```
import numpy as np
import pandas as pd
import warnings
warnings.filterwarnings('ignore')
```

```
from google.colab import drive
drive.mount('/content/drive')
```

```
FDI = pd.read_csv('/content/drive/MyDrive/FDI data.csv')
```

```
FDI.shape
```

(63, 18)

```
FDI.head()
```

```
FDI.style.set_caption('Amount in US$ Millions').format(precision=2)
```

```
FDI.columns
```

```
Index(['Sector', '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')
```

```
Year = ['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']
Sectors = ['Sector']
```

```
# extracting detailed information
FDI.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
```

```
pd.isnull(FDI).sum()
```

```
# checking for duplicate values
FDI.duplicated().value_counts()
```

```
False    63
Name: count, dtype: int64
```

```
Rates = [45.68,47.69,48.39,45.95,44.93,44.27,45.24,40.26,45.99,
         47.44,45.56,47.92,54.40,60.50,61.14,65.46,67.07]
```

```
def multiply_columns(df, col_list,num):
    for col in col_list:
        df[col] = df[col] * Rates[col_list.index(col)]/10
    return df

FDI_InUSD=FDI.copy()
FDI_02 = multiply_columns(FDI, Year, Rates)
```

```
FDI_02.style.set_caption("FDI INFLOWS (Amount in ₹ Crores)").format(precision=2)
```

```
def multiply_columns(df, col_list,num):
    for col in col_list:
        df[col] = df[col] * Rates[col_list.index(col)]/10
    return df

FDI_InUSD=FDI.copy()
FDI_02 = multiply_columns(FDI, Year, Rates)
```

```
FDI_02.style.set_caption("FDI INFLOWS (Amount in ₹ Crores)").format(precision=2)
```

37	FOOD PROCESSING INDUSTRIES	208.99	1046.27	178.46	501.87	197.60	184.78	461.45	282.50	472.36	1323.0
38	VEGETABLE OILS AND VANASPATI	0.00	0.00	0.00	7.77	40.84	54.50	73.38	6.16	197.21	330.8
39	SOAPS, COSMETICS & TOILET PREPARATIONS	0.00	0.00	0.00	0.00	4.00	387.01	27.32	25.69	101.32	116.6
40	RUBBER GOODS	0.46	221.23	79.46	29.27	179.99	150.92	84.83	60.87	390.36	114.4
41	LEATHER,LEATHER GOODS AND PICKERS	44.54	0.95	0.05	34.69	1.98	4.91	37.37	30.03	15.27	24.0

	Sector										
42	GLUE AND GELATIN	0.00	4.48	25.26	0.00	0.00	0.00	0.00	9.18	0.00	1.2
43	GLASS	154.72	39.92	217.66	24.08	37.56	3.59	6.47	44.45	106.51	13.4
44	CERAMICS	18.41	3.72	1.02	6.75	120.37	25.10	552.97	463.43	912.58	34.2
45	CEMENT AND GYPSUM PRODUCTS	309.34	667.18	102.01	44.02	0.72	2001.36	1096.93	68.00	3333.36	160.3
46	TIMBER PRODUCTS	0.00	0.24	0.19	0.51	0.31	1.46	0.00	1.57	51.83	31.0
47	DEFENCE INDUSTRIES	0.00	0.00	0.00	0.00	0.22	0.00	0.00	0.00	0.00	0.0
48	CONSULTANCY SERVICES	19.41	315.80	124.36	212.29	1134.12	209.84	518.81	939.35	1180.06	1619.1
49	SERVICES SECTOR (Fin.,Banking,Insurance,Non Fin/Business,Outsourcing,R&D,Courier,Tech. Testing and Analysis, Other)	326.06	896.33	1433.99	1245.93	2049.48	2428.70	21325.14	28126.32	28437.87	19803.9
50	HOSPITAL & DIAGNOSTIC CENTRES	0.00	33.05	140.96	110.65	117.67	144.01	172.55	973.33	1102.43	643.1
51	EDUCATION	0.00	0.00	0.00	0.87	8.85	13.99	189.15	175.49	986.58	300.5
52	HOTEL & TOURISM	60.30	153.18	163.32	226.81	166.29	317.77	885.17	1696.84	2133.57	3572.3
53	TRADING	52.49	206.35	184.51	143.00	63.89	128.07	518.68	1389.05	2960.10	3500.8
54	RETAIL TRADING	0.00	0.00	0.00	0.00	0.00	0.00	0.00	5.11	0.41	65.1
55	AGRICULTURE SERVICES	80.03	67.05	53.28	2.71	17.21	40.20	56.69	234.03	24.60	5798.2
56	DIAMOND,GOLD ORNAMENTS	86.02	1.72	6.29	9.01	38.55	68.71	280.35	238.14	384.02	147.4
57	TEA AND COFFEE (PROCESSING & WAREHOUSING COFFEE & RUBBER)	92.41	0.67	0.00	1.47	0.04	6.33	28.05	76.25	170.53	38.6
58	PRINTING OF BOOKS (INCLUDING LITHO PRINTING INDUSTRY)	0.00	0.00	30.49	0.00	0.27	43.83	90.66	143.08	145.37	334.5
59	COIR	0.00	0.00	0.00	0.00	2.11	2.61	0.18	0.04	0.00	1.1
60	CONSTRUCTION (INFRASTRUCTURE) ACTIVITIES	0.00	0.00	0.00	0.00	0.00	4.12	289.81	736.44	794.25	1539.7
61	CONSTRUCTION DEVELOPMENT: Townships, housing, built-up infrastructure and construction-development projects	111.14	246.80	174.69	216.15	683.21	1012.50	6301.71	15650.39	21419.89	25931.3
62	MISCELLANEOUS INDUSTRIES	3800.90	1055.71	1058.58	1082.03	547.38	729.39	1379.23	2127.42	7127.07	5444.0

```python
melt = pd.melt(FDI_InUSD, id_vars = Sectors, value_vars = Year, var_name='Year',
    value_name='FDI(US$ Million)',ignore_index=True)
melt=round(melt,2)
melt
```

	Sector	Year	FDI(US$ Million)
0	METALLURGICAL INDUSTRIES	2000-01	22.69
1	MINING	2000-01	1.32
2	POWER	2000-01	89.42
3	NON-CONVENTIONAL ENERGY	2000-01	0.00
4	COAL PRODUCTION	2000-01	0.00
...	...	...	...
1066	PRINTING OF BOOKS (INCLUDING LITHO PRINTING IN...	2016-17	53.17
1067	COIR	2016-17	0.00
1068	CONSTRUCTION (INFRASTRUCTURE) ACTIVITIES	2016-17	1860.73
1069	CONSTRUCTION DEVELOPMENT: Townships, housing, ...	2016-17	105.14
1070	MISCELLANEOUS INDUSTRIES	2016-17	296.40

1071 rows × 3 columns

Next steps: Generate code with `melt` | View recommended plots | New interactive sheet

```python
melt01 = pd.melt(FDI_02, id_vars = Sectors, value_vars = Year, var_name='Year',
    value_name='FDI(₹ Crores)',ignore_index=True)
melt01=round(melt01,2)
melt01
```