In [81]:
```
!java -version
```

```
java version "1.8.0_471"
Java(TM) SE Runtime Environment (build 1.8.0_471-b09)
Java HotSpot(TM) 64-Bit Server VM (build 25.471-b09, mixed mode)
```

In [82]:
```
!pip install tabula-py
!pip install pandas
```

```
Requirement already satisfied: tabula-py in c:\users\hp\anaconda3\lib\site-packages
(2.10.0)
Requirement already satisfied: pandas>=0.25.3 in c:\users\hp\anaconda3\lib\site-pack
ages (from tabula-py) (2.3.3)
Requirement already satisfied: numpy>1.24.4 in c:\users\hp\anaconda3\lib\site-packag
es (from tabula-py) (2.3.5)
Requirement already satisfied: distro in c:\users\hp\anaconda3\lib\site-packages (fr
om tabula-py) (1.9.0)
Requirement already satisfied: python-dateutil>=2.8.2 in c:\users\hp\anaconda3\lib\s
ite-packages (from pandas>=0.25.3->tabula-py) (2.9.0.post0)
Requirement already satisfied: pytz>=2020.1 in c:\users\hp\anaconda3\lib\site-packag
es (from pandas>=0.25.3->tabula-py) (2025.2)
Requirement already satisfied: tzdata>=2022.7 in c:\users\hp\anaconda3\lib\site-pack
ages (from pandas>=0.25.3->tabula-py) (2025.2)
Requirement already satisfied: six>=1.5 in c:\users\hp\anaconda3\lib\site-packages
(from python-dateutil>=2.8.2->pandas>=0.25.3->tabula-py) (1.17.0)
Requirement already satisfied: pandas in c:\users\hp\anaconda3\lib\site-packages (2.
3.3)
Requirement already satisfied: numpy>=1.26.0 in c:\users\hp\anaconda3\lib\site-packa
ges (from pandas) (2.3.5)
Requirement already satisfied: python-dateutil>=2.8.2 in c:\users\hp\anaconda3\lib\s
ite-packages (from pandas) (2.9.0.post0)
Requirement already satisfied: pytz>=2020.1 in c:\users\hp\anaconda3\lib\site-packag
es (from pandas) (2025.2)
Requirement already satisfied: tzdata>=2022.7 in c:\users\hp\anaconda3\lib\site-pack
ages (from pandas) (2025.2)
Requirement already satisfied: six>=1.5 in c:\users\hp\anaconda3\lib\site-packages
(from python-dateutil>=2.8.2->pandas) (1.17.0)
```

In [83]:
```
import tabula as tb
import pandas as pd
```

In [84]:
```
data_area= [0,0,100,85]
dflist1= tb.read_pdf('Bulletin_269_ Aug 2019_347.pdf', pages=46, area= data_area,re
df= dflist1[0]
df
```

Out[84]:

	Unnamed: 0	Unnamed: 1	21-Balance of Payments	Unnamed: 2	Unnamed: 3	Unnamed: 4	Unnamed: 5
0	NaN	NaN	NaN	(US$ mn)	NaN	NaN	NaN
1	NaN	NaN	NaN	□□□□□□□□ا □□□□ا	NaN	NaN	(+) 2016/2017* 2017/2018* (+)
2	NaN	During	NaN	Fiscal Years	NaN	NaN	□□ا□□ا □□□□ل□□□□ و ا □□□□ ا□ □□□□ا□م□ □□□□ا □□□□□ا □..
3	NaN	NaN	2013/2014 2014/2015	2015/2016 2016/2017*	(+)2017/2018*	NaN	Q4 Q1 Q2 Q3
4	Trade Balance	NaN	-34159.3 -39060.4	-38683.1 -37274.8	-37276.0	NaN	-8698.4 -8908.2 -9839.6 -9255.6
5	Export proceeds **	NaN	26022.6 22245.1	18704.6 21728.2	25827.0	NaN	5798.2 5839.4 6215.7 6755.8
6	Petroleum exports	NaN	12355.9 8891.9	5674.3 6589.5	8773.0	NaN	1996.9 1782.5 2029.1 2202.5
7	Other exports	NaN	13666.7 13353.2	13030.3 15138.7	17054.0	NaN	3801.3 4056.9 4186.6 4553.3
8	Import payments**	NaN	-60181.9 -61305.5	-57387.7 -59003.0	-63103.0	NaN	-14496.6 -14747.6 -16055.3 -16011.4
9	Petroleum imports	NaN	-13246.9 -12366.1	-9293.6 -12015.5	-12489.8	NaN	-3337.6 -2752.9 -3231.7 -3410.0
10	Other imports	NaN	-46935.0 -48939.4	-48094.1 -46987.5	-50613.2	NaN	-11159.0 -11994.7 -12823.6 -12601.4

	Unnamed: 0	Unnamed: 1	21-Balance of Payments	Unnamed: 2	Unnamed: 3	Unnamed: 4	Unnamed: 5
11	Services Balance	NaN	8274.4 10742.9	6533.0 5614.2	11122.4	NaN	2323.1 2847.3 2462.7 2528.8
12	Receipts	NaN	17437.2 21811.8	16079.3 15400.1	21486.9	NaN	5080.8 5678.2 5068.2 5038.1
13	Transportation	NaN	9466.0 9850.3	9534.6 7911.2	8707.9	NaN	2429.7 2267.9 2087.1 2029.5
14	Of which: Suez Canal dues	NaN	5369.1 5361.7	5121.6 4945.3	5706.7	NaN	1228.7 1382.2 1386.3 1389.7
15	Travel (tourism revenues)	NaN	5073.3 7370.4	3767.5 4379.7	9804.3	NaN	1539.0 2696.7 2282.5 2271.4
16	Government receipts	NaN	654.4 1381.5	378.0 776.4	636.7	NaN	585.1 131.7 138.0 163.6
17	Others	NaN	2243.5 3209.6	2399.2 2332.8	2338.0	NaN	527.0 581.9 560.6 573.6
18	Payments	NaN	9162.8 11068.9	9546.3 9785.9	10364.5	NaN	2757.7 2830.9 2605.5 2509.3
19	Transportation	NaN	1717.2 1535.0	1339.1 1332.1	1480.2	NaN	400.6 382.6 351.3 368.6
20	Travel	NaN	3044.5 3338.2	4091.0 2739.9	2451.5	NaN	550.2 649.3 511.9 542.2
21	Government expenditures	NaN	1073.9 854.1	777.1 1124.1	1493.5	NaN	424.0 449.0 540.1 238.4
22	Others	NaN	3327.2 5341.6	3339.1 4589.8	4939.3	NaN	1382.9 1350.0 1202.2 1360.1

	Unnamed: 0	Unnamed: 1	21-Balance of Payments	Unnamed: 2	Unnamed: 3	Unnamed: 4	Unnamed: 5
23	Investment Income Balance	NaN	-7262.7 -5700.9	-4471.7 -4568.5	-6279.6	NaN	-1249.8 -1519.3 -1520.3 -1664.3
24	Receipts	NaN	194.2 212.8	396.9 497.9	835.4	NaN	192.6 229.0 183.7 212.0
25	Payments	NaN	7456.9 5913.7	4868.6 5066.4	7115.0	NaN	1442.4 1748.3 1704.0 1876.3
26	Of which: Interest paid	NaN	652.5 643.6	752.0 1231.9	1616.1	NaN	343.3 415.5 401.8 399.6
27	Current Transfers	NaN	30367.9 21875.8	16790.7 21835.1	26470.9	NaN	6071.9 5826.0 7112.4 6460.6
28	Private (net),	NaN	18447.7 19205.4	16689.2 21686.1	26264.7	NaN	6005.4 5782.9 7087.2 6435.4
29	NaN	NaN	(1)	NaN	NaN	NaN	NaN
30	Of which: Remittances of Egyptians working ab...	NaN	18518.7 19330.0	17077.4 21816.3	26392.9	NaN	6033.9 5824.3 7098.8 6464.4
31	Official (net)	NaN	11920.2 2670.4	101.5 149.0	206.2	NaN	66.5 43.1 25.2 25.2
32	Balance of Current Account	NaN	-2779.7 -12142.6	-19831.1 -14394.0	-5962.3	NaN	-1553.2 -1754.2 -1784.8 -1930.5
33	Capital & Financial Account	NaN	5189.5 17928.9	21176.7 31015.1	21996.5	NaN	4496.9 6247.4 4180.1 8617.1
34	Capital Account	NaN	194.1 -122.9	-141.4 -113.3	-150.7	NaN	-14.7 -40.3 -41.0 -37.0
35	Financial Account	NaN	4995.4 18051.8	21318.1 31128.4	22147.2	NaN	4511.6 6287.7 4221.1 8654.1

	Unnamed: 0	Unnamed: 1	21-Balance of Payments	Unnamed: 2	Unnamed: 3	Unnamed: 4	Unnamed: 5
36	Direct investment abroad	NaN	-326.6 -223.3	-164.2 -175.1	-271.2	NaN	-27.4 -52.4 -79.5 -68.1
37	Direct investment in Egypt (net)	NaN	4178.2 6379.8	6932.6 7932.8	7719.5	NaN	1367.8 1843.0 1919.9 2256.3
38	Portfolio investment abroad	NaN	65.9 47.2	192.1 208.4	-20.8	NaN	29.7 13.9 10.7 -49.7
39	Portfolio investment in Egypt (Net)	NaN	1237.2 -638.6	-1286.8 15985.3	12094.8	NaN	8184.1 7478.5 540.7 6905.5
40	Of which:Bonds	NaN	926.7 -1147.5	-1444.8 5491.5	5293.2	NaN	2301.3 5.8 -108.9 3294.5
41	Other Investments (Net)	NaN	-159.3 12486.7	15644.4 7177.0	2624.9	NaN	-5042.6 -2995.3 1829.3 -389.9
42	Net Borrowing	NaN	207.2 5036.2	7102.7 9699.2	10278.8	NaN	587.5 657.3 3889.5 2402.6
43	Medium- and Long-Term Loans	NaN	-956.2 -482.5	-186.3 5156.8	6738.5	NaN	-9.7 993.6 2065.2 1238.4
44	Disbursements	NaN	1153.4 1753.5	2523.4 7641.1	8846.4	NaN	503.3 1563.0 2603.5 1879.4
45	Repayments	NaN	-2109.6 -2236.0	-2709.7 -2484.3	-2107.9	NaN	-513.0 -569.4 -538.3 -641.0
46	Medium- and Long-Term Suppliers' Credit	NaN	-56.3 257.9	1505.3 2795.1	1118.5	NaN	627.9 235.1 175.5 122.0
47	Disbursements	NaN	8.1 313.0	1560.7 2912.2	1313.6	NaN	664.7 275.5 233.2 175.1

	Unnamed: 0	Unnamed: 1	21- Balance of Payments	Unnamed: 2	Unnamed: 3	Unnamed: 4	Unnamed: 5
48	Repayments	NaN	-64.4 -55.1	-55.4 -117.1	-195.1	NaN	-36.8 -40.4 -57.7 -53.1
49	NaN	NaN	- 76 -	NaN	NaN	NaN	NaN

In [85]:
```
dfcle= df.iloc[4:].copy()
dfcle= dfcle.dropna(axis=1,how= 'all')
newnames= ['Category','2013/2014','2014/2015','2015/2016','2016/2017','2017/2018','
```

In [86]:
```
dfcle.columns= newnames[:len(dfcle.columns)]
dfcle= dfcle.reindex(columns=newnames)
dfcle
```

Out[86]:

	Category	2013/2014	2014/2015	2015/2016	2016/2017	2017/2018	2016/2017(
4	Trade Balance	-34159.3 -39060.4	-38683.1 -37274.8	-37276.0	-8698.4 -8908.2 -9839.6 -9255.6	-9272.6	-9812.6 -943
5	Export proceeds **	26022.6 22245.1	18704.6 21728.2	25827.0	5798.2 5839.4 6215.7 6755.8	7016.1	6785.2 748
6	Petroleum exports	12355.9 8891.9	5674.3 6589.5	8773.0	1996.9 1782.5 2029.1 2202.5	2758.9	2810.0 319
7	Other exports	13666.7 13353.2	13030.3 15138.7	17054.0	3801.3 4056.9 4186.6 4553.3	4257.2	3975.2 429
8	Import payments**	-60181.9 -61305.5	-57387.7 -59003.0	-63103.0	-14496.6 -14747.6 -16055.3 -16011.4	-16288.7	-1659 -1692
9	Petroleum imports	-13246.9 -12366.1	-9293.6 -12015.5	-12489.8	-3337.6 -2752.9 -3231.7 -3410.0	-3095.2	-3415.6 -244
10	Other imports	-46935.0 -48939.4	-48094.1 -46987.5	-50613.2	-11159.0 -11994.7 -12823.6 -12601.4	-13193.5	-1318 -1448
11	Services Balance	8274.4 10742.9	6533.0 5614.2	11122.4	2323.1 2847.3 2462.7 2528.8	3283.6	4283.1 297
12	Receipts	17437.2 21811.8	16079.3 15400.1	21486.9	5080.8 5678.2 5068.2 5038.1	5702.4	6938.4 589
13	Transportation	9466.0 9850.3	9534.6 7911.2	8707.9	2429.7 2267.9 2087.1 2029.5	2323.4	2242.8 224
14	Of which: Suez Canal dues	5369.1 5361.7	5121.6 4945.3	5706.7	1228.7 1382.2 1386.3 1389.7	1548.5	1441.2 148

	Category	2013/2014	2014/2015	2015/2016	2016/2017	2017/2018	2016/2017(
15	Travel (tourism revenues)	5073.3 7370.4	3767.5 4379.7	9804.3	1539.0 2696.7 2282.5 2271.4	2553.7	3930.9 285
16	Government receipts	654.4 1381.5	378.0 776.4	636.7	585.1 131.7 138.0 163.6	203.4	166.0 12
17	Others	2243.5 3209.6	2399.2 2332.8	2338.0	527.0 581.9 560.6 573.6	621.9	598.7 66
18	Payments	9162.8 11068.9	9546.3 9785.9	10364.5	2757.7 2830.9 2605.5 2509.3	2418.8	2655.3 291
19	Transportation	1717.2 1535.0	1339.1 1332.1	1480.2	400.6 382.6 351.3 368.6	377.7	448.5 41
20	Travel	3044.5 3338.2	4091.0 2739.9	2451.5	550.2 649.3 511.9 542.2	748.1	717.0 65
21	Government expenditures	1073.9 854.1	777.1 1124.1	1493.5	424.0 449.0 540.1 238.4	266.0	183.1 17
22	Others	3327.2 5341.6	3339.1 4589.8	4939.3	1382.9 1350.0 1202.2 1360.1	1027.0	1306.7 166
23	Investment Income Balance	-7262.7 -5700.9	-4471.7 -4568.5	-6279.6	-1249.8 -1519.3 -1520.3 -1664.3	-1575.7	-2048.0 -176
24	Receipts	194.2 212.8	396.9 497.9	835.4	192.6 229.0 183.7 212.0	210.7	227.7 25
25	Payments	7456.9 5913.7	4868.6 5066.4	7115.0	1442.4 1748.3 1704.0 1876.3	1786.4	2275.7 202
26	Of which: Interest paid	652.5 643.6	752.0 1231.9	1616.1	343.3 415.5 401.8 399.6	399.2	508.3 56
27	Current Transfers	30367.9 21875.8	16790.7 21835.1	26470.9	6071.9 5826.0 7112.4 6460.6	7071.9	5908.9 604
28	Private (net),	18447.7 19205.4	16689.2 21686.1	26264.7	6005.4 5782.9 7087.2 6435.4	6959.2	5861.2 593
29	NaN	(1)	NaN	NaN	NaN	NaN	N

	Category	2013/2014	2014/2015	2015/2016	2016/2017	2017/2018	2016/2017(
30	Of which: Remittances of Egyptians working ab...	18518.7 19330.0	17077.4 21816.3	26392.9	6033.9 5824.3 7098.8 6464.4	7005.4	5909.0 613
31	Official (net)	11920.2 2670.4	101.5 149.0	206.2	66.5 43.1 25.2 25.2	112.7	47.7 11
32	Balance of Current Account	-2779.7 -12142.6	-19831.1 -14394.0	-5962.3	-1553.2 -1754.2 -1784.8 -1930.5	-492.8	-1668.6 -218
33	Capital & Financial Account	5189.5 17928.9	21176.7 31015.1	21996.5	4496.9 6247.4 4180.1 8617.1	2951.9	1475.6 31
34	Capital Account	194.1 -122.9	-141.4 -113.3	-150.7	-14.7 -40.3 -41.0 -37.0	-32.4	-35.0 -2
35	Financial Account	4995.4 18051.8	21318.1 31128.4	22147.2	4511.6 6287.7 4221.1 8654.1	2984.3	1510.6 33
36	Direct investment abroad	-326.6 -223.3	-164.2 -175.1	-271.2	-27.4 -52.4 -79.5 -68.1	-71.2	-65.8 -11
37	Direct investment in Egypt (net)	4178.2 6379.8	6932.6 7932.8	7719.5	1367.8 1843.0 1919.9 2256.3	1700.3	1099.9 174
38	Portfolio investment abroad	65.9 47.2	192.1 208.4	-20.8	29.7 13.9 10.7 -49.7	4.3	-75.4 2
39	Portfolio investment in Egypt (Net)	1237.2 -638.6	-1286.8 15985.3	12094.8	8184.1 7478.5 540.7 6905.5	-2829.9	-3240.3 -264
40	Of which:Bonds	926.7 -1147.5	-1444.8 5491.5	5293.2	2301.3 5.8 -108.9 3294.5	2101.8	-121.3 -18
41	Other Investments (Net)	-159.3 12486.7	15644.4 7177.0	2624.9	-5042.6 -2995.3 1829.3 -389.9	4180.8	3792.2 134
42	Net Borrowing	207.2 5036.2	7102.7 9699.2	10278.8	587.5 657.3 3889.5 2402.6	3329.4	998.8 7

	Category	2013/2014	2014/2015	2015/2016	2016/2017	2017/2018	2016/2017(
43	Medium- and Long-Term Loans	-956.2 -482.5	-186.3 5156.8	6738.5	-9.7 993.6 2065.2 1238.4	2441.3	-488.8 80
44	Disbursements	1153.4 1753.5	2523.4 7641.1	8846.4	503.3 1563.0 2603.5 1879.4	2800.5	153.3 137
45	Repayments	-2109.6 -2236.0	-2709.7 -2484.3	-2107.9	-513.0 -569.4 -538.3 -641.0	-359.2	-642.1 -57
46	Medium- and Long-Term Suppliers' Credit	-56.3 257.9	1505.3 2795.1	1118.5	627.9 235.1 175.5 122.0	585.9	291.3 26
47	Disbursements	8.1 313.0	1560.7 2912.2	1313.6	664.7 275.5 233.2 175.1	629.8	328.8 30
48	Repayments	-64.4 -55.1	-55.4 -117.1	-195.1	-36.8 -40.4 -57.7 -53.1	-43.9	-37.5 -3
49	NaN	- 76 -	NaN	NaN	NaN	NaN	- 7

In [87]:
```
dfcle.Category
```

Out[87]:
```
4                                        Trade Balance
5                                   Export proceeds **
6                                    Petroleum exports
7                                        Other exports
8                                    Import payments**
9                                    Petroleum imports
10                                       Other imports
11                                    Services Balance
12                                            Receipts
13                                      Transportation
14                           Of which:  Suez Canal dues
15                          Travel ( tourism revenues )
16                                  Government receipts
17                                              Others
18                                            Payments
19                                      Transportation
20                                              Travel
21                                 Government expenditures
22                                              Others
23                           Investment Income Balance
24                                            Receipts
25                                            Payments
26                             Of which:  Interest paid
27                                    Current Transfers
28                                       Private (net),
29                                                 NaN
30    Of  which: Remittances of Egyptians working ab...
31                                         Official  (net)
32                           Balance of Current Account
33                          Capital & Financial Account
34                                     Capital Account
35                                   Financial Account
36                               Direct investment abroad
37                      Direct investment in Egypt (net)
38                           Portfolio investment abroad
39                  Portfolio investment in Egypt (Net)
40                                      Of which:Bonds
41                               Other Investments (Net)
42                                        Net Borrowing
43                           Medium- and Long-Term Loans
44                                        Disbursements
45                                           Repayments
46              Medium- and Long-Term Suppliers' Credit
47                                        Disbursements
48                                           Repayments
49                                                 NaN
Name: Category, dtype: object
```

In [88]:
```
dfcle.set_index('Category')
```

Out[88]:

Category	2013/2014	2014/2015	2015/2016	2016/2017	2017/2018	2016/2017(Q4)
Trade Balance	-34159.3 -39060.4	-38683.1 -37274.8	-37276.0	-8698.4 -8908.2 -9839.6 -9255.6	-9272.6	-9812.6 -9438.9
Export proceeds **	26022.6 22245.1	18704.6 21728.2	25827.0	5798.2 5839.4 6215.7 6755.8	7016.1	6785.2 7488.7
Petroleum exports	12355.9 8891.9	5674.3 6589.5	8773.0	1996.9 1782.5 2029.1 2202.5	2758.9	2810.0 3198.7
Other exports	13666.7 13353.2	13030.3 15138.7	17054.0	3801.3 4056.9 4186.6 4553.3	4257.2	3975.2 4290.0
Import payments**	-60181.9 -61305.5	-57387.7 -59003.0	-63103.0	-14496.6 -14747.6 -16055.3 -16011.4	-16288.7	-16597.8 -16927.6
Petroleum imports	-13246.9 -12366.1	-9293.6 -12015.5	-12489.8	-3337.6 -2752.9 -3231.7 -3410.0	-3095.2	-3415.6 -2442.3
Other imports	-46935.0 -48939.4	-48094.1 -46987.5	-50613.2	-11159.0 -11994.7 -12823.6 -12601.4	-13193.5	-13182.2 -14485.3
Services Balance	8274.4 10742.9	6533.0 5614.2	11122.4	2323.1 2847.3 2462.7 2528.8	3283.6	4283.1 2975.6
Receipts	17437.2 21811.8	16079.3 15400.1	21486.9	5080.8 5678.2 5068.2 5038.1	5702.4	6938.4 5894.8
Transportation	9466.0 9850.3	9534.6 7911.2	8707.9	2429.7 2267.9 2087.1 2029.5	2323.4	2242.8 2247.7
Of which: Suez Canal dues	5369.1 5361.7	5121.6 4945.3	5706.7	1228.7 1382.2 1386.3 1389.7	1548.5	1441.2 1487.1

Category	2013/2014	2014/2015	2015/2016	2016/2017	2017/2018	2016/2017(Q4)
Travel (tourism revenues)	5073.3 7370.4	3767.5 4379.7	9804.3	1539.0 2696.7 2282.5 2271.4	2553.7	3930.9 2859.1
Government receipts	654.4 1381.5	378.0 776.4	636.7	585.1 131.7 138.0 163.6	203.4	166.0 127.7
Others	2243.5 3209.6	2399.2 2332.8	2338.0	527.0 581.9 560.6 573.6	621.9	598.7 660.3
Payments	9162.8 11068.9	9546.3 9785.9	10364.5	2757.7 2830.9 2605.5 2509.3	2418.8	2655.3 2919.2
Transportation	1717.2 1535.0	1339.1 1332.1	1480.2	400.6 382.6 351.3 368.6	377.7	448.5 416.1
Travel	3044.5 3338.2	4091.0 2739.9	2451.5	550.2 649.3 511.9 542.2	748.1	717.0 659.5
Government expenditures	1073.9 854.1	777.1 1124.1	1493.5	424.0 449.0 540.1 238.4	266.0	183.1 175.3
Others	3327.2 5341.6	3339.1 4589.8	4939.3	1382.9 1350.0 1202.2 1360.1	1027.0	1306.7 1668.3
Investment Income Balance	-7262.7 -5700.9	-4471.7 -4568.5	-6279.6	-1249.8 -1519.3 -1520.3 -1664.3	-1575.7	-2048.0 -1769.5
Receipts	194.2 212.8	396.9 497.9	835.4	192.6 229.0 183.7 212.0	210.7	227.7 259.2
Payments	7456.9 5913.7	4868.6 5066.4	7115.0	1442.4 1748.3 1704.0 1876.3	1786.4	2275.7 2028.7
Of which: Interest paid	652.5 643.6	752.0 1231.9	1616.1	343.3 415.5 401.8 399.6	399.2	508.3 565.9
Current Transfers	30367.9 21875.8	16790.7 21835.1	26470.9	6071.9 5826.0 7112.4 6460.6	7071.9	5908.9 6047.8
Private (net),	18447.7 19205.4	16689.2 21686.1	26264.7	6005.4 5782.9 7087.2 6435.4	6959.2	5861.2 5932.0

Category	2013/2014	2014/2015	2015/2016	2016/2017	2017/2018	2016/2017(Q4)
NaN	(1)	NaN	NaN	NaN	NaN	NaN
Of which: Remittances of Egyptians working abroad	18518.7 19330.0	17077.4 21816.3	26392.9	6033.9 5824.3 7098.8 6464.4	7005.4	5909.0 6136.9
Official (net)	11920.2 2670.4	101.5 149.0	206.2	66.5 43.1 25.2 25.2	112.7	47.7 115.8
Balance of Current Account	-2779.7 -12142.6	-19831.1 -14394.0	-5962.3	-1553.2 -1754.2 -1784.8 -1930.5	-492.8	-1668.6 -2185.0
Capital & Financial Account	5189.5 17928.9	21176.7 31015.1	21996.5	4496.9 6247.4 4180.1 8617.1	2951.9	1475.6 310.5
Capital Account	194.1 -122.9	-141.4 -113.3	-150.7	-14.7 -40.3 -41.0 -37.0	-32.4	-35.0 -28.7
Financial Account	4995.4 18051.8	21318.1 31128.4	22147.2	4511.6 6287.7 4221.1 8654.1	2984.3	1510.6 339.2
Direct investment abroad	-326.6 -223.3	-164.2 -175.1	-271.2	-27.4 -52.4 -79.5 -68.1	-71.2	-65.8 -118.4
Direct investment in Egypt (net)	4178.2 6379.8	6932.6 7932.8	7719.5	1367.8 1843.0 1919.9 2256.3	1700.3	1099.9 1741.1
Portfolio investment abroad	65.9 47.2	192.1 208.4	-20.8	29.7 13.9 10.7 -49.7	4.3	-75.4 24.7
Portfolio investment in Egypt (Net)	1237.2 -638.6	-1286.8 15985.3	12094.8	8184.1 7478.5 540.7 6905.5	-2829.9	-3240.3 -2649.8
Of which:Bonds	926.7 -1147.5	-1444.8 5491.5	5293.2	2301.3 5.8 -108.9 3294.5	2101.8	-121.3 -182.3

Category	2013/2014	2014/2015	2015/2016	2016/2017	2017/2018	2016/2017(Q4)
Other Investments (Net)	-159.3 12486.7	15644.4 7177.0	2624.9	-5042.6 -2995.3 1829.3 -389.9	4180.8	3792.2 1341.6
Net Borrowing	207.2 5036.2	7102.7 9699.2	10278.8	587.5 657.3 3889.5 2402.6	3329.4	998.8 72.5
Medium- and Long-Term Loans	-956.2 -482.5	-186.3 5156.8	6738.5	-9.7 993.6 2065.2 1238.4	2441.3	-488.8 803.7
Disbursements	1153.4 1753.5	2523.4 7641.1	8846.4	503.3 1563.0 2603.5 1879.4	2800.5	153.3 1379.4
Repayments	-2109.6 -2236.0	-2709.7 -2484.3	-2107.9	-513.0 -569.4 -538.3 -641.0	-359.2	-642.1 -575.7
Medium- and Long-Term Suppliers' Credit	-56.3 257.9	1505.3 2795.1	1118.5	627.9 235.1 175.5 122.0	585.9	291.3 266.1
Disbursements	8.1 313.0	1560.7 2912.2	1313.6	664.7 275.5 233.2 175.1	629.8	328.8 305.6
Repayments	-64.4 -55.1	-55.4 -117.1	-195.1	-36.8 -40.4 -57.7 -53.1	-43.9	-37.5 -39.5
NaN	- 76 -	NaN	NaN	NaN	NaN	- 75 -

In [89]:
```
keep= [10,26,33]
dfc= dfcle.iloc[keep].copy()
zi= dfc.reset_index(drop=True)
zi
```

Out[89]:

	Category	2013/2014	2014/2015	2015/2016	2016/2017	2017/2018	2016/2017(Q4)
0	Of which: Suez Canal dues	5369.1 5361.7	5121.6 4945.3	5706.7	1228.7 1382.2 1386.3 1389.7	1548.5	1441.2 1487.1
1	Of which: Remittances of Egyptians working ab...	18518.7 19330.0	17077.4 21816.3	26392.9	6033.9 5824.3 7098.8 6464.4	7005.4	5909.0 6136.9
2	Direct investment in Egypt (net)	4178.2 6379.8	6932.6 7932.8	7719.5	1367.8 1843.0 1919.9 2256.3	1700.3	1099.9 1741.1

In [90]:
```
zi
```

Out[90]:

	Category	2013/2014	2014/2015	2015/2016	2016/2017	2017/2018	2016/2017(Q4)
0	Of which: Suez Canal dues	5369.1 5361.7	5121.6 4945.3	5706.7	1228.7 1382.2 1386.3 1389.7	1548.5	1441.2 1487.1
1	Of which: Remittances of Egyptians working ab...	18518.7 19330.0	17077.4 21816.3	26392.9	6033.9 5824.3 7098.8 6464.4	7005.4	5909.0 6136.9
2	Direct investment in Egypt (net)	4178.2 6379.8	6932.6 7932.8	7719.5	1367.8 1843.0 1919.9 2256.3	1700.3	1099.9 1741.1

In [91]:
```
import numpy as np
df=zi
cols_to_fix = [
    '2013/2014', '2014/2015', '2015/2016', '2016/2017', '2017/2018',
    '2016/2017(Q4)', '2017/2018(Q1)', '2017/2018(Q2)', '2017/2018(Q3)','2017/2018(Q
]

for col in cols_to_fix:
    df[col] = df[col].astype(str).replace('nan', '')

for i in range(len(cols_to_fix) - 1):
    current_col = cols_to_fix[i]
```

```
    next_col = cols_to_fix[i+1]


    for row_idx in df.index:
        cell_parts = df.at[row_idx, current_col].split()

        if len(cell_parts) > 1:
            df.at[row_idx, current_col] = cell_parts[0]
            extra_values = cell_parts[1:]
            existing_next_val = df.at[row_idx, next_col].split()
            new_next_content = extra_values + existing_next_val
            df.at[row_idx, next_col] = " ".join(new_next_content)

df.replace('', np.nan, inplace=True)
```

In [92]:
```
df1=df

cols_to_remove = ['2013/2014', '2014/2015', '2015/2016', '2016/2017', '2017/2018']
df1 = df1.drop(columns=cols_to_remove)
df1
```

Out[92]:

	Category	2016/2017(Q4)	2017/2018(Q1)	2017/2018(Q2)	2017/2018(Q3)	2017/2018
0	Of which: Suez Canal dues	1228.7	1382.2	1386.3	1389.7	15
1	Of which: Remittances of Egyptians working ab...	6033.9	5824.3	7098.8	6464.4	70
2	Direct investment in Egypt (net)	1367.8	1843.0	1919.9	2256.3	17

In [93]:
```
data_area= [0,0,100,85]
dflist2= tb.read_pdf('Bulletin_2018_8_Aug 2018_335.pdf', pages=46, area= data_area,
df2= dflist2[0]
df2
```

Out[93]:

	Unnamed: 0	Unnamed: 1	21-Balance of Payments	Unnamed: 2	Unnamed: 3	Unnamed: 4	Unnamed: 5	
0	NaN	NaN	NaN	(US$ mn)	NaN	NaN	NaN	
1	NaN	NaN	NaN	□□□□□□□ا□□□□ا	NaN	2015/2016*	NaN	2
2	NaN	During	NaN	Fiscal Years	NaN	□□ا□□ا□□□□ا	ا□ ول□□□□ا	
3	NaN	NaN	2012/2013 2013/2014	2014/2015 (1) 2015/2016* 2016/2017*	NaN	Q4	Q1	
4	Trade Balance	NaN	-30694.7 -34159.3	-39060.4 -38683.1 -35435.1	NaN	-8834.5	-9416.7	
5	Export proceeds **	NaN	26988.1 26022.6	22245.1 18704.6 21687.0	NaN	5298.9	5261.4	
6	Petroleum exports	NaN	13023.0 12355.9	8891.9 5674.3 6548.3	NaN	1463.2	1525.9	
7	Other exports	NaN	13965.1 13666.7	13353.2 13030.3 15138.7	NaN	3835.7	3735.5	
8	Import payments**	NaN	-57682.8 -60181.9	-61305.5 -57387.7 -57122.1	NaN	-14133.4	-14678.1	
9	Petroleum imports	NaN	-12124.2 -13246.9	-12366.1 -9293.6 -11196.7	NaN	-2221.5	-2746.7	
10	Other imports	NaN	-45558.6 -46935.0	-48939.4 -48094.1 -45925.4	NaN	-11911.9	-11931.4	
11	Services Balance	NaN	12445.8 8274.4	10742.9 6533.0 6811.1	NaN	1012.7	974.1	
12	Receipts	NaN	22026.6 17437.2	21811.8 16079.3 16597.0	NaN	3564.1	3327.7	

	Unnamed: 0	Unnamed: 1	21- Balance of Payments	Unnamed: 2	Unnamed: 3	Unnamed: 4	Unnamed: 5	
13	Transportation	NaN	9187.5 9466.0	9850.3 9534.6 9108.1	NaN	2281.6	1904.0	
14	Of which: Suez Canal dues	NaN	5031.8 5369.1	5361.7 5121.6 4945.3	NaN	1243.9	1300.4	
15	Travel (tourism revenues)	NaN	9751.8 5073.3	7370.4 3767.5 4379.7	NaN	510.4	758.2	
16	Government receipts	NaN	437.6 654.4	1381.5 378.0 776.4	NaN	99.3	62.5	
17	Others	NaN	2649.7 2243.5	3209.6 2399.2 2332.8	NaN	672.8	603.0	6
18	Payments	NaN	9580.8 9162.8	11068.9 9546.3 9785.9	NaN	2551.4	2353.6	
19	Transportation	NaN	1658.7 1717.2	1535.0 1339.1 1332.1	NaN	322.8	306.2	2
20	Travel	NaN	2928.8 3044.5	3338.2 4091.0 2739.9	NaN	1177.3	1105.2	6
21	Government expenditures	NaN	1243.7 1073.9	854.1 777.1 1124.1	NaN	327.2	157.0	2
22	Others	NaN	3749.6 3327.2	5341.6 3339.1 4589.8	NaN	724.1	785.2	
23	Investment Income Balance	NaN	-7406.4 -7262.7	-5700.9 -4471.7 -4423.0	NaN	-1357.0	-1129.5	
24	Receipts	NaN	197.8 194.2	212.8 396.9 497.9	NaN	123.7	81.6	
25	Payments	NaN	7604.2 7456.9	5913.7 4868.6 4920.9	NaN	1480.7	1211.1	
26	Of which: Interest paid	NaN	755.1 652.5	643.6 752.0 1143.5	NaN	227.6	258.9	3
27	Current Transfers	NaN	19264.9 30367.9	21875.8 16790.7	NaN	4388.2	4352.8	

	Unnamed: 0	Unnamed: 1	21-Balance of Payments	Unnamed: 2	Unnamed: 3	Unnamed: 4	Unnamed: 5
				17471.8			
28	Private (net),	NaN	18429.3 18447.7	19205.4 16689.2 17322.8	NaN	4347.4	4319.0
29	NaN	NaN	(2)	NaN	NaN	NaN	NaN
30	Of which: Remittances of Egyptians working ab...	NaN	18668.0 18518.7	19330.0 17077.4 17453.0	NaN	4417.8	4354.9
31	Official (net)	NaN	835.6 11920.2	2670.4 101.5 149.0	NaN	40.8	33.8
32	Balance of Current Account	NaN	-6390.4 -2779.7	-12142.6 -19831.1 -15575.2	NaN	-4790.6	-5219.3
33	Capital & Financial Account	NaN	9773.0 5189.5	17928.9 21176.7 29034.2	NaN	6626.6	8029.7
34	Capital Account	NaN	-86.8 194.1	-122.9 -141.4 -113.3	NaN	-10.6	-9.4
35	Financial Account	NaN	9859.8 4995.4	18051.8 21318.1 29147.5	NaN	6637.2	8039.1
36	Direct investment abroad	NaN	-183.6 -326.6	-223.3 -164.2 -175.1	NaN	-50.7	-62.0
37	Direct investment in Egypt (net)	NaN	3753.3 4178.2	6379.8 6932.6 7915.8	NaN	1046.9	1872.2
38	Portfolio investment abroad	NaN	22.4 65.9	47.2 192.1 208.4	NaN	43.5	27.7
39	Portfolio investment in Egypt (Net)	NaN	1477.4 1237.2	-638.6 -1286.8 15985.3	NaN	214.6	-840.9
40	Of which:Bonds	NaN	2257.9 926.7	-1147.5 -1444.8 5491.5	NaN	-21.1	-832.5
41	Other Investments	NaN	4790.3 -159.3	12486.7 15644.4	NaN	5382.9	7042.1

	Unnamed: 0	Unnamed: 1	21- Balance of Payments	Unnamed: 2	Unnamed: 3	Unnamed: 4	Unnamed: 5
	(Net)			5213.1			
42	Net Borrowing	NaN	1174.1 207.2	5036.2 7102.7 7735.3	NaN	1830.3	2240.8
43	Medium- and Long-Term Loans	NaN	750.4 -956.2	-482.5 -186.3 4133.4	NaN	-8.9	992.6
44	Disbursements	NaN	2709.6 1153.4	1753.5 2523.4 6679.1	NaN	447.2	1918.6
45	Repayments	NaN	-1959.2 -2109.6	-2236.0 -2709.7 -2545.7	NaN	-456.1	-926.0
46	Medium- and Long-Term Suppliers' Credit	NaN	-18.2 -56.3	257.9 1505.3 1516.4	NaN	823.8	677.4
47	Disbursements	NaN	43.4 8.1	313.0 1560.7 1637.3	NaN	857.2	694.8
48	Repayments	NaN	-61.6 -64.4	-55.1 -55.4 -120.9	NaN	-33.4	-17.4
49	NaN	NaN	- 76 -	NaN	NaN	NaN	NaN

In [94]:
```
df22= df2.iloc[4:].copy()
df22= df22.dropna(axis=1,how= 'all')
newnames= ['Category',"2012/2013", "2013/2014", "2014/2015", "2015/2016", "2016/201
df22
```

Out[94]:

	Unnamed: 0	21-Balance of Payments	Unnamed: 2	Unnamed: 4	Unnamed: 5	Unnamed: 6	Unnamed: 7	ات
4	Trade Balance	-30694.7 -34159.3	-39060.4 -38683.1 -35435.1	-8834.5	-9416.7	-9599.6 -9347.0	-8387.5	
5	Export proceeds **	26988.1 26022.6	22245.1 18704.6 21687.0	5298.9	5261.4	5185.1 5547.5	5693.0	
6	Petroleum exports	13023.0 12355.9	8891.9 5674.3 6548.3	1463.2	1525.9	1409.4 1721.3	1891.7	
7	Other exports	13965.1 13666.7	13353.2 13030.3 15138.7	3835.7	3735.5	3775.7 3826.2	3801.3	
8	Import payments**	-57682.8 -60181.9	-61305.5 -57387.7 -57122.1	-14133.4	-14678.1	-14784.7 -14894.5	-14080.5	
9	Petroleum imports	-12124.2 -13246.9	-12366.1 -9293.6 -11196.7	-2221.5	-2746.7	-2589.5 -3192.6	-2999.6	
10	Other imports	-45558.6 -46935.0	-48939.4 -48094.1 -45925.4	-11911.9	-11931.4	-12195.2 -11701.9	-11080.9	
11	Services Balance	12445.8 8274.4	10742.9 6533.0 6811.1	1012.7	974.1	784.1 1532.9	2323.1	
12	Receipts	22026.6 17437.2	21811.8 16079.3 16597.0	3564.1	3327.7	3283.5 3708.1	5080.8	
13	Transportation	9187.5 9466.0	9850.3 9534.6 9108.1	2281.6	1904.0	1762.4 1815.1	2429.7	
14	Of which: Suez Canal dues	5031.8 5369.1	5361.7 5121.6 4945.3	1243.9	1300.4	1214.2 1202.0	1228.7	
15	Travel (tourism revenues)	9751.8 5073.3	7370.4 3767.5 4379.7	510.4	758.2	825.8 1256.7	1539.0	
16	Government receipts	437.6 654.4	1381.5 378.0 776.4	99.3	62.5	60.9 67.9	585.1	1

	Unnamed: 0	21-Balance of Payments	Unnamed: 2	Unnamed: 4	Unnamed: 5	Unnamed: 6	Unnamed: 7	ت
17	Others	2649.7 2243.5	3209.6 2399.2 2332.8	672.8	603.0	634.4 568.4	527.0	5
18	Payments	9580.8 9162.8	11068.9 9546.3 9785.9	2551.4	2353.6	2499.4 2175.2	2757.7	
19	Transportation	1658.7 1717.2	1535.0 1339.1 1332.1	322.8	306.2	291.1 334.2	400.6	3
20	Travel	2928.8 3044.5	3338.2 4091.0 2739.9	1177.3	1105.2	636.2 448.3	550.2	6
21	Government expenditures	1243.7 1073.9	854.1 777.1 1124.1	327.2	157.0	217.8 325.3	424.0	4
22	Others	3749.6 3327.2	5341.6 3339.1 4589.8	724.1	785.2	1354.3 1067.4	1382.9	
23	Investment Income Balance	-7406.4 -7262.7	-5700.9 -4471.7 -4423.0	-1357.0	-1129.5	-1134.8 -1054.4	-1197.1	
24	Receipts	197.8 194.2	212.8 396.9 497.9	123.7	81.6	94.4 129.3	192.6	2
25	Payments	7604.2 7456.9	5913.7 4868.6 4920.9	1480.7	1211.1	1229.2 1183.7	1389.7	
26	Of which: Interest paid	755.1 652.5	643.6 752.0 1143.5	227.6	258.9	306.4 323.3	318.9	4
27	Current Transfers	19264.9 30367.9	21875.8 16790.7 17471.8	4388.2	4352.8	5755.3 5764.0	4865.1	
28	Private (net),	18429.3 18447.7	19205.4 16689.2 17322.8	4347.4	4319.0	5716.2 5754.4	4798.6	
29	NaN	(2)	NaN	NaN	NaN	NaN	NaN	

	Unnamed: 0	21- Balance of Payments	Unnamed: 2	Unnamed: 4	Unnamed: 5	Unnamed: 6	Unnamed: 7	ات
30	Of which: Remittances of Egyptians working ab...	18668.0 18518.7	19330.0 17077.4 17453.0	4417.8	4354.9	5756.0 5780.4	4827.1	
31	Official (net)	835.6 11920.2	2670.4 101.5 149.0	40.8	33.8	39.1 9.6	66.5	
32	Balance of Current Account	-6390.4 -2779.7	-12142.6 -19831.1 -15575.2	-4790.6	-5219.3	-4195.0 -3104.5	-2396.4	
33	Capital & Financial Account	9773.0 5189.5	17928.9 21176.7 29034.2	6626.6	8029.7	10687.8 8358.9	4395.3	
34	Capital Account	-86.8 194.1	-122.9 -141.4 -113.3	-10.6	-9.4	-29.6 -59.6	-14.7	-
35	Financial Account	9859.8 4995.4	18051.8 21318.1 29147.5	6637.2	8039.1	10717.4 8418.5	4410.0	
36	Direct investment abroad	-183.6 -326.6	-223.3 -164.2 -175.1	-50.7	-62.0	-46.0 -39.7	-27.4	-
37	Direct investment in Egypt (net)	3753.3 4178.2	6379.8 6932.6 7915.8	1046.9	1872.2	2414.8 2278.0	1350.8	
38	Portfolio investment abroad	22.4 65.9	47.2 192.1 208.4	43.5	27.7	107.0 44.0	29.7	
39	Portfolio investment in Egypt (Net)	1477.4 1237.2	-638.6 -1286.8 15985.3	214.6	-840.9	1053.8 7588.3	8184.1	74
40	Of which:Bonds	2257.9 926.7	-1147.5 -1444.8 5491.5	-21.1	-832.5	27.0 3995.7	2301.3	
41	Other Investments (Net)	4790.3 -159.3	12486.7 15644.4 5213.1	5382.9	7042.1	7187.8 -1452.1	-5127.2	
42	Net Borrowing	1174.1 207.2	5036.2 7102.7 7735.3	1830.3	2240.8	4842.6 2586.5	502.9	65

	Unnamed: 0	21-Balance of Payments	Unnamed: 2	Unnamed: 4	Unnamed: 5	Unnamed: 6	Unnamed: 7	ت
43	Medium- and Long-Term Loans	750.4 -956.2	-482.5 -186.3 4133.4	-8.9	992.6	2854.0 1319.9	-47.3	99
44	Disbursements	2709.6 1153.4	1753.5 2523.4 6679.1	447.2	1918.6	3348.2 1871.0	465.0	
45	Repayments	-1959.2 -2109.6	-2236.0 -2709.7 -2545.7	-456.1	-926.0	-494.2 -551.1	-512.3	
46	Medium- and Long-Term Suppliers' Credit	-18.2 -56.3	257.9 1505.3 1516.4	823.8	677.4	516.5 973.3	586.7	2
47	Disbursements	43.4 8.1	313.0 1560.7 1637.3	857.2	694.8	553.6 999.1	625.0	2
48	Repayments	-61.6 -64.4	-55.1 -55.4 -120.9	-33.4	-17.4	-37.1 -25.8	-38.3	-
49	NaN	- 76 -	NaN	NaN	NaN	NaN	NaN	

In [95]:
```
df22.columns= newnames[:len(df22.columns)]
df22= df22.reindex(columns=newnames)
df22.set_index('Category')
df22
```

Out[95]:

	Category	2012/2013	2013/2014	2014/2015	2015/2016	2016/2017	2015/2016(
4	Trade Balance	-30694.7 -34159.3	-39060.4 -38683.1 -35435.1	-8834.5	-9416.7	-9599.6 -9347.0	-838
5	Export proceeds **	26988.1 26022.6	22245.1 18704.6 21687.0	5298.9	5261.4	5185.1 5547.5	569
6	Petroleum exports	13023.0 12355.9	8891.9 5674.3 6548.3	1463.2	1525.9	1409.4 1721.3	189
7	Other exports	13965.1 13666.7	13353.2 13030.3 15138.7	3835.7	3735.5	3775.7 3826.2	380
8	Import payments**	-57682.8 -60181.9	-61305.5 -57387.7 -57122.1	-14133.4	-14678.1	-14784.7 -14894.5	-1408
9	Petroleum imports	-12124.2 -13246.9	-12366.1 -9293.6 -11196.7	-2221.5	-2746.7	-2589.5 -3192.6	-299
10	Other imports	-45558.6 -46935.0	-48939.4 -48094.1 -45925.4	-11911.9	-11931.4	-12195.2 -11701.9	-1108
11	Services Balance	12445.8 8274.4	10742.9 6533.0 6811.1	1012.7	974.1	784.1 1532.9	232
12	Receipts	22026.6 17437.2	21811.8 16079.3 16597.0	3564.1	3327.7	3283.5 3708.1	508
13	Transportation	9187.5 9466.0	9850.3 9534.6 9108.1	2281.6	1904.0	1762.4 1815.1	242
14	Of which: Suez Canal dues	5031.8 5369.1	5361.7 5121.6 4945.3	1243.9	1300.4	1214.2 1202.0	122
15	Travel (tourism revenues)	9751.8 5073.3	7370.4 3767.5 4379.7	510.4	758.2	825.8 1256.7	153
16	Government receipts	437.6 654.4	1381.5 378.0 776.4	99.3	62.5	60.9 67.9	58
17	Others	2649.7 2243.5	3209.6 2399.2 2332.8	672.8	603.0	634.4 568.4	52
18	Payments	9580.8 9162.8	11068.9 9546.3	2551.4	2353.6	2499.4 2175.2	275

	Category	2012/2013	2013/2014	2014/2015	2015/2016	2016/2017	2015/2016(
			9785.9				
19	Transportation	1658.7 1717.2	1535.0 1339.1 1332.1	322.8	306.2	291.1 334.2	40
20	Travel	2928.8 3044.5	3338.2 4091.0 2739.9	1177.3	1105.2	636.2 448.3	55
21	Government expenditures	1243.7 1073.9	854.1 777.1 1124.1	327.2	157.0	217.8 325.3	42
22	Others	3749.6 3327.2	5341.6 3339.1 4589.8	724.1	785.2	1354.3 1067.4	138
23	Investment Income Balance	-7406.4 -7262.7	-5700.9 -4471.7 -4423.0	-1357.0	-1129.5	-1134.8 -1054.4	-119
24	Receipts	197.8 194.2	212.8 396.9 497.9	123.7	81.6	94.4 129.3	19
25	Payments	7604.2 7456.9	5913.7 4868.6 4920.9	1480.7	1211.1	1229.2 1183.7	138
26	Of which: Interest paid	755.1 652.5	643.6 752.0 1143.5	227.6	258.9	306.4 323.3	31
27	Current Transfers	19264.9 30367.9	21875.8 16790.7 17471.8	4388.2	4352.8	5755.3 5764.0	486
28	Private (net),	18429.3 18447.7	19205.4 16689.2 17322.8	4347.4	4319.0	5716.2 5754.4	479
29	NaN	(2)	NaN	NaN	NaN	NaN	N
30	Of which: Remittances of Egyptians working ab...	18668.0 18518.7	19330.0 17077.4 17453.0	4417.8	4354.9	5756.0 5780.4	482
31	Official (net)	835.6 11920.2	2670.4 101.5 149.0	40.8	33.8	39.1 9.6	6
32	Balance of Current Account	-6390.4 -2779.7	-12142.6 -19831.1 -15575.2	-4790.6	-5219.3	-4195.0 -3104.5	-239
33	Capital & Financial Account	9773.0 5189.5	17928.9 21176.7 29034.2	6626.6	8029.7	10687.8 8358.9	439

	Category	2012/2013	2013/2014	2014/2015	2015/2016	2016/2017	2015/2016(
34	Capital Account	-86.8 194.1	-122.9 -141.4 -113.3	-10.6	-9.4	-29.6 -59.6	-1
35	Financial Account	9859.8 4995.4	18051.8 21318.1 29147.5	6637.2	8039.1	10717.4 8418.5	441
36	Direct investment abroad	-183.6 -326.6	-223.3 -164.2 -175.1	-50.7	-62.0	-46.0 -39.7	-2
37	Direct investment in Egypt (net)	3753.3 4178.2	6379.8 6932.6 7915.8	1046.9	1872.2	2414.8 2278.0	135
38	Portfolio investment abroad	22.4 65.9	47.2 192.1 208.4	43.5	27.7	107.0 44.0	2
39	Portfolio investment in Egypt (Net)	1477.4 1237.2	-638.6 -1286.8 15985.3	214.6	-840.9	1053.8 7588.3	818
40	Of which:Bonds	2257.9 926.7	-1147.5 -1444.8 5491.5	-21.1	-832.5	27.0 3995.7	230
41	Other Investments (Net)	4790.3 -159.3	12486.7 15644.4 5213.1	5382.9	7042.1	7187.8 -1452.1	-512
42	Net Borrowing	1174.1 207.2	5036.2 7102.7 7735.3	1830.3	2240.8	4842.6 2586.5	50
43	Medium- and Long-Term Loans	750.4 -956.2	-482.5 -186.3 4133.4	-8.9	992.6	2854.0 1319.9	-4
44	Disbursements	2709.6 1153.4	1753.5 2523.4 6679.1	447.2	1918.6	3348.2 1871.0	46
45	Repayments	-1959.2 -2109.6	-2236.0 -2709.7 -2545.7	-456.1	-926.0	-494.2 -551.1	-51
46	Medium- and Long-Term Suppliers' Credit	-18.2 -56.3	257.9 1505.3 1516.4	823.8	677.4	516.5 973.3	58
47	Disbursements	43.4 8.1	313.0 1560.7 1637.3	857.2	694.8	553.6 999.1	62

	Category	2012/2013	2013/2014	2014/2015	2015/2016	2016/2017	2015/2016(
48	Repayments	-61.6 -64.4	-55.1 -55.4 -120.9	-33.4	-17.4	-37.1 -25.8	-3
49	NaN	- 76 -	NaN	NaN	NaN	NaN	N

In [96]:
```
keep= [10,26,33]
d23= df22.iloc[keep].copy()
d24= d23.reset_index(drop=True)
d24
```

Out[96]:

	Category	2012/2013	2013/2014	2014/2015	2015/2016	2016/2017	2015/2016(Q4)
0	Of which: Suez Canal dues	5031.8 5369.1	5361.7 5121.6 4945.3	1243.9	1300.4	1214.2 1202.0	1228.7
1	Of which: Remittances of Egyptians working ab...	18668.0 18518.7	19330.0 17077.4 17453.0	4417.8	4354.9	5756.0 5780.4	4827.1
2	Direct investment in Egypt (net)	3753.3 4178.2	6379.8 6932.6 7915.8	1046.9	1872.2	2414.8 2278.0	1350.8

In [97]:
```
df2=d24
cols_to_fix = [
    '2012/2013', "2013/2014", "2014/2015", "2015/2016", "2016/2017", "2015/2016(Q4)"
]

for col in cols_to_fix:
    df2[col] = df2[col].astype(str).replace('nan', '')

for i in range(len(cols_to_fix) - 1):
    current_col = cols_to_fix[i]
    next_col = cols_to_fix[i+1]


    for row_idx in df2.index:
        cell_parts = df2.at[row_idx, current_col].split()

        if len(cell_parts) > 1:
            df2.at[row_idx, current_col] = cell_parts[0]
            extra_values = cell_parts[1:]
            existing_next_val = df2.at[row_idx, next_col].split()
            new_next_content = extra_values + existing_next_val
            df2.at[row_idx, next_col] = " ".join(new_next_content)
```

```
df2.replace('', np.nan, inplace=True)
df2
```

Out[97]:

	Category	2012/2013	2013/2014	2014/2015	2015/2016	2016/2017	2015/2016(Q4)
0	Of which: Suez Canal dues	5031.8	5369.1	5361.7	5121.6	4945.3	1243.9
1	Of which: Remittances of Egyptians working ab...	18668.0	18518.7	19330.0	17077.4	17453.0	4417.8
2	Direct investment in Egypt (net)	3753.3	4178.2	6379.8	6932.6	7915.8	1046.9

In [98]:

```
df2
cols_to_remove = ['2012/2013','2013/2014', '2014/2015', '2015/2016', '2016/2017']
df2 = df2.drop(columns=cols_to_remove)
df2
```

Out[98]:

	Category	2015/2016(Q4)	2016/2017(Q1)	2016/2017(Q2)	2016/2017(Q3)	2016/2017
0	Of which: Suez Canal dues	1243.9	1300.4	1214.2	1202.0	12
1	Of which: Remittances of Egyptians working ab...	4417.8	4354.9	5756.0	5780.4	48
2	Direct investment in Egypt (net)	1046.9	1872.2	2414.8	2278.0	13

In [99]:

```
data_area= [0,0,100,85]
dflist3= tb.read_pdf('Bulletin_281_Aug 2020_359.pdf', pages=46, area= data_area,rel
df3= dflist3[0]
df3
```

Out[99]:

	Unnamed: 0	Unnamed: 1	Unnamed: 2	Unnamed: 3	21-Balance of Payments	Unnamed: 4	Unnamed: 5
0	NaN	NaN	NaN	NaN	(US$ mn)	NaN	NaN
1	NaN	NaN	NaN	NaN	□□□□□□□ا □□□□ا	NaN	NaN
2	NaN	During	NaN	NaN	Fiscal Years	NaN	NaN
3	NaN	NaN	NaN	2014/2015	2015/2016 2016/2017 2017/2018* (+)	NaN	(+)2018/2019*
4	Trade Balance	NaN	NaN	-39060.4	-38683.1 -37274.8 -37276.0	NaN	-38034.4
5	Export proceeds **	NaN	NaN	22245.1	18704.6 21728.2 25827.0	NaN	28495.0
6	Petroleum exports	NaN	NaN	8891.9	5674.3 6589.5 8773.0	NaN	11557.0
7	Other exports	NaN	NaN	13353.2	13030.3 15138.7 17054.0	NaN	16938.0
8	Import payments**	NaN	NaN	-61305.5	-57387.7 -59003.0 -63103.0	NaN	-66529.4
9	Petroleum imports	NaN	NaN	-12366.1	-9293.6 -12015.5 -12489.8	NaN	-11548.9
10	Other imports	NaN	NaN	-48939.4	-48094.1 -46987.5 -50613.2	NaN	-54980.5
11	Services Balance	NaN	NaN	10742.9	6533.0 5614.2 11122.4	NaN	13036.5
12	Receipts	NaN	NaN	21811.8	16079.3 15400.1 21486.9	NaN	24423.6

	Unnamed: 0	Unnamed: 1	Unnamed: 2	Unnamed: 3	21-Balance of Payments	Unnamed: 4	Unnamed: 5
13	Transportation	NaN	NaN	9850.3	9534.6 7911.2 8707.9	NaN	8600.3
14	Of which: Suez Canal dues	NaN	NaN	5361.7	5121.6 4945.3 5706.7	NaN	5730.7
15	Travel (tourism revenues)	NaN	NaN	7370.4	3767.5 4379.7 9804.3	NaN	12570.6
16	Government receipts	NaN	NaN	1381.5	378.0 776.4 636.7	NaN	718.8
17	Others	NaN	NaN	3209.6	2399.2 2332.8 2338.0	NaN	2533.9
18	Payments	NaN	NaN	11068.9	9546.3 9785.9 10364.5	NaN	11387.1
19	Transportation	NaN	NaN	1535.0	1339.1 1332.1 1480.2	NaN	1792.4
20	Travel	NaN	NaN	3338.2	4091.0 2739.9 2451.5	NaN	2902.9
21	Government expenditures	NaN	NaN	854.1	777.1 1124.1 1493.5	NaN	692.4
22	Others	NaN	NaN	5341.6	3339.1 4589.8 4939.3	NaN	5999.4
23	Investment Income Balance	NaN	NaN	-5700.9	-4471.7 -4568.5 -6279.6	NaN	-11009.6
24	Receipts	NaN	NaN	212.8	396.9 497.9 835.4	NaN	1014.1
25	(++)	NaN	NaN	NaN	NaN	NaN	NaN
26	Payments	NaN	NaN	5913.7	4868.6 5066.4 7115.0	NaN	12023.7
27	Of which: Interest paid	NaN	NaN	643.6	752.0 1231.9 1616.1	NaN	2574.1

	Unnamed: 0	Unnamed: 1	Unnamed: 2	Unnamed: 3	21-Balance of Payments	Unnamed: 4	Unnamed: 5
28	Current Transfers	NaN	NaN	21875.8	16790.7 21835.1 26470.9	NaN	25113.6
29	Private (net),	NaN	NaN	19205.4	16689.2 21686.1 26264.7	NaN	24763.1
30	NaN	(1)	NaN	NaN	NaN	NaN	NaN
31	Of which: Remittances of Egyptians working ab...	NaN	NaN	19330.0	17077.4 21816.3 26392.9	NaN	25150.8
32	Official (net)	NaN	NaN	2670.4	101.5 149.0 206.2	NaN	350.5
33	Balance of Current Account	NaN	NaN	-12142.6	-19831.1 -14394.0 -5962.3	NaN	-10893.9
34	Capital & Financial Account	NaN	NaN	17928.9	21176.7 31015.1 21996.5	NaN	10856.9
35	Capital Account	NaN	NaN	-122.9	-141.4 -113.3 -150.7	NaN	-129.2
36	Financial Account	NaN	NaN	18051.8	21318.1 31128.4 22147.2	NaN	10986.1
37	Direct investment abroad	NaN	NaN	-223.3	-164.2 -175.1 -271.2	NaN	-374.0
38	NaN	(++)	NaN	NaN	NaN	NaN	NaN
39	Direct investment in Egypt (net)	NaN	NaN	6379.8	6932.6 7932.8 7719.5	NaN	8236.3
40	Portfolio investment abroad	NaN	NaN	47.2	192.1 208.4 -20.8	NaN	-96.4
41	Portfolio investment in Egypt (Net)	NaN	NaN	-638.6	-1286.8 15985.3 12094.8	NaN	4230.1
42	Of which:Bonds	NaN	NaN	-1147.5	-1444.8 5491.5 5293.2	NaN	5094.2

	Unnamed: 0	Unnamed: 1	Unnamed: 2	Unnamed: 3	21-Balance of Payments	Unnamed: 4	Unnamed: 5
43	Other Investments (Net)	NaN	NaN	12486.7	15644.4 7177.0 2624.9	NaN	-1009.9
44	Net Borrowing	NaN	NaN	5036.2	7102.7 9699.2 10278.8	NaN	6253.4
45	Medium- and Long-Term Loans	NaN	NaN	-482.5	-186.3 5156.8 6738.5	NaN	3333.7
46	Disbursements	NaN	NaN	1753.5	2523.4 7641.1 8846.4	NaN	5525.2
47	Repayments	NaN	NaN	-2236.0	-2709.7 -2484.3 -2107.9	NaN	-2191.5
48	Medium- and Long-Term Suppliers' Credit	NaN	NaN	257.9	1505.3 2795.1 1118.5	NaN	828.8
49	Disbursements	NaN	NaN	313.0	1560.7 2912.2 1313.6	NaN	1160.8
50	Repayments	NaN	NaN	-55.1	-55.4 -117.1 -195.1	NaN	-332.0
51	NaN	NaN	NaN	NaN	- 76 -	NaN	NaN

In [100…

```
df31= df3.iloc[4:].copy()
df31= df31.dropna(axis=1,how= 'all')
newnames= ['Category',"2014/2015", "2015/2016", "2016/2017", "2017/2018", "2018/201
df31
```

Out[100...

	Unnamed: 0	Unnamed: 1	Unnamed: 3	21-Balance of Payments	Unnamed: 5	Unnamed: 7	Unnamed: 9	Un
4	Trade Balance	NaN	-39060.4	-38683.1 -37274.8 -37276.0	-38034.4	-9272.6	-9812.6	
5	Export proceeds **	NaN	22245.1	18704.6 21728.2 25827.0	28495.0	7016.1	6785.2	
6	Petroleum exports	NaN	8891.9	5674.3 6589.5 8773.0	11557.0	2758.9	2810.0	
7	Other exports	NaN	13353.2	13030.3 15138.7 17054.0	16938.0	4257.2	3975.2	
8	Import payments**	NaN	-61305.5	-57387.7 -59003.0 -63103.0	-66529.4	-16288.7	-16597.8	
9	Petroleum imports	NaN	-12366.1	-9293.6 -12015.5 -12489.8	-11548.9	-3095.2	-3415.6	
10	Other imports	NaN	-48939.4	-48094.1 -46987.5 -50613.2	-54980.5	-13193.5	-13182.2	
11	Services Balance	NaN	10742.9	6533.0 5614.2 11122.4	13036.5	3283.6	4283.1	
12	Receipts	NaN	21811.8	16079.3 15400.1 21486.9	24423.6	5702.4	6938.4	
13	Transportation	NaN	9850.3	9534.6 7911.2 8707.9	8600.3	2323.4	2242.8	
14	Of which: Suez Canal dues	NaN	5361.7	5121.6 4945.3 5706.7	5730.7	1548.5	1441.2	
15	Travel (tourism revenues)	NaN	7370.4	3767.5 4379.7 9804.3	12570.6	2553.7	3930.9	
16	Government receipts	NaN	1381.5	378.0 776.4 636.7	718.8	203.4	166.0	

	Unnamed: 0	Unnamed: 1	Unnamed: 3	21-Balance of Payments	Unnamed: 5	Unnamed: 7	Unnamed: 9	Un
17	Others	NaN	3209.6	2399.2 2332.8 2338.0	2533.9	621.9	598.7	
18	Payments	NaN	11068.9	9546.3 9785.9 10364.5	11387.1	2418.8	2655.3	
19	Transportation	NaN	1535.0	1339.1 1332.1 1480.2	1792.4	377.7	448.5	
20	Travel	NaN	3338.2	4091.0 2739.9 2451.5	2902.9	748.1	717.0	
21	Government expenditures	NaN	854.1	777.1 1124.1 1493.5	692.4	266.0	183.1	
22	Others	NaN	5341.6	3339.1 4589.8 4939.3	5999.4	1027.0	1306.7	
23	Investment Income Balance	NaN	-5700.9	-4471.7 -4568.5 -6279.6	-11009.6	-1575.7	-2391.4	
24	Receipts	NaN	212.8	396.9 497.9 835.4	1014.1	210.7	227.7	
25	(++)	NaN	NaN	NaN	NaN	NaN	NaN	
26	Payments	NaN	5913.7	4868.6 5066.4 7115.0	12023.7	1786.4	2619.1	
27	Of which: Interest paid	NaN	643.6	752.0 1231.9 1616.1	2574.1	399.2	508.3	
28	Current Transfers	NaN	21875.8	16790.7 21835.1 26470.9	25113.6	7071.9	5908.9	
29	Private (net),	NaN	19205.4	16689.2 21686.1 26264.7	24763.1	6959.2	5861.2	
30	NaN	(1)	NaN	NaN	NaN	NaN	NaN	
31	Of which: Remittances of	NaN	19330.0	17077.4 21816.3	25150.8	7005.4	5909.0	

	Unnamed: 0	Unnamed: 1	Unnamed: 3	21-Balance of Payments	Unnamed: 5	Unnamed: 7	Unnamed: 9	Un
	Egyptians working ab...			26392.9				
32	Official (net)	NaN	2670.4	101.5 149.0 206.2	350.5	112.7	47.7	
33	Balance of Current Account	NaN	-12142.6	-19831.1 -14394.0 -5962.3	-10893.9	-492.8	-2012.0	
34	Capital & Financial Account	NaN	17928.9	21176.7 31015.1 21996.5	10856.9	2951.9	1791.1	
35	Capital Account	NaN	-122.9	-141.4 -113.3 -150.7	-129.2	-32.4	-35.0	-2
36	Financial Account	NaN	18051.8	21318.1 31128.4 22147.2	10986.1	2984.3	1826.1	
37	Direct investment abroad	NaN	-223.3	-164.2 -175.1 -271.2	-374.0	-71.2	-65.8	
38	NaN	(++)	NaN	NaN	NaN	NaN	NaN	
39	Direct investment in Egypt (net)	NaN	6379.8	6932.6 7932.8 7719.5	8236.3	1700.3	1415.4	
40	Portfolio investment abroad	NaN	47.2	192.1 208.4 -20.8	-96.4	4.3	-75.4	
41	Portfolio investment in Egypt (Net)	NaN	-638.6	-1286.8 15985.3 12094.8	4230.1	-2829.9	-3240.3	
42	Of which:Bonds	NaN	-1147.5	-1444.8 5491.5 5293.2	5094.2	2101.8	-121.3	
43	Other Investments (Net)	NaN	12486.7	15644.4 7177.0 2624.9	-1009.9	4180.8	3792.2	
44	Net Borrowing	NaN	5036.2	7102.7 9699.2 10278.8	6253.4	3329.4	998.8	

	Unnamed: 0	Unnamed: 1	Unnamed: 3	21- Balance of Payments	Unnamed: 5	Unnamed: 7	Unnamed: 9	Un
45	Medium- and Long-Term Loans	NaN	-482.5	-186.3 5156.8 6738.5	3333.7	2441.3	-488.8	
46	Disbursements	NaN	1753.5	2523.4 7641.1 8846.4	5525.2	2800.5	153.3	
47	Repayments	NaN	-2236.0	-2709.7 -2484.3 -2107.9	-2191.5	-359.2	-642.1	
48	Medium- and Long-Term Suppliers' Credit	NaN	257.9	1505.3 2795.1 1118.5	828.8	585.9	291.3	
49	Disbursements	NaN	313.0	1560.7 2912.2 1313.6	1160.8	629.8	328.8	
50	Repayments	NaN	-55.1	-55.4 -117.1 -195.1	-332.0	-43.9	-37.5	
51	NaN	NaN	NaN	- 76 -	NaN	NaN	NaN	

In [101…

```
df31.columns= newnames[:len(df31.columns)]
df31= df31.reindex(columns=newnames)
df31.set_index('Category')
df31
```

Out[101...

	Category	2014/2015	2015/2016	2016/2017	2017/2018	2018/2019	2017/2018(
4	Trade Balance	NaN	-39060.4	-38683.1 -37274.8 -37276.0	-38034.4	-9272.6	-981
5	Export proceeds **	NaN	22245.1	18704.6 21728.2 25827.0	28495.0	7016.1	678
6	Petroleum exports	NaN	8891.9	5674.3 6589.5 8773.0	11557.0	2758.9	281
7	Other exports	NaN	13353.2	13030.3 15138.7 17054.0	16938.0	4257.2	397
8	Import payments**	NaN	-61305.5	-57387.7 -59003.0 -63103.0	-66529.4	-16288.7	-1659
9	Petroleum imports	NaN	-12366.1	-9293.6 -12015.5 -12489.8	-11548.9	-3095.2	-341
10	Other imports	NaN	-48939.4	-48094.1 -46987.5 -50613.2	-54980.5	-13193.5	-1318
11	Services Balance	NaN	10742.9	6533.0 5614.2 11122.4	13036.5	3283.6	428
12	Receipts	NaN	21811.8	16079.3 15400.1 21486.9	24423.6	5702.4	693
13	Transportation	NaN	9850.3	9534.6 7911.2 8707.9	8600.3	2323.4	224
14	Of which: Suez Canal dues	NaN	5361.7	5121.6 4945.3 5706.7	5730.7	1548.5	144
15	Travel (tourism revenues)	NaN	7370.4	3767.5 4379.7 9804.3	12570.6	2553.7	393
16	Government receipts	NaN	1381.5	378.0 776.4 636.7	718.8	203.4	16
17	Others	NaN	3209.6	2399.2 2332.8 2338.0	2533.9	621.9	59
18	Payments	NaN	11068.9	9546.3 9785.9	11387.1	2418.8	265

	Category	2014/2015	2015/2016	2016/2017	2017/2018	2018/2019	2017/2018(
				10364.5			
19	Transportation	NaN	1535.0	1339.1 1332.1 1480.2	1792.4	377.7	44
20	Travel	NaN	3338.2	4091.0 2739.9 2451.5	2902.9	748.1	71
21	Government expenditures	NaN	854.1	777.1 1124.1 1493.5	692.4	266.0	18
22	Others	NaN	5341.6	3339.1 4589.8 4939.3	5999.4	1027.0	130
23	Investment Income Balance	NaN	-5700.9	-4471.7 -4568.5 -6279.6	-11009.6	-1575.7	-239
24	Receipts	NaN	212.8	396.9 497.9 835.4	1014.1	210.7	22
25	(++)	NaN	NaN	NaN	NaN	NaN	N
26	Payments	NaN	5913.7	4868.6 5066.4 7115.0	12023.7	1786.4	261
27	Of which: Interest paid	NaN	643.6	752.0 1231.9 1616.1	2574.1	399.2	50
28	Current Transfers	NaN	21875.8	16790.7 21835.1 26470.9	25113.6	7071.9	590
29	Private (net),	NaN	19205.4	16689.2 21686.1 26264.7	24763.1	6959.2	586
30	NaN	(1)	NaN	NaN	NaN	NaN	N
31	Of which: Remittances of Egyptians working ab...	NaN	19330.0	17077.4 21816.3 26392.9	25150.8	7005.4	590
32	Official (net)	NaN	2670.4	101.5 149.0 206.2	350.5	112.7	4
33	Balance of Current Account	NaN	-12142.6	-19831.1 -14394.0 -5962.3	-10893.9	-492.8	-201

	Category	2014/2015	2015/2016	2016/2017	2017/2018	2018/2019	2017/2018(
34	Capital & Financial Account	NaN	17928.9	21176.7 31015.1 21996.5	10856.9	2951.9	179
35	Capital Account	NaN	-122.9	-141.4 -113.3 -150.7	-129.2	-32.4	-3
36	Financial Account	NaN	18051.8	21318.1 31128.4 22147.2	10986.1	2984.3	182
37	Direct investment abroad	NaN	-223.3	-164.2 -175.1 -271.2	-374.0	-71.2	-6
38	NaN	(++)	NaN	NaN	NaN	NaN	N
39	Direct investment in Egypt (net)	NaN	6379.8	6932.6 7932.8 7719.5	8236.3	1700.3	141
40	Portfolio investment abroad	NaN	47.2	192.1 208.4 -20.8	-96.4	4.3	-7
41	Portfolio investment in Egypt (Net)	NaN	-638.6	-1286.8 15985.3 12094.8	4230.1	-2829.9	-324
42	Of which:Bonds	NaN	-1147.5	-1444.8 5491.5 5293.2	5094.2	2101.8	-12
43	Other Investments (Net)	NaN	12486.7	15644.4 7177.0 2624.9	-1009.9	4180.8	379
44	Net Borrowing	NaN	5036.2	7102.7 9699.2 10278.8	6253.4	3329.4	99
45	Medium- and Long-Term Loans	NaN	-482.5	-186.3 5156.8 6738.5	3333.7	2441.3	-48
46	Disbursements	NaN	1753.5	2523.4 7641.1 8846.4	5525.2	2800.5	15
47	Repayments	NaN	-2236.0	-2709.7 -2484.3 -2107.9	-2191.5	-359.2	-64
48	Medium- and Long-Term	NaN	257.9	1505.3 2795.1 1118.5	828.8	585.9	29

	Category	2014/2015	2015/2016	2016/2017	2017/2018	2018/2019	2017/2018(
	Suppliers' Credit						
49	Disbursements	NaN	313.0	1560.7 2912.2 1313.6	1160.8	629.8	32
50	Repayments	NaN	-55.1	-55.4 -117.1 -195.1	-332.0	-43.9	-3
51	NaN	NaN	NaN	- 76 -	NaN	NaN	N

In [102…

```
keep= [10,27,35]
df32= df31.iloc[keep].copy()
df33= df32.reset_index(drop=True)
df33
```

Out[102…

	Category	2014/2015	2015/2016	2016/2017	2017/2018	2018/2019	2017/2018(Q4)
0	Of which: Suez Canal dues	NaN	5361.7	5121.6 4945.3 5706.7	5730.7	1548.5	1441.2
1	Of which: Remittances of Egyptians working ab...	NaN	19330.0	17077.4 21816.3 26392.9	25150.8	7005.4	5909.0
2	Direct investment in Egypt (net)	NaN	6379.8	6932.6 7932.8 7719.5	8236.3	1700.3	1415.4

In [103…

```
df3=df33
cols_to_fix = [
    "2014/2015", "2015/2016", "2016/2017", "2017/2018", "2018/2019", "2017/2018(Q4)"
]

for col in cols_to_fix:
    df3[col] = df3[col].astype(str).replace('nan', '')

for i in range(len(cols_to_fix) - 1):
    current_col = cols_to_fix[i]
    next_col = cols_to_fix[i+1]


    for row_idx in df3.index:
        cell_parts = df3.at[row_idx, current_col].split()
```

```
        if len(cell_parts) > 1:
            df3.at[row_idx, current_col] = cell_parts[0]
            extra_values = cell_parts[1:]
            existing_next_val = df3.at[row_idx, next_col].split()
            new_next_content = extra_values + existing_next_val
            df3.at[row_idx, next_col] = " ".join(new_next_content)

df3.replace('', np.nan, inplace=True)
df3
```

```
C:\Users\hp\AppData\Local\Temp\ipykernel_17944\3976044703.py:24: FutureWarning: Down
casting behavior in `replace` is deprecated and will be removed in a future version.
To retain the old behavior, explicitly call `result.infer_objects(copy=False)`. To o
pt-in to the future behavior, set `pd.set_option('future.no_silent_downcasting', Tru
e)`
  df3.replace('', np.nan, inplace=True)
```

Out[103…

	Category	2014/2015	2015/2016	2016/2017	2017/2018	2018/2019	2017/2018(Q4)
0	Of which: Suez Canal dues	NaN	5361.7	5121.6	4945.3	5706.7	5730.7
1	Of which: Remittances of Egyptians working ab...	NaN	19330.0	17077.4	21816.3	26392.9	25150.8
2	Direct investment in Egypt (net)	NaN	6379.8	6932.6	7932.8	7719.5	8236.3

In [104…

```
df3.to_dict()
```

Out[104…
```
{'Category': {0: 'Of which:  Suez Canal dues',
  1: 'Of  which: Remittances of Egyptians working abroad',
  2: 'Direct investment in Egypt (net)'},
 '2014/2015': {0: nan, 1: nan, 2: nan},
 '2015/2016': {0: '5361.7', 1: '19330.0', 2: '6379.8'},
 '2016/2017': {0: '5121.6', 1: '17077.4', 2: '6932.6'},
 '2017/2018': {0: '4945.3', 1: '21816.3', 2: '7932.8'},
 '2018/2019': {0: '5706.7', 1: '26392.9', 2: '7719.5'},
 '2017/2018(Q4)': {0: '5730.7', 1: '25150.8', 2: '8236.3'},
 '2018/2019(Q1)': {0: '1548.5', 1: '7005.4', 2: '1700.3'},
 '2018/2019(Q2)': {0: '1441.2', 1: '5909.0', 2: '1415.4'},
 '2018/2019(Q3)': {0: '1487.1', 1: '6136.9', 2: '2769.3'},
 '2018/2019(Q4)': {0: '1344.7', 1: '6165.5', 2: '2339.3'},
 '2019/2020(Q1)': {0: '1457.7', 1: '6939.4', 2: '1712.3'},
 '2019/2020(Q2)': {0: '1507.3', 1: '6712.6', 2: '2352.6'},
 '2019/2020(Q3)': {0: '1524.8 1429.1 ويس',
  1: '6963.9 7869.0 ارج',
  2: '2605.9 970.5'}}
```

In [105…
```
import re
bad_col = '2019/2020(Q3)'
cols = [c for c in df3.columns if c != 'Category']


bad_col_idx = cols.index(bad_col)
extracted = df3[bad_col].astype(str).str.findall(r'[\d\.]+')
df3[bad_col] = extracted.str[1]
value_to_move_left = extracted.str[0]

for i in range(bad_col_idx - 1, -1, -1):
    current_col_name = cols[i]
    next_value_to_move = df3[current_col_name].copy()
    df3[current_col_name] = value_to_move_left
    value_to_move_left = next_value_to_move
for col in cols:
    df3[col] = pd.to_numeric(df3[col], errors='coerce')
print(df3[cols].tail())
```

```
   2014/2015  2015/2016  2016/2017  2017/2018  2018/2019  2017/2018(Q4)  \
0     5361.7     5121.6     4945.3     5706.7     5730.7         1548.5
1    19330.0    17077.4    21816.3    26392.9    25150.8         7005.4
2     6379.8     6932.6     7932.8     7719.5     8236.3         1700.3

   2018/2019(Q1)  2018/2019(Q2)  2018/2019(Q3)  2018/2019(Q4)  2019/2020(Q1)  \
0         1441.2         1487.1         1344.7         1457.7         1507.3
1         5909.0         6136.9         6165.5         6939.4         6712.6
2         1415.4         2769.3         2339.3         1712.3         2352.6

   2019/2020(Q2)  2019/2020(Q3)
0         1524.8         1429.1
1         6963.9         7869.0
2         2605.9          970.5
```

In [106…
```
df3
```

Out[106…

	Category	2014/2015	2015/2016	2016/2017	2017/2018	2018/2019	2017/2018(Q4)
0	Of which: Suez Canal dues	5361.7	5121.6	4945.3	5706.7	5730.7	1548.5
1	Of which: Remittances of Egyptians working ab...	19330.0	17077.4	21816.3	26392.9	25150.8	7005.4
2	Direct investment in Egypt (net)	6379.8	6932.6	7932.8	7719.5	8236.3	1700.3

In [107…

```
df3
cols_to_remove = ['2018/2019','2017/2018', '2014/2015', '2015/2016', '2016/2017']
df3 = df3.drop(columns=cols_to_remove)
df3
```

Out[107…

	Category	2017/2018(Q4)	2018/2019(Q1)	2018/2019(Q2)	2018/2019(Q3)	2018/2019
0	Of which: Suez Canal dues	1548.5	1441.2	1487.1	1344.7	14
1	Of which: Remittances of Egyptians working ab...	7005.4	5909.0	6136.9	6165.5	69
2	Direct investment in Egypt (net)	1700.3	1415.4	2769.3	2339.3	17

In [108…

```
data_area= [0,0,100,85]
dflist4= tb.read_pdf('Bulletin_293_Aug 2021_371.pdf', pages=46, area= data_area,rel
df4= dflist4[0]
df4
```

Out[108...

	Unnamed: 0	Unnamed: 1	Unnamed: 2	Unnamed: 3	21-Balance of Payments	Unnamed: 4	Unnamed: 5
0	NaN	NaN	NaN	NaN	(US$ mn)	NaN	NaN
1	NaN	NaN	NaN	NaN	□□□□□□□l□□□□l	NaN	NaN
2	NaN	During	NaN	NaN	Fiscal Years	NaN	NaN
3	NaN	NaN	NaN	2015/2016	2016/2017 2017/2018* (+) 2018/2019* (+)	NaN	(+)2019/2020*
4	Trade Balance	NaN	NaN	-38683.1	-37274.8 -37276.0 -38034.4	NaN	-36465.1
5	Export proceeds **	NaN	NaN	18704.6	21728.2 25827.0 28495.0	NaN	26376.0
6	Petroleum exports	NaN	NaN	5674.3	6589.5 8773.0 11557.0	NaN	8479.9
7	Other exports	NaN	NaN	13030.3	15138.7 17054.0 16938.0	NaN	17896.1
8	Import payments**	NaN	NaN	-57387.7	-59003.0 -63103.0 -66529.4	NaN	-62841.1
9	Petroleum imports	NaN	NaN	-9293.6	-12015.5 -12489.8 -11548.9	NaN	-8900.9
10	Other imports	NaN	NaN	-48094.1	-46987.5 -50613.2 -54980.5	NaN	-53940.2
11	Services Balance	NaN	NaN	6533.0	5614.2 11122.4 13036.5	NaN	8972.5
12	Receipts	NaN	NaN	16079.3	15400.1 21486.9 24423.6	NaN	21288.9

	Unnamed: 0	Unnamed: 1	Unnamed: 2	Unnamed: 3	21-Balance of Payments	Unnamed: 4	Unnamed: 5
13	Transportation	NaN	NaN	9534.6	7911.2 8707.9 8600.3	NaN	7881.1
14	Of which: Suez Canal dues	NaN	NaN	5121.6	4945.3 5706.7 5730.7	NaN	5805.7
15	Travel (tourism revenues)	NaN	NaN	3767.5	4379.7 9804.3 12570.6	NaN	9859.4
16	Government receipts	NaN	NaN	378.0	776.4 636.7 718.8	NaN	758.5
17	Others	NaN	NaN	2399.2	2332.8 2338.0 2533.9	NaN	2789.9
18	Payments	NaN	NaN	9546.3	9785.9 10364.5 11387.1	NaN	12316.4
19	Transportation	NaN	NaN	1339.1	1332.1 1480.2 1792.4	NaN	2050.1
20	Travel	NaN	NaN	4091.0	2739.9 2451.5 2902.9	NaN	3213.0
21	Government expenditures	NaN	NaN	777.1	1124.1 1493.5 692.4	NaN	975.8
22	Others	NaN	NaN	3339.1	4589.8 4939.3 5999.4	NaN	6077.5
23	Investment Income Balance	NaN	NaN	-4471.7	-4568.5 -6279.6 -11009.6	NaN	-11354.0
24	Receipts	NaN	NaN	396.9	497.9 835.4 1014.1	NaN	942.1
25	(++)	NaN	NaN	NaN	NaN	NaN	NaN
26	Payments	NaN	NaN	4868.6	5066.4 7115.0 12023.7	NaN	12296.1
27	Of which: Interest paid	NaN	NaN	752.0	1231.9 1616.1 2574.1	NaN	2947.7

	Unnamed: 0	Unnamed: 1	Unnamed: 2	Unnamed: 3	21-Balance of Payments	Unnamed: 4	Unnamed: 5
28	Current Transfers	NaN	NaN	16790.7	21835.1 26470.9 25113.6	NaN	27679.9
29	Private (net),	NaN	NaN	16689.2	21686.1 26264.7 24763.1	NaN	27461.8
30	NaN	(1)	NaN	NaN	NaN	NaN	NaN
31	Of which: Remittances of Egyptians working ab...	NaN	NaN	17077.4	21816.3 26392.9 25150.8	NaN	27758.0
32	Official (net)	NaN	NaN	101.5	149.0 206.2 350.5	NaN	218.1
33	Balance of Current Account	NaN	NaN	-19831.1	-14394.0 -5962.3 -10893.9	NaN	-11166.7
34	Capital & Financial Account	NaN	NaN	21176.7	31015.1 21996.5 10856.9	NaN	5374.6
35	Capital Account	NaN	NaN	-141.4	-113.3 -150.7 -129.2	NaN	-248.5
36	Financial Account	NaN	NaN	21318.1	31128.4 22147.2 10986.1	NaN	5623.1
37	Direct investment abroad	NaN	NaN	-164.2	-175.1 -271.2 -374.0	NaN	-351.2
38	NaN	(++)	NaN	NaN	NaN	NaN	NaN
39	Direct investment in Egypt (net)	NaN	NaN	6932.6	7932.8 7719.5 8236.3	NaN	7453.0
40	Portfolio investment abroad	NaN	NaN	192.1	208.4 -20.8 -96.4	NaN	-818.1
41	Portfolio investment in Egypt (Net)	NaN	NaN	-1286.8	15985.3 12094.8 4230.1	NaN	-7307.3
42	Of which:Bonds	NaN	NaN	-1444.8	5491.5 5293.2 5094.2	NaN	4594.9

	Unnamed: 0	Unnamed: 1	Unnamed: 2	Unnamed: 3	21-Balance of Payments	Unnamed: 4	Unnamed: 5
43	Other Investments (Net)	NaN	NaN	15644.4	7177.0 2624.9 -1009.9	NaN	6646.7
44	Net Borrowing	NaN	NaN	7102.7	9699.2 10278.8 6253.4	NaN	4541.6
45	Medium- and Long-Term Loans	NaN	NaN	-186.3	5156.8 6738.5 3333.7	NaN	7216.8
46	Disbursements	NaN	NaN	2523.4	7641.1 8846.4 5525.2	NaN	9253.1
47	Repayments	NaN	NaN	-2709.7	-2484.3 -2107.9 -2191.5	NaN	-2036.3
48	Medium- and Long-Term Suppliers' Credit	NaN	NaN	1505.3	2795.1 1118.5 828.8	NaN	-644.9
49	Disbursements	NaN	NaN	1560.7	2912.2 1313.6 1160.8	NaN	34.3
50	Repayments	NaN	NaN	-55.4	-117.1 -195.1 -332.0	NaN	-679.2
51	NaN	NaN	NaN	NaN	- 76 -	NaN	NaN

In [109…

```
df41= df4.iloc[4:].copy()
df41= df41.dropna(axis=1,how= 'all')
newnames= ['Category',"2015/2016", "2016/2017", "2017/2018", "2018/2019", "2019/202
df41
```

Out[109...

	Unnamed: 0	Unnamed: 1	Unnamed: 3	21-Balance of Payments	Unnamed: 5	Unnamed: 7	Unnamed: 9	Un
4	Trade Balance	NaN	-38683.1	-37274.8 -37276.0 -38034.4	-36465.1	-8287.5	-8783.2	
5	Export proceeds **	NaN	18704.6	21728.2 25827.0 28495.0	26376.0	7583.0	7120.8	
6	Petroleum exports	NaN	5674.3	6589.5 8773.0 11557.0	8479.9	3043.9	2438.3	
7	Other exports	NaN	13030.3	15138.7 17054.0 16938.0	17896.1	4539.1	4682.5	
8	Import payments**	NaN	-57387.7	-59003.0 -63103.0 -66529.4	-62841.1	-15870.5	-15904.0	
9	Petroleum imports	NaN	-9293.6	-12015.5 -12489.8 -11548.9	-8900.9	-2741.5	-3044.5	
10	Other imports	NaN	-48094.1	-46987.5 -50613.2 -54980.5	-53940.2	-13129.0	-12859.5	
11	Services Balance	NaN	6533.0	5614.2 11122.4 13036.5	8972.5	3275.1	4035.0	
12	Receipts	NaN	16079.3	15400.1 21486.9 24423.6	21288.9	6267.7	7436.6	
13	Transportation	NaN	9534.6	7911.2 8707.9 8600.3	7881.1	2139.0	2262.8	
14	Of which: Suez Canal dues	NaN	5121.6	4945.3 5706.7 5730.7	5805.7	1457.7	1507.3	
15	Travel (tourism revenues)	NaN	3767.5	4379.7 9804.3 12570.6	9859.4	3179.0	4193.6	
16	Government receipts	NaN	378.0	776.4 636.7 718.8	758.5	298.8	219.9	

	Unnamed: 0	Unnamed: 1	Unnamed: 3	21-Balance of Payments	Unnamed: 5	Unnamed: 7	Unnamed: 9	Un
17	Others	NaN	2399.2	2332.8 2338.0 2533.9	2789.9	650.9	760.3	
18	Payments	NaN	9546.3	9785.9 10364.5 11387.1	12316.4	2992.6	3401.6	
19	Transportation	NaN	1339.1	1332.1 1480.2 1792.4	2050.1	516.0	522.8	
20	Travel	NaN	4091.0	2739.9 2451.5 2902.9	3213.0	824.0	955.2	
21	Government expenditures	NaN	777.1	1124.1 1493.5 692.4	975.8	126.2	227.3	
22	Others	NaN	3339.1	4589.8 4939.3 5999.4	6077.5	1526.4	1696.3	
23	Investment Income Balance	NaN	-4471.7	-4568.5 -6279.6 -11009.6	-11354.0	-3002.9	-3328.0	
24	Receipts	NaN	396.9	497.9 835.4 1014.1	942.1	294.0	300.8	
25	(++)	NaN	NaN	NaN	NaN	NaN	NaN	
26	Payments	NaN	4868.6	5066.4 7115.0 12023.7	12296.1	3296.9	3628.8	
27	Of which: Interest paid	NaN	752.0	1231.9 1616.1 2574.1	2947.7	786.6	828.4	
28	Current Transfers	NaN	16790.7	21835.1 26470.9 25113.6	27679.9	6927.6	6694.0	
29	Private (net),	NaN	16689.2	21686.1 26264.7 24763.1	27461.8	6877.4	6630.5	
30	NaN	(1)	NaN	NaN	NaN	NaN	NaN	
31	Of which: Remittances of	NaN	17077.4	21816.3 26392.9	27758.0	6939.4	6712.6	

	Unnamed: 0	Unnamed: 1	Unnamed: 3	21-Balance of Payments	Unnamed: 5	Unnamed: 7	Unnamed: 9	Un
	Egyptians working ab...			25150.8				
32	Official (net)	NaN	101.5	149.0 206.2 350.5	218.1	50.2	63.5	
33	Balance of Current Account	NaN	-19831.1	-14394.0 -5962.3 -10893.9	-11166.7	-1087.7	-1382.2	
34	Capital & Financial Account	NaN	21176.7	31015.1 21996.5 10856.9	5374.6	1220.4	657.9	
35	Capital Account	NaN	-141.4	-113.3 -150.7 -129.2	-248.5	-31.8	-37.2	-6
36	Financial Account	NaN	21318.1	31128.4 22147.2 10986.1	5623.1	1252.2	695.1	
37	Direct investment abroad	NaN	-164.2	-175.1 -271.2 -374.0	-351.2	-85.6	-70.6	
38	NaN	(++)	NaN	NaN	NaN	NaN	NaN	
39	Direct investment in Egypt (net)	NaN	6932.6	7932.8 7719.5 8236.3	7453.0	1712.3	2352.6	
40	Portfolio investment abroad	NaN	192.1	208.4 -20.8 -96.4	-818.1	-85.1	123.2	-1
41	Portfolio investment in Egypt (Net)	NaN	-1286.8	15985.3 12094.8 4230.1	-7307.3	3178.3	-1981.5	
42	Of which:Bonds	NaN	-1444.8	5491.5 5293.2 5094.2	4594.9	2050.3	-300.2	
43	Other Investments (Net)	NaN	15644.4	7177.0 2624.9 -1009.9	6646.7	-3467.7	271.4	
44	Net Borrowing	NaN	7102.7	9699.2 10278.8 6253.4	4541.6	1338.7	3954.9	

	Unnamed: 0	Unnamed: 1	Unnamed: 3	21- Balance of Payments	Unnamed: 5	Unnamed: 7	Unnamed: 9	Un
45	Medium- and Long-Term Loans	NaN	-186.3	5156.8 6738.5 3333.7	7216.8	770.6	2290.8	
46	Disbursements	NaN	2523.4	7641.1 8846.4 5525.2	9253.1	1112.3	2964.8	
47	Repayments	NaN	-2709.7	-2484.3 -2107.9 -2191.5	-2036.3	-341.7	-674.0	
48	Medium- and Long-Term Suppliers' Credit	NaN	1505.3	2795.1 1118.5 828.8	-644.9	347.4	-170.3	
49	Disbursements	NaN	1560.7	2912.2 1313.6 1160.8	34.3	430.4	2.9	
50	Repayments	NaN	-55.4	-117.1 -195.1 -332.0	-679.2	-83.0	-173.2	
51	NaN	NaN	NaN	- 76 -	NaN	NaN	NaN	

In [110…

```
df41.columns= newnames[:len(df41.columns)]
df41= df41.reindex(columns=newnames)
df41.set_index('Category')
df41
```

Out[110…

	Category	2015/2016	2016/2017	2017/2018	2018/2019	2019/2020	2018/2019(
4	Trade Balance	NaN	-38683.1	-37274.8 -37276.0 -38034.4	-36465.1	-8287.5	-878
5	Export proceeds **	NaN	18704.6	21728.2 25827.0 28495.0	26376.0	7583.0	712
6	Petroleum exports	NaN	5674.3	6589.5 8773.0 11557.0	8479.9	3043.9	243
7	Other exports	NaN	13030.3	15138.7 17054.0 16938.0	17896.1	4539.1	468
8	Import payments**	NaN	-57387.7	-59003.0 -63103.0 -66529.4	-62841.1	-15870.5	-1590
9	Petroleum imports	NaN	-9293.6	-12015.5 -12489.8 -11548.9	-8900.9	-2741.5	-304
10	Other imports	NaN	-48094.1	-46987.5 -50613.2 -54980.5	-53940.2	-13129.0	-1285
11	Services Balance	NaN	6533.0	5614.2 11122.4 13036.5	8972.5	3275.1	403
12	Receipts	NaN	16079.3	15400.1 21486.9 24423.6	21288.9	6267.7	743
13	Transportation	NaN	9534.6	7911.2 8707.9 8600.3	7881.1	2139.0	226
14	Of which: Suez Canal dues	NaN	5121.6	4945.3 5706.7 5730.7	5805.7	1457.7	150
15	Travel (tourism revenues)	NaN	3767.5	4379.7 9804.3 12570.6	9859.4	3179.0	419
16	Government receipts	NaN	378.0	776.4 636.7 718.8	758.5	298.8	21
17	Others	NaN	2399.2	2332.8 2338.0 2533.9	2789.9	650.9	76
18	Payments	NaN	9546.3	9785.9 10364.5	12316.4	2992.6	340

	Category	2015/2016	2016/2017	2017/2018	2018/2019	2019/2020	2018/2019(
				11387.1			
19	Transportation	NaN	1339.1	1332.1 1480.2 1792.4	2050.1	516.0	52
20	Travel	NaN	4091.0	2739.9 2451.5 2902.9	3213.0	824.0	95
21	Government expenditures	NaN	777.1	1124.1 1493.5 692.4	975.8	126.2	22
22	Others	NaN	3339.1	4589.8 4939.3 5999.4	6077.5	1526.4	169
23	Investment Income Balance	NaN	-4471.7	-4568.5 -6279.6 -11009.6	-11354.0	-3002.9	-332
24	Receipts	NaN	396.9	497.9 835.4 1014.1	942.1	294.0	30
25	(++)	NaN	NaN	NaN	NaN	NaN	N
26	Payments	NaN	4868.6	5066.4 7115.0 12023.7	12296.1	3296.9	362
27	Of which: Interest paid	NaN	752.0	1231.9 1616.1 2574.1	2947.7	786.6	82
28	Current Transfers	NaN	16790.7	21835.1 26470.9 25113.6	27679.9	6927.6	669
29	Private (net),	NaN	16689.2	21686.1 26264.7 24763.1	27461.8	6877.4	663
30	NaN	(1)	NaN	NaN	NaN	NaN	N
31	Of which: Remittances of Egyptians working ab...	NaN	17077.4	21816.3 26392.9 25150.8	27758.0	6939.4	671
32	Official (net)	NaN	101.5	149.0 206.2 350.5	218.1	50.2	6
33	Balance of Current Account	NaN	-19831.1	-14394.0 -5962.3 -10893.9	-11166.7	-1087.7	-138

	Category	2015/2016	2016/2017	2017/2018	2018/2019	2019/2020	2018/2019(
34	Capital & Financial Account	NaN	21176.7	31015.1 21996.5 10856.9	5374.6	1220.4	65
35	Capital Account	NaN	-141.4	-113.3 -150.7 -129.2	-248.5	-31.8	-3
36	Financial Account	NaN	21318.1	31128.4 22147.2 10986.1	5623.1	1252.2	69
37	Direct investment abroad	NaN	-164.2	-175.1 -271.2 -374.0	-351.2	-85.6	-7
38	NaN	(++)	NaN	NaN	NaN	NaN	N
39	Direct investment in Egypt (net)	NaN	6932.6	7932.8 7719.5 8236.3	7453.0	1712.3	235
40	Portfolio investment abroad	NaN	192.1	208.4 -20.8 -96.4	-818.1	-85.1	12
41	Portfolio investment in Egypt (Net)	NaN	-1286.8	15985.3 12094.8 4230.1	-7307.3	3178.3	-198
42	Of which:Bonds	NaN	-1444.8	5491.5 5293.2 5094.2	4594.9	2050.3	-30
43	Other Investments (Net)	NaN	15644.4	7177.0 2624.9 -1009.9	6646.7	-3467.7	27
44	Net Borrowing	NaN	7102.7	9699.2 10278.8 6253.4	4541.6	1338.7	395
45	Medium- and Long-Term Loans	NaN	-186.3	5156.8 6738.5 3333.7	7216.8	770.6	229
46	Disbursements	NaN	2523.4	7641.1 8846.4 5525.2	9253.1	1112.3	296
47	Repayments	NaN	-2709.7	-2484.3 -2107.9 -2191.5	-2036.3	-341.7	-67
48	Medium- and Long-Term	NaN	1505.3	2795.1 1118.5 828.8	-644.9	347.4	-17

	Category	2015/2016	2016/2017	2017/2018	2018/2019	2019/2020	2018/2019(
	Suppliers' Credit						
49	Disbursements	NaN	1560.7	2912.2 1313.6 1160.8	34.3	430.4	
50	Repayments	NaN	-55.4	-117.1 -195.1 -332.0	-679.2	-83.0	-17
51	NaN	NaN	NaN	- 76 -	NaN	NaN	N

In [111…

```
keep= [10,27,35]
d42= df41.iloc[keep].copy()
d43= d42.reset_index(drop=True)
d43
```

Out[111…

	Category	2015/2016	2016/2017	2017/2018	2018/2019	2019/2020	2018/2019(Q4)
0	Of which: Suez Canal dues	NaN	5121.6	4945.3 5706.7 5730.7	5805.7	1457.7	1507.3
1	Of which: Remittances of Egyptians working ab...	NaN	17077.4	21816.3 26392.9 25150.8	27758.0	6939.4	6712.6
2	Direct investment in Egypt (net)	NaN	6932.6	7932.8 7719.5 8236.3	7453.0	1712.3	2352.6

In [112…

```
df4=d43
cols_to_fix = [
    "2015/2016", "2016/2017", "2017/2018", "2018/2019", "2019/2020", "2018/2019(Q4)"
]

for col in cols_to_fix:
    df4[col] = df4[col].astype(str).replace('nan', '')

for i in range(len(cols_to_fix) - 1):
    current_col = cols_to_fix[i]
    next_col = cols_to_fix[i+1]


    for row_idx in df4.index:
        cell_parts = df4.at[row_idx, current_col].split()
```

```
        if len(cell_parts) > 1:
            df4.at[row_idx, current_col] = cell_parts[0]
            extra_values = cell_parts[1:]
            existing_next_val = df4.at[row_idx, next_col].split()
            new_next_content = extra_values + existing_next_val
            df4.at[row_idx, next_col] = " ".join(new_next_content)

df4.replace('', np.nan, inplace=True)
df4
```

```
C:\Users\hp\AppData\Local\Temp\ipykernel_17944\2770791749.py:24: FutureWarning: Down
casting behavior in `replace` is deprecated and will be removed in a future version.
To retain the old behavior, explicitly call `result.infer_objects(copy=False)`. To o
pt-in to the future behavior, set `pd.set_option('future.no_silent_downcasting', Tru
e)`
  df4.replace('', np.nan, inplace=True)
```

Out[112…

	Category	2015/2016	2016/2017	2017/2018	2018/2019	2019/2020	2018/2019(Q4)
0	Of which: Suez Canal dues	NaN	5121.6	4945.3	5706.7	5730.7	5805.7
1	Of which: Remittances of Egyptians working ab...	NaN	17077.4	21816.3	26392.9	25150.8	27758.0
2	Direct investment in Egypt (net)	NaN	6932.6	7932.8	7719.5	8236.3	7453.0

In [113…

```
bad_col = '2020/2021(Q3)'
cols = [c for c in df4.columns if c != 'Category']


bad_col_idx = cols.index(bad_col)
extracted = df4[bad_col].astype(str).str.findall(r'[\d\.]+')
df4[bad_col] = extracted.str[1]
value_to_move_left = extracted.str[0]

for i in range(bad_col_idx - 1, -1, -1):
    current_col_name = cols[i]
    next_value_to_move = df4[current_col_name].copy()
    df4[current_col_name] = value_to_move_left
    value_to_move_left = next_value_to_move
for col in cols:
    df4[col] = pd.to_numeric(df4[col], errors='coerce')
print(df4[cols].tail())
```

```
   2015/2016  2016/2017  2017/2018  2018/2019  2019/2020  2018/2019(Q4)  \
0     5121.6     4945.3     5706.7     5730.7     5805.7         1457.7
1    17077.4    21816.3    26392.9    25150.8    27758.0         6939.4
2     6932.6     7932.8     7719.5     8236.3     7453.0         1712.3

   2019/2020(Q1)  2019/2020(Q2)  2019/2020(Q3)  2019/2020(Q4)  2020/2021(Q1)  \
0         1507.3         1524.8         1429.1         1344.5         1380.7
1         6712.6         6963.9         7869.0         6212.5         8028.1
2         2352.6         2605.9          970.5         1524.0         1605.1

   2020/2021(Q2)  2020/2021(Q3)
0         1516.6         1452.4
1         7493.3         7849.6
2         1752.2         1429.7
```

In [114…

```
cols_to_remove = ['2019/2020','2018/2019', '2015/2016', '2016/2017', '2017/2018']
df4 = df4.drop(columns=cols_to_remove)
df4
```

Out[114…

	Category	2018/2019(Q4)	2019/2020(Q1)	2019/2020(Q2)	2019/2020(Q3)	2019/2020
0	Of which: Suez Canal dues	1457.7	1507.3	1524.8	1429.1	13
1	Of which: Remittances of Egyptians working ab...	6939.4	6712.6	6963.9	7869.0	62
2	Direct investment in Egypt (net)	1712.3	2352.6	2605.9	970.5	15

In [115…

```
data_area= [0,0,100,83]
dflist5= tb.read_pdf('Monthly Statistical Bulletin 305.pdf', pages=46, area= data_a
df5= dflist5[0]
df5
```

Out[115...

	Unnamed: 0	Unnamed: 1	Unnamed: 2	21-Balance of Payments	Unnamed: 3	Unnamed: 4	Unnamed: 5
0	NaN	NaN	NaN	(US$ mn)	NaN	NaN	NaN
1	NaN	NaN	NaN	□□□□□□ا□ □□□□ا	NaN	NaN	NaN
2	NaN	During	NaN	Fiscal Years	NaN	NaN	NaN
3	NaN	NaN	2016/2017	(+) 2017/2018 2018/2019 (+) 2019/2020* (+)	NaN	(+)2020/2021*	NaN
4	Trade Balance	NaN	-37274.8	-37276.0 -38034.4 -36465.1	NaN	-42059.6	NaN
5	Export proceeds **	NaN	21728.2	25827.0 28495.0 26376.0	NaN	28676.5	NaN
6	Petroleum exports	NaN	6589.5	8773.0 11557.0 8479.9	NaN	8597.2	NaN
7	Other exports	NaN	15138.7	17054.0 16938.0 17896.1	NaN	20079.3	NaN
8	Import payments**	NaN	-59003.0	-63103.0 -66529.4 -62841.1	NaN	-70736.1	NaN
9	Petroleum imports	NaN	-12015.5	-12489.8 -11548.9 -8900.9	NaN	-8603.9	NaN
10	Other imports	NaN	-46987.5	-50613.2 -54980.5 -53940.2	NaN	-62132.2	NaN
11	Services Balance	NaN	5614.2	11122.4 13036.5 8972.5	NaN	5119.0	NaN
12	Receipts	NaN	15400.1	21486.9 24423.6 21288.9	NaN	15995.1	NaN

	Unnamed: 0	Unnamed: 1	Unnamed: 2	21-Balance of Payments	Unnamed: 3	Unnamed: 4	Unnamed: 5
13	Transportation	NaN	7911.2	8707.9 8600.3 7881.1	NaN	7527.7	NaN
14	Of which: Suez Canal dues	NaN	4945.3	5706.7 5730.7 5805.7	NaN	5911.2	NaN
15	Travel (tourism revenues)	NaN	4379.7	9804.3 12570.6 9859.4	NaN	4861.5	NaN
16	Government receipts	NaN	776.4	636.7 718.8 758.5	NaN	513.1	NaN
17	Others	NaN	2332.8	2338.0 2533.9 2789.9	NaN	3092.8	NaN
18	Payments	NaN	9785.9	10364.5 11387.1 12316.4	NaN	10876.1	NaN
19	Transportation	NaN	1332.1	1480.2 1792.4 2050.1	NaN	1812.2	NaN
20	Travel	NaN	2739.9	2451.5 2902.9 3213.0	NaN	2708.2	NaN
21	Government expenditures	NaN	1124.1	1493.5 692.4 975.8	NaN	1246.6	NaN
22	Others	NaN	4589.8	4939.3 5999.4 6077.5	NaN	5109.1	NaN
23	Investment Income Balance	NaN	-4568.5	-6279.6 -11009.6 -11354.0	NaN	-12399.2	NaN
24	Receipts	NaN	497.9	835.4 1014.1 942.1	NaN	572.9	NaN
25	(++)	NaN	NaN	NaN	NaN	NaN	NaN
26	Payments	NaN	5066.4	7115.0 12023.7 12296.1	NaN	12972.1	NaN
27	Of which: Interest paid	NaN	1231.9	1616.1 2574.1 2947.7	NaN	2518.7	NaN

	Unnamed: 0	Unnamed: 1	Unnamed: 2	21-Balance of Payments	Unnamed: 3	Unnamed: 4	Unnamed: 5
28	Current Transfers	NaN	21835.1	26470.9 25113.6 27679.9	NaN	30903.4	NaN
29	Private (net),	NaN	21686.1	26264.7 24763.1 27461.8	NaN	31180.3	NaN
30	Of which: Remittances of Egyptians working ab...	NaN	21816.3	26392.9 25150.8 27758.0	NaN	31425.3	NaN
31	Official (net)	NaN	149.0	206.2 350.5 218.1	NaN	-276.9	NaN
32	Balance of Current Account	NaN	-14394.0	-5962.3 -10893.9 -11166.7	NaN	-18436.4	NaN
33	Capital & Financial Account	NaN	31015.1	21996.5 10856.9 5374.6	NaN	23374.0	NaN
34	Capital Account	NaN	-113.3	-150.7 -129.2 -248.5	NaN	-153.0	NaN
35	Financial Account	NaN	31128.4	22147.2 10986.1 5623.1	NaN	23527.0	NaN
36	Direct investment abroad	NaN	-175.1	-271.2 -374.0 -351.2	NaN	-379.1	NaN
37	NaN	(++)	NaN	NaN	NaN	NaN	NaN
38	Direct investment in Egypt (net)	NaN	7932.8	7719.5 8236.3 7453.0	NaN	5214.2	NaN
39	Portfolio investment abroad	NaN	208.4	-20.8 -96.4 -818.1	NaN	-750.7	NaN
40	Portfolio investment in Egypt (Net)	NaN	15985.3	12094.8 4230.1 -7307.3	NaN	18742.4	NaN
41	Of which:Bonds	NaN	5491.5	5293.2 5094.2 4594.9	NaN	4548.9	NaN

	Unnamed: 0	Unnamed: 1	Unnamed: 2	21-Balance of Payments	Unnamed: 3	Unnamed: 4	Unnamed: 5
42	Other Investments (Net)	NaN	7177.0	2624.9 -1009.9 6646.7	NaN	700.2	NaN
43	Net Borrowing	NaN	9699.2	10278.8 6253.4 4541.6	NaN	7964.7	NaN
44	Medium- and Long-Term Loans	NaN	5156.8	6738.5 3333.7 7216.8	NaN	4263.7	NaN
45	Disbursements	NaN	7641.1	8846.4 5525.2 9253.1	NaN	6502.4	NaN
46	Repayments	NaN	-2484.3	-2107.9 -2191.5 -2036.3	NaN	-2238.7	NaN
47	Medium- and Long-Term Suppliers' Credit	NaN	2795.1	1118.5 828.8 -644.9	NaN	2173.6	NaN
48	Disbursements	NaN	2912.2	1313.6 1160.8 34.3	NaN	3304.1	NaN
49	Repayments	NaN	-117.1	-195.1 -332.0 -679.2	NaN	-1130.5	NaN
50	NaN	NaN	NaN	- 76 -	NaN	NaN	NaN

In [116…
```
df51= df5.iloc[4:].copy()
df51= df51.dropna(axis=1,how= 'all')
newnames= ['Category',"2016/2017", "2017/2018", "2018/2019", "2019/2020", "2020/202
```

In [117…
```
df51.columns= newnames[:len(df51.columns)]
df51= df51.reindex(columns=newnames)
df51.set_index('Category')
```

Out[117...

Category	2016/2017	2017/2018	2018/2019	2019/2020	2020/2021	2019/2020(Q4)
Trade Balance	NaN	-37274.8	-37276.0 -38034.4 -36465.1	-42059.6	-8406.6	-8559.3
Export proceeds **	NaN	21728.2	25827.0 28495.0 26376.0	28676.5	5422.4	6280.6
Petroleum exports	NaN	6589.5	8773.0 11557.0 8479.9	8597.2	1153.0	1600.1
Other exports	NaN	15138.7	17054.0 16938.0 17896.1	20079.3	4269.4	4680.5
Import payments**	NaN	-59003.0	-63103.0 -66529.4 -62841.1	-70736.1	-13829.0	-14839.9
Petroleum imports	NaN	-12015.5	-12489.8 -11548.9 -8900.9	-8603.9	-800.7	-1456.4
Other imports	NaN	-46987.5	-50613.2 -54980.5 -53940.2	-62132.2	-13028.3	-13383.5
Services Balance	NaN	5614.2	11122.4 13036.5 8972.5	5119.0	550.1	876.3
Receipts	NaN	15400.1	21486.9 24423.6 21288.9	15995.1	2738.9	3397.2
Transportation	NaN	7911.2	8707.9 8600.3 7881.1	7527.7	1585.0	1738.3
Of which: Suez Canal dues	NaN	4945.3	5706.7 5730.7 5805.7	5911.2	1344.5	1380.7
Travel (tourism revenues)	NaN	4379.7	9804.3 12570.6 9859.4	4861.5	305.0	801.0
Government receipts	NaN	776.4	636.7 718.8 758.5	513.1	193.5	137.7
Others	NaN	2332.8	2338.0 2533.9 2789.9	3092.8	655.4	720.2

Category	2016/2017	2017/2018	2018/2019	2019/2020	2020/2021	2019/2020(Q4)
Payments	NaN	9785.9	10364.5 11387.1 12316.4	10876.1	2188.8	2520.9
Transportation	NaN	1332.1	1480.2 1792.4 2050.1	1812.2	466.4	437.5
Travel	NaN	2739.9	2451.5 2902.9 3213.0	2708.2	380.4	575.8
Government expenditures	NaN	1124.1	1493.5 692.4 975.8	1246.6	357.0	230.7
Others	NaN	4589.8	4939.3 5999.4 6077.5	5109.1	985.0	1276.9
Investment Income Balance	NaN	-4568.5	-6279.6 -11009.6 -11354.0	-12399.2	-2175.3	-3066.8
Receipts	NaN	497.9	835.4 1014.1 942.1	572.9	259.2	57.5
(++)	NaN	NaN	NaN	NaN	NaN	NaN
Payments	NaN	5066.4	7115.0 12023.7 12296.1	12972.1	2434.5	3124.3
Of which: Interest paid	NaN	1231.9	1616.1 2574.1 2947.7	2518.7	570.8	706.1
Current Transfers	NaN	21835.1	26470.9 25113.6 27679.9	30903.4	6204.0	7964.9
Private (net),	NaN	21686.1	26264.7 24763.1 27461.8	31180.3	6155.0	7947.9
Of which: Remittances of Egyptians working abroad	NaN	21816.3	26392.9 25150.8 27758.0	31425.3	6212.5	8028.1
Official (net)	NaN	149.0	206.2 350.5 218.1	-276.9	49.0	17.0

Category	2016/2017	2017/2018	2018/2019	2019/2020	2020/2021	2019/2020(Q4)
Balance of Current Account	NaN	-14394.0	-5962.3 -10893.9 -11166.7	-18436.4	-3827.8	-2784.9
Capital & Financial Account	NaN	31015.1	21996.5 10856.9 5374.6	23374.0	1284.0	3917.6
Capital Account	NaN	-113.3	-150.7 -129.2 -248.5	-153.0	-62.6	-44.3
Financial Account	NaN	31128.4	22147.2 10986.1 5623.1	23527.0	1346.6	3961.9
Direct investment abroad	NaN	-175.1	-271.2 -374.0 -351.2	-379.1	-56.3	-78.5
NaN	(++)	NaN	NaN	NaN	NaN	NaN
Direct investment in Egypt (net)	NaN	7932.8	7719.5 8236.3 7453.0	5214.2	1524.0	1605.1
Portfolio investment abroad	NaN	208.4	-20.8 -96.4 -818.1	-750.7	-846.2	-85.9
Portfolio investment in Egypt (Net)	NaN	15985.3	12094.8 4230.1 -7307.3	18742.4	636.8	6686.4
Of which:Bonds	NaN	5491.5	5293.2 5094.2 4594.9	4548.9	3742.5	29.8
Other Investments (Net)	NaN	7177.0	2624.9 -1009.9 6646.7	700.2	88.3	-4165.2
Net Borrowing	NaN	9699.2	10278.8 6253.4 4541.6	7964.7	1286.0	2189.2
Medium- and Long-Term Loans	NaN	5156.8	6738.5 3333.7 7216.8	4263.7	4687.6	331.8
Disbursements	NaN	7641.1	8846.4 5525.2 9253.1	6502.4	5046.7	956.1

Category	2016/2017	2017/2018	2018/2019	2019/2020	2020/2021	2019/2020(Q4)
Repayments	NaN	-2484.3	-2107.9 -2191.5 -2036.3	-2238.7	-359.1	-624.3
Medium- and Long-Term Suppliers' Credit	NaN	2795.1	1118.5 828.8 -644.9	2173.6	-154.9	1884.9
Disbursements	NaN	2912.2	1313.6 1160.8 34.3	3304.1	19.0	2053.0
Repayments	NaN	-117.1	-195.1 -332.0 -679.2	-1130.5	-173.9	-168.1
NaN	NaN	NaN	- 76 -	NaN	NaN	NaN

In [118…

```
keep= [10,26,34]
d52= df51.iloc[keep].copy()
d53= d52.reset_index(drop=True)
d53
```

Out[118…

	Category	2016/2017	2017/2018	2018/2019	2019/2020	2020/2021	2019/2020(Q4)
0	Of which: Suez Canal dues	NaN	4945.3	5706.7 5730.7 5805.7	5911.2	1344.5	1380.7
1	Of which: Remittances of Egyptians working ab...	NaN	21816.3	26392.9 25150.8 27758.0	31425.3	6212.5	8028.1
2	Direct investment in Egypt (net)	NaN	7932.8	7719.5 8236.3 7453.0	5214.2	1524.0	1605.1

In [119…

```
df5=d53
cols_to_fix = [
    "2016/2017", "2017/2018", "2018/2019", "2019/2020", "2020/2021", "2019/2020(Q4)"
]

for col in cols_to_fix:
    df5[col] = df5[col].astype(str).replace('nan', '')
```

```
for i in range(len(cols_to_fix) - 1):
    current_col = cols_to_fix[i]
    next_col = cols_to_fix[i+1]


    for row_idx in df5.index:
        cell_parts = df5.at[row_idx, current_col].split()

        if len(cell_parts) > 1:
            df5.at[row_idx, current_col] = cell_parts[0]
            extra_values = cell_parts[1:]
            existing_next_val = df5.at[row_idx, next_col].split()
            new_next_content = extra_values + existing_next_val
            df5.at[row_idx, next_col] = " ".join(new_next_content)

df5.replace('', np.nan, inplace=True)
df5
```

```
C:\Users\hp\AppData\Local\Temp\ipykernel_17944\1203110805.py:24: FutureWarning: Down
casting behavior in `replace` is deprecated and will be removed in a future version.
To retain the old behavior, explicitly call `result.infer_objects(copy=False)`. To o
pt-in to the future behavior, set `pd.set_option('future.no_silent_downcasting', Tru
e)`
  df5.replace('', np.nan, inplace=True)
```

Out[119…

	Category	2016/2017	2017/2018	2018/2019	2019/2020	2020/2021	2019/2020(Q4)
0	Of which: Suez Canal dues	NaN	4945.3	5706.7	5730.7	5805.7	5911.2
1	Of which: Remittances of Egyptians working ab...	NaN	21816.3	26392.9	25150.8	27758.0	31425.3
2	Direct investment in Egypt (net)	NaN	7932.8	7719.5	8236.3	7453.0	5214.2

In [120…

```
bad_col = '2021/2022(Q3)'
cols = [c for c in df5.columns if c != 'Category']


bad_col_idx = cols.index(bad_col)
extracted = df5[bad_col].astype(str).str.findall(r'[\d\.]+')
df5[bad_col] = extracted.str[1]
value_to_move_left = extracted.str[0]

for i in range(bad_col_idx - 1, -1, -1):
    current_col_name = cols[i]
    next_value_to_move = df5[current_col_name].copy()
```

```
    df5[current_col_name] = value_to_move_left
    value_to_move_left = next_value_to_move
for col in cols:
    df5[col] = pd.to_numeric(df5[col], errors='coerce')
print(df5[cols].tail())
```

```
   2016/2017  2017/2018  2018/2019  2019/2020  2020/2021  2019/2020(Q4)  \
0     4945.3     5706.7     5730.7     5805.7     5911.2         1344.5
1    21816.3    26392.9    25150.8    27758.0    31425.3         6212.5
2     7932.8     7719.5     8236.3     7453.0     5214.2         1524.0

   2020/2021(Q1)  2020/2021(Q2)  2020/2021(Q3)  2020/2021(Q4)  2021/2022(Q1)  \
0         1380.7         1516.6         1452.4         1561.5         1688.2
1         8028.1         7493.3         7849.6         8054.3         8145.9
2         1605.1         1752.2         1429.7          427.2         1664.9

   2021/2022(Q2)  2021/2022(Q3)
0         1690.8         1705.9
1         7437.2         8045.7
2         1600.5         4083.1
```

In [121…

```
cols_to_remove = ['2019/2020','2018/2019', '2017/2018', '2020/2021', '2016/2017']
df5 = df5.drop(columns=cols_to_remove)
df5
```

Out[121…

	Category	2019/2020(Q4)	2020/2021(Q1)	2020/2021(Q2)	2020/2021(Q3)	2020/2021
0	Of which: Suez Canal dues	1344.5	1380.7	1516.6	1452.4	15
1	Of which: Remittances of Egyptians working ab...	6212.5	8028.1	7493.3	7849.6	80
2	Direct investment in Egypt (net)	1524.0	1605.1	1752.2	1429.7	4

In [122…

```
data_area= [0,0,100,83]
dflist6= tb.read_pdf('Monthly Statistical Bulletin 317.pdf', pages=46, area= data_a
df6= dflist6[0]
```

In [123…

```
df61= df6.iloc[4:].copy()
df61= df61.dropna(axis=1,how= 'all')
newnames= ['Category',"2017/2018", "2018/2019", "2019/2020", "2020/2021", "2021/202
```

In [124…

```
df61.columns= newnames[:len(df61.columns)]
df61= df61.reindex(columns=newnames)
df61.set_index('Category')
```

Out[124...

Category	2017/2018	2018/2019	2019/2020	2020/2021	2021/2022	2020/2021(Q4)
NaN	NaN	2017/2018	2018/2019 2019/2020 2020/2021*	2021/2022*	Q4	Q1
Trade Balance	NaN	-37276.0	-38034.4 -36465.1 -42059.6	-43396.0	-11485.1	-11074.5
Export proceeds **	NaN	25827.0	28495.0 26376.0 28676.5	43906.4	8103.0	8852.3
Petroleum exports	NaN	8773.0	11557.0 8479.9 8597.2	17977.2	2666.1	2901.0
Other exports	NaN	17054.0	16938.0 17896.1 20079.3	25929.2	5436.9	5951.3
Import payments**	NaN	-63103.0	-66529.4 -62841.1 -70736.1	-87302.4	-19588.1	-19926.8
Petroleum imports	NaN	-12489.8	-11548.9 -8900.9 -8603.9	-13544.6	-2847.7	-3002.1
Other imports	NaN	-50613.2	-54980.5 -53940.2 -62132.2	-73757.8	-16740.4	-16924.7
Services Balance	NaN	11122.4	13036.5 8972.5 5119.0	11158.7	1933.9	2937.2
Receipts	NaN	21486.9	24423.6 21288.9 15995.1	26925.7	4683.7	6204.2
Transportation	NaN	8707.9	8600.3 7881.1 7527.7	9734.2	2043.7	2276.7
Of which: Suez Canal dues	NaN	5706.7	5730.7 5805.7 5911.2	6996.8	1561.5	1688.2
Travel (tourism revenues)	NaN	9804.3	12570.6 9859.4 4861.5	10748.1	1748.9	2836.8
Government receipts	NaN	636.7	718.8 758.5 513.1	2736.3	106.2	160.7

Category	2017/2018	2018/2019	2019/2020	2020/2021	2021/2022	2020/2021(Q4)
Others	NaN	2338.0	2533.9 2789.9 3092.8	3707.1	784.9	930.0
Payments	NaN	10364.5	11387.1 12316.4 10876.1	15767.0	2749.8	3267.0
Transportation	NaN	1480.2	1792.4 2050.1 1812.2	3023.6	546.9	653.4
Travel	NaN	2451.5	2902.9 3213.0 2708.2	4479.8	739.4	823.7
Government expenditures	NaN	1493.5	692.4 975.8 1246.6	2340.0	348.0	291.6
Others	NaN	4939.3	5999.4 6077.5 5109.1	5923.6	1115.5	1498.3
Investment Income Balance	NaN	-6279.6	-11009.6 -11354.0 -12399.2	-15763.2	-3546.9	-3883.7
Receipts	NaN	835.4	1014.1 942.1 572.9	996.5	252.8	112.0
(++)	NaN	NaN	NaN	NaN	NaN	NaN
Payments	NaN	7115.0	12023.7 12296.1 12972.1	16759.7	3799.7	3995.7
Of which: Interest paid	NaN	1616.1	2574.1 2947.7 2518.7	2777.6	614.1	704.4
Current Transfers	NaN	26470.9	25113.6 27679.9 30903.4	31449.2	7963.3	8020.3
Private (net),	NaN	26264.7	24763.1 27461.8 31180.3	31719.8	7991.7	8080.3
Of which: Remittances of Egyptians working abroad	NaN	26392.9	25150.8 27758.0 31425.3	31923.5	8054.3	8145.9

Category	2017/2018	2018/2019	2019/2020	2020/2021	2021/2022	2020/2021(Q4)
Official (net)	NaN	206.2	350.5 218.1 -276.9	-270.6	-28.4	-60.0
Balance of Current Account	NaN	-5962.3	-10893.9 -11166.7 -18436.4	-16551.3	-5134.8	-4000.7
Capital & Financial Account	NaN	21996.5	10856.9 5374.6 23374.0	11805.6	6311.8	6049.5
Capital Account	NaN	-150.7	-129.2 -248.5 -153.0	-77.8	-35.0	-36.3
Financial Account	NaN	22147.2	10986.1 5623.1 23527.0	11883.4	6346.8	6085.8
Direct investment abroad	NaN	-271.2	-374.0 -351.2 -379.1	-346.4	-76.7	-84.2
NaN	(++)	NaN	NaN	NaN	NaN	NaN
Direct investment in Egypt (net)	NaN	7719.5	8236.3 7453.0 5214.2	8937.4	427.2	1664.9
Portfolio investment abroad	NaN	-20.8	-96.4 -818.1 -750.7	-139.8	-23.6	-0.9
Portfolio investment in Egypt (Net)	NaN	12094.8	4230.1 -7307.3 18742.4	-20983.3	2755.5	3560.8
Of which:Bonds	NaN	5293.2	5094.2 4594.9 4548.9	1014.3	-32.1	3092.2
Other Investments (Net)	NaN	2624.9	-1009.9 6646.7 700.2	24415.5	3264.4	945.2
Net Borrowing	NaN	10278.8	6253.4 4541.6 7964.7	-1446.9	1747.2	-2426.0
Medium- and Long-Term Loans	NaN	6738.5	3333.7 7216.8 4263.7	710.4	1508.2	-96.7
Disbursements	NaN	8846.4	5525.2 9253.1	3661.1	2129.8	563.2

Category	2017/2018	2018/2019	2019/2020	2020/2021	2021/2022	2020/2021(Q4)
			6502.4			
Repayments	NaN	-2107.9	-2191.5 -2036.3 -2238.7	-2950.7	-621.6	-659.9
Medium- and Long-Term Suppliers' and Buyers' Credit	NaN	1118.5	828.8 -644.9 2173.6	749.5	-37.2	-1976.1
Disbursements	NaN	1313.6	1160.8 34.3 3304.1	3973.5	323.5	272.8
Repayments	NaN	-195.1	-332.0 -679.2 -1130.5	-3224.0	-360.7	-2248.9
NaN	NaN	- 76 -	NaN	NaN	NaN	NaN

In [125…
```
keep= [11,26,35]
df62= df61.iloc[keep].copy()
df63= df62.reset_index(drop=True)
```

In [126…
```
df6=df63
cols_to_fix = [
   "2017/2018", "2018/2019", "2019/2020", "2020/2021", "2021/2022", "2020/2021(Q4)"
]

for col in cols_to_fix:
    df6[col] = df6[col].astype(str).replace('nan', '')

for i in range(len(cols_to_fix) - 1):
    current_col = cols_to_fix[i]
    next_col = cols_to_fix[i+1]


    for row_idx in df6.index:
        cell_parts = df6.at[row_idx, current_col].split()

        if len(cell_parts) > 1:
            df6.at[row_idx, current_col] = cell_parts[0]
            extra_values = cell_parts[1:]
            existing_next_val = df6.at[row_idx, next_col].split()
            new_next_content = extra_values + existing_next_val
            df6.at[row_idx, next_col] = " ".join(new_next_content)

df6.replace('', np.nan, inplace=True)
df6
```

```
C:\Users\hp\AppData\Local\Temp\ipykernel_17944\696352640.py:24: FutureWarning: Downc
asting behavior in `replace` is deprecated and will be removed in a future version.
To retain the old behavior, explicitly call `result.infer_objects(copy=False)`. To o
pt-in to the future behavior, set `pd.set_option('future.no_silent_downcasting', Tru
e)`
  df6.replace('', np.nan, inplace=True)
```

Out[126…

	Category	2017/2018	2018/2019	2019/2020	2020/2021	2021/2022	2020/2021(Q4)
0	Of which: Suez Canal dues	NaN	5706.7	5730.7	5805.7	5911.2	6996.8
1	Private (net),	NaN	26264.7	24763.1	27461.8	31180.3	31719.8
2	Direct investment in Egypt (net)	NaN	7719.5	8236.3	7453.0	5214.2	8937.4

In [127…

```
bad_col = '2022/2023(Q3)'
cols = [c for c in df6.columns if c != 'Category']


bad_col_idx = cols.index(bad_col)
extracted = df6[bad_col].astype(str).str.findall(r'[\d\.]+')
df6[bad_col] = extracted.str[1]
value_to_move_left = extracted.str[0]

for i in range(bad_col_idx - 1, -1, -1):
    current_col_name = cols[i]
    next_value_to_move = df6[current_col_name].copy()
    df6[current_col_name] = value_to_move_left
    value_to_move_left = next_value_to_move
for col in cols:
    df6[col] = pd.to_numeric(df6[col], errors='coerce')
print(df6[cols].tail())
```

```
   2017/2018  2018/2019  2019/2020  2020/2021  2021/2022  2020/2021(Q4)  \
0     5706.7     5730.7     5805.7     5911.2     6996.8         1561.5
1    26264.7    24763.1    27461.8    31180.3    31719.8         7991.7
2     7719.5     8236.3     7453.0     5214.2     8937.4          427.2

   2021/2022(Q1)  2021/2022(Q2)  2021/2022(Q3)  2021/2022(Q4)  2022/2023(Q1)  \
0         1688.2         1690.8         1705.9         1911.9         2010.2
1         8080.3         7398.4         8008.3         8232.8         6380.7
2         1664.9         1600.5         4083.1         1588.9         3296.8

   2022/2023(Q2)  2022/2023(Q3)
0         1970.2         2240.0
1         5511.0         5399.5
2         2431.1         2217.5
```

In [128…

```
df6
```

Out[128...

	Category	2017/2018	2018/2019	2019/2020	2020/2021	2021/2022	2020/2021(Q4)
0	Of which: Suez Canal dues	5706.7	5730.7	5805.7	5911.2	6996.8	1561.5
1	Private (net),	26264.7	24763.1	27461.8	31180.3	31719.8	7991.7
2	Direct investment in Egypt (net)	7719.5	8236.3	7453.0	5214.2	8937.4	427.2

In [129...

```
cols_to_remove = ['2019/2020','2018/2019', '2017/2018', '2021/2022', '2020/2021']
df6 = df6.drop(columns=cols_to_remove)
df6
```

Out[129...

	Category	2020/2021(Q4)	2021/2022(Q1)	2021/2022(Q2)	2021/2022(Q3)	2021/2022(
0	Of which: Suez Canal dues	1561.5	1688.2	1690.8	1705.9	191
1	Private (net),	7991.7	8080.3	7398.4	8008.3	823
2	Direct investment in Egypt (net)	427.2	1664.9	1600.5	4083.1	158

In [130...

```
data_area= [0,0,100,83]
dflist7= tb.read_pdf('Monthly Statistical Bulletin 329.pdf', pages=46, area= data_a
df7= dflist7[0]
```

In [131...

```
df71= df7.iloc[4:].copy()
df71= df71.dropna(axis=1,how= 'all')
newnames= ['Category',"2018/2019", "2019/2020", "2020/2021", "2021/2022", "2022/202
```

In [132...

```
df71.columns= newnames[:len(df71.columns)]
df71= df71.reindex(columns=newnames)
df71.set_index('Category')
```

Out[132...

Category	2018/2019	2019/2020	2020/2021	2021/2022	2022/2023	2021/2022(Q4)
Trade Balance	NaN	-38034.4 -36465.1	-42059.6 -43396.0	-31159.6	-9858.3	-9102.2
Export proceeds **	NaN	28495.0 26376.0	28676.5 43906.4	39624.0	11436.9	9965.2
Petroleum exports	NaN	11557.0 8479.9	8597.2 17977.2	13816.5	4904.5	3708.3
Other exports	NaN	16938.0 17896.1	20079.3 25929.2	25807.5	6532.4	6256.9
Import payments**	NaN	-66529.4 -62841.1	-70736.1 -87302.4	-70783.6	-21295.2	-19067.4
Petroleum imports	NaN	-11548.9 -8900.9	-8603.9 -13544.6	-13406.5	-4606.7	-3814.3
Other imports	NaN	-54980.5 -53940.2	-62132.2 -73757.8	-57377.1	-16688.5	-15253.1
Services Balance	NaN	13036.5 8972.5	5119.0 11158.7	21926.5	3211.1	4053.7
Receipts	NaN	24423.6 21288.9	15995.1 26925.7	34562.1	7398.2	8044.2
Transportation	NaN	8600.3 7881.1	7527.7 9734.2	14000.3	2723.9	3043.7
Of which: Suez Canal dues	NaN	5730.7 5805.7	5911.2 6996.8	8759.6	1911.9	2010.2
Travel (tourism revenues)	NaN	12570.6 9859.4	4861.5 10748.1	13629.3	2545.8	4071.3
Government receipts	NaN	718.8 758.5	513.1 2736.3	2355.2	1176.8	139.9
Others	NaN	2533.9 2789.9	3092.8 3707.1	4577.3	951.7	789.3
Payments	NaN	11387.1 12316.4	10876.1 15767.0	12635.6	4187.1	3990.5
Transportation	NaN	1792.4 2050.1	1812.2 3023.6	2750.0	826.3	884.8
Travel	NaN	2902.9 3213.0	2708.2 4479.8	4990.2	1455.0	1592.0
Government expenditures	NaN	692.4 975.8	1246.6 2340.0	1248.4	515.6	471.8

Category	2018/2019	2019/2020	2020/2021	2021/2022	2022/2023	2021/2022(Q4)
Others	NaN	5999.4 6077.5	5109.1 5923.6	3647.0	1390.2	1041.9
Investment Income Balance	NaN	-11009.6 -11354.0	-12399.2 -15763.2	-17318.4	-4503.6	-4535.3
Receipts	NaN	1014.1 942.1	572.9 996.5	2137.8	417.3	275.8
(++)	NaN	NaN	NaN	NaN	NaN	NaN
Payments	NaN	12023.7 12296.1	12972.1 16759.7	19456.2	4920.9	4811.1
Of which: Interest paid	NaN	2574.1 2947.7	2518.7 2777.6	6150.4	838.4	1219.5
Current Transfers	NaN	25113.6 27679.9	30903.4 31449.2	21841.0	8193.0	6391.5
Private (net),	NaN	24763.1 27461.8	31180.3 31719.8	21897.4	8232.8	6380.7
Of which: Remittances of Egyptians working abroad	NaN	25150.8 27758.0	31425.3 31923.5	22076.4	8294.7	6442.1
Official (net)	NaN	350.5 218.1	-276.9 -270.6	-56.4	-39.8	10.8
Balance of Current Account	NaN	-10893.9 -11166.7	-18436.4 -16551.3	-4710.5	-2957.8	-3192.3
Capital & Financial Account	NaN	10856.9 5374.6	23374.0 11805.6	8931.5	988.3	4417.0
Capital Account	NaN	-129.2 -248.5	-153.0 -77.8	-54.2	46.9	14.5
Financial Account	NaN	10986.1 5623.1	23527.0 11883.4	8985.7	941.4	4402.5
Direct investment abroad	NaN	-374.0 -351.2	-379.1 -346.4	-337.7	-85.0	-68.5
NaN	(++)	NaN	NaN	NaN	NaN	NaN

Category	2018/2019	2019/2020	2020/2021	2021/2022	2022/2023	2021/2022(Q4)
Direct investment in Egypt (net)	NaN	8236.3 7453.0	5214.2 8937.4	10038.6	1588.9	3296.8
Portfolio investment abroad	NaN	-96.4 -818.1	-750.7 -139.8	-328.1	-129.5	-50.5
Portfolio investment in Egypt (Net)	NaN	4230.1 -7307.3	18742.4 -20983.3	-3765.7	-3735.2	-2158.3
Of which:Bonds	NaN	5094.2 4594.9	4548.9 1014.3	344.7	-42.6	-2.0
Other Investments (Net)	NaN	-1009.9 6646.7	700.2 24415.5	3378.6	3302.2	3383.0
Net Borrowing	NaN	6253.4 4541.6	7964.7 -1446.9	1434.4	2.4	2283.1
Medium- and Long-Term Loans	NaN	3333.7 7216.8	4263.7 710.4	-226.0	407.1	-283.6
Disbursements	NaN	5525.2 9253.1	6502.4 3661.1	3195.2	1200.5	391.1
Repayments	NaN	-2191.5 -2036.3	-2238.7 -2950.7	-3421.2	-793.4	-674.7
Medium- and Long-Term Suppliers' and Buyers' Credit	NaN	828.8 -644.9	2173.6 749.5	1678.2	-210.7	347.7
Disbursements	NaN	1160.8 34.3	3304.1 3973.5	2861.6	147.1	581.0
Repayments	NaN	-332.0 -679.2	-1130.5 -3224.0	-1183.4	-357.8	-233.3
NaN	NaN	- 76 -	NaN	NaN	NaN	NaN

In [133…

```
keep= [10,26,34]
df72= df71.iloc[keep].copy()
df73= df72.reset_index(drop=True)
df73
```

Out[133...

	Category	2018/2019	2019/2020	2020/2021	2021/2022	2022/2023	2021/2022(Q4)
0	Of which: Suez Canal dues	NaN	5730.7 5805.7	5911.2 6996.8	8759.6	1911.9	2010.2
1	Of which: Remittances of Egyptians working ab...	NaN	25150.8 27758.0	31425.3 31923.5	22076.4	8294.7	6442.1
2	Direct investment in Egypt (net)	NaN	8236.3 7453.0	5214.2 8937.4	10038.6	1588.9	3296.8

In [134...

```
df7=df73
cols_to_fix = [
"2018/2019", "2019/2020", "2020/2021", "2021/2022", "2022/2023", "2021/2022(Q4)", "
]

for col in cols_to_fix:
    df7[col] = df7[col].astype(str).replace('nan', '')

for i in range(len(cols_to_fix) - 1):
    current_col = cols_to_fix[i]
    next_col = cols_to_fix[i+1]


    for row_idx in df7.index:
        cell_parts = df7.at[row_idx, current_col].split()

        if len(cell_parts) > 1:
            df7.at[row_idx, current_col] = cell_parts[0]
            extra_values = cell_parts[1:]
            existing_next_val = df7.at[row_idx, next_col].split()
            new_next_content = extra_values + existing_next_val
            df7.at[row_idx, next_col] = " ".join(new_next_content)

df7.replace('', np.nan, inplace=True)
df7
```

```
C:\Users\hp\AppData\Local\Temp\ipykernel_17944\2498804823.py:24: FutureWarning: Down
casting behavior in `replace` is deprecated and will be removed in a future version.
To retain the old behavior, explicitly call `result.infer_objects(copy=False)`. To o
pt-in to the future behavior, set `pd.set_option('future.no_silent_downcasting', Tru
e)`
  df7.replace('', np.nan, inplace=True)
```

Out[134...

	Category	2018/2019	2019/2020	2020/2021	2021/2022	2022/2023	2021/2022(Q4)
0	Of which: Suez Canal dues	NaN	5730.7	5805.7	5911.2	6996.8	8759.6
1	Of which: Remittances of Egyptians working ab...	NaN	25150.8	27758.0	31425.3	31923.5	22076.4
2	Direct investment in Egypt (net)	NaN	8236.3	7453.0	5214.2	8937.4	10038.6

In [135...

```
bad_col = '2023/2024(Q3)'
cols = [c for c in df7.columns if c != 'Category']


bad_col_idx = cols.index(bad_col)
extracted = df7[bad_col].astype(str).str.findall(r'[\d\.]+')
df7[bad_col] = extracted.str[1]
value_to_move_left = extracted.str[0]

for i in range(bad_col_idx - 1, -1, -1):
    current_col_name = cols[i]
    next_value_to_move = df7[current_col_name].copy()
    df7[current_col_name] = value_to_move_left
    value_to_move_left = next_value_to_move
for col in cols:
    df7[col] = pd.to_numeric(df7[col], errors='coerce')
print(df7[cols].tail())
```

```
   2018/2019  2019/2020  2020/2021  2021/2022  2022/2023  2021/2022(Q4)  \
0     5730.7     5805.7     5911.2     6996.8     8759.6         1911.9
1    25150.8    27758.0    31425.3    31923.5    22076.4         8294.7
2     8236.3     7453.0     5214.2     8937.4    10038.6         1588.9

   2022/2023(Q1)  2022/2023(Q2)  2022/2023(Q3)  2022/2023(Q4)  2023/2024(Q1)  \
0         2010.2         1970.2         2240.0         2539.2         2399.3
1         6442.1         5550.3         5457.7         4626.3         4516.3
2         3296.8         2431.1         2217.5         2093.2         2321.7

   2023/2024(Q2)  2023/2024(Q3)
0         2403.4          959.3
1         4931.8         5022.4
2         3208.2        18182.4
```

In [136...

```
cols_to_remove = ['2019/2020','2018/2019', '2022/2023', '2021/2022', '2020/2021']
df7 = df7.drop(columns=cols_to_remove)
df7
```

Out[136...

	Category	2021/2022(Q4)	2022/2023(Q1)	2022/2023(Q2)	2022/2023(Q3)	2022/2023
0	Of which: Suez Canal dues	1911.9	2010.2	1970.2	2240.0	25
1	Of which: Remittances of Egyptians working ab...	8294.7	6442.1	5550.3	5457.7	46
2	Direct investment in Egypt (net)	1588.9	3296.8	2431.1	2217.5	20

In [137...

```
data_area= [0,0,100,83]
dflist8= tb.read_pdf('Monthly Statistical Bulletin 341.pdf', pages=46, area= data_a
df81= dflist8[0]
df81= df81.iloc[4:].copy()
df81= df81.dropna(axis=1,how= 'all')
newnames= [
    "Category",
    "2019/2020",
    "2020/2021",
    "2021/2022",
    "2022/2023",
    "2023/2024",
    "2022/2023(Q4)",
    "2023/2024(Q1)",
    "2023/2024(Q2)",
    "2023/2024(Q3)",
    "2023/2024(Q4)",
    "2024/2025(Q1)",
    "2024/2025(Q2)",
    "2024/2025(Q3)"
]
df81.columns= newnames[:len(df81.columns)]
df81= df81.reindex(columns=newnames)
df81.set_index('Category')
keep= [10,26,33]
d82= df81.iloc[keep].copy()
d82= d82.reset_index(drop=True)
d82

df8=d82
cols_to_fix = [
    "2019/2020",
    "2020/2021",
    "2021/2022",
    "2022/2023",
    "2023/2024",
```

```
    "2022/2023(Q4)",
    "2023/2024(Q1)",
    "2023/2024(Q2)",
    "2023/2024(Q3)",
    "2023/2024(Q4)",
    "2024/2025(Q1)",
    "2024/2025(Q2)",
    "2024/2025(Q3)"]

for col in cols_to_fix:
    df8[col] = df8[col].astype(str).replace('nan', '')

for i in range(len(cols_to_fix) - 1):
    current_col = cols_to_fix[i]
    next_col = cols_to_fix[i+1]


    for row_idx in df8.index:
        cell_parts = df8.at[row_idx, current_col].split()

        if len(cell_parts) > 1:
            df8.at[row_idx, current_col] = cell_parts[0]
            extra_values = cell_parts[1:]
            existing_next_val = df8.at[row_idx, next_col].split()
            new_next_content = extra_values + existing_next_val
            df8.at[row_idx, next_col] = " ".join(new_next_content)

df8.replace('', np.nan, inplace=True)
df8
```

Out[137…

	Category	2019/2020	2020/2021	2021/2022	2022/2023	2023/2024	2022/2023(Q4)
0	Of which: Suez Canal dues	5805.7	5911.2	6996.8	8759.6	6632.4	2539.2
1	Of which: Remittances of Egyptians working ab...	27758.0	31425.3	31923.5	22076.4	21937.8	4626.3
2	Direct investment in Egypt (net)	7453.0	5214.2	8937.4	10038.6	46064.5	2093.2

In [138…

```
dfs = [df1, df2, df3, df4, df5, df6, df7,df8]
dfs_indexed = [d.set_index('Category') for d in dfs]
final_df = pd.concat(dfs_indexed, axis=1, join='outer')
final_df = final_df.reset_index()
final_df = dfs
final_df
```

Out[138… [

	Category	2016/2017(Q4)	2017/2018(Q1)	2017/2018(Q2)	2017/2018(Q3)	2017/2018(Q4)	2018/2019(Q1)	2018/2019(Q2)	2018/2019(Q3)
0	Of which: Suez Canal dues	1228.7	1382.2	1386.3	1389.7	1548.5	1441.2	1487.1	1344.7
1	Of which: Remittances of Egyptians working ab...	6033.9	5824.3	7098.8	6464.4	7005.4	5909.0	6136.9	6165.5
2	Direct investment in Egypt (net)	1367.8	1843.0	1919.9	2256.3	1700.3	1099.9	1741.1	1805.4

,

	Category	2015/2016(Q4)	2016/2017(Q1)	2016/2017(Q2)	2016/2017(Q3)	2016/2017(Q4)	2017/2018(Q1)	2017/2018(Q2)	2017/2018(Q3)
0	Of which: Suez Canal dues	1243.9	1300.4	1214.2	1202.0	1228.7	1382.2	1386.3	1389.7
1	Of which: Remittances of Egyptians working ab...	4417.8	4354.9	5756.0	5780.4	4827.1	5973.6	7098.9	6464.4 ج
2	Direct investment in Egypt (net)	1046.9	1872.2	2414.8	2278.0	1350.8	1843.0	1919.9	2256.3

,

	Category	2017/2018(Q4)	2018/2019(Q1)	2018/2019(Q2)	2018/2019(Q3)	2018/2019(Q4)	2019/2020(Q1)	2019/2020(Q2)	2019/2020(Q3)
0	Of which: Suez Canal dues	1548.5	1441.2	1487.1	1344.7	1457.7	1507.3	1524.8	1429.1
1	Of which: Remittances of Egyptians working ab...	7005.4	5909.0	6136.9	6165.5	6939.4	6712.6	6963.9	7869.0
2	Direct investment in Egypt (net)	1700.3	1415.4	2769.3	2339.3	1712.3	2352.6	2605.9	970.5

,

	Category	2018/2019(Q4)	2019/2020(Q1)	2019/2020(Q2)	2019/2020(Q3)	2019/2020(Q4)	2020/2021(Q1)	2020/2021(Q2)	2020/2021(Q3)
0	Of which: Suez Canal dues	1457.7	1507.3	1524.8	1429.1	1344.5	1380.7	1516.6	1452.4
1	Of which: Remittances of Egyptians working ab...	6939.4	6712.6	6963.9	7869.0	6212.5	8028.1	7493.3	7849.6
2	Direct investment in Egypt (net)	1712.3	2352.6	2605.9	970.5	1524.0	1605.1	1752.2	1429.7

,

```
                                          Category  2019/2020(Q4)  \
0                       Of which:  Suez Canal dues         1344.5
1  Of  which: Remittances of Egyptians working ab...         6212.5
2                   Direct investment in Egypt (net)         1524.0

   2020/2021(Q1)  2020/2021(Q2)  2020/2021(Q3)  2020/2021(Q4)  2021/2022(Q1)  \
0         1380.7         1516.6         1452.4         1561.5         1688.2
1         8028.1         7493.3         7849.6         8054.3         8145.9
2         1605.1         1752.2         1429.7          427.2         1664.9

   2021/2022(Q2)  2021/2022(Q3)
0         1690.8         1705.9
1         7437.2         8045.7
2         1600.5         4083.1  ,
                         Category  2020/2021(Q4)  2021/2022(Q1)  \
0       Of which:  Suez Canal dues         1561.5         1688.2
1                   Private (net),         7991.7         8080.3
2  Direct investment in Egypt (net)          427.2         1664.9

   2021/2022(Q2)  2021/2022(Q3)  2021/2022(Q4)  2022/2023(Q1)  2022/2023(Q2)  \
0         1690.8         1705.9         1911.9         2010.2         1970.2
1         7398.4         8008.3         8232.8         6380.7         5511.0
2         1600.5         4083.1         1588.9         3296.8         2431.1

   2022/2023(Q3)
0         2240.0
1         5399.5
2         2217.5  ,
                                          Category  2021/2022(Q4)  \
0                       Of which:  Suez Canal dues         1911.9
1  Of  which: Remittances of Egyptians working ab...         8294.7
2                   Direct investment in Egypt (net)         1588.9

   2022/2023(Q1)  2022/2023(Q2)  2022/2023(Q3)  2022/2023(Q4)  2023/2024(Q1)  \
0         2010.2         1970.2         2240.0         2539.2         2399.3
1         6442.1         5550.3         5457.7         4626.3         4516.3
2         3296.8         2431.1         2217.5         2093.2         2321.7

   2023/2024(Q2)  2023/2024(Q3)
0         2403.4          959.3
1         4931.8         5022.4
2         3208.2        18182.4  ,
                                          Category 2019/2020 2020/2021  \
0                       Of which:  Suez Canal dues    5805.7    5911.2
1  Of  which: Remittances of Egyptians working ab...   27758.0   31425.3
2                   Direct investment in Egypt (net)    7453.0    5214.2

  2021/2022 2022/2023 2023/2024 2022/2023(Q4) 2023/2024(Q1) 2023/2024(Q2)  \
0    6996.8    8759.6    6632.4        2539.2        2399.3        2403.4
1   31923.5   22076.4   21937.8        4626.3        4516.3        4931.8
2    8937.4   10038.6   46064.5        2093.2        2321.7        3208.2

  2023/2024(Q3) 2023/2024(Q4) 2024/2025(Q1) 2024/2025(Q2) 2024/2025(Q3)
0         959.3         870.4         931.2         880.9         830.1
1        5022.4        7467.3        8325.9        8743.5        9373.5
2       18182.4       22352.2        2717.1        3326.3        3769.7  ]
```

In [139…

```
cleaned_dfs = []
for d in dfs:
    temp_df = d.copy()
    temp_df['Category'] = temp_df['Category'].astype(str).str.strip()
    temp_df = temp_df[temp_df['Category'] != 'Private (net)']
    temp_df = temp_df.set_index('Category')
    cleaned_dfs.append(temp_df)
final_df = pd.concat(cleaned_dfs, axis=1, join='outer')
final_df = final_df.reset_index()
print(final_df.head())
```

	Category	2016/2017(Q4)	\
0	Of which: Suez Canal dues	1228.7	
1	Of which: Remittances of Egyptians working ab...	6033.9	
2	Direct investment in Egypt (net)	1367.8	
3	Private (net),	NaN	

	2017/2018(Q1)	2017/2018(Q2)	2017/2018(Q3)	2017/2018(Q4)	2018/2019(Q1)	\
0	1382.2	1386.3	1389.7	1548.5	1441.2	
1	5824.3	7098.8	6464.4	7005.4	5909.0	
2	1843.0	1919.9	2256.3	1700.3	1099.9	
3	NaN	NaN	NaN	NaN	NaN	

	2018/2019(Q2)	2018/2019(Q3)	2015/2016(Q4)	2016/2017(Q1)	2016/2017(Q2)	\
0	1487.1	1344.7	1243.9	1300.4	1214.2	
1	6136.9	6165.5	4417.8	4354.9	5756.0	
2	1741.1	1805.4	1046.9	1872.2	2414.8	
3	NaN	NaN	NaN	NaN	NaN	

	2016/2017(Q3)	2016/2017(Q4)	2017/2018(Q1)	2017/2018(Q2)	2017/2018(Q3)	\
0	1202.0	1228.7	1382.2	1386.3	1389.7	
1	5780.4	4827.1	5973.6	7098.9	6464.4 ج	
2	2278.0	1350.8	1843.0	1919.9	2256.3	
3	NaN	NaN	NaN	NaN	NaN	

	2017/2018(Q4)	2018/2019(Q1)	2018/2019(Q2)	2018/2019(Q3)	2018/2019(Q4)	\
0	1548.5	1441.2	1487.1	1344.7	1457.7	
1	7005.4	5909.0	6136.9	6165.5	6939.4	
2	1700.3	1415.4	2769.3	2339.3	1712.3	
3	NaN	NaN	NaN	NaN	NaN	

	2019/2020(Q1)	2019/2020(Q2)	2019/2020(Q3)	2018/2019(Q4)	2019/2020(Q1)	\
0	1507.3	1524.8	1429.1	1457.7	1507.3	
1	6712.6	6963.9	7869.0	6939.4	6712.6	
2	2352.6	2605.9	970.5	1712.3	2352.6	
3	NaN	NaN	NaN	NaN	NaN	

	2019/2020(Q2)	2019/2020(Q3)	2019/2020(Q4)	2020/2021(Q1)	2020/2021(Q2)	\
0	1524.8	1429.1	1344.5	1380.7	1516.6	
1	6963.9	7869.0	6212.5	8028.1	7493.3	
2	2605.9	970.5	1524.0	1605.1	1752.2	
3	NaN	NaN	NaN	NaN	NaN	

	2020/2021(Q3)	2019/2020(Q4)	2020/2021(Q1)	2020/2021(Q2)	2020/2021(Q3)	\
0	1452.4	1344.5	1380.7	1516.6	1452.4	
1	7849.6	6212.5	8028.1	7493.3	7849.6	
2	1429.7	1524.0	1605.1	1752.2	1429.7	
3	NaN	NaN	NaN	NaN	NaN	

	2020/2021(Q4)	2021/2022(Q1)	2021/2022(Q2)	2021/2022(Q3)	2020/2021(Q4)	\
0	1561.5	1688.2	1690.8	1705.9	1561.5	
1	8054.3	8145.9	7437.2	8045.7	NaN	
2	427.2	1664.9	1600.5	4083.1	427.2	
3	NaN	NaN	NaN	NaN	7991.7	

	2021/2022(Q1)	2021/2022(Q2)	2021/2022(Q3)	2021/2022(Q4)	2022/2023(Q1)	\
0	1688.2	1690.8	1705.9	1911.9	2010.2	

```
1            NaN            NaN            NaN            NaN            NaN
2         1664.9         1600.5         4083.1         1588.9         3296.8
3         8080.3         7398.4         8008.3         8232.8         6380.7

   2022/2023(Q2)  2022/2023(Q3)  2021/2022(Q4)  2022/2023(Q1)  2022/2023(Q2)  \
0         1970.2         2240.0         1911.9         2010.2         1970.2
1            NaN            NaN         8294.7         6442.1         5550.3
2         2431.1         2217.5         1588.9         3296.8         2431.1
3         5511.0         5399.5            NaN            NaN            NaN

   2022/2023(Q3) 2022/2023(Q4) 2023/2024(Q1) 2023/2024(Q2) 2023/2024(Q3)  \
0         2240.0        2539.2        2399.3        2403.4         959.3
1         5457.7        4626.3        4516.3        4931.8        5022.4
2         2217.5        2093.2        2321.7        3208.2       18182.4
3            NaN           NaN           NaN           NaN           NaN

  2019/2020 2020/2021 2021/2022 2022/2023 2023/2024 2022/2023(Q4)  \
0    5805.7    5911.2    6996.8    8759.6    6632.4        2539.2
1   27758.0   31425.3   31923.5   22076.4   21937.8        4626.3
2    7453.0    5214.2    8937.4   10038.6   46064.5        2093.2
3       NaN       NaN       NaN       NaN       NaN           NaN

  2023/2024(Q1) 2023/2024(Q2) 2023/2024(Q3) 2023/2024(Q4) 2024/2025(Q1)  \
0        2399.3        2403.4         959.3         870.4         931.2
1        4516.3        4931.8        5022.4        7467.3        8325.9
2        2321.7        3208.2       18182.4       22352.2        2717.1
3           NaN           NaN           NaN           NaN           NaN

  2024/2025(Q2) 2024/2025(Q3)
0         880.9         830.1
1        8743.5        9373.5
2        3326.3        3769.7
3           NaN           NaN
```

In [140…

```
final_df
```

Out[140…

	Category	2016/2017(Q4)	2017/2018(Q1)	2017/2018(Q2)	2017/2018(Q3)	2017/2018
0	Of which: Suez Canal dues	1228.7	1382.2	1386.3	1389.7	15
1	Of which: Remittances of Egyptians working ab...	6033.9	5824.3	7098.8	6464.4	70
2	Direct investment in Egypt (net)	1367.8	1843.0	1919.9	2256.3	17
3	Private (net),	NaN	NaN	NaN	NaN	

In [141…

```
final_df['Category'] = final_df['Category'].astype(str).str.strip()
final_df = final_df[final_df['Category'] != 'Private (net),']
print(final_df['Category'].unique())
```

```
['Of which:  Suez Canal dues'
 'Of  which: Remittances of Egyptians working abroad'
 'Direct investment in Egypt (net)']
```

In [142…

```
final_df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Index: 3 entries, 0 to 2
Data columns (total 70 columns):
 #   Column         Non-Null Count  Dtype
---  ------         --------------  -----
 0   Category       3 non-null      object
 1   2016/2017(Q4)  3 non-null      object
 2   2017/2018(Q1)  3 non-null      object
 3   2017/2018(Q2)  3 non-null      object
 4   2017/2018(Q3)  3 non-null      object
 5   2017/2018(Q4)  3 non-null      object
 6   2018/2019(Q1)  3 non-null      object
 7   2018/2019(Q2)  3 non-null      object
 8   2018/2019(Q3)  3 non-null      object
 9   2015/2016(Q4)  3 non-null      object
 10  2016/2017(Q1)  3 non-null      object
 11  2016/2017(Q2)  3 non-null      object
 12  2016/2017(Q3)  3 non-null      object
 13  2016/2017(Q4)  3 non-null      object
 14  2017/2018(Q1)  3 non-null      object
 15  2017/2018(Q2)  3 non-null      object
 16  2017/2018(Q3)  3 non-null      object
 17  2017/2018(Q4)  3 non-null      float64
 18  2018/2019(Q1)  3 non-null      float64
 19  2018/2019(Q2)  3 non-null      float64
 20  2018/2019(Q3)  3 non-null      float64
 21  2018/2019(Q4)  3 non-null      float64
 22  2019/2020(Q1)  3 non-null      float64
 23  2019/2020(Q2)  3 non-null      float64
 24  2019/2020(Q3)  3 non-null      float64
 25  2018/2019(Q4)  3 non-null      float64
 26  2019/2020(Q1)  3 non-null      float64
 27  2019/2020(Q2)  3 non-null      float64
 28  2019/2020(Q3)  3 non-null      float64
 29  2019/2020(Q4)  3 non-null      float64
 30  2020/2021(Q1)  3 non-null      float64
 31  2020/2021(Q2)  3 non-null      float64
 32  2020/2021(Q3)  3 non-null      float64
 33  2019/2020(Q4)  3 non-null      float64
 34  2020/2021(Q1)  3 non-null      float64
 35  2020/2021(Q2)  3 non-null      float64
 36  2020/2021(Q3)  3 non-null      float64
 37  2020/2021(Q4)  3 non-null      float64
 38  2021/2022(Q1)  3 non-null      float64
 39  2021/2022(Q2)  3 non-null      float64
 40  2021/2022(Q3)  3 non-null      float64
 41  2020/2021(Q4)  2 non-null      float64
 42  2021/2022(Q1)  2 non-null      float64
 43  2021/2022(Q2)  2 non-null      float64
 44  2021/2022(Q3)  2 non-null      float64
 45  2021/2022(Q4)  2 non-null      float64
 46  2022/2023(Q1)  2 non-null      float64
 47  2022/2023(Q2)  2 non-null      float64
 48  2022/2023(Q3)  2 non-null      float64
 49  2021/2022(Q4)  3 non-null      float64
 50  2022/2023(Q1)  3 non-null      float64
```

```
 51  2022/2023(Q2)  3 non-null      float64
 52  2022/2023(Q3)  3 non-null      float64
 53  2022/2023(Q4)  3 non-null      float64
 54  2023/2024(Q1)  3 non-null      float64
 55  2023/2024(Q2)  3 non-null      float64
 56  2023/2024(Q3)  3 non-null      float64
 57  2019/2020      3 non-null      object
 58  2020/2021      3 non-null      object
 59  2021/2022      3 non-null      object
 60  2022/2023      3 non-null      object
 61  2023/2024      3 non-null      object
 62  2022/2023(Q4)  3 non-null      object
 63  2023/2024(Q1)  3 non-null      object
 64  2023/2024(Q2)  3 non-null      object
 65  2023/2024(Q3)  3 non-null      object
 66  2023/2024(Q4)  3 non-null      object
 67  2024/2025(Q1)  3 non-null      object
 68  2024/2025(Q2)  3 non-null      object
 69  2024/2025(Q3)  3 non-null      object
dtypes: float64(40), object(30)
memory usage: 1.7+ KB
```

In [143…

```
df2
```

Out[143…

	Category	2015/2016(Q4)	2016/2017(Q1)	2016/2017(Q2)	2016/2017(Q3)	2016/2017
0	Of which: Suez Canal dues	1243.9	1300.4	1214.2	1202.0	12
1	Of which: Remittances of Egyptians working ab...	4417.8	4354.9	5756.0	5780.4	48
2	Direct investment in Egypt (net)	1046.9	1872.2	2414.8	2278.0	13

In [144…

```
df4
```

Out[144...

	Category	2018/2019(Q4)	2019/2020(Q1)	2019/2020(Q2)	2019/2020(Q3)	2019/2020
0	Of which: Suez Canal dues	1457.7	1507.3	1524.8	1429.1	13
1	Of which: Remittances of Egyptians working ab...	6939.4	6712.6	6963.9	7869.0	62
2	Direct investment in Egypt (net)	1712.3	2352.6	2605.9	970.5	15

In [145...

```
final_df = final_df.groupby(level=0, axis=1).first()
cols = [c for c in final_df.columns if c != 'Category']
sorted_cols = sorted(cols, key=lambda x: (x.split('(')[0], x.split('(')[1] if '(' i
final_df = final_df[['Category'] + sorted_cols]
print(final_df.columns)
```

```
Index(['Category', '2015/2016(Q4)', '2016/2017(Q1)', '2016/2017(Q2)',
       '2016/2017(Q3)', '2016/2017(Q4)', '2017/2018(Q1)', '2017/2018(Q2)',
       '2017/2018(Q3)', '2017/2018(Q4)', '2018/2019(Q1)', '2018/2019(Q2)',
       '2018/2019(Q3)', '2018/2019(Q4)', '2019/2020', '2019/2020(Q1)',
       '2019/2020(Q2)', '2019/2020(Q3)', '2019/2020(Q4)', '2020/2021',
       '2020/2021(Q1)', '2020/2021(Q2)', '2020/2021(Q3)', '2020/2021(Q4)',
       '2021/2022', '2021/2022(Q1)', '2021/2022(Q2)', '2021/2022(Q3)',
       '2021/2022(Q4)', '2022/2023', '2022/2023(Q1)', '2022/2023(Q2)',
       '2022/2023(Q3)', '2022/2023(Q4)', '2023/2024', '2023/2024(Q1)',
       '2023/2024(Q2)', '2023/2024(Q3)', '2023/2024(Q4)', '2024/2025(Q1)',
       '2024/2025(Q2)', '2024/2025(Q3)'],
      dtype='object')
```

```
C:\Users\hp\AppData\Local\Temp\ipykernel_17944\1319700087.py:1: FutureWarning: DataF
rame.groupby with axis=1 is deprecated. Do `frame.T.groupby(...)` without axis inste
ad.
  final_df = final_df.groupby(level=0, axis=1).first()
```

In [146...

```
final_df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Index: 3 entries, 0 to 2
Data columns (total 42 columns):
 #   Column         Non-Null Count  Dtype
---  ------         --------------  -----
 0   Category       3 non-null      object
 1   2015/2016(Q4)  3 non-null      object
 2   2016/2017(Q1)  3 non-null      object
 3   2016/2017(Q2)  3 non-null      object
 4   2016/2017(Q3)  3 non-null      object
 5   2016/2017(Q4)  3 non-null      object
 6   2017/2018(Q1)  3 non-null      object
 7   2017/2018(Q2)  3 non-null      object
 8   2017/2018(Q3)  3 non-null      object
 9   2017/2018(Q4)  3 non-null      object
 10  2018/2019(Q1)  3 non-null      object
 11  2018/2019(Q2)  3 non-null      object
 12  2018/2019(Q3)  3 non-null      object
 13  2018/2019(Q4)  3 non-null      float64
 14  2019/2020      3 non-null      object
 15  2019/2020(Q1)  3 non-null      float64
 16  2019/2020(Q2)  3 non-null      float64
 17  2019/2020(Q3)  3 non-null      float64
 18  2019/2020(Q4)  3 non-null      float64
 19  2020/2021      3 non-null      object
 20  2020/2021(Q1)  3 non-null      float64
 21  2020/2021(Q2)  3 non-null      float64
 22  2020/2021(Q3)  3 non-null      float64
 23  2020/2021(Q4)  3 non-null      float64
 24  2021/2022      3 non-null      object
 25  2021/2022(Q1)  3 non-null      float64
 26  2021/2022(Q2)  3 non-null      float64
 27  2021/2022(Q3)  3 non-null      float64
 28  2021/2022(Q4)  3 non-null      float64
 29  2022/2023      3 non-null      object
 30  2022/2023(Q1)  3 non-null      float64
 31  2022/2023(Q2)  3 non-null      float64
 32  2022/2023(Q3)  3 non-null      float64
 33  2022/2023(Q4)  3 non-null      float64
 34  2023/2024      3 non-null      object
 35  2023/2024(Q1)  3 non-null      float64
 36  2023/2024(Q2)  3 non-null      float64
 37  2023/2024(Q3)  3 non-null      float64
 38  2023/2024(Q4)  3 non-null      object
 39  2024/2025(Q1)  3 non-null      object
 40  2024/2025(Q2)  3 non-null      object
 41  2024/2025(Q3)  3 non-null      object
dtypes: float64(20), object(22)
memory usage: 1.0+ KB
```

In [147…

```
final_df.drop(columns=['2015/2016(Q4)'], inplace=True)
final_df
```

Out[147...

	Category	2016/2017(Q1)	2016/2017(Q2)	2016/2017(Q3)	2016/2017(Q4)	2017/2018
0	Of which: Suez Canal dues	1300.4	1214.2	1202.0	1228.7	13
1	Of which: Remittances of Egyptians working ab...	4354.9	5756.0	5780.4	6033.9	58
2	Direct investment in Egypt (net)	1872.2	2414.8	2278.0	1367.8	18

In [148...

```
pd.set_option('display.max_columns', None)
final_df
```

Out[148...

	Category	2016/2017(Q1)	2016/2017(Q2)	2016/2017(Q3)	2016/2017(Q4)	2017/2018
0	Of which: Suez Canal dues	1300.4	1214.2	1202.0	1228.7	13
1	Of which: Remittances of Egyptians working ab...	4354.9	5756.0	5780.4	6033.9	58
2	Direct investment in Egypt (net)	1872.2	2414.8	2278.0	1367.8	18

In [173...

```
# This flips the table so Variables are Columns and Quarters are Rows
df_master = final_df.T

# Rename columns to the list above
df_master.columns = ["Suez Revenue Revenue Revenue", "Remittances", "FDI"]
df_master
```

Out[173...

	Suez Revenue Revenue Revenue	Remittances	FDI
Category	Of which: Suez Canal dues	Of which: Remittances of Egyptians working ab...	Direct investment in Egypt (net)
2016/2017(Q1)	1300.4	4354.9	1872.2
2016/2017(Q2)	1214.2	5756.0	2414.8
2016/2017(Q3)	1202.0	5780.4	2278.0
2016/2017(Q4)	1228.7	6033.9	1367.8
2017/2018(Q1)	1382.2	5824.3	1843.0
2017/2018(Q2)	1386.3	7098.8	1919.9
2017/2018(Q3)	1389.7	6464.4	2256.3
2017/2018(Q4)	1548.5	7005.4	1700.3
2018/2019(Q1)	1441.2	5909.0	1099.9
2018/2019(Q2)	1487.1	6136.9	1741.1
2018/2019(Q3)	1344.7	6165.5	1805.4
2018/2019(Q4)	1457.7	6939.4	1712.3
2019/2020	5805.7	27758.0	7453.0
2019/2020(Q1)	1507.3	6712.6	2352.6
2019/2020(Q2)	1524.8	6963.9	2605.9
2019/2020(Q3)	1429.1	7869.0	970.5
2019/2020(Q4)	1344.5	6212.5	1524.0
2020/2021	5911.2	31425.3	5214.2
2020/2021(Q1)	1380.7	8028.1	1605.1
2020/2021(Q2)	1516.6	7493.3	1752.2
2020/2021(Q3)	1452.4	7849.6	1429.7
2020/2021(Q4)	1561.5	8054.3	427.2
2021/2022	6996.8	31923.5	8937.4
2021/2022(Q1)	1688.2	8145.9	1664.9
2021/2022(Q2)	1690.8	7437.2	1600.5
2021/2022(Q3)	1705.9	8045.7	4083.1
2021/2022(Q4)	1911.9	8294.7	1588.9
2022/2023	8759.6	22076.4	10038.6

	Suez Revenue Revenue Revenue	Remittances	FDI
2022/2023(Q1)	2010.2	6442.1	3296.8
2022/2023(Q2)	1970.2	5550.3	2431.1
2022/2023(Q3)	2240.0	5457.7	2217.5
2022/2023(Q4)	2539.2	4626.3	2093.2
2023/2024	6632.4	21937.8	46064.5
2023/2024(Q1)	2399.3	4516.3	2321.7
2023/2024(Q2)	2403.4	4931.8	3208.2
2023/2024(Q3)	959.3	5022.4	18182.4
2023/2024(Q4)	870.4	7467.3	22352.2
2024/2025(Q1)	931.2	8325.9	2717.1
2024/2025(Q2)	880.9	8743.5	3326.3
2024/2025(Q3)	830.1	9373.5	3769.7

In [179…

```
df_master
```

Out[179...

	Suez Revenue Revenue Revenue	Remittances	FDI
Category	Of which: Suez Canal dues	Of which: Remittances of Egyptians working ab...	Direct investment in Egypt (net)
2016/2017(Q1)	1300.4	4354.9	1872.2
2016/2017(Q2)	1214.2	5756.0	2414.8
2016/2017(Q3)	1202.0	5780.4	2278.0
2016/2017(Q4)	1228.7	6033.9	1367.8
2017/2018(Q1)	1382.2	5824.3	1843.0
2017/2018(Q2)	1386.3	7098.8	1919.9
2017/2018(Q3)	1389.7	6464.4	2256.3
2017/2018(Q4)	1548.5	7005.4	1700.3
2018/2019(Q1)	1441.2	5909.0	1099.9
2018/2019(Q2)	1487.1	6136.9	1741.1
2018/2019(Q3)	1344.7	6165.5	1805.4
2018/2019(Q4)	1457.7	6939.4	1712.3
2019/2020	5805.7	27758.0	7453.0
2019/2020(Q1)	1507.3	6712.6	2352.6
2019/2020(Q2)	1524.8	6963.9	2605.9
2019/2020(Q3)	1429.1	7869.0	970.5
2019/2020(Q4)	1344.5	6212.5	1524.0
2020/2021	5911.2	31425.3	5214.2
2020/2021(Q1)	1380.7	8028.1	1605.1
2020/2021(Q2)	1516.6	7493.3	1752.2
2020/2021(Q3)	1452.4	7849.6	1429.7
2020/2021(Q4)	1561.5	8054.3	427.2
2021/2022	6996.8	31923.5	8937.4
2021/2022(Q1)	1688.2	8145.9	1664.9
2021/2022(Q2)	1690.8	7437.2	1600.5
2021/2022(Q3)	1705.9	8045.7	4083.1
2021/2022(Q4)	1911.9	8294.7	1588.9
2022/2023	8759.6	22076.4	10038.6

	Suez Revenue Revenue Revenue	Remittances	FDI
2022/2023(Q1)	2010.2	6442.1	3296.8
2022/2023(Q2)	1970.2	5550.3	2431.1
2022/2023(Q3)	2240.0	5457.7	2217.5
2022/2023(Q4)	2539.2	4626.3	2093.2
2023/2024	6632.4	21937.8	46064.5
2023/2024(Q1)	2399.3	4516.3	2321.7
2023/2024(Q2)	2403.4	4931.8	3208.2
2023/2024(Q3)	959.3	5022.4	18182.4
2023/2024(Q4)	870.4	7467.3	22352.2
2024/2025(Q1)	931.2	8325.9	2717.1
2024/2025(Q2)	880.9	8743.5	3326.3
2024/2025(Q3)	830.1	9373.5	3769.7

In [189…

```
df= df_master
df
```

Out[189...

	Suez Revenue Revenue Revenue		Remittances	FDI
Category	Of which: Suez Canal dues	Of which: Remittances of Egyptians working ab...	Direct investment in Egypt (net)
2016/2017(Q1)	1300.4	4354.9	1872.2
2016/2017(Q2)	1214.2	5756.0	2414.8
2016/2017(Q3)	1202.0	5780.4	2278.0
2016/2017(Q4)	1228.7	6033.9	1367.8
2017/2018(Q1)	1382.2	5824.3	1843.0
2017/2018(Q2)	1386.3	7098.8	1919.9
2017/2018(Q3)	1389.7	6464.4	2256.3
2017/2018(Q4)	1548.5	7005.4	1700.3
2018/2019(Q1)	1441.2	5909.0	1099.9
2018/2019(Q2)	1487.1	6136.9	1741.1
2018/2019(Q3)	1344.7	6165.5	1805.4
2018/2019(Q4)	1457.7	6939.4	1712.3
2019/2020	5805.7	27758.0	7453.0
2019/2020(Q1)	1507.3	6712.6	2352.6
2019/2020(Q2)	1524.8	6963.9	2605.9
2019/2020(Q3)	1429.1	7869.0	970.5
2019/2020(Q4)	1344.5	6212.5	1524.0
2020/2021	5911.2	31425.3	5214.2
2020/2021(Q1)	1380.7	8028.1	1605.1
2020/2021(Q2)	1516.6	7493.3	1752.2
2020/2021(Q3)	1452.4	7849.6	1429.7
2020/2021(Q4)	1561.5	8054.3	427.2
2021/2022	6996.8	31923.5	8937.4
2021/2022(Q1)	1688.2	8145.9	1664.9
2021/2022(Q2)	1690.8	7437.2	1600.5
2021/2022(Q3)	1705.9	8045.7	4083.1
2021/2022(Q4)	1911.9	8294.7	1588.9
2022/2023	8759.6	22076.4	10038.6

	Suez Revenue Revenue Revenue	Remittances	FDI
2022/2023(Q1)	2010.2	6442.1	3296.8
2022/2023(Q2)	1970.2	5550.3	2431.1
2022/2023(Q3)	2240.0	5457.7	2217.5
2022/2023(Q4)	2539.2	4626.3	2093.2
2023/2024	6632.4	21937.8	46064.5
2023/2024(Q1)	2399.3	4516.3	2321.7
2023/2024(Q2)	2403.4	4931.8	3208.2
2023/2024(Q3)	959.3	5022.4	18182.4
2023/2024(Q4)	870.4	7467.3	22352.2
2024/2025(Q1)	931.2	8325.9	2717.1
2024/2025(Q2)	880.9	8743.5	3326.3
2024/2025(Q3)	830.1	9373.5	3769.7

In [186…

```
from statsmodels.tsa.vector_ar.vecm import VECM, select_order, select_coint_rank
import matplotlib.pyplot as plt
```

In [187…

```
import numpy as np
import pandas as pd

# 1) If you still have a "Quarter" or "Category" column, remove it from logging
# (keep it separately)
if "Quarter" in df.columns:
    quarter_labels = df["Quarter"].astype(str).values
    df_num = df.drop(columns=["Quarter"]).copy()
elif "Category" in df.columns:
    quarter_labels = df["Category"].astype(str).values
    df_num = df.drop(columns=["Category"]).copy()
else:
    # if Quarter/Category is already the index
    quarter_labels = df.index.astype(str).values
    df_num = df.copy()

# 2) Remove the descriptive text row if it exists (it produces many NaNs when numer
df_num = df_num.apply(pd.to_numeric, errors="coerce")
df_num = df_num.dropna(how="all")  # drops fully-empty rows after coercion

# (optional but recommended) drop rows that are "mostly missing" (like the subtitle
min_numeric = max(1, df_num.shape[1] // 2)
df_num = df_num[df_num.notna().sum(axis=1) >= min_numeric]

# 3) Log safely (only for strictly positive values)
df_num = df_num.where(df_num > 0)  # non-positive becomes NaN
df_log = np.log(df_num)
```

```
# 4) Add labels back
df_log["Label"] = quarter_labels[:len(df_log)]  # align length safely

df_log
df = df.drop(columns=["Label"], errors="ignore")
df
```

Out[187...

	Suez Revenue Revenue Revenue	Remittances	FDI
Category	Of which: Suez Canal dues	Of which: Remittances of Egyptians working ab...	Direct investment in Egypt (net)
2016/2017(Q1)	1300.4	4354.9	1872.2
2016/2017(Q2)	1214.2	5756.0	2414.8
2016/2017(Q3)	1202.0	5780.4	2278.0
2016/2017(Q4)	1228.7	6033.9	1367.8
2017/2018(Q1)	1382.2	5824.3	1843.0
2017/2018(Q2)	1386.3	7098.8	1919.9
2017/2018(Q3)	1389.7	6464.4	2256.3
2017/2018(Q4)	1548.5	7005.4	1700.3
2018/2019(Q1)	1441.2	5909.0	1099.9
2018/2019(Q2)	1487.1	6136.9	1741.1
2018/2019(Q3)	1344.7	6165.5	1805.4
2018/2019(Q4)	1457.7	6939.4	1712.3
2019/2020	5805.7	27758.0	7453.0
2019/2020(Q1)	1507.3	6712.6	2352.6
2019/2020(Q2)	1524.8	6963.9	2605.9
2019/2020(Q3)	1429.1	7869.0	970.5
2019/2020(Q4)	1344.5	6212.5	1524.0
2020/2021	5911.2	31425.3	5214.2
2020/2021(Q1)	1380.7	8028.1	1605.1
2020/2021(Q2)	1516.6	7493.3	1752.2
2020/2021(Q3)	1452.4	7849.6	1429.7
2020/2021(Q4)	1561.5	8054.3	427.2
2021/2022	6996.8	31923.5	8937.4
2021/2022(Q1)	1688.2	8145.9	1664.9
2021/2022(Q2)	1690.8	7437.2	1600.5
2021/2022(Q3)	1705.9	8045.7	4083.1
2021/2022(Q4)	1911.9	8294.7	1588.9
2022/2023	8759.6	22076.4	10038.6

	Suez Revenue Revenue Revenue	Remittances	FDI
2022/2023(Q1)	2010.2	6442.1	3296.8
2022/2023(Q2)	1970.2	5550.3	2431.1
2022/2023(Q3)	2240.0	5457.7	2217.5
2022/2023(Q4)	2539.2	4626.3	2093.2
2023/2024	6632.4	21937.8	46064.5
2023/2024(Q1)	2399.3	4516.3	2321.7
2023/2024(Q2)	2403.4	4931.8	3208.2
2023/2024(Q3)	959.3	5022.4	18182.4
2023/2024(Q4)	870.4	7467.3	22352.2
2024/2025(Q1)	931.2	8325.9	2717.1
2024/2025(Q2)	880.9	8743.5	3326.3
2024/2025(Q3)	830.1	9373.5	3769.7

In [190…

```
import pandas as pd
import numpy as np

df = pd.read_excel("Project_All.xlsx", sheet_name="Sheet1", header=0)

# Standardize first column name (Quarter)
df = df.rename(columns={df.columns[0]: "Quarter"})

# Remove junk rows like "Category" and subtitle rows
df["Quarter"] = df["Quarter"].astype(str).str.strip()
df = df[~df["Quarter"].isin(["Category", "nan", "NaN", "None"])].copy()

# Keep only quarter-form rows like 2016/2017(Q1)
df = df[df["Quarter"].str.contains(r"\(Q[1-4]\)", regex=True)].copy()

# Convert numeric columns
for c in df.columns[1:]:
    df[c] = pd.to_numeric(df[c], errors="coerce")
df = df.rename(columns={"Sum": "Manufacturing"})

df
```

Out[190…

	Quarter	Suez Revenue	Remittances	FDI	REER	Manufacturing	CPI
1	2016/2017(Q1)	1300.4	4354.9	1872.2	165.309001	19551.566763	180.502876
2	2016/2017(Q2)	1214.2	5756.0	2414.8	121.698440	19919.117737	194.320319
3	2016/2017(Q3)	1202.0	5780.4	2278.0	101.380384	20203.072006	214.662163
4	2016/2017(Q4)	1228.7	6033.9	1367.8	102.093663	20584.274341	226.116436
5	2017/2018(Q1)	1382.2	5824.3	1843.0	104.980889	20981.818514	238.527280
6	2017/2018(Q2)	1386.3	7098.8	1919.9	107.663741	21165.244379	245.070583
7	2017/2018(Q3)	1389.7	6464.4	2256.3	105.670227	20822.227254	246.644765
8	2017/2018(Q4)	1548.5	7005.4	1700.3	110.505014	21080.729550	255.485367
9	2018/2019(Q1)	1441.2	5909.0	1099.9	119.744964	21277.006253	273.308174
10	2018/2019(Q2)	1487.1	6136.9	1741.1	124.425648	21084.789503	282.061914
11	2018/2019(Q3)	1344.7	6165.5	1805.4	126.664588	21125.455525	280.544454
12	2018/2019(Q4)	1457.7	6939.4	1712.3	133.460586	21851.644656	286.469174
13	2019/2020(Q1)	1507.3	6712.6	2352.6	140.289180	21247.394172	292.261334
14	2019/2020(Q2)	1524.8	6963.9	2605.9	144.536253	21301.211580	295.016134
15	2019/2020(Q3)	1429.1	7869.0	970.5	150.652481	21304.320489	296.956946
16	2019/2020(Q4)	1344.5	6212.5	1524.0	153.369168	17375.436451	301.939198
17	2020/2021(Q1)	1380.7	8028.1	1605.1	147.895995	18096.209101	303.230304
18	2020/2021(Q2)	1516.6	7493.3	1752.2	151.065943	19087.689011	310.397861
19	2020/2021(Q3)	1452.4	7849.6	1429.7	148.080979	20261.175894	310.028269
20	2020/2021(Q4)	1561.5	8054.3	427.2	148.677238	20987.599453	315.951016
21	2021/2022(Q1)	1688.2	8145.9	1664.9	152.209662	21090.030054	321.095275
22	2021/2022(Q2)	1690.8	7437.2	1600.5	157.449515	21506.385868	328.671365
23	2021/2022(Q3)	1705.9	8045.7	4083.1	156.824290	22093.783593	337.463371
24	2021/2022(Q4)	1911.9	8294.7	1588.9	143.780321	22569.991889	357.759810
25	2022/2023(Q1)	2010.2	6442.1	3296.8	147.494642	22393.960884	367.393603
26	2022/2023(Q2)	1970.2	5550.3	2431.1	130.129110	22345.582278	390.402470
27	2022/2023(Q3)	2240.0	5457.7	2217.5	106.132478	22268.919973	439.413225
28	2022/2023(Q4)	2539.2	4626.3	2093.2	109.381845	22647.956723	475.890696
29	2023/2024(Q1)	2399.3	4516.3	2321.7	117.543918	22548.167224	504.324417
30	2023/2024(Q2)	2403.4	4931.8	3208.2	124.149225	22078.968462	525.743240

	Quarter	Suez Revenue	Remittances	FDI	REER	Manufacturing	CPI
31	2023/2024(Q3)	959.3	5022.4	18182.4	119.708983	21713.967708	584.481321
32	2023/2024(Q4)	870.4	7467.3	22352.2	91.041521	21778.649241	615.721001
33	2024/2025(Q1)	931.2	8325.9	2717.1	91.548031	22931.477467	636.110972
34	2024/2025(Q2)	880.9	8743.5	3326.3	95.423731	22871.369214	659.026306
35	2024/2025(Q3)	830.1	9373.5	3769.7	95.319006	23204.544441	681.099852

In [ ]:

In [206…

```
# ============================================================
# FINAL PREPROCESSING CELL (Run once at top)
# - Loads Project_All.xlsx (Sheet1)
# - Keeps ONLY quarterly rows 2016/2017(Q1) ... 2024/2025(Q3)
# - Converts numeric columns safely (removes tabs/commas/spaces)
# - Builds a proper fiscal-quarter DatetimeIndex (QS-JUL) => no "ns" warnings
# - Log-transforms safely (optional)
# - Builds Rent_Total correctly
# - Outputs: df_levels, df_log, df (analysis df), model_data, model_data_rent
# ============================================================

import pandas as pd
import numpy as np
import re

EXCEL_PATH = "Project_All.xlsx"
SHEET_NAME = "Sheet1"

# ---- Choose your analysis space ----
# True  => df becomes LOG-LEVELS (recommended for VAR interpretation)
# False => df stays in LEVELS (then your diffs are level changes, not growth rates)
USE_LOGS = True

FISCAL_FREQ = "QS-JUL"  # fiscal quarters: Q1=Jul, Q2=Oct, Q3=Jan, Q4=Apr

# ----------------------------
# 1) Load
# ----------------------------
raw = pd.read_excel(EXCEL_PATH, sheet_name=SHEET_NAME, header=0)
raw = raw.rename(columns={raw.columns[0]: "Quarter"})
raw["Quarter"] = raw["Quarter"].astype(str).str.strip()

# ----------------------------
# 2) Keep ONLY quarter rows (drop annual totals & junk)
# ----------------------------
raw = raw[raw["Quarter"].str.contains(r"\(Q[1-4]\)", regex=True, na=False)].copy()

# ----------------------------
# 3) Clean column names & normalize key columns
```

```
# ----------------------------
raw.columns = [re.sub(r"\s+", " ", str(c)).strip() for c in raw.columns]

# Manufacturing sometimes imported as "Sum"
raw = raw.rename(columns={"Sum": "Manufacturing", "Manufacturing Sum": "Manufacturi

# Optional: handle alternate Suez naming if it ever appears
raw = raw.rename(columns={"Suez": "Suez Revenue"})

# ----------------------------
# 4) Convert all non-Quarter columns to numeric safely
# ----------------------------
for c in raw.columns:
    if c == "Quarter":
        continue
    raw[c] = (
        raw[c].astype(str)
              .str.replace(",", "", regex=False)
              .str.replace("\t", "", regex=False)
              .str.strip()
    )
    raw[c] = pd.to_numeric(raw[c], errors="coerce")

# ----------------------------
# 5) Convert fiscal-quarter label -> real date
#    2016/2017(Q1) -> 2016-07-01
#    2016/2017(Q2) -> 2016-10-01
#    2016/2017(Q3) -> 2017-01-01
#    2016/2017(Q4) -> 2017-04-01
# ----------------------------
def fiscal_q_to_date(lbl: str) -> pd.Timestamp:
    m = re.match(r"(\d{4})/(\d{4})\(Q([1-4])\)", str(lbl))
    if not m:
        return pd.NaT
    start_year = int(m.group(1))
    q = int(m.group(3))
    month = {1: 7, 2: 10, 3: 1, 4: 4}[q]
    year = start_year if q in (1, 2) else start_year + 1
    return pd.Timestamp(year, month, 1)

dates = raw["Quarter"].map(fiscal_q_to_date)

df_levels = raw.drop(columns=["Quarter"]).copy()
df_levels.index = pd.DatetimeIndex(dates)
df_levels = df_levels.sort_index()

# Drop rows with bad/unparsed quarter labels (should be none)
df_levels = df_levels[~df_levels.index.isna()].copy()

# Remove duplicated quarter dates if any (keep first)
if df_levels.index.duplicated().any():
    print("WARNING: duplicated quarter dates detected; keeping first occurrence.")
    df_levels = df_levels[~df_levels.index.duplicated(keep="first")].copy()

# Force quarterly frequency (this removes statsmodels 'ns' warnings)
# If a quarter is missing, asfreq will insert it as an all-NaN row (we warn below).
```

```
df_levels = df_levels.asfreq(FISCAL_FREQ)

missing_q = df_levels.index[df_levels.isna().all(axis=1)]
if len(missing_q) > 0:
    print("WARNING: Missing quarters detected (inserted by asfreq as all-NaN rows):
    print(missing_q)

# ----------------------------
# 6) Verify required columns exist (fail early, clearly)
# ----------------------------
required = {"Suez Revenue", "Remittances", "FDI", "REER", "Manufacturing", "CPI"}
missing_cols = required - set(df_levels.columns)
if missing_cols:
    raise KeyError(f"Missing columns in Excel sheet: {missing_cols}. "
                   f"Found columns: {list(df_levels.columns)}")

# ----------------------------
# 7) Build df in chosen space + set DATA_IN_LOGS for your later code
# ----------------------------
# ----------------------------
# LOG TRANSFORMATION (FINAL)
# ----------------------------

# Ensure numeric (extra safety)
df_levels = df_levels.apply(pd.to_numeric, errors="coerce")

if USE_LOGS:
    # Check non-positive values BEFORE log
    bad = (df_levels <= 0).sum()
    if bad.any():
        print("WARNING: non-positive values detected; they will become NaN under lo
        print(bad[bad > 0])

    # Safe log: only positive values logged
    df_log = np.log(df_levels.where(df_levels > 0))

    df = df_log.copy()
    DATA_IN_LOGS = True

else:
    df_log = None
    df = df_levels.copy()
    DATA_IN_LOGS = False


# ----------------------------
# 8) Build Rent_Total correctly
# ----------------------------
if DATA_IN_LOGS:
    # log(Suez + Remittances) from logSuez and logRemit
    df["Rent_Total"] = np.logaddexp(df["Suez Revenue"], df["Remittances"])
else:
    # level sum (if you later want log, do np.log(df["Rent_Total"]))
    df["Rent_Total"] = df["Suez Revenue"] + df["Remittances"]

# ----------------------------
```

```
# 9) Convenience datasets used by your later analysis cells
# ----------------------------
model_data = df[["Suez Revenue", "REER", "Manufacturing", "CPI"]].dropna()
model_data_rent = df[["Rent_Total", "REER", "Manufacturing", "CPI"]].dropna()
def find_duplicate_blocks(df_in, cols, block_len=3):
    vals = df_in[cols].round(6).to_numpy()
    for i in range(len(df_in) - block_len + 1):
        block = vals[i:i+block_len]
        for j in range(i + block_len, len(df_in) - block_len + 1):
            if np.all(block == vals[j:j+block_len]):
                return (df_in.index[i], df_in.index[j], block_len)
    return None

dup = find_duplicate_blocks(model_data, cols=["Suez Revenue","REER","Manufacturing"
if dup:
    print("WARNING: duplicated 3-quarter block detected:", dup)


print("✅ Preprocessing complete")
print("df_levels shape:", df_levels.shape)
print("df (analysis space) shape:", df.shape)
print("Date range:", df.index.min(), "→", df.index.max())
print("DATA_IN_LOGS =", DATA_IN_LOGS)

model_data.tail(3)
if len(missing_q) > 0:
    raise ValueError("Missing fiscal quarters detected. Fix the source data before
```

```
✅ Preprocessing complete
df_levels shape: (35, 6)
df (analysis space) shape: (35, 7)
Date range: 2016-07-01 00:00:00 → 2025-01-01 00:00:00
DATA_IN_LOGS = True
```

In [207…

```
# ============================================================
# CLEANED NOTEBOOK (AFTER PREPROCESSING CELL)
# Suez Revenue / REER / Manufacturing / CPI (+ Rent)
# ============================================================

import numpy as np
import pandas as pd

from statsmodels.tsa.stattools import adfuller, kpss, zivot_andrews
from statsmodels.tsa.vector_ar.vecm import (
    select_order as vecm_select_order,
    select_coint_rank,
    VECM
)
from statsmodels.tsa.api import VAR
from statsmodels.stats.diagnostic import acorr_ljungbox

import matplotlib.pyplot as plt

import warnings
from statsmodels.tools.sm_exceptions import InterpolationWarning
warnings.filterwarnings("ignore", category=InterpolationWarning)
```

```
# ----------------------------
# 0) USER SETTINGS (DO NOT override DATA_IN_LOGS here)
# ----------------------------
MAXLAGS_VAR = 4
HORIZON_IRF = 12
HORIZON_FEVD = 10
ALPHA = 0.05

# ----------------------------
# 1) HELPERS
# ----------------------------
def adf_pvalue(x: pd.Series, regression: str = "c") -> float:
    x = pd.Series(x).dropna()
    if len(x) < 8:
        return np.nan
    try:
        return adfuller(x, regression=regression, autolag="AIC")[1]
    except Exception:
        return np.nan

def kpss_pvalue(x: pd.Series, regression: str = "c") -> float:
    x = pd.Series(x).dropna()
    if len(x) < 8:
        return np.nan
    try:
        stat, pval, lags, crit = kpss(x, regression=regression, nlags="auto")
        return float(pval)
    except Exception:
        return np.nan

def zivot_pvalue(x: pd.Series, regression: str = "c") -> float:
    x = pd.Series(x).dropna()
    if len(x) < 20:
        return np.nan
    try:
        za_stat, za_p, za_cv, bp = zivot_andrews(x, regression=regression, autolag=
        return float(za_p) if za_p is not None else np.nan
    except Exception:
        return np.nan

def stationarity_report(df_in: pd.DataFrame, name: str = "DATA", regressions=("c","
    print(f"\n================ STATIONARITY REPORT: {name} ================")
    for col in df_in.columns:
        s = df_in[col].dropna()
        print(f"\n--- {col} ---")
        for reg in regressions:
            p_lvl = adf_pvalue(s, regression=reg)
            p_d1  = adf_pvalue(s.diff().dropna(), regression=reg)
            p_d2  = adf_pvalue(s.diff().diff().dropna(), regression=reg)
            print(f"ADF({reg}): level p={p_lvl:.4f} | d1 p={p_d1:.4f} | d2 p={p_d2:
        kp_c  = kpss_pvalue(s, regression="c")
        kp_ct = kpss_pvalue(s, regression="ct")
        print(f"KPSS: level p(c)={kp_c:.4f} | p(ct)={kp_ct:.4f}")
        za = zivot_pvalue(s, regression="c")
        if not np.isnan(za):
```

```
            print(f"Zivot-Andrews (approx p): {za:.4f}")

def choose_var_lag(var_model: VAR, maxlags: int = 4, criterion: str = "bic") -> int
    sel = var_model.select_order(maxlags=maxlags)
    print(sel.summary())
    try:
        lag = int(sel.selected_orders.get(criterion, None))
        if lag is None:
            raise ValueError("selected_orders missing criterion")
    except Exception:
        lag = int(getattr(sel, criterion, sel.aic))
    return max(1, lag)

def var_diagnostics(res, name="VAR"):
    print(f"\n================ {name}: STABILITY & DIAGNOSTICS ================")

    try:
        print("Stable?", res.is_stable(verbose=True))
        print("Roots:", res.roots)
    except Exception as e:
        print("Stability check failed:", e)

    try:
        print("\n--- Portmanteau / whiteness test ---")
        print(res.test_whiteness(nlags=8).summary())
    except Exception as e:
        print("Whiteness test failed:", e)

    try:
        print("\n--- Normality test ---")
        print(res.test_normality().summary())
        print("\nNOTE: Normality rejected -> prefer GIRF and bootstrap IRFs for inf

    except Exception as e:
        print("Normality test failed:", e)

    print("\n--- Ljung-Box (per equation residual) ---")
    resid = res.resid
    for col in resid.columns:
        try:
            lb = acorr_ljungbox(resid[col], lags=[4, 8], return_df=True)
            print(f"\n{col}\n{lb}")
        except Exception as e:
            print(f"{col}: Ljung-Box failed:", e)

def generalized_irf(var_res, horizon: int = 12):
    Psi = np.asarray(var_res.ma_rep(horizon))   # (h+1, k, k)
    Sigma = np.asarray(var_res.sigma_u)         # (k, k)
    k = Sigma.shape[0]
    girf = np.zeros((horizon + 1, k, k))

    for h in range(horizon + 1):
        for j in range(k):
            ej = np.zeros((k, 1))
            ej[j, 0] = 1.0
            denom = np.sqrt(Sigma[j, j])
```

```
            if denom == 0 or np.isnan(denom):
                raise ValueError(f"Sigma[{j},{j}] is zero/NaN; cannot scale GIRF.")
            shock = (Sigma @ ej) / denom
            resp = Psi[h] @ shock
            girf[h, :, j] = resp[:, 0]
    return girf

def plot_girf(var_res, impulse: str, response: str, horizon: int = 12, title_prefix
    names = list(var_res.names)
    j = names.index(impulse)
    i = names.index(response)

    girf = generalized_irf(var_res, horizon=horizon)
    y = girf[:, i, j]
    x = np.arange(horizon + 1)

    plt.figure()
    plt.plot(x, y)
    plt.axhline(0, linewidth=1)
    plt.title(f"{title_prefix}: {impulse} → {response}")
    plt.xlabel("Horizon")
    plt.ylabel("Response")
    plt.show()

# ----------------------------
# 2) INPUTS (from preprocessing cell)
# ----------------------------
# preprocessing cell must define:
# - df (analysis dataset; logs if DATA_IN_LOGS True)
# - DATA_IN_LOGS (bool)
# - model_data, model_data_rent (levels in analysis space, includes Rent_Total)
#
# If not present, stop early with a clear message
required_vars = ["Suez Revenue", "REER", "Manufacturing", "CPI", "Rent_Total"]
for v in required_vars:
    if v not in df.columns:
        raise KeyError(f"Expected '{v}' in df.columns, but it is missing. "
                       f"Run the preprocessing cell first. Columns are: {list(df.co

# Baseline + Rent datasets (already created in preprocessing cell, but safe re-crea
model_data = df[["Suez Revenue", "REER", "Manufacturing", "CPI"]].dropna()
model_data_rent = df[["Rent_Total", "REER", "Manufacturing", "CPI"]].dropna()

stationarity_report(model_data, name="Baseline (analysis space)")
stationarity_report(model_data_rent, name="Rent proxy (analysis space)")

def decide_reer_transform(reer, alpha=0.05):
    # ADF constant
    c_d1  = adf_pvalue(reer.diff().dropna(), "c")
    c_d2  = adf_pvalue(reer.diff().diff().dropna(), "c")
    # ADF with trend
    ct_d1 = adf_pvalue(reer.diff().dropna(), "ct")
    ct_d2 = adf_pvalue(reer.diff().diff().dropna(), "ct")

    # if BOTH specs say d1 non-stationary AND d2 stationary → treat as I(2)
    is_i2 = (c_d1 >= alpha and c_d2 < alpha) and (ct_d1 >= alpha and ct_d2 < alpha)
```

```
    if is_i2:
        return "ddREER"
    else:
        return "dREER"


# ----------------------------
# 4) VECM branch (ONLY if all series look I(1) and REER is I(1)
# ----------------------------
def looks_I1(series: pd.Series, alpha=0.05) -> bool:
    p_lvl = adf_pvalue(series, regression="c")
    p_d1  = adf_pvalue(series.diff().dropna(), regression="c")
    return (p_lvl >= alpha) and (p_d1 < alpha)

can_vecm = (reer_var == "dREER") and all(looks_I1(model_data[c], alpha=ALPHA) for c

if can_vecm:
    print("\n================ VECM / COINTEGRATION CHECK ================")
    lag_sel = vecm_select_order(model_data, maxlags=3, deterministic="ci")
    print(lag_sel.summary())

    try:
        k_ar_diff = int(lag_sel.selected_orders["bic"])
    except Exception:
        k_ar_diff = int(lag_sel.aic)

    print(f"Selected k_ar_diff (BIC): {k_ar_diff}")

    rank_test = select_coint_rank(
        model_data,
        det_order=0,
        k_ar_diff=k_ar_diff,
        method="trace"
    )
    print(rank_test.summary())

    rank = int(rank_test.rank)
    print("Suggested cointegration rank:", rank)

    if rank >= 1:
        vecm_res = VECM(
            model_data,
            k_ar_diff=k_ar_diff,
            coint_rank=rank,
            deterministic="ci"
        ).fit()
        print(vecm_res.summary())

        beta = vecm_res.beta[:, 0]
        beta_norm = beta / beta[0]
        print("\nNormalized cointegrating vector (normalized on first variable):")
        print(pd.Series(beta_norm, index=model_data.columns))
    else:
        print("\nRank = 0 → No cointegration. Skipping VECM.")
else:
```

```
    print("\nSkipping VECM (either REER looks I(2) or not all baseline vars look I(

# ----------------------------
# 5) Build STATIONARY VAR dataset (Baseline)
# ----------------------------
if reer_var == "ddREER":
    var_data_base = pd.DataFrame({
        "dSuez Revenue": model_data["Suez Revenue"].diff(),
        "ddREER": model_data["REER"].diff().diff(),
        "dManufacturing": model_data["Manufacturing"].diff(),
        "dCPI": model_data["CPI"].diff()
    }).dropna()
else:
    var_data_base = pd.DataFrame({
        "dSuez Revenue": model_data["Suez Revenue"].diff(),
        "dREER": model_data["REER"].diff(),
        "dManufacturing": model_data["Manufacturing"].diff(),
        "dCPI": model_data["CPI"].diff()
    }).dropna()

print("\nADF check (baseline VAR data):")
for col in var_data_base.columns:
    print(f"{col}: p-value = {adf_pvalue(var_data_base[col]):.4f}")

# ----------------------------
# 6) Fit VAR (Baseline) + diagnostics
# ----------------------------
print("\n================ BASELINE VAR FIT ================")
var_model = VAR(var_data_base)
selected_lag = choose_var_lag(var_model, maxlags=MAXLAGS_VAR, criterion="bic")
print("Selected lag (baseline):", selected_lag)

var_res = var_model.fit(selected_lag, trend="c")
print(var_res.summary())
var_diagnostics(var_res, name="Baseline VAR")

# ----------------------------
# 7) Granger causality (Baseline)
# ----------------------------
print("\n================ GRANGER CAUSALITY (Baseline) ================")
print("\n--- GRANGER: dSuez Revenue → REER dynamics ---")
print(var_res.test_causality(caused=reer_var, causing=["dSuez Revenue"], kind="f").

print("\n--- GRANGER: REER dynamics → dManufacturing ---")
print(var_res.test_causality(caused="dManufacturing", causing=[reer_var], kind="f")

# ----------------------------
# 8) IRFs + FEVD (Baseline)
# ----------------------------
print("\n================ IRF / FEVD (Baseline) ================")

irf = var_res.irf(HORIZON_IRF)
irf.plot(impulse="dSuez Revenue", response=reer_var, signif=0.05, repl=1000)
irf.plot(impulse=reer_var, response="dManufacturing", signif=0.05, repl=1000)
plt.show()
```

```
plot_girf(var_res, impulse="dSuez Revenue", response=reer_var, horizon=HORIZON_IRF,
plot_girf(var_res, impulse=reer_var, response="dManufacturing", horizon=HORIZON_IRF

fevd = var_res.fevd(HORIZON_FEVD)
print(fevd.summary())
fevd.plot()
plt.show()

# ============================================================
# 9) ROBUSTNESS: Rent proxy VAR
# ============================================================
print("\n\n================ RENT PROXY VAR (ROBUSTNESS) ================")

rent_lvl = model_data_rent["Rent_Total"].dropna()
rent_p = adf_pvalue(rent_lvl, regression="c")

if rent_p < ALPHA:
    rent_name = "Rent"
    rent_series = rent_lvl
else:
    rent_name = "dRent"
    rent_series = rent_lvl.diff()

if reer_var == "ddREER":
    reer_series = model_data_rent["REER"].diff().diff()
else:
    reer_series = model_data_rent["REER"].diff()

var_data_rent = pd.DataFrame({
    rent_name: rent_series,
    reer_var: reer_series,
    "dManufacturing": model_data_rent["Manufacturing"].diff(),
    "dCPI": model_data_rent["CPI"].diff()
}).dropna()

print("\nADF check (rent VAR data):")
for col in var_data_rent.columns:
    print(f"{col}: p-value = {adf_pvalue(var_data_rent[col]):.4f}")

print("\n================ RENT VAR FIT ================")
var_model_rent = VAR(var_data_rent)
selected_lag_rent = choose_var_lag(var_model_rent, maxlags=MAXLAGS_VAR, criterion="
print("Selected lag (rent):", selected_lag_rent)

var_res_rent = var_model_rent.fit(selected_lag_rent, trend="c")
print(var_res_rent.summary())
var_diagnostics(var_res_rent, name="Rent proxy VAR")

print("\n================ GRANGER CAUSALITY (Rent VAR) ================")
print(f"\n--- GRANGER: {rent_name} → {reer_var} ---")
print(var_res_rent.test_causality(caused=reer_var, causing=[rent_name], kind="f").s

print(f"\n--- GRANGER: {reer_var} → dManufacturing ---")
print(var_res_rent.test_causality(caused="dManufacturing", causing=[reer_var], kind
```

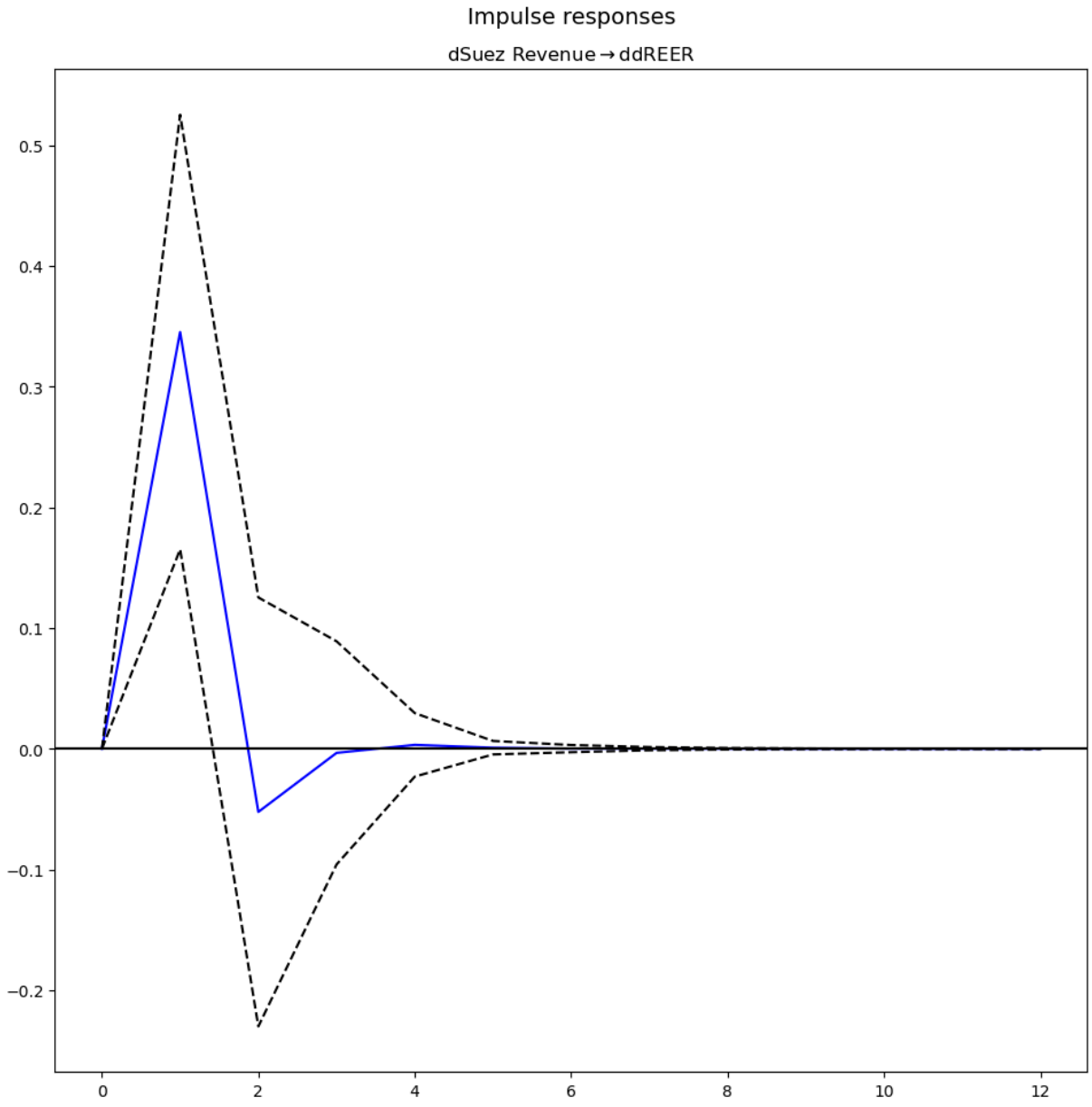
```
print("\n================ IRF / FEVD (Rent VAR) ================")
irf_rent = var_res_rent.irf(HORIZON_IRF)
irf_rent.plot(impulse=rent_name, response=reer_var)
irf_rent.plot(impulse=reer_var, response="dManufacturing")
plt.show()

plot_girf(var_res_rent, impulse=rent_name, response=reer_var, horizon=HORIZON_IRF,
plot_girf(var_res_rent, impulse=reer_var, response="dManufacturing", horizon=HORIZO

fevd_rent = var_res_rent.fevd(HORIZON_FEVD)
print(fevd_rent.summary())
fevd_rent.plot()
plt.show()
```

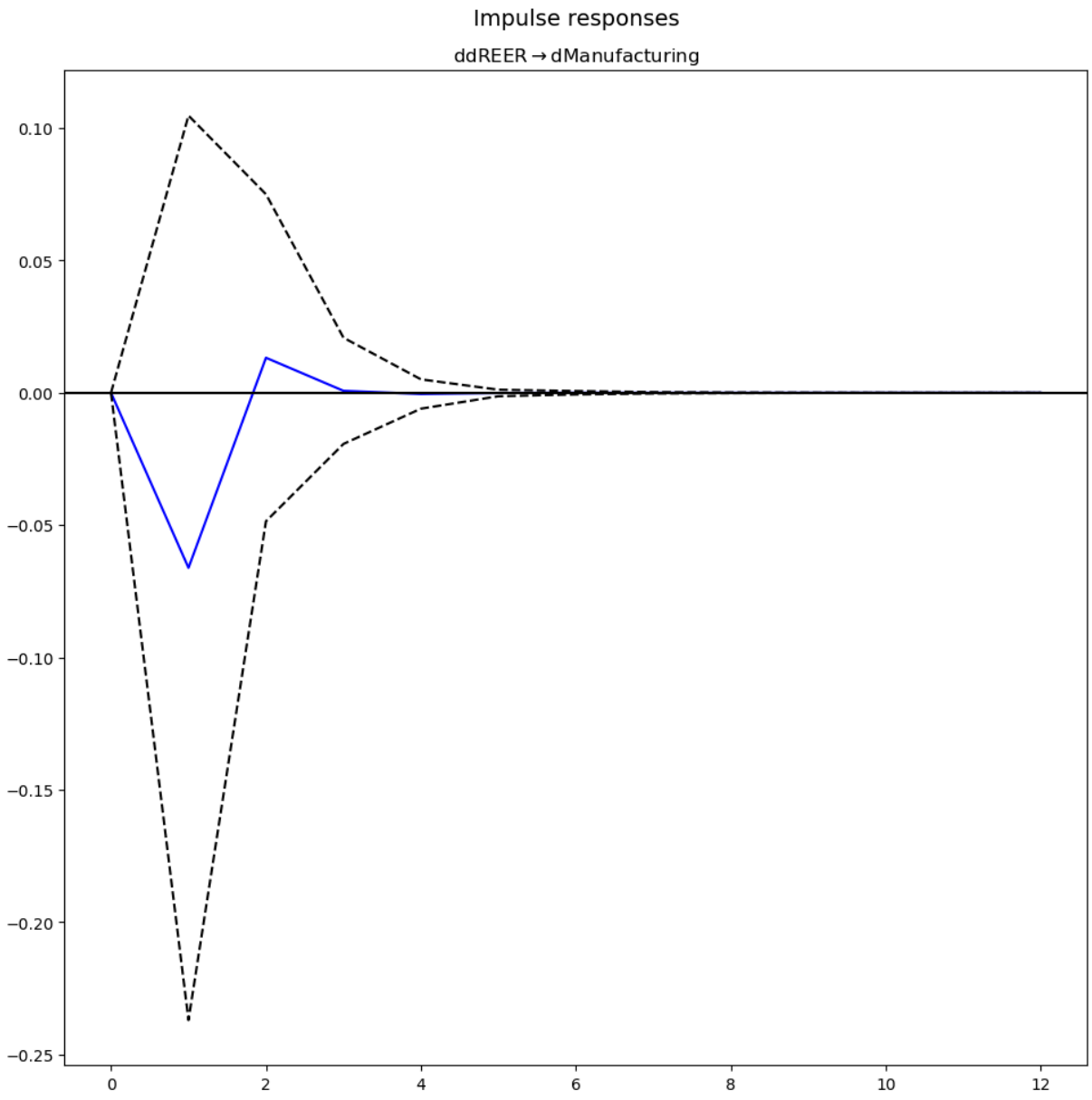
```
================ STATIONARITY REPORT: Baseline (analysis space) ================

--- Suez Revenue ---
ADF(c): level p=0.6351 | d1 p=0.0000 | d2 p=0.0000
ADF(ct): level p=0.9273 | d1 p=0.0000 | d2 p=0.0002
KPSS: level p(c)=0.1000 | p(ct)=0.0882

--- REER ---
ADF(c): level p=0.4831 | d1 p=0.3236 | d2 p=0.0000
ADF(ct): level p=0.9552 | d1 p=0.0332 | d2 p=0.0000
KPSS: level p(c)=0.1000 | p(ct)=0.0149

--- Manufacturing ---
ADF(c): level p=0.3003 | d1 p=0.0000 | d2 p=0.0000
ADF(ct): level p=0.3804 | d1 p=0.0000 | d2 p=0.0000
KPSS: level p(c)=0.0504 | p(ct)=0.1000

--- CPI ---
ADF(c): level p=0.9830 | d1 p=0.0242 | d2 p=0.0000
ADF(ct): level p=0.2921 | d1 p=0.0821 | d2 p=0.3235
KPSS: level p(c)=0.0100 | p(ct)=0.0421

================ STATIONARITY REPORT: Rent proxy (analysis space) ================

--- Rent_Total ---
ADF(c): level p=0.0897 | d1 p=0.0000 | d2 p=0.0000
ADF(ct): level p=0.4087 | d1 p=0.1654 | d2 p=0.0000
KPSS: level p(c)=0.1000 | p(ct)=0.0753

--- REER ---
ADF(c): level p=0.4831 | d1 p=0.3236 | d2 p=0.0000
ADF(ct): level p=0.9552 | d1 p=0.0332 | d2 p=0.0000
KPSS: level p(c)=0.1000 | p(ct)=0.0149

--- Manufacturing ---
ADF(c): level p=0.3003 | d1 p=0.0000 | d2 p=0.0000
ADF(ct): level p=0.3804 | d1 p=0.0000 | d2 p=0.0000
KPSS: level p(c)=0.0504 | p(ct)=0.1000

--- CPI ---
ADF(c): level p=0.9830 | d1 p=0.0242 | d2 p=0.0000
ADF(ct): level p=0.2921 | d1 p=0.0821 | d2 p=0.3235
KPSS: level p(c)=0.0100 | p(ct)=0.0421

Skipping VECM (either REER looks I(2) or not all baseline vars look I(1)).

ADF check (baseline VAR data):
dSuez Revenue: p-value = 0.0000
ddREER: p-value = 0.0000
dManufacturing: p-value = 0.0000
dCPI: p-value = 0.0072

================ BASELINE VAR FIT ================
 VAR Order Selection (* highlights the minimums)
=================================================
      AIC         BIC         FPE         HQIC
```

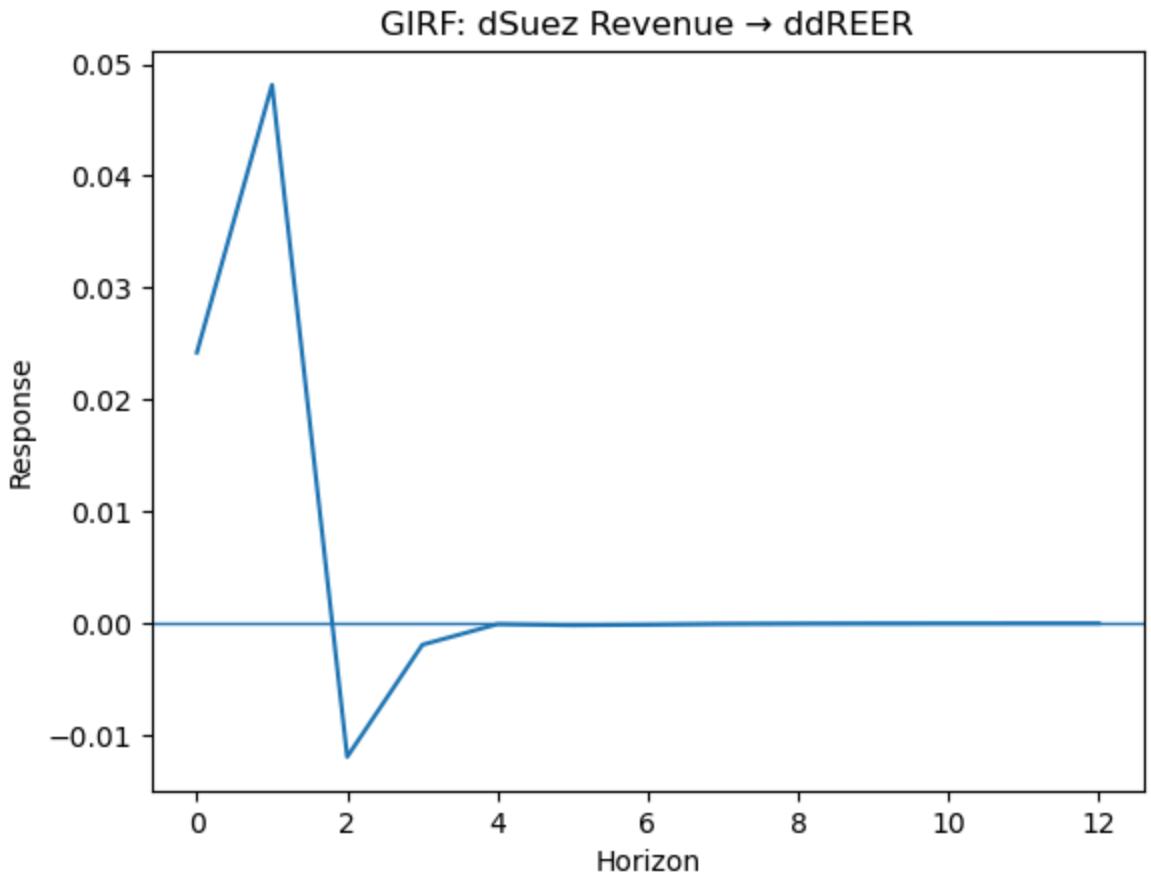
```
------------------------------------------------
0      -21.31     -21.12*   5.588e-10     -21.25*
1      -21.36      -20.42  5.361e-10*      -21.07
2      -21.03      -19.33   8.032e-10      -20.50
3     -21.46*      -19.01   6.300e-10      -20.69
4      -21.20      -17.99   1.234e-09      -20.19
------------------------------------------------
Selected lag (baseline): 1
  Summary of Regression Results
==================================
Model:                         VAR
Method:                        OLS
Date:           Thu, 29, Jan, 2026
Time:                      15:49:02
----------------------------------------------------------------------
No. of Equations:         4.00000    BIC:                   -20.6342
Nobs:                     32.0000    HQIC:                  -21.2466
Log likelihood:           183.180    FPE:                4.41899e-10
AIC:                     -21.5503    Det(Omega_mle):     2.47239e-10
----------------------------------------------------------------------
Results for equation dSuez Revenue
================================================================================
                       coefficient       std. error           t-stat            prob
--------------------------------------------------------------------------------
const                    -0.011142         0.054470           -0.205           0.838
L1.dSuez Revenue          0.093587         0.210340            0.445           0.656
L1.ddREER                -0.237682         0.355881           -0.668           0.504
L1.dManufacturing         0.509639         0.801075            0.636           0.525
L1.dCPI                   0.023425         1.150334            0.020           0.984
================================================================================

Results for equation ddREER
================================================================================
                       coefficient       std. error           t-stat            prob
--------------------------------------------------------------------------------
const                    -0.034464         0.023788           -1.449           0.147
L1.dSuez Revenue          0.345307         0.091857            3.759           0.000
L1.ddREER                -0.331687         0.155416           -2.134           0.033
L1.dManufacturing         0.296930         0.349836            0.849           0.396
L1.dCPI                   1.203283         0.502360            2.395           0.017
================================================================================

Results for equation dManufacturing
================================================================================
                       coefficient       std. error           t-stat            prob
--------------------------------------------------------------------------------
const                     0.001265         0.013346            0.095           0.924
L1.dSuez Revenue          0.026372         0.051535            0.512           0.609
L1.ddREER                -0.066142         0.087194           -0.759           0.448
L1.dManufacturing         0.002287         0.196270            0.012           0.991
L1.dCPI                   0.105778         0.281842            0.375           0.707
================================================================================

Results for equation dCPI
================================================================================
                       coefficient       std. error           t-stat            prob
```

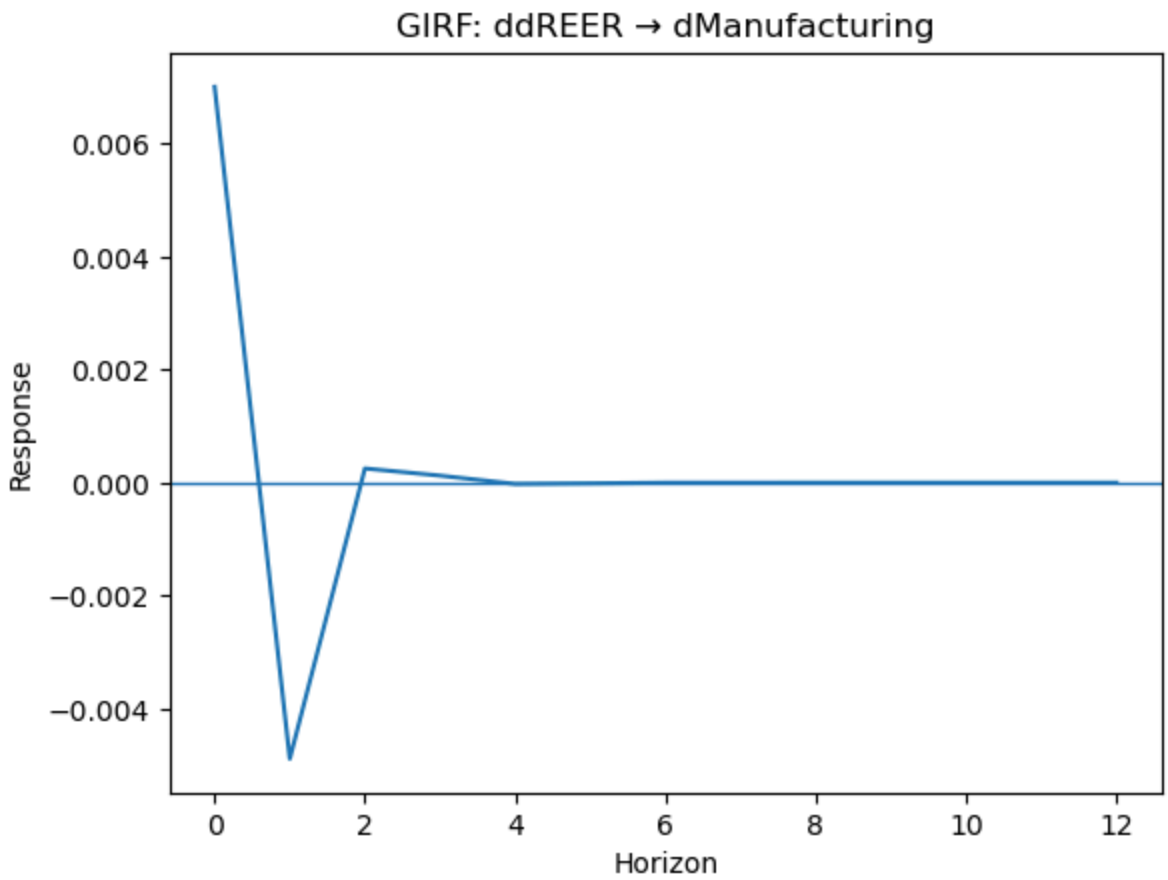
```
--------------------------------------------------------------------------------
const                     0.016572         0.007526            2.202           0.028
L1.dSuez Revenue          0.018379         0.029061            0.632           0.527
L1.ddREER                -0.021564         0.049168           -0.439           0.661
L1.dManufacturing         0.021384         0.110677            0.193           0.847
L1.dCPI                   0.519790         0.158930            3.271           0.001
================================================================================

Correlation matrix of residuals
                  dSuez Revenue    ddREER  dManufacturing      dCPI
dSuez Revenue          1.000000  0.294046        0.180812 -0.364052
ddREER                 0.294046  1.000000        0.151640 -0.334039
dManufacturing         0.180812  0.151640        1.000000 -0.071039
dCPI                  -0.364052 -0.334039       -0.071039  1.000000




================ Baseline VAR: STABILITY & DIAGNOSTICS ================
Eigenvalues of VAR(1) rep
0.1484916505960538
0.21334814276094827
0.21334814276094827
0.4821070016538394
Stable? True
Roots: [-6.73438537-0.j         -0.54527386+4.65534986j -0.54527386-4.65534986j
  2.07422833-0.j        ]

--- Portmanteau / whiteness test ---
Portmanteau-test for residual autocorrelation. H_0: residual autocorrelation up to l
ag 8 is zero. Conclusion: fail to reject H_0 at 5% significance level.
=========================================
Test statistic Critical value p-value  df
-----------------------------------------
         91.18          137.7   0.925 112
-----------------------------------------

--- Normality test ---
normality (skew and kurtosis) test. H_0: data generated by normally-distributed proc
ess. Conclusion: reject H_0 at 5% significance level.
=======================================
Test statistic Critical value p-value df
---------------------------------------
         1019.          15.51   0.000  8
---------------------------------------

NOTE: Normality rejected -> prefer GIRF and bootstrap IRFs for inference.

--- Ljung-Box (per equation residual) ---

dSuez Revenue
    lb_stat  lb_pvalue
4  0.139931   0.997664
8  0.351436   0.999965

ddREER
```

```
     lb_stat  lb_pvalue
4   6.165864   0.187099
8  13.401967   0.098747

dManufacturing
    lb_stat  lb_pvalue
4  2.365481   0.668874
8  4.186060   0.839958

dCPI
    lb_stat  lb_pvalue
4  8.288385   0.081568
8  9.518328   0.300472

================ GRANGER CAUSALITY (Baseline) ================

--- GRANGER: dSuez Revenue → REER dynamics ---
Granger causality F-test. H_0: dSuez Revenue does not Granger-cause ddREER. Conclusi
on: reject H_0 at 5% significance level.
================================================
Test statistic Critical value p-value    df
------------------------------------------------
         14.13          3.929   0.000 (1, 108)
------------------------------------------------

--- GRANGER: REER dynamics → dManufacturing ---
Granger causality F-test. H_0: ddREER does not Granger-cause dManufacturing. Conclus
ion: fail to reject H_0 at 5% significance level.
================================================
Test statistic Critical value p-value    df
------------------------------------------------
        0.5754          3.929   0.450 (1, 108)
------------------------------------------------

================ IRF / FEVD (Baseline) ================
```

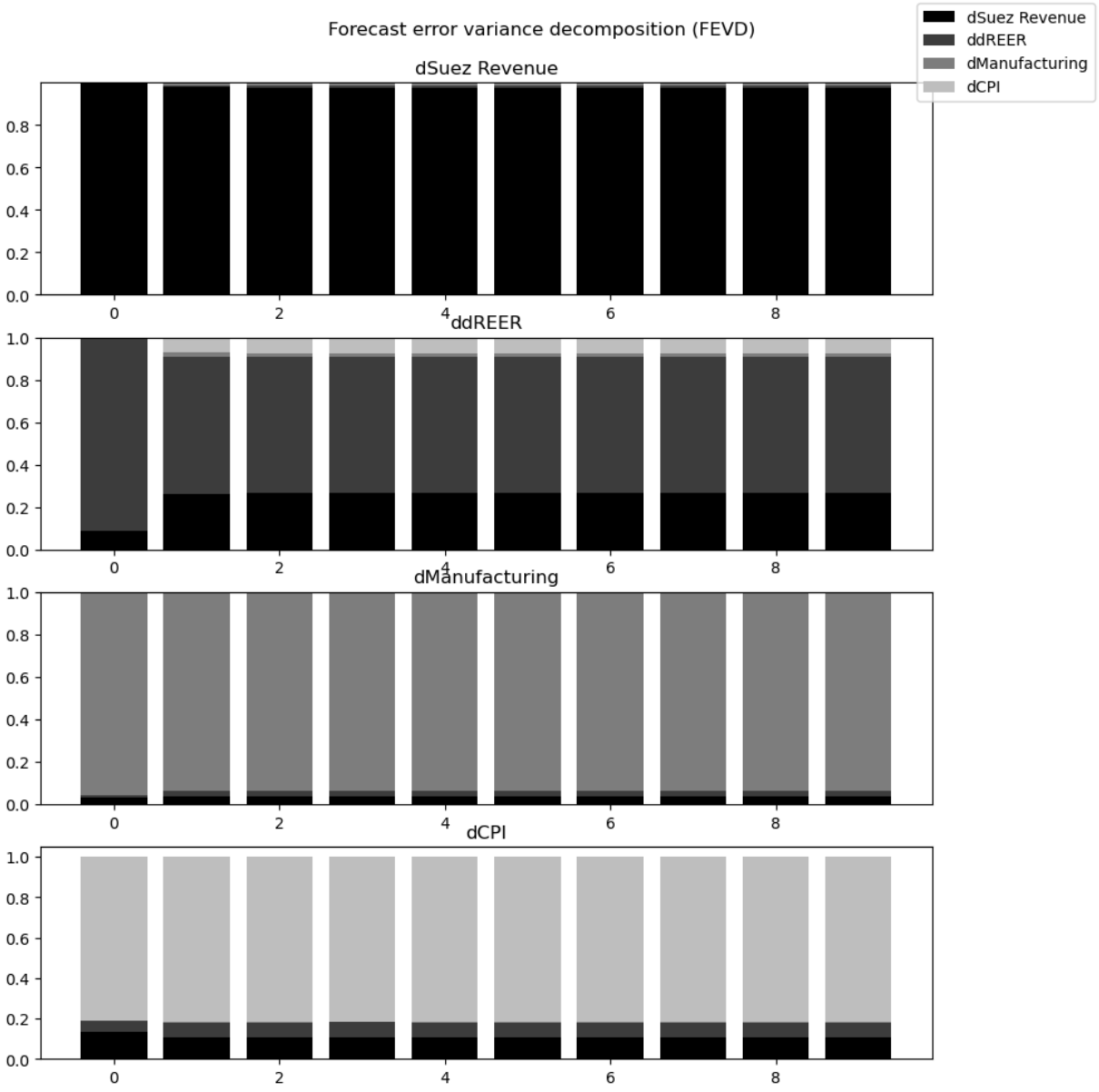

Impulse responses
dSuez Revenue → ddREER
0.5
0.4
0.3
0.2
0.1
0.0
−0.1
−0.2
0
2
4
6
8
10
12


Impulse responses
ddREER → dManufacturing
0.10
0.05
0.00
−0.05
−0.10
−0.15
−0.20
−0.25
0
2
4
6
8
10
12


GIRF: dSuez Revenue → ddREER
Response
Horizon
0.05
0.04
0.03
0.02
0.01
0.00
−0.01
0
2
4
6
8
10
12


GIRF: ddREER → dManufacturing
Response
Horizon
0.006
0.004
0.002
0.000
−0.002
−0.004
0
2
4
6
8
10
12

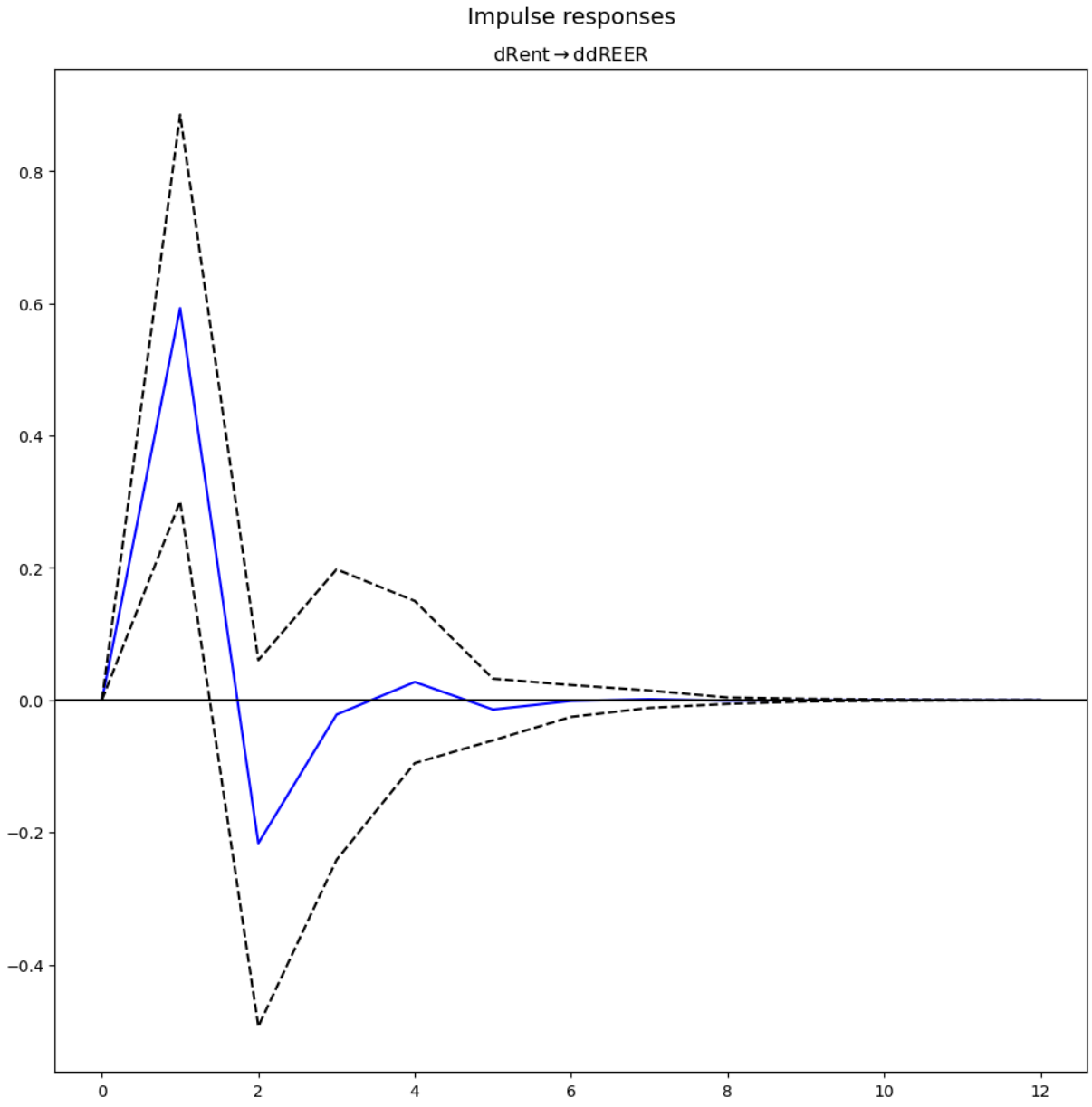
```
FEVD for dSuez Revenue
     dSuez Revenue    ddREER  dManufacturing      dCPI
0         1.000000  0.000000        0.000000  0.000000
1         0.978120  0.007370        0.014502  0.000008
2         0.977200  0.007594        0.014488  0.000718
3         0.976981  0.007647        0.014533  0.000839
4         0.976962  0.007649        0.014532  0.000857
5         0.976957  0.007649        0.014532  0.000862
6         0.976956  0.007649        0.014532  0.000863
7         0.976956  0.007649        0.014532  0.000863
8         0.976956  0.007649        0.014532  0.000863
9         0.976956  0.007649        0.014532  0.000863

FEVD for ddREER
     dSuez Revenue    ddREER  dManufacturing      dCPI
0         0.086463  0.913537        0.000000  0.000000
1         0.261072  0.649620        0.017704  0.071603
2         0.268778  0.638089        0.019434  0.073699
3         0.268771  0.637324        0.019518  0.074387
4         0.268699  0.637182        0.019515  0.074603
5         0.268683  0.637148        0.019515  0.074655
6         0.268680  0.637139        0.019515  0.074666
7         0.268679  0.637138        0.019515  0.074668
8         0.268679  0.637137        0.019515  0.074669
9         0.268679  0.637137        0.019515  0.074669

FEVD for dManufacturing
     dSuez Revenue    ddREER  dManufacturing      dCPI
0         0.032693  0.010615        0.956693  0.000000
1         0.034615  0.026082        0.936480  0.002823
2         0.038424  0.026583        0.932042  0.002951
3         0.038485  0.026580        0.931982  0.002953
4         0.038486  0.026581        0.931979  0.002954
5         0.038486  0.026581        0.931979  0.002954
6         0.038486  0.026581        0.931979  0.002954
7         0.038486  0.026581        0.931979  0.002954
8         0.038486  0.026581        0.931979  0.002954
9         0.038486  0.026581        0.931979  0.002954

FEVD for dCPI
     dSuez Revenue    ddREER  dManufacturing      dCPI
0         0.132534  0.056402        0.000387  0.810677
1         0.108991  0.071822        0.002085  0.817102
2         0.106889  0.074044        0.002640  0.816427
3         0.106508  0.074433        0.002693  0.816366
4         0.106399  0.074526        0.002706  0.816370
5         0.106373  0.074548        0.002709  0.816370
6         0.106367  0.074553        0.002710  0.816369
7         0.106366  0.074555        0.002710  0.816369
8         0.106365  0.074555        0.002710  0.816369
9         0.106365  0.074555        0.002710  0.816369


None
```

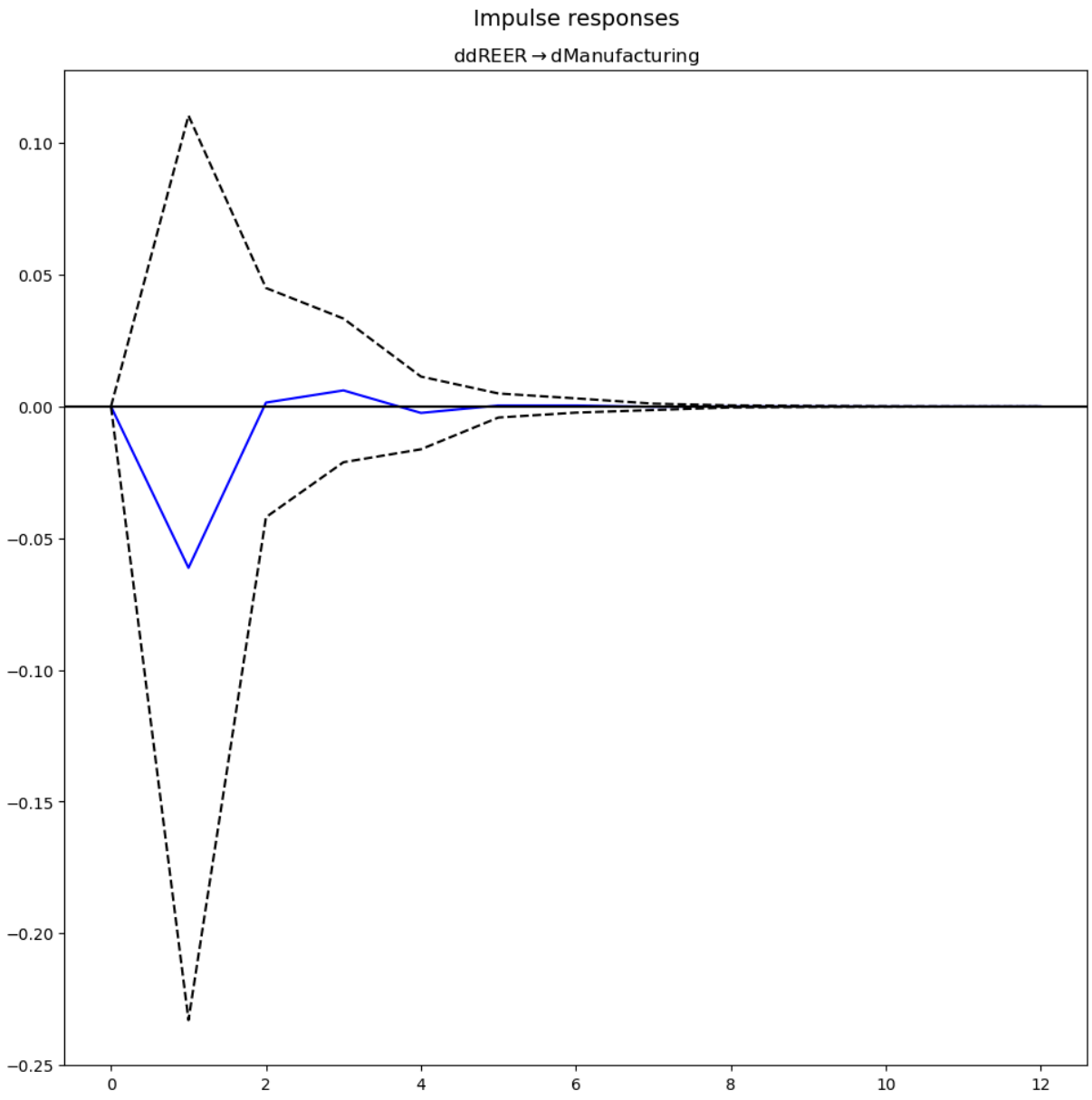

Forecast error variance decomposition (FEVD)
dSuez Revenue
ddREER
dManufacturing
dCPI
dSuez Revenue
ddREER
dManufacturing
dCPI
0.0
0.2
0.4
0.6
0.8
1.0
0
2
4
6
8

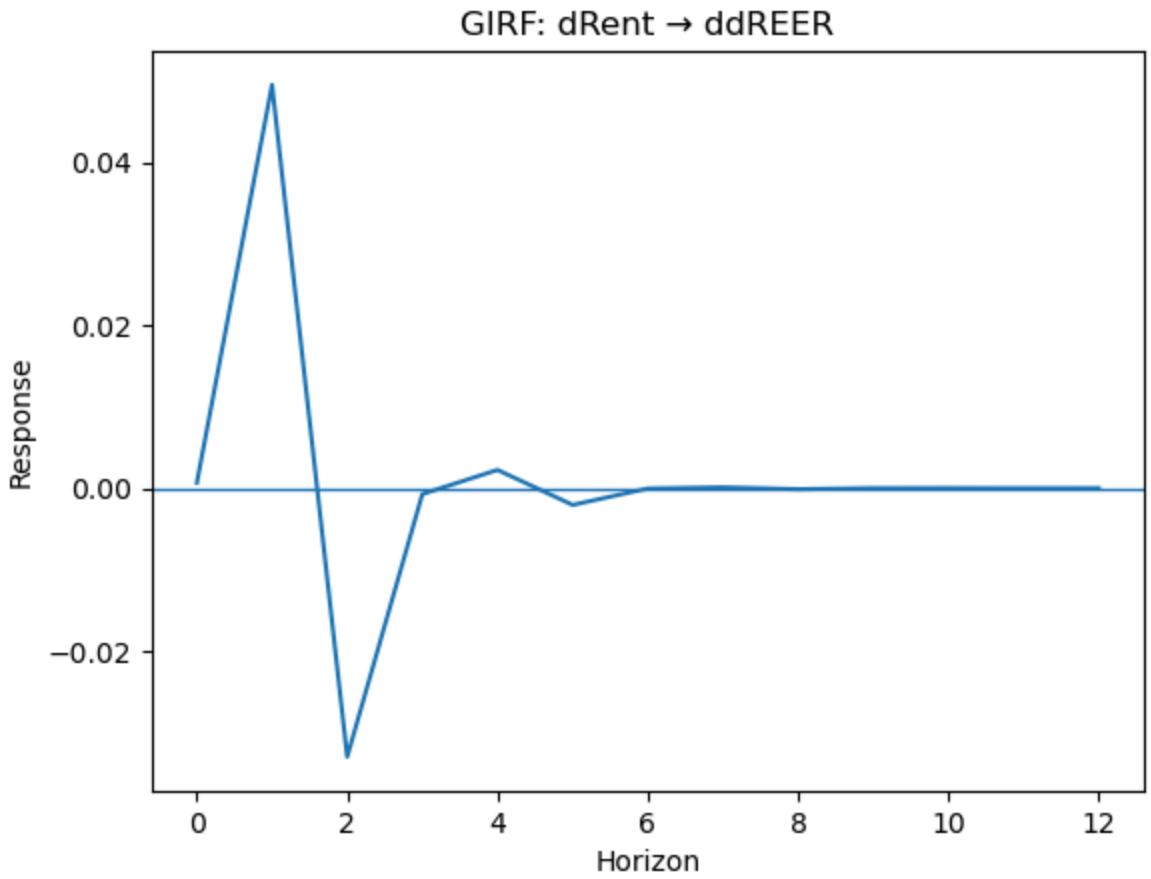
```
================ RENT PROXY VAR (ROBUSTNESS) ================

ADF check (rent VAR data):
dRent: p-value = 0.0000
ddREER: p-value = 0.0000
dManufacturing: p-value = 0.0000
dCPI: p-value = 0.0072

================ RENT VAR FIT ================
 VAR Order Selection (* highlights the minimums)
=================================================
      AIC         BIC         FPE         HQIC
-------------------------------------------------
0      -22.40     -22.21*   1.868e-10     -22.34*
1     -22.58*      -21.64  1.585e-10*      -22.28
2      -21.94      -20.24   3.226e-10      -21.41
3      -22.11      -19.66   3.284e-10      -21.34
4      -21.84      -18.63   6.528e-10      -20.83
-------------------------------------------------
Selected lag (rent): 1
  Summary of Regression Results
==================================
Model:                         VAR
Method:                        OLS
Date:           Thu, 29, Jan, 2026
Time:                      15:49:04
----------------------------------------------------------------------
No. of Equations:         4.00000    BIC:                   -21.9346
Nobs:                     32.0000    HQIC:                  -22.5470
Log likelihood:           203.987    FPE:                1.20383e-10
AIC:                     -22.8507    Det(Omega_mle):     6.73531e-11
----------------------------------------------------------------------
Results for equation dRent
================================================================================================
                       coefficient       std. error           t-stat            prob
------------------------------------------------------------------------------------------------
const                     0.012101         0.032747            0.370           0.712
L1.dRent                 -0.344903         0.206968           -1.666           0.096
L1.ddREER                -0.232428         0.211649           -1.098           0.272
L1.dManufacturing        -0.260431         0.540595           -0.482           0.630
L1.dCPI                   0.180327         0.669986            0.269           0.788
================================================================================================

Results for equation ddREER
================================================================================================
                       coefficient       std. error           t-stat            prob
------------------------------------------------------------------------------------------------
const                    -0.040714         0.023611           -1.724           0.085
L1.dRent                  0.593206         0.149227            3.975           0.000
L1.ddREER                 0.011186         0.152603            0.073           0.942
L1.dManufacturing        -0.270423         0.389778           -0.694           0.488
L1.dCPI                   1.086167         0.483071            2.248           0.025
================================================================================================

Results for equation dManufacturing
```

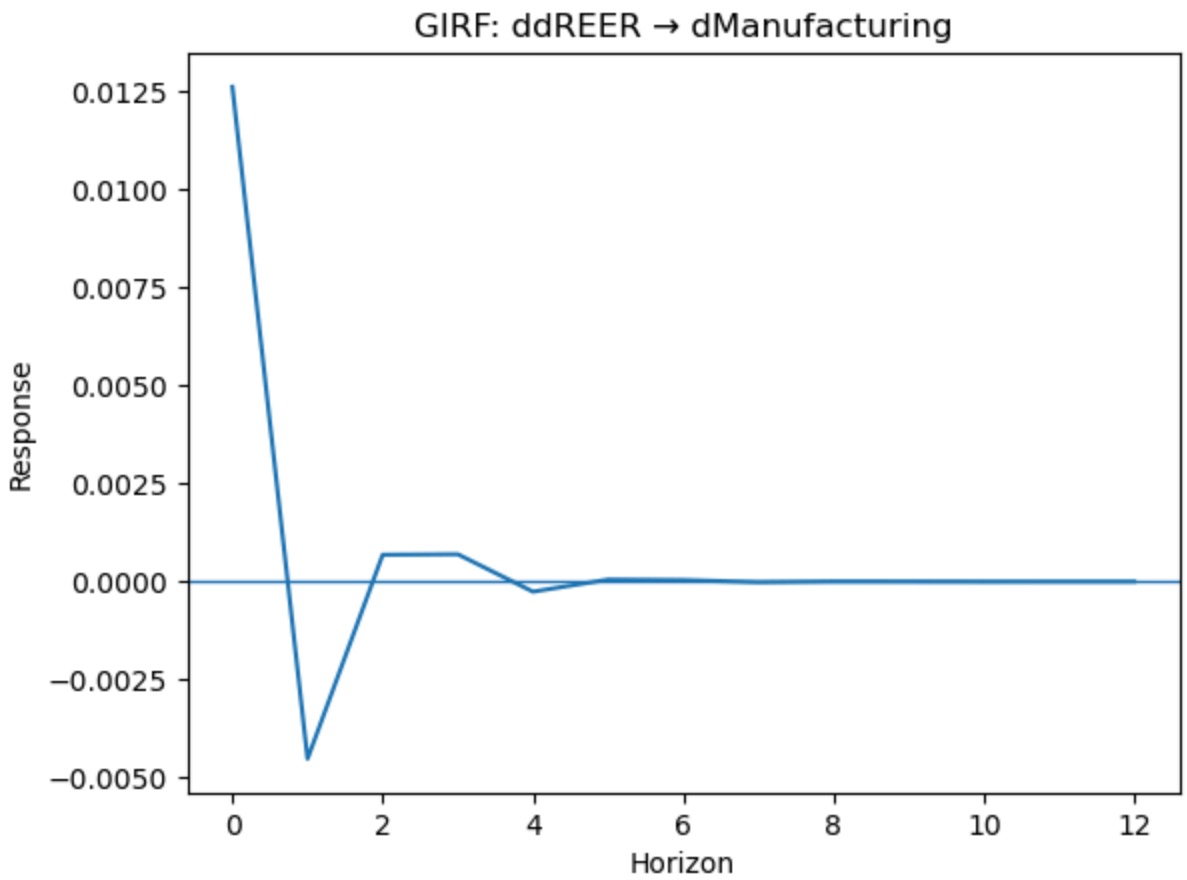
```
================================================================================
                       coefficient       std. error           t-stat            prob
--------------------------------------------------------------------------------
const                     0.003442         0.013555            0.254           0.800
L1.dRent                 -0.026391         0.085668           -0.308           0.758
L1.ddREER                -0.061254         0.087606           -0.699           0.484
L1.dManufacturing         0.051314         0.223763            0.229           0.819
L1.dCPI                   0.041842         0.277320            0.151           0.880
================================================================================

Results for equation dCPI
================================================================================
                       coefficient       std. error           t-stat            prob
--------------------------------------------------------------------------------
const                     0.018286         0.007642            2.393           0.017
L1.dRent                 -0.023722         0.048300           -0.491           0.623
L1.ddREER                -0.019740         0.049392           -0.400           0.689
L1.dManufacturing         0.062417         0.126158            0.495           0.621
L1.dCPI                   0.471145         0.156354            3.013           0.003
================================================================================

Correlation matrix of residuals
                     dRent    ddREER  dManufacturing      dCPI
dRent             1.000000  0.008238        0.447840 -0.399597
ddREER            0.008238  1.000000        0.272028 -0.177895
dManufacturing    0.447840  0.272028        1.000000 -0.064305
dCPI             -0.399597 -0.177895       -0.064305  1.000000




================ Rent proxy VAR: STABILITY & DIAGNOSTICS ================
Eigenvalues of VAR(1) rep
0.38542083129860766
0.38542083129860766
0.1688495594385326
0.42042684184915363
Stable? True
Roots: [ 5.9224318 -0.j         -1.34815606+2.21681097j -1.34815606-2.21681097j
  2.3785351 -0.j        ]

--- Portmanteau / whiteness test ---
Portmanteau-test for residual autocorrelation. H_0: residual autocorrelation up to l
ag 8 is zero. Conclusion: fail to reject H_0 at 5% significance level.
========================================
Test statistic Critical value p-value  df
----------------------------------------
         91.34          137.7   0.924 112
----------------------------------------

--- Normality test ---
normality (skew and kurtosis) test. H_0: data generated by normally-distributed proc
ess. Conclusion: reject H_0 at 5% significance level.
======================================
Test statistic Critical value p-value df
--------------------------------------
```

```
         176.8          15.51   0.000  8
----------------------------------------

NOTE: Normality rejected -> prefer GIRF and bootstrap IRFs for inference.

--- Ljung-Box (per equation residual) ---

dRent
    lb_stat  lb_pvalue
4  1.973852   0.740568
8  9.050594   0.338046

ddREER
    lb_stat  lb_pvalue
4  1.474045   0.831229
8  5.133748   0.743190

dManufacturing
    lb_stat  lb_pvalue
4  2.034817   0.729355
8  3.800008   0.874702

dCPI
    lb_stat  lb_pvalue
4  7.502175   0.111613
8  8.958287   0.345827

================ GRANGER CAUSALITY (Rent VAR) ================

--- GRANGER: dRent → ddREER ---
Granger causality F-test. H_0: dRent does not Granger-cause ddREER. Conclusion: reje
ct H_0 at 5% significance level.
=============================================
Test statistic Critical value p-value    df
---------------------------------------------
         15.80          3.929   0.000 (1, 108)
---------------------------------------------

--- GRANGER: ddREER → dManufacturing ---
Granger causality F-test. H_0: ddREER does not Granger-cause dManufacturing. Conclus
ion: fail to reject H_0 at 5% significance level.
=============================================
Test statistic Critical value p-value    df
---------------------------------------------
        0.4889          3.929   0.486 (1, 108)
---------------------------------------------

================ IRF / FEVD (Rent VAR) ================
```

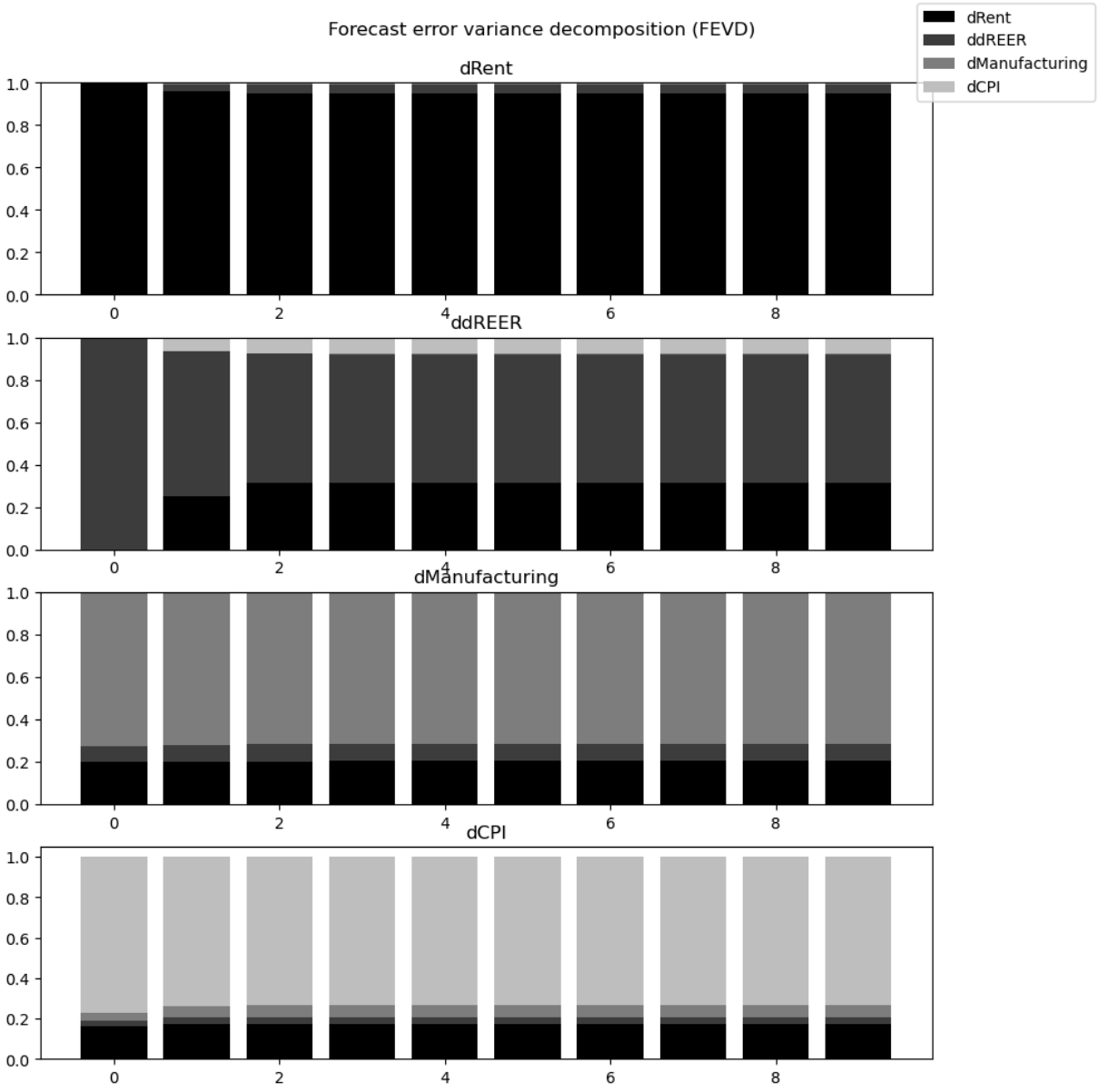

Impulse responses
dRent → ddREER
0.8
0.6
0.4
0.2
0.0
−0.2
−0.4
0
2
4
6
8
10
12


Impulse responses
ddREER → dManufacturing
0.10
0.05
0.00
−0.05
−0.10
−0.15
−0.20
−0.25
0
2
4
6
8
10
12


GIRF: dRent → ddREER
Response
Horizon
0.04
0.02
0.00
−0.02
0
2
4
6
8
10
12


GIRF: ddREER → dManufacturing
Response
Horizon
0.0125
0.0100
0.0075
0.0050
0.0025
0.0000
−0.0025
−0.0050
0
2
4
6
8
10
12

```
FEVD for dRent
       dRent    ddREER  dManufacturing      dCPI
0    1.000000  0.000000        0.000000  0.000000
1    0.959008  0.034085        0.005783  0.001124
2    0.949247  0.040477        0.007185  0.003090
3    0.949291  0.040372        0.007250  0.003087
4    0.949191  0.040446        0.007277  0.003086
5    0.949158  0.040468        0.007280  0.003094
6    0.949158  0.040468        0.007280  0.003094
7    0.949157  0.040468        0.007280  0.003094
8    0.949157  0.040468        0.007280  0.003094
9    0.949157  0.040468        0.007280  0.003094

FEVD for ddREER
       dRent    ddREER  dManufacturing      dCPI
0    0.000068  0.999932        0.000000  0.000000
1    0.253116  0.679504        0.002921  0.064460
2    0.315419  0.608235        0.002615  0.073731
3    0.313864  0.607046        0.005323  0.073768
4    0.314016  0.606524        0.005363  0.074097
5    0.314222  0.606266        0.005364  0.074147
6    0.314215  0.606258        0.005378  0.074149
7    0.314215  0.606256        0.005378  0.074151
8    0.314215  0.606255        0.005378  0.074152
9    0.314215  0.606255        0.005378  0.074152

FEVD for dManufacturing
       dRent    ddREER  dManufacturing      dCPI
0    0.200561  0.072010        0.727429  0.000000
1    0.200210  0.080242        0.719123  0.000424
2    0.201744  0.080195        0.717048  0.001013
3    0.202695  0.080274        0.715859  0.001172
4    0.202685  0.080299        0.715842  0.001175
5    0.202687  0.080299        0.715835  0.001180
6    0.202690  0.080299        0.715830  0.001180
7    0.202690  0.080299        0.715830  0.001180
8    0.202690  0.080299        0.715830  0.001180
9    0.202690  0.080299        0.715830  0.001180

FEVD for dCPI
       dRent    ddREER  dManufacturing      dCPI
0    0.159678  0.030488        0.035858  0.773976
1    0.170540  0.033986        0.054513  0.740961
2    0.173231  0.033634        0.060399  0.732736
3    0.173091  0.033538        0.061677  0.731694
4    0.173124  0.033541        0.061879  0.731457
5    0.173135  0.033539        0.061920  0.731407
6    0.173134  0.033538        0.061928  0.731400
7    0.173134  0.033538        0.061929  0.731399
8    0.173134  0.033538        0.061929  0.731399
9    0.173134  0.033538        0.061929  0.731399


None
```


Forecast error variance decomposition (FEVD)
dRent
ddREER
dManufacturing
dCPI
dRent
ddREER
dManufacturing
dCPI
0.0
0.2
0.4
0.6
0.8
1.0
0
2
4
6
8

In [ ]: