
GDP_Data.drop(['Jammu & Kashmir'],axis = 1, inplace = True)
GDP_Data.drop(['Delhi'],axis = 1, inplace = True)

#Drop Nan as 0
GDP_Data.dropna(axis = 1,inplace= True)
#Transposing data for easier plots
GDP_Data = GDP_Data.T.reset_index()
#Adding column names
GDP_Data.columns= ["States","GDSP"]
#Sorting data ascending
GDP_Data = GDP_Data.sort_values(by ='GDSP', ascending=True)
print(GDP_Data)
```

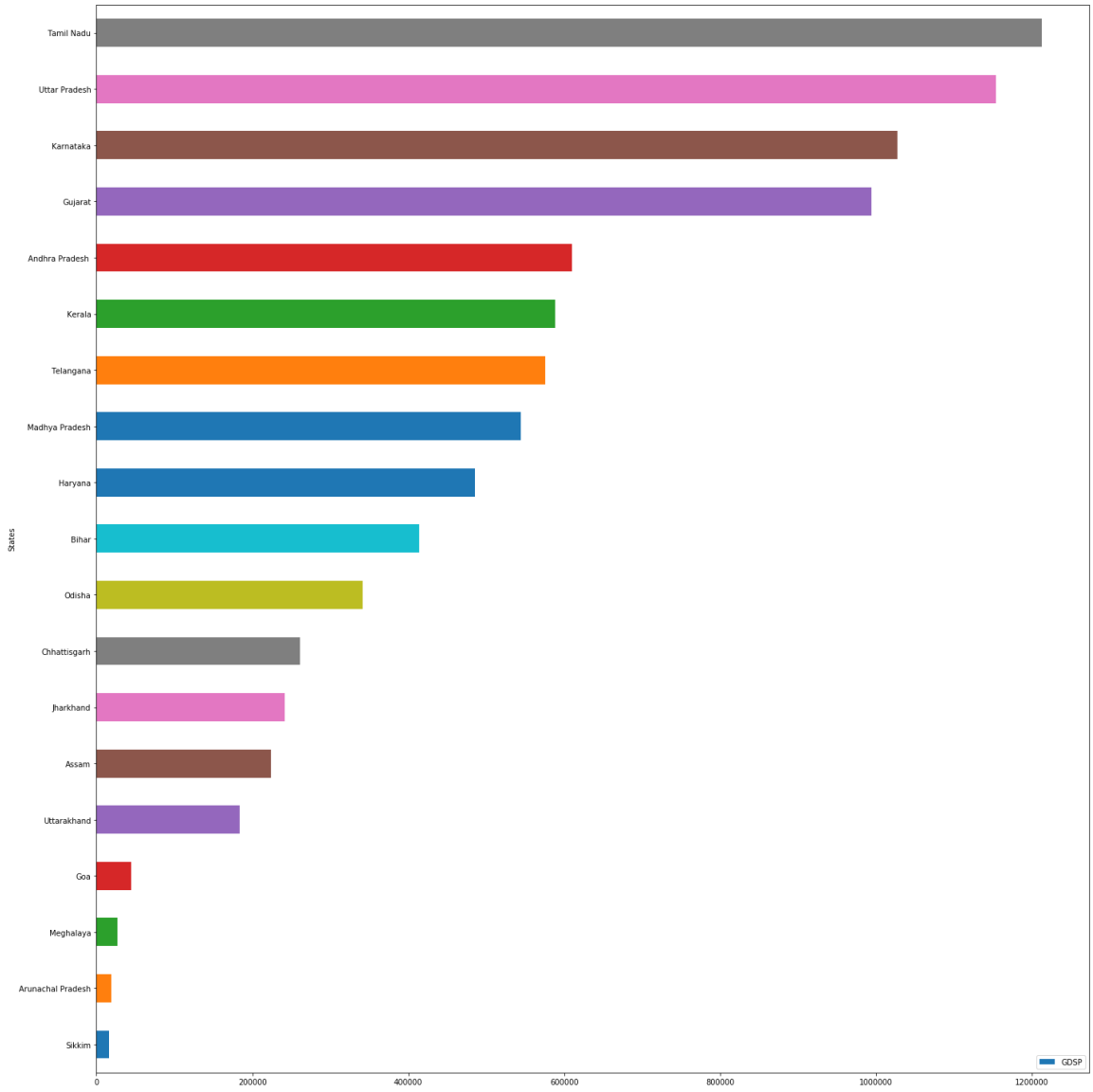
```
               States       GDSP
14             Sikkim    16637.0
1   Arunachal Pradesh    18784.0
12          Meghalaya    26745.0
5                 Goa    45002.0
18        Uttarakhand   184091.0
2               Assam   224234.0
8           Jharkhand   241955.0
4        Chhattisgarh   260776.0
13             Odisha   341887.0
3               Bihar   413503.0
7             Haryana   485184.0
11     Madhya Pradesh   543975.0
16          Telangana   575631.0
10             Kerala   588337.0
0     Andhra Pradesh    609934.0
6               Gujarat   994316.0
9           Karnataka  1027068.0
17      Uttar Pradesh  1153795.0
15         Tamil Nadu  1212668.0
```

Which Plot will you use for this? Why? (Remeber to plot the graph in a way such as it is easier to read and compare)

I will use the horizontal barplot because of its readability and simplicity

In [22]:

```
GDP_Data.plot.barh(x = "States", y = "GDSP", figsize=(20,20))
plt.tight_layout()
```



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

Top 5 states- States GDSP

Andhra Pradesh 609934.0

Gujarat 994316.0

Karnataka 1027068.0

Uttar Pradesh 1153795.0

Tamil Nadu 1212668.0

Bottom 5 states- States GDSP

Sikkim 16637.0

Arunachal Pradesh 18784.0

Meghalaya 26745.0

Goa 45002.0

Uttarakhand 184091.0

In [23]:

```
#Bottom 5 states based on GDP
print(GDP_Data[0:5])
#Top 5 states based on GDP
print(GDP_Data[-5:])
```

```
               States      GDSP
14             Sikkim   16637.0
1   Arunachal Pradesh   18784.0
12          Meghalaya   26745.0
5                Goa    45002.0
18         Uttarakhand  184091.0
             States       GDSP
0   Andhra Pradesh    609934.0
6           Gujarat   994316.0
9          Karnataka  1027068.0
17    Uttar Pradesh  1153795.0
15       Tamil Nadu  1212668.0
```

What insights can you draw from this graph? What states are performing poorly?

Tamil Nadu, Uttar Pradesh, Karnataka & Gujarat contributes most towards the GDP in the year 2015-16

While Meghalaya, Arunachal Pradesh, Goa & Sikkim contributes the least towards the GDP