Project-4 FDI Analysis

In [1]:
```
# Submitted by: Umang Parti
# Submitted to: Unified Mentor
```

In [2]:
```
# importing libraries
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
from IPython.display import FileLink
```

In [3]:
```
# importing data
fdi_df = pd.read_csv(r"C:\Users\umang\Desktop\unified mentor\project - 4\FDI data.csv")
print(fdi_df.head())
```

```
                    Sector  2000-01  2001-02  2002-03  2003-04  2004-05  \
0  METALLURGICAL INDUSTRIES    22.69    14.14    36.61     8.11   200.38
1                    MINING     1.32     6.52    10.06    23.48     9.92
2                     POWER    89.42   757.44    59.11    27.09    43.37
3   NON-CONVENTIONAL ENERGY     0.00     0.00     1.70     4.14     1.27
4           COAL PRODUCTION     0.00     0.00     0.00     0.04     0.00

   2005-06  2006-07  2007-08  2008-09  2009-10  2010-11  2011-12  2012-13  \
0   149.13   169.94  1175.75   959.94   419.88  1098.14  1786.14  1466.23
1     7.40     6.62   444.36    34.16   174.40    79.51   142.65    57.89
2    72.69   157.15   988.68   907.66  1271.79  1271.77  1652.38   535.68
3     1.35     2.44    58.82   125.88   622.52   214.40   452.17  1106.52
4     9.14     1.30    14.08     0.22     0.00     0.00     0.00     0.00

   2013-14  2014-15  2015-16  2016-17
0   567.63   359.34   456.31  1440.18
1    12.73   684.39   520.67    55.75
2  1066.08   707.04   868.80  1112.98
3   414.25   615.95   776.51   783.57
4     2.96     0.00     0.00     0.00
```

Exploratory Analysis

In [4]:
```
print(fdi_df.shape)
```

```
(63, 18)
```

In [5]:
```
fdi_df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 63 entries, 0 to 62
Data columns (total 18 columns):
 #   Column   Non-Null Count  Dtype  
---  ------   --------------  -----  
 0   Sector   63 non-null     object 
 1   2000-01  63 non-null     float64
 2   2001-02  63 non-null     float64
 3   2002-03  63 non-null     float64
 4   2003-04  63 non-null     float64
 5   2004-05  63 non-null     float64
 6   2005-06  63 non-null     float64
 7   2006-07  63 non-null     float64
 8   2007-08  63 non-null     float64
 9   2008-09  63 non-null     float64
 10  2009-10  63 non-null     float64
 11  2010-11  63 non-null     float64
 12  2011-12  63 non-null     float64
 13  2012-13  63 non-null     float64
 14  2013-14  63 non-null     float64
 15  2014-15  63 non-null     float64
 16  2015-16  63 non-null     float64
 17  2016-17  63 non-null     float64
dtypes: float64(17), object(1)
memory usage: 9.0+ KB
```

In [6]:
```
fdi_df.isnull().sum()
```

Out[6]:
```
Sector     0
2000-01    0
2001-02    0
2002-03    0
2003-04    0
2004-05    0
2005-06    0
2006-07    0
2007-08    0
2008-09    0
2009-10    0
2010-11    0
2011-12    0
2012-13    0
2013-14    0
2014-15    0
2015-16    0
2016-17    0
dtype: int64
```

In [7]:
```
# check duplicated data
fdi_df.duplicated().sum()
```

Out[7]:
```
0
```

- We have 63 sectors and 17 years of data
- Only 1 column containing sectors in Object Others are Float
- Our data neither has any null values nor duplicate records

Descriptive Analysis

In [8]:
```
print(fdi_df.describe().T)
```

	count	mean	std	min	25%	50%	75%	max
2000-01	63.0	37.757302	112.227860	0.0	0.000	4.03	23.510	832.07
2001-02	63.0	63.931587	157.878737	0.0	0.000	5.07	44.830	873.23
2002-03	63.0	42.925714	86.606439	0.0	0.200	11.01	36.555	419.96
2003-04	63.0	34.727778	67.653735	0.0	0.215	6.37	38.660	368.32
2004-05	63.0	51.090317	101.934873	0.0	0.715	9.09	43.205	527.90
2005-06	63.0	87.932540	206.436967	0.0	1.230	22.62	63.855	1359.97
2006-07	63.0	198.281905	686.783115	0.0	4.160	25.82	108.325	4713.78
2007-08	63.0	390.085714	1026.249935	0.0	9.950	58.82	279.270	6986.17
2008-09	63.0	498.348571	1134.649040	0.0	11.950	84.88	383.320	6183.49
2009-10	63.0	410.069524	926.814626	0.0	7.880	69.74	341.595	5466.13
2010-11	63.0	339.413810	627.141139	0.0	8.430	58.07	304.280	3296.09
2011-12	63.0	557.472698	1031.474056	0.0	22.720	129.36	593.525	5215.98
2012-13	63.0	355.930000	778.091368	0.0	15.115	95.41	288.025	4832.98
2013-14	63.0	385.703492	658.429944	0.0	16.610	113.78	473.060	3982.89
2014-15	63.0	490.959841	837.787060	0.0	33.800	177.22	595.390	4443.26
2015-16	63.0	634.936349	1335.307706	0.0	30.000	159.13	519.070	6889.46
2016-17	63.0	690.131111	1411.965354	0.0	19.905	110.86	741.220	8684.07

- We notice that Mean FDI rose significantly from 37 in 2000-01 to 690 in 2016-17
- We Maximum FDI is 8684 in the year 2016-17
- The Median Value ie 50th percentile is less than Mean value for all years dipicting that data is Negatively skewed.

Sector-Wise Analysis of Total FDI

In [9]:
```
# copying the columns in new table
fdi_df_total= fdi_df.copy()
```

In [10]:
```
# selecting the columns
fdi_columns = fdi_df.columns[1:]
fdi_columns
```

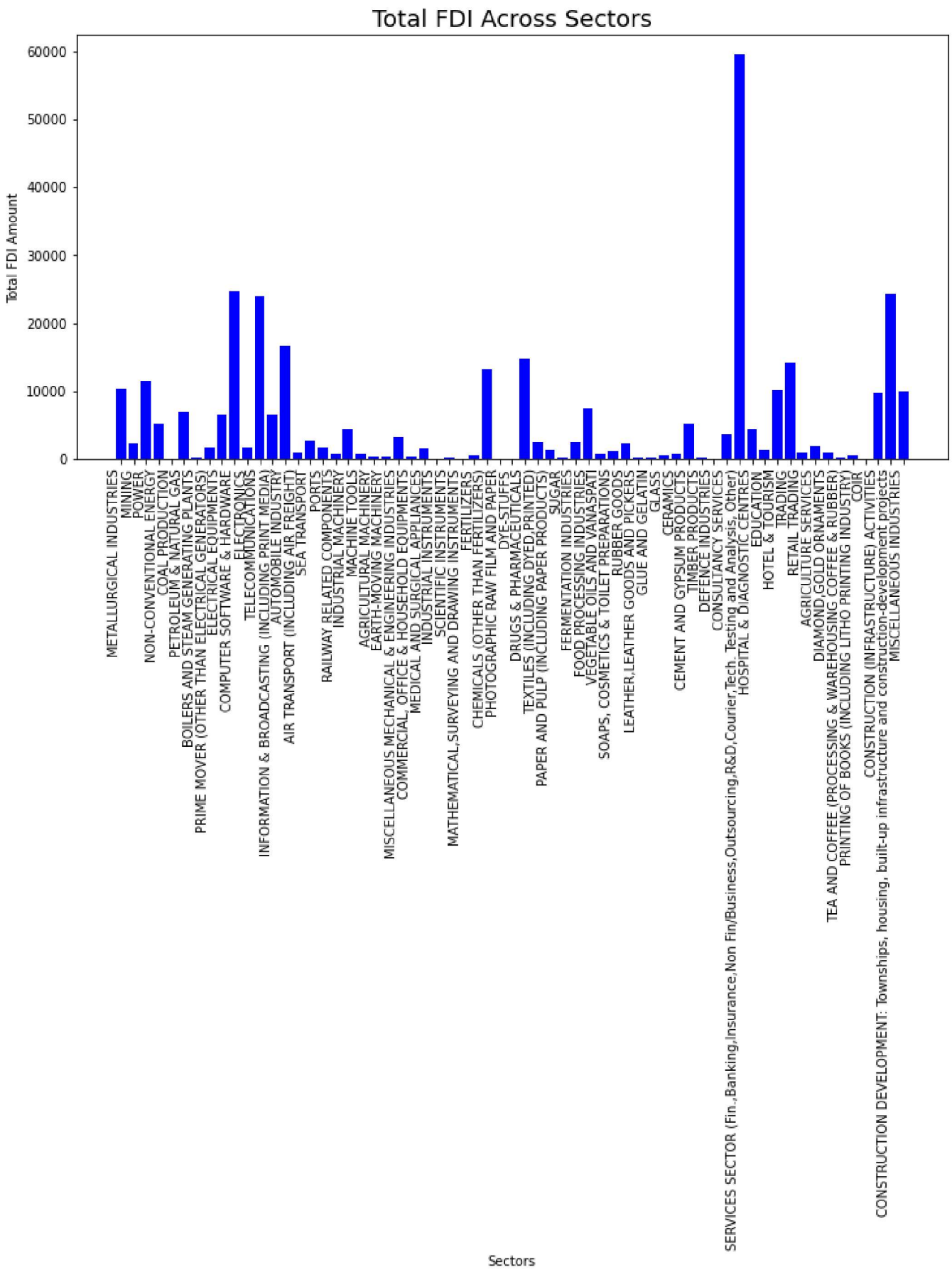
Out[10]:
```
Index(['2000-01', '2001-02', '2002-03', '2003-04', '2004-05', '2005-06',
       '2006-07', '2007-08', '2008-09', '2009-10', '2010-11', '2011-12',
       '2012-13', '2013-14', '2014-15', '2015-16', '2016-17'],
      dtype='object')
```

In [11]:
```
# create a new column of total and average
fdi_df_total['Total'] = fdi_df_total[fdi_columns].sum(axis=1)
fdi_df_total['Average'] = fdi_df_total[fdi_columns].mean(axis=1)
print(fdi_df_total)
```

```
                                          Sector  2000-01  2001-02  \
0                          METALLURGICAL INDUSTRIES    22.69    14.14
1                                          MINING     1.32     6.52
2                                           POWER    89.42   757.44
3                          NON-CONVENTIONAL ENERGY     0.00     0.00
4                                  COAL PRODUCTION     0.00     0.00
..                                             ...      ...      ...
58  PRINTING OF BOOKS (INCLUDING LITHO PRINTING IN...     0.00     0.00
59                                            COIR     0.00     0.00
60          CONSTRUCTION (INFRASTRUCTURE) ACTIVITIES     0.00     0.00
61  CONSTRUCTION DEVELOPMENT: Townships, housing, ...    24.33    51.75
62                          MISCELLANEOUS INDUSTRIES   832.07   221.37

    2002-03  2003-04  2004-05  2005-06  2006-07  2007-08  2008-09  2009-10  \
0     36.61     8.11   200.38   149.13   169.94  1175.75   959.94   419.88
1     10.06    23.48     9.92     7.40     6.62   444.36    34.16   174.40
2     59.11    27.09    43.37    72.69   157.15   988.68   907.66  1271.79
3      1.70     4.14     1.27     1.35     2.44    58.82   125.88   622.52
4      0.00     0.04     0.00     9.14     1.30    14.08     0.22     0.00
..      ...      ...      ...      ...      ...      ...      ...      ...
58     6.30     0.00     0.06     9.90    20.04    35.54    31.61    70.51
59     0.00     0.00     0.47     0.59     0.04     0.01     0.00     0.25
60     0.00     0.00     0.00     0.93    64.06   182.92   172.70   324.56
61    36.10    47.04   152.06   228.71  1392.95  3887.33  4657.51  5466.13
62   218.76   235.48   121.83   164.76   304.87   528.42  1549.70  1147.56

    2010-11  2011-12  2012-13  2013-14  2014-15  2015-16  2016-17     Total  \
0   1098.14  1786.14  1466.23   567.63   359.34   456.31  1440.18  10330.54
1     79.51   142.65    57.89    12.73   684.39   520.67    55.75   2271.83
2   1271.77  1652.38   535.68  1066.08   707.04   868.80  1112.98  11589.13
3    214.40   452.17  1106.52   414.25   615.95   776.51   783.57   5181.49
4      0.00     0.00     0.00     2.96     0.00     0.00     0.00     27.74
..      ...      ...      ...      ...      ...      ...      ...       ...
58    36.63    47.39    14.34   113.78    72.58   122.81    53.17    634.66
59     0.10     0.55     0.15     0.54     1.36     0.00     0.00      4.06
60   675.07   386.28   283.89   485.37   870.25  4510.71  1860.73   9817.47
61  1663.03  3140.78  1332.49  1226.05   769.14   112.55   105.14  24293.09
62  1475.97   813.38   229.49   468.74   765.88   668.77   296.40  10043.45

        Average
0    607.678824
1    133.637059
2    681.713529
3    304.793529
4      1.631765
..          ...
58    37.332941
59     0.238824
60   577.498235
61  1429.005294
62   590.791176

[63 rows x 20 columns]
```

In [12]:

```python
# plotting bar graph
plt.figure(figsize=(12, 6))
plt.bar(fdi_df_total['Sector'], fdi_df_total['Total'], color='blue')
plt.title('Total FDI Across Sectors',fontsize=18)
plt.xlabel('Sectors')
plt.ylabel('Total FDI Amount')
plt.xticks(rotation=90, ha='right')
plt.savefig('my_chart.png')
plt.show()
```

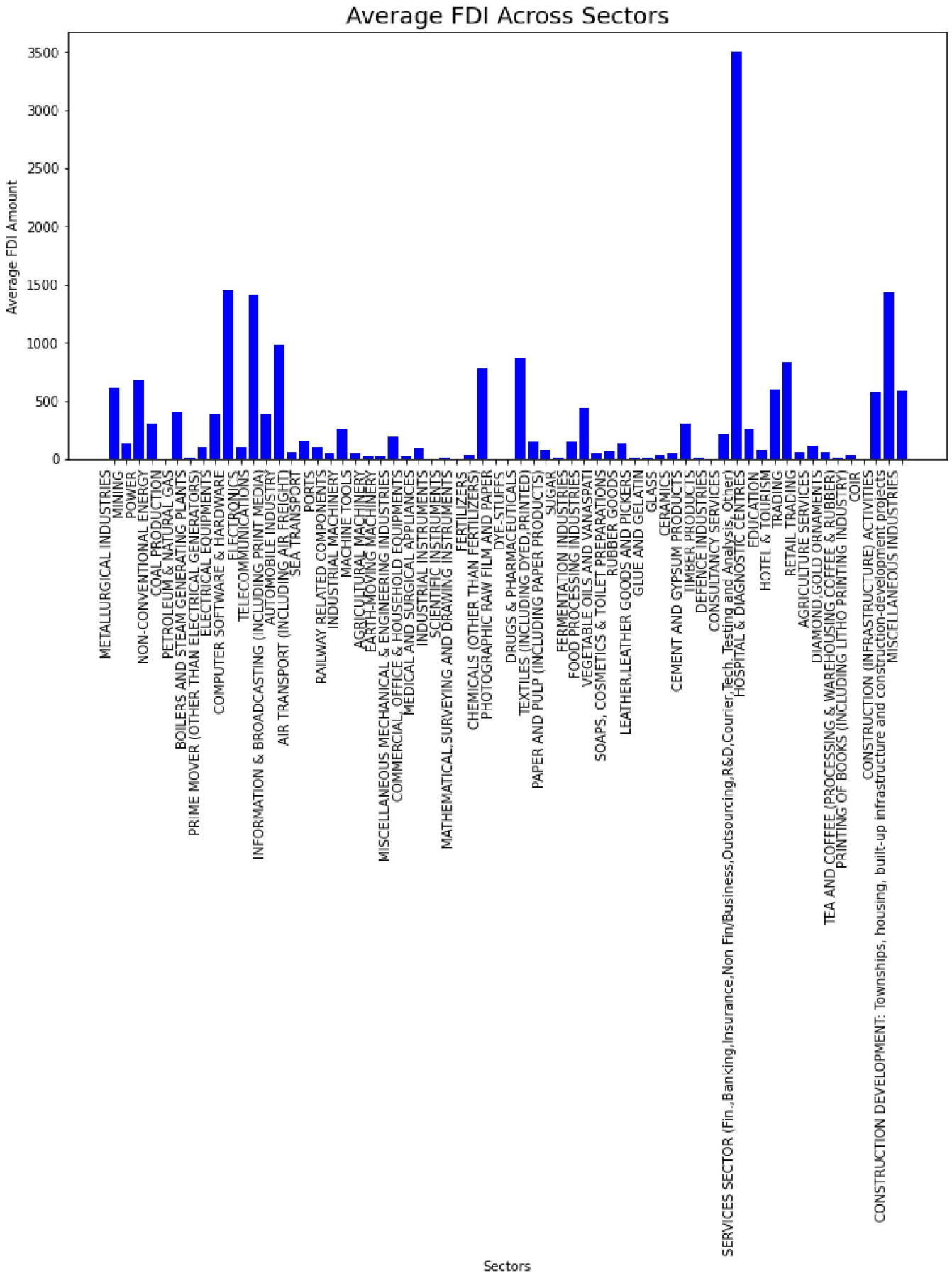


In [13]:
```
FileLink(r'my_chart.png')
```

Out[13]: my_chart.png

In [14]:
```
# plotting bar graph
plt.figure(figsize=(12, 6))
plt.bar(fdi_df_total['Sector'], fdi_df_total['Average'], color='blue')
plt.title('Average FDI Across Sectors',fontsize=18)
plt.xlabel('Sectors')
plt.ylabel('Average FDI Amount')
plt.xticks(rotation=90, ha='right')
```

```
plt.savefig('my_chart2.png')
plt.show()
```



In [15]:
```
FileLink(r'my_chart2.png')
```

Out[15]: my_chart2.png

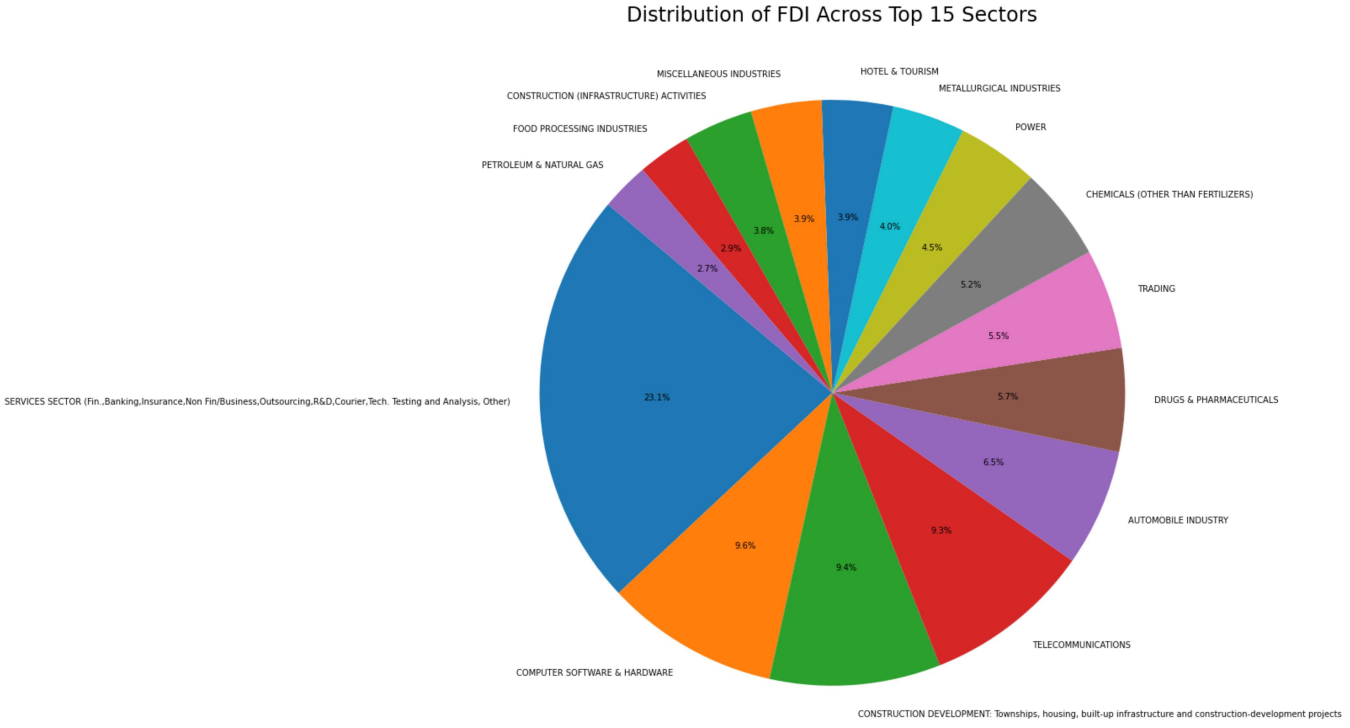
In [16]:
```
fdi_df_total = fdi_df_total.sort_values(by='Total', ascending=False)
fdi_df_total
```

Out[16]:

	Sector	2000-01	2001-02	2002-03	2003-04	2004-05	2005-06	2006-07	2007-08	2008-09	200
49	SERVICES SECTOR (Fin.,Banking,Insurance,Non Fi...	71.38	187.95	296.34	271.15	456.15	548.61	4713.78	6986.17	6183.49	4174.
9	COMPUTER SOFTWARE & HARDWARE	228.39	419.39	314.24	368.32	527.90	1359.97	2613.33	1382.25	1543.34	871.
61	CONSTRUCTION DEVELOPMENT: Townships, housing, ...	24.33	51.75	36.10	47.04	152.06	228.71	1392.95	3887.33	4657.51	5466.
11	TELECOMMUNICATIONS	177.69	873.23	191.60	86.49	118.33	617.98	476.51	1260.70	2548.63	2539.
13	AUTOMOBILE INDUSTRY	195.33	235.76	419.96	119.09	121.97	139.93	260.72	656.10	1150.03	1236.
...	...	...	...	...	...	...	...	...	...	...	
30	PHOTOGRAPHIC RAW FILM AND PAPER	0.00	0.00	0.60	0.24	6.16	0.00	2.81	54.86	1.05	0.
4	COAL PRODUCTION	0.00	0.00	0.00	0.04	0.00	9.14	1.30	14.08	0.22	0.
27	MATHEMATICAL,SURVEYING AND DRAWING INSTRUMENTS	0.00	0.00	0.00	0.00	0.00	0.00	0.00	1.27	0.00	0.
47	DEFENCE INDUSTRIES	0.00	0.00	0.00	0.00	0.05	0.00	0.00	0.00	0.00	0.
59	COIR	0.00	0.00	0.00	0.00	0.47	0.59	0.04	0.01	0.00	0.

63 rows × 20 columns

In [17]:
```
# Plotting a pie chart of top 15 sectors
plt.figure(figsize=(16,16))
plt.pie(fdi_df_total['Total'].head(15), labels=fdi_df_total['Sector'].head(15),
        autopct='%1.1f%%', startangle=140)
plt.title('Distribution of FDI Across Top 15 Sectors',fontsize=24)
plt.savefig('my_chart3.png')
plt.show()
```

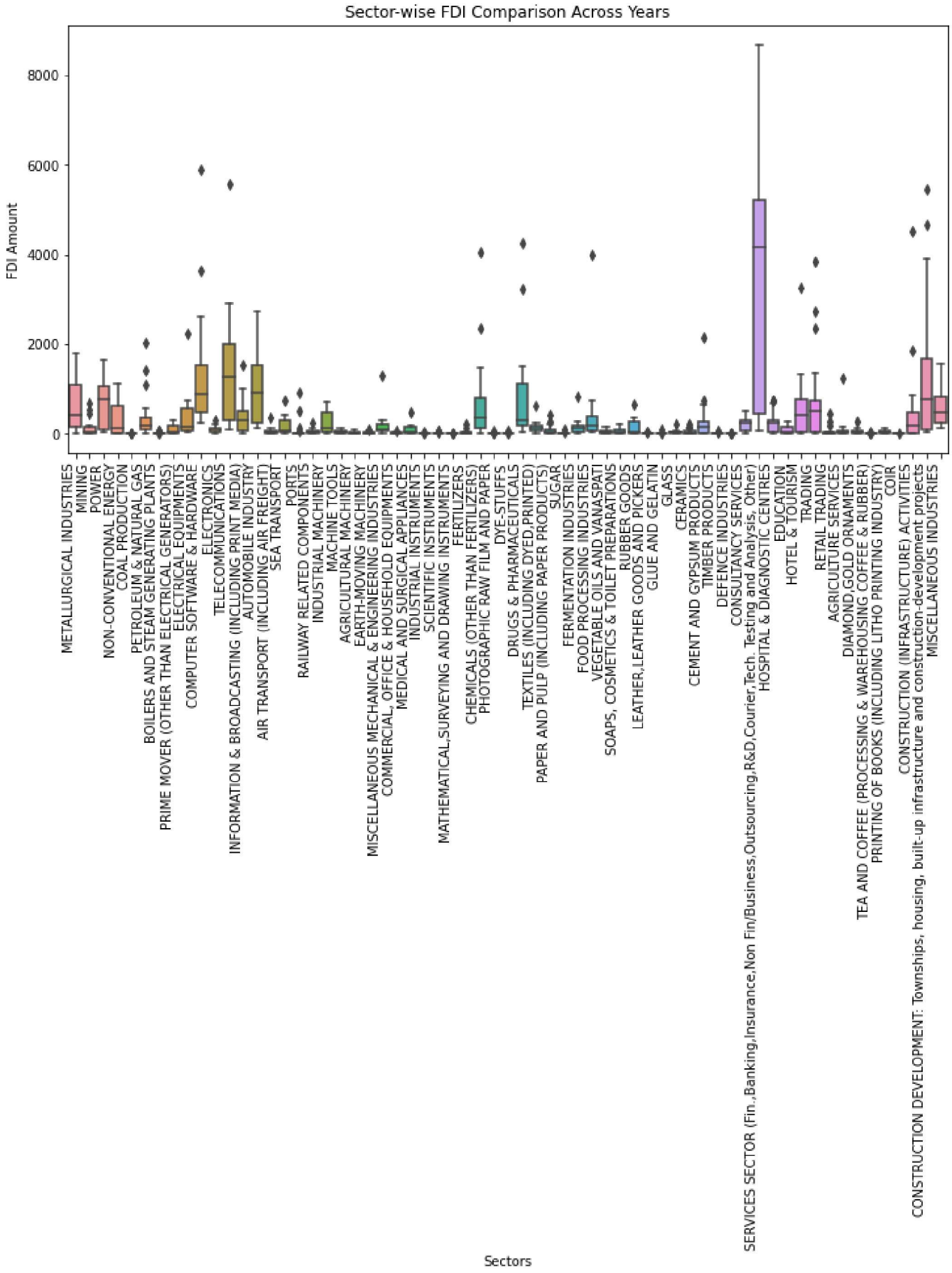


In [18]:
```
FileLink(r'my_chart3.png')
```

Out[18]: my_chart3.png

In [19]:
```
fdi_df_trans = fdi_df.set_index('Sector').T
```

In [20]:
```
# Box Plot
plt.figure(figsize=(12, 6))
sns.boxplot(data=fdi_df_trans)
plt.title('Sector-wise FDI Comparison Across Years')
plt.xlabel('Sectors')
plt.ylabel('FDI Amount')
plt.xticks(rotation=90, ha='right')
plt.savefig('my_chart4.png')
plt.show()
```



In [21]:
```
FileLink(r'my_chart4.png')
```

Out[21]: my_chart4.png

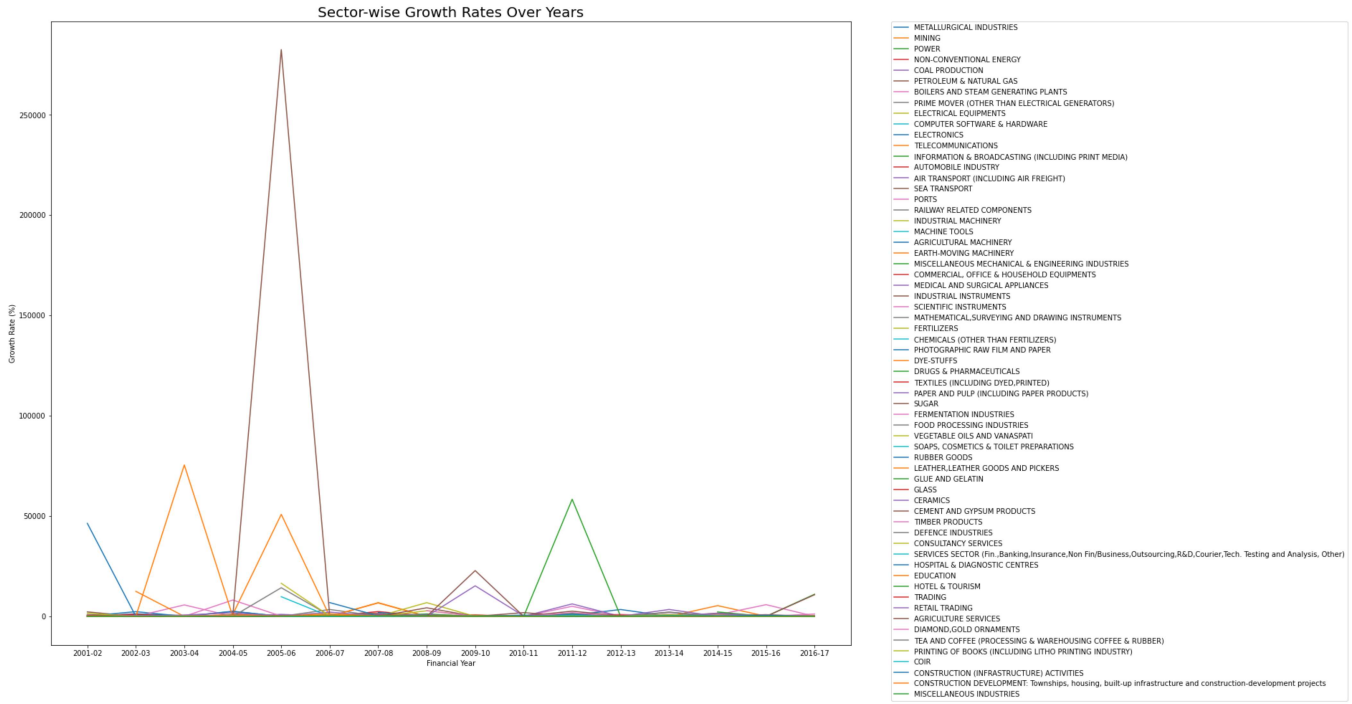
- We notice that Service Sector has highest Total FDI as well as Average FDI
- Through the bar graph we also notice that approx 15 Sectors are the top performing ones, In order to analyse them we create a pie chart.
- We notice that among those top 15 Sectors Service Sector contributes around 23% of FDI followed by Computer, Construction, Telecom, Automobile, Drugs, Trading, and Chemicals

- We use boxplot to compare among the sectors

Sector-Wise Growth Rate

In [22]:
```
fdi_df_trans_growth = fdi_df_trans.pct_change() * 100
```

In [23]:
```
plt.figure(figsize=(20, 16))
for sector in fdi_df_trans_growth.columns:
    plt.plot(fdi_df_trans_growth.index, fdi_df_trans_growth[sector], label=sector)
plt.title('Sector-wise Growth Rates Over Years',fontsize=20)
plt.xlabel('Financial Year')
plt.ylabel('Growth Rate (%)')
plt.legend(bbox_to_anchor=(1.05, 1), loc='best', borderaxespad=0.)
plt.savefig('my_chart5.png')
plt.show()
```



In [24]:
```
FileLink(r'my_chart5.png')
```

Out[24]: my_chart5.png

In [25]:
```
# Calculate standard deviation or coefficient of variation for each sector
sector_std = fdi_df.std(axis=1)
sector_cv = fdi_df.std(axis=1) * 100  / fdi_df.mean(axis=1)
print(sector_cv, sector_cv.max(), sector_cv.mean())
```

```
0      97.143204
1     156.417720
2      76.931446
3     118.019870
4     240.796784
         ...
58    103.732678
59    153.125352
60    193.553190
61    124.340435
62     76.612449
Length: 63, dtype: float64 348.7944783501986 149.70394143539468
```

```
C:\Users\umang\anaconda3\envs\rstudio\lib\site-packages\ipykernel_launcher.py:2: FutureWarnin
g: Dropping of nuisance columns in DataFrame reductions (with 'numeric_only=None') is depreca
ted; in a future version this will raise TypeError.  Select only valid columns before calling
the reduction.

C:\Users\umang\anaconda3\envs\rstudio\lib\site-packages\ipykernel_launcher.py:3: FutureWarnin
g: Dropping of nuisance columns in DataFrame reductions (with 'numeric_only=None') is depreca
ted; in a future version this will raise TypeError.  Select only valid columns before calling
the reduction.
  This is separate from the ipykernel package so we can avoid doing imports until
```

In [26]:
```
# Define a threshold for consistency
consistency_threshold_1 = 200
consistency_threshold_2 = 100
```

In [27]:
```
# Identify sectors with consistently high or low FDI values
high_fdi_sectors_index = sector_cv[sector_cv > consistency_threshold_1].index
consistent_fdi_sectors_index = sector_cv[(sector_cv > consistency_threshold_2) & (sector_cv <
low_fdi_sectors_index = sector_cv[sector_cv < consistency_threshold_2].index
high_fdi_sectors_index,consistent_fdi_sectors_index,low_fdi_sectors_index
```

Out[27]:
```
(Int64Index([4, 16, 27, 30, 31, 35, 37, 42, 47, 54, 55], dtype='int64'),
 Int64Index([ 1,  3,  5,  6,  7,  8,  9, 11, 12, 14, 15, 17, 20, 21, 22, 24, 25,
             26, 28, 29, 32, 34, 36, 38, 39, 40, 41, 43, 44, 45, 46, 50, 51, 52,
             53, 56, 57, 58, 59, 60, 61],
            dtype='int64'),
 Int64Index([0, 2, 10, 13, 18, 19, 23, 33, 48, 49, 62], dtype='int64'))
```

In [28]:
```
high_fdi_sectors = fdi_df_trans.iloc[:, high_fdi_sectors_index]
print("high FDI sectors: ", high_fdi_sectors.columns.tolist())
```

```
high FDI sectors:  ['COAL PRODUCTION', 'PORTS', 'MATHEMATICAL,SURVEYING AND DRAWING INSTRUMEN
TS', 'PHOTOGRAPHIC RAW FILM AND PAPER', 'DYE-STUFFS', 'SUGAR', 'FOOD PROCESSING INDUSTRIES',
'GLUE AND GELATIN', 'DEFENCE INDUSTRIES', 'RETAIL TRADING', 'AGRICULTURE SERVICES']
```

In [29]:
```
consistent_fdi_sectors = fdi_df_trans.iloc[:, high_fdi_sectors_index]
print("Consistent FDI Sectors: ", consistent_fdi_sectors.columns.tolist())
```

```
Consistent FDI Sectors:  ['COAL PRODUCTION', 'PORTS', 'MATHEMATICAL,SURVEYING AND DRAWING INS
TRUMENTS', 'PHOTOGRAPHIC RAW FILM AND PAPER', 'DYE-STUFFS', 'SUGAR', 'FOOD PROCESSING INDUSTR
IES', 'GLUE AND GELATIN', 'DEFENCE INDUSTRIES', 'RETAIL TRADING', 'AGRICULTURE SERVICES']
```

In [30]:
```
low_fdi_sectors = fdi_df_trans.iloc[:, low_fdi_sectors_index]
print("Low FDI Sectors:", low_fdi_sectors.columns.tolist())
```

```
Low FDI Sectors: ['METALLURGICAL INDUSTRIES', 'POWER', 'ELECTRONICS', 'AUTOMOBILE INDUSTRY',
'INDUSTRIAL MACHINERY', 'MACHINE TOOLS', 'COMMERCIAL, OFFICE & HOUSEHOLD EQUIPMENTS', 'TEXTIL
ES (INCLUDING DYED,PRINTED)', 'CONSULTANCY SERVICES', 'SERVICES SECTOR (Fin.,Banking,Insuranc
e,Non Fin/Business,Outsourcing,R&D,Courier,Tech. Testing and Analysis, Other)', 'MISCELLANEOU
S INDUSTRIES']
```

- After plotting the graph it wasn't much clear about which sector grew by how much
- So, we create a threshold to find consistent, High and Low FDI performing Sectors
- high FDI sectors: 'COAL PRODUCTION', 'PORTS', 'MATHEMATICAL,SURVEYING AND DRAWING INSTRUMENTS', 'PHOTOGRAPHIC RAW FILM AND PAPER', 'DYE-STUFFS', 'SUGAR', 'FOOD PROCESSING INDUSTRIES', 'GLUE AND GELATIN', 'DEFENCE INDUSTRIES', 'RETAIL TRADING', 'AGRICULTURE SERVICES'
- Consistent FDI Sectors: 'COAL PRODUCTION', 'PORTS', 'MATHEMATICAL,SURVEYING AND DRAWING INSTRUMENTS', 'PHOTOGRAPHIC RAW FILM AND PAPER', 'DYE-STUFFS', 'SUGAR', 'FOOD PROCESSING INDUSTRIES', 'GLUE AND GELATIN', 'DEFENCE INDUSTRIES', 'RETAIL TRADING', 'AGRICULTURE SERVICES'
- Low FDI Sectors:'METALLURGICAL INDUSTRIES', 'POWER', 'ELECTRONICS', 'AUTOMOBILE INDUSTRY', 'INDUSTRIAL MACHINERY', 'MACHINE TOOLS', 'COMMERCIAL, OFFICE & HOUSEHOLD EQUIPMENTS', 'TEXTILES (INCLUDING DYED,PRINTED)', 'CONSULTANCY SERVICES', 'SERVICES

SECTOR (Fin.,Banking,Insurance,Non Fin/Business,Outsourcing,R&D,Courier,Tech. Testing and Analysis, Other)', 'MISCELLANEOUS INDUSTRIES'

Comparative Analysis

In [31]:
```
fdi_df_compare=fdi_df_total.head(9)
fdi_df_compare=fdi_df_compare.drop(columns=['Total','Average'])
fdi_df_compare=fdi_df_compare.set_index('Sector').T
fdi_df_compare
```

Out[31]:

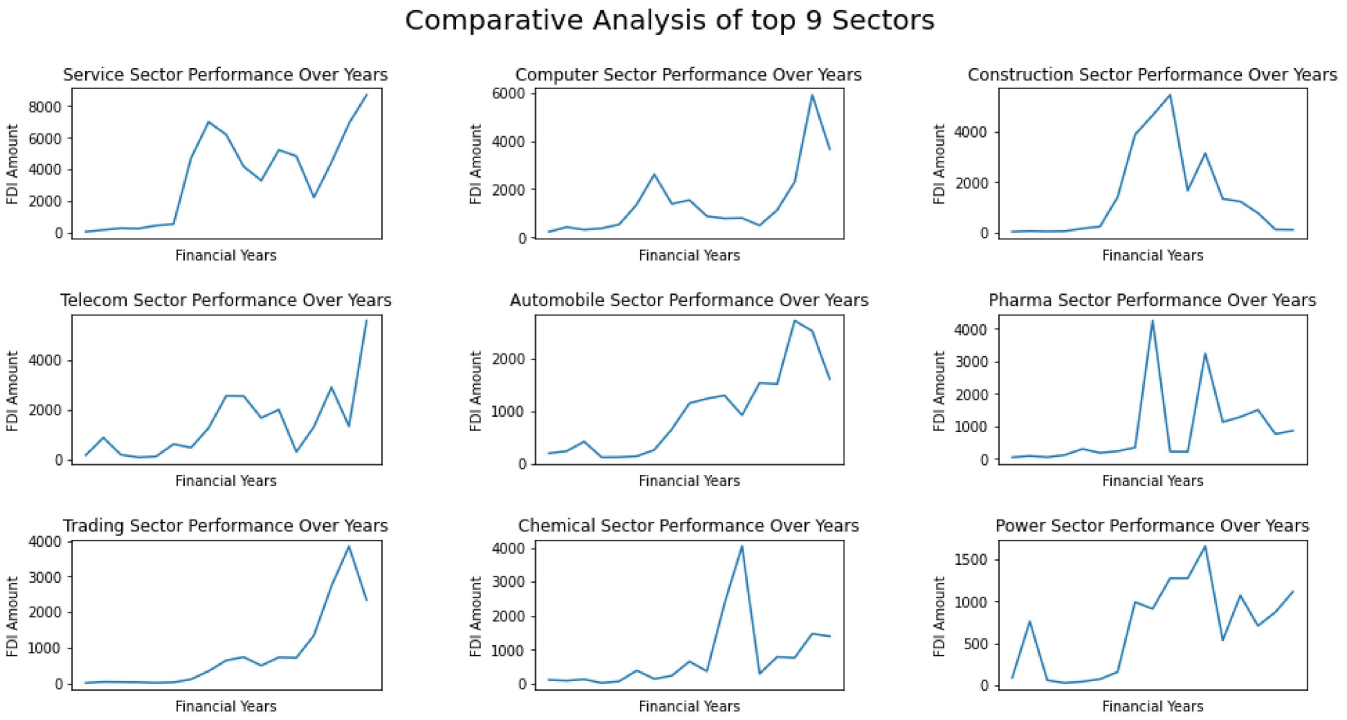
Sector	SERVICES SECTOR (Fin.,Banking,Insurance,Non Fin/Business,Outsourcing,R&D,Courier,Tech. Testing and Analysis, Other)	COMPUTER SOFTWARE & HARDWARE	CONSTRUCTION DEVELOPMENT: Townships, housing, built-up infrastructure and construction-development projects	TELECOMMUNICATIONS	AUT I
2000-01	71.38	228.39	24.33	177.69	
2001-02	187.95	419.39	51.75	873.23	
2002-03	296.34	314.24	36.10	191.60	
2003-04	271.15	368.32	47.04	86.49	
2004-05	456.15	527.90	152.06	118.33	
2005-06	548.61	1359.97	228.71	617.98	
2006-07	4713.78	2613.33	1392.95	476.51	
2007-08	6986.17	1382.25	3887.33	1260.70	
2008-09	6183.49	1543.34	4657.51	2548.63	
2009-10	4174.53	871.86	5466.13	2539.26	
2010-11	3296.09	779.81	1663.03	1664.50	
2011-12	5215.98	796.35	3140.78	1997.24	
2012-13	4832.98	485.96	1332.49	303.87	
2013-14	2225.10	1126.27	1226.05	1306.95	
2014-15	4443.26	2296.04	769.14	2894.94	
2015-16	6889.46	5904.36	112.55	1324.40	
2016-17	8684.07	3651.71	105.14	5563.69	

In [32]:

```
# Subplots
plt.figure(figsize=(16, 8))
plt.subplots_adjust(wspace=0.5,hspace=0.5)
plt.suptitle('Comparative Analysis of top 9 Sectors',fontsize=20)

plt.subplot(3, 3, 1)
plt.plot(fdi_df_compare.index, fdi_df_compare['SERVICES SECTOR (Fin.,Banking,Insurance,Non Fi
plt.title('Service Sector Performance Over Years')
plt.xlabel('Financial Years')
```

```
plt.ylabel('FDI Amount')
plt.gca().axes.get_xaxis().set_ticks([])
plt.gca().axes.get_xaxis().set_ticklabels([])

plt.subplot(3, 3, 2)
plt.plot(fdi_df_compare.index, fdi_df_compare['COMPUTER SOFTWARE & HARDWARE'], label='COMPUTE
plt.title('Computer Sector Performance Over Years')
plt.xlabel('Financial Years')
plt.ylabel('FDI Amount')
plt.gca().axes.get_xaxis().set_ticks([])
plt.gca().axes.get_xaxis().set_ticklabels([])

plt.subplot(3, 3, 3)
plt.plot(fdi_df_compare.index, fdi_df_compare['CONSTRUCTION DEVELOPMENT: Townships, housing,
plt.title('Construction Sector Performance Over Years')
plt.xlabel('Financial Years')
plt.ylabel('FDI Amount')
plt.gca().axes.get_xaxis().set_ticks([])
plt.gca().axes.get_xaxis().set_ticklabels([])

plt.subplot(3, 3, 4)
plt.plot(fdi_df_compare.index, fdi_df_compare['TELECOMMUNICATIONS'], label='TELECOMMUNICATION
plt.title('Telecom Sector Performance Over Years')
plt.xlabel('Financial Years')
plt.ylabel('FDI Amount')
plt.gca().axes.get_xaxis().set_ticks([])
plt.gca().axes.get_xaxis().set_ticklabels([])

plt.subplot(3, 3, 5)
plt.plot(fdi_df_compare.index, fdi_df_compare['AUTOMOBILE INDUSTRY'], label='AUTOMOBILE INDUS
plt.title('Automobile Sector Performance Over Years')
plt.xlabel('Financial Years')
plt.ylabel('FDI Amount')
plt.gca().axes.get_xaxis().set_ticks([])
plt.gca().axes.get_xaxis().set_ticklabels([])

plt.subplot(3, 3, 6)
plt.plot(fdi_df_compare.index, fdi_df_compare['DRUGS & PHARMACEUTICALS'], label='DRUGS & PHAR
plt.title('Pharma Sector Performance Over Years')
plt.xlabel('Financial Years')
plt.ylabel('FDI Amount')
plt.gca().axes.get_xaxis().set_ticks([])
plt.gca().axes.get_xaxis().set_ticklabels([])

plt.subplot(3, 3, 7)
plt.plot(fdi_df_compare.index, fdi_df_compare['TRADING'], label='TRADING')
plt.title('Trading Sector Performance Over Years')
plt.xlabel('Financial Years')
plt.ylabel('FDI Amount')
plt.gca().axes.get_xaxis().set_ticks([])
plt.gca().axes.get_xaxis().set_ticklabels([])

plt.subplot(3, 3, 8)
plt.plot(fdi_df_compare.index, fdi_df_compare['CHEMICALS (OTHER THAN FERTILIZERS)'], label='C
plt.title('Chemical Sector Performance Over Years')
plt.xlabel('Financial Years')
plt.ylabel('FDI Amount')
plt.gca().axes.get_xaxis().set_ticks([])
plt.gca().axes.get_xaxis().set_ticklabels([])

plt.subplot(3, 3, 9)
plt.plot(fdi_df_compare.index, fdi_df_compare['POWER'], label='POWER')
plt.title('Power Sector Performance Over Years')
plt.xlabel('Financial Years')
plt.ylabel('FDI Amount')
plt.gca().axes.get_xaxis().set_ticks([])
plt.gca().axes.get_xaxis().set_ticklabels([])
```

```
plt.savefig('my_chart6.png')
plt.show()
```



In [33]:
```
FileLink(r'my_chart6.png')
```

Out[33]: my_chart6.png

- We notice that some sectors like Power were already at a rise during 2000s but fell and then grew in 2007
- Sectors like Service, Construction, and Pharma saw a high sudden increase during years 2007-09
- Sectors like Telecom, Computer, and Automobile grew slowly and gradually
- Sector like Trading grew much later.
- We also observe that in 2017 sectors like Telecom, Service, Chemical and Power are rising
- While Sectors like Trading, Automobile, Computer, and Construction are falling

Year- Wise (Time Series) Analysis

In [34]:
```
fdi_trans_columns = fdi_df_trans.columns[0:]
```

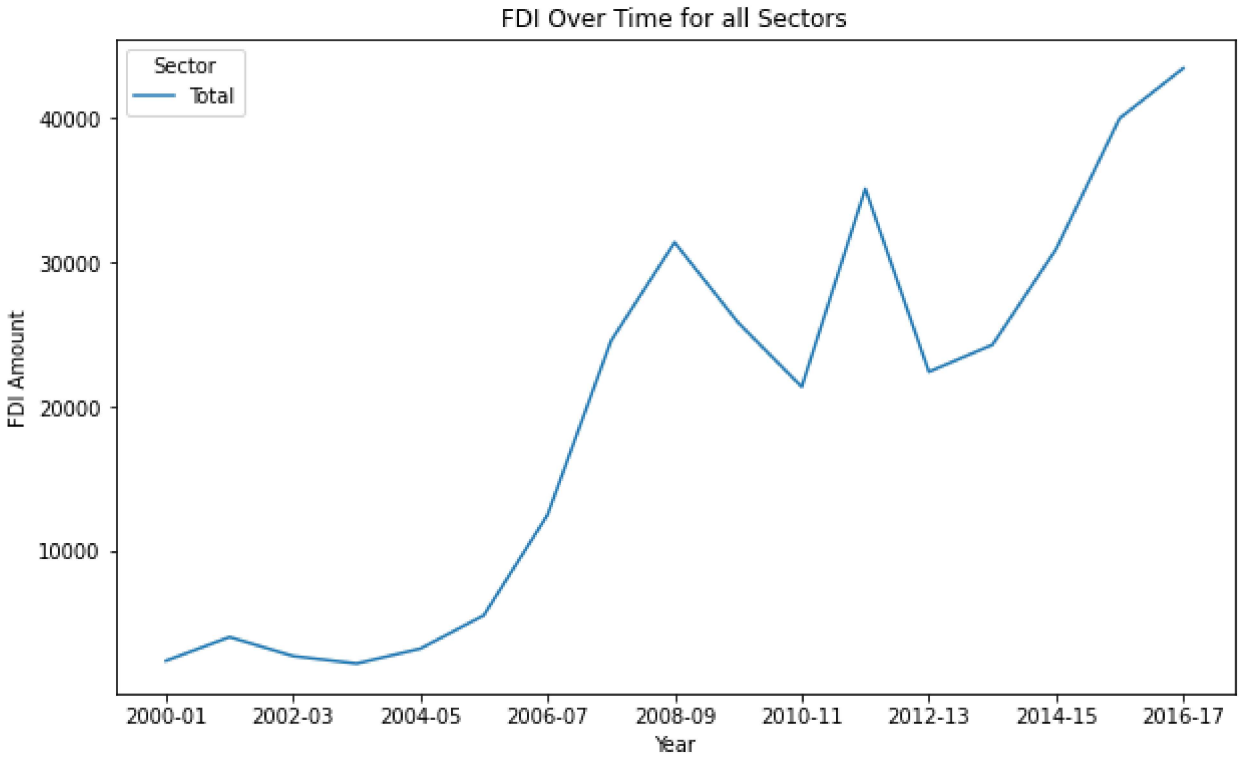
In [35]:
```
fdi_df_trans['Total']=fdi_df_trans[fdi_trans_columns].sum(axis=1)
fdi_df_trans=fdi_df_trans.drop(columns=fdi_trans_columns)
fdi_df_trans
```

Out[35]:

Sector	Total
2000-01	2378.71
2001-02	4027.69
2002-03	2704.32
2003-04	2187.85
2004-05	3218.69
2005-06	5539.75
2006-07	12491.76
2007-08	24575.40
2008-09	31395.96
2009-10	25834.38
2010-11	21383.07
2011-12	35120.78
2012-13	22423.59
2013-14	24299.32
2014-15	30930.47
2015-16	40000.99
2016-17	43478.26

In [36]:

```
# Total FDI and its change over time
fdi_df_trans.plot(figsize=(10, 6))
plt.title('FDI Over Time for all Sectors')
plt.xlabel('Year')
plt.ylabel('FDI Amount')
plt.savefig('my_chart7.png')
plt.show()
```



In [37]:

```
FileLink(r'my_chart7.png')
```

Out[37]: my_chart7.png

- We notice that overall FDI of India in total for all sectors has risen with a great amount from 2000-01 to 2016-17
- We notice that it rose most significantly in 2007-2009
- Though it fell during 2010-11 and increased during 2011-12 fell again in 2013-14. But is constantly rising since then which is a positive growth sign.

In [38]:
```
# Thank You
```