In [1]:
```
import numpy as np
import pandas as pd
```

One-Sheet EXCEL|CSV

In [2]:
```
interest_rate = pd.read_csv(r"D:\Program Files (x86)\Project\PythonProject\Python T
real_wage = pd.read_csv(r"D:\Program Files (x86)\Project\PythonProject\Python Time
adjusted_reserves = pd.read_csv(r"D:\Program Files (x86)\Project\PythonProject\Pyth
ppic = pd.read_csv(r"D:\Program Files (x86)\Project\PythonProject\Python Time Serie
```

Combine One-Sheet EXCEL|CSV

In [3]:
```
names = [interest_rate, real_wage, adjusted_reserves, ppic]
one_sheet_combined_data = pd.DataFrame({'observation_date': adjusted_reserves['obse
for i in names:
    one_sheet_combined_data = pd.merge(one_sheet_combined_data, i, on='observation_
one_sheet_combined_data["observation_date"] = pd.to_datetime(one_sheet_combined_dat
one_sheet_combined_data.head()
```

Out[3]:

	observation_date	FEDFUNDS	COMPRNFB	ADJRESSL	PPICRM
0	1918Q1	NaN	NaN	1.290	NaN
1	1918Q2	NaN	NaN	1.291	NaN
2	1918Q3	NaN	NaN	1.207	NaN
3	1918Q4	NaN	NaN	1.357	NaN
4	1919Q1	NaN	NaN	1.345	NaN

Multi-Sheet EXCEL

GDP

In [4]:
```
gdp = pd.read_excel(r"D:\Program Files (x86)\Project\PythonProject\Python Time Seri
gdp = gdp.iloc[[0], :].T.iloc[3:,].reset_index()
gdp.columns = ['time', 'GDP']
gdp["GDP"] = gdp["GDP"] * 1000000
gdp.head()
```

Out[4]:

	time	GDP
0	1947Q1	243164000000
1	1947Q2	245968000000
2	1947Q3	249585000000
3	1947Q4	259745000000
4	1948Q1	265742000000

Private Consumption

In [5]:
```
private_consumption = pd.read_excel(r"D:\Program Files (x86)\Project\PythonProject\
private_consumption = private_consumption.iloc[[1], :].T.iloc[3:,].reset_index()
private_consumption.columns = ['time', 'private_consumption']
private_consumption["private_consumption"] = private_consumption["private_consumpti
private_consumption.head()
```

Out[5]:

	time	private_consumption
0	1947Q1	156161000000
1	1947Q2	160031000000
2	1947Q3	163543000000
3	1947Q4	167672000000
4	1948Q1	170372000000

Total Government Expenditure

In [6]:
```
government_expenditure = pd.read_excel(r"D:\Program Files (x86)\Project\PythonProje
government_expenditure = government_expenditure.iloc[[21], :].T.iloc[3:,].reset_ind
government_expenditure.columns = ['time', 'government_expenditure']
government_expenditure["government_expenditure"] = government_expenditure["governme
government_expenditure.head()
```

Out[6]:

	time	government_expenditure
0	1947Q1	40274000000
1	1947Q2	40138000000
2	1947Q3	39361000000
3	1947Q4	39534000000
4	1948Q1	40875000000

Total Government Revenue

In [7]:
```
government_revenue = pd.read_excel(r"D:\Program Files (x86)\Project\PythonProject\P
government_current_receipts = government_revenue.iloc[[0], :].T.iloc[3:,].reset_ind
transfers_payments = government_revenue.iloc[[14], :].T.iloc[3:,].reset_index()
interest_payments = government_revenue.iloc[[10], :].T.iloc[3:,].reset_index()



cashe = government_current_receipts.iloc[:, 1] - transfers_payments.iloc[:, 1] - in

government_revenue = pd.DataFrame({'time': government_expenditure['time'], 'governm
government_revenue["government_revenue"] = government_revenue["government_revenue"]
government_revenue.head()
```

Out[7]:

	time	government_revenue
0	1947Q1	54752000000
1	1947Q2	54115000000
2	1947Q3	53708000000
3	1947Q4	56337000000
4	1948Q1	57153000000

Private Non-Residential Investment

In [8]:
```
private_investment = pd.read_excel(r"D:\Program Files (x86)\Project\PythonProject\P
gross_private_investment = (private_investment.iloc[[6], :]).T.iloc[3:,].reset_inde
private_residential_investment = (private_investment.iloc[[12], :]).T.iloc[3:,].res
cashe = gross_private_investment.iloc[:, 1] - private_residential_investment.iloc[:
private_non_residential_investment = pd.DataFrame({'time': government_expenditure['
private_non_residential_investment["private_non_residential_investment"] = private_
private_non_residential_investment.head()
```

Out[8]:

	time	private_non_residential_investment
0	1947Q1	25313000000
1	1947Q2	23932000000
2	1947Q3	22445000000
3	1947Q4	27996000000
4	1948Q1	31857000000

GDP Deflator

In [9]:
```
deflator = pd.read_excel(r"D:\Program Files (x86)\Project\PythonProject\Python Time
deflator = deflator.iloc[[0], :].T.iloc[3:,].reset_index()
deflator.columns = ['time', 'gdp_deflator']
deflator.head()
```

Out[9]:

	time	gdp_deflator
0	1947Q1	11.141
1	1947Q2	11.299
2	1947Q3	11.489
3	1947Q4	11.772
4	1948Q1	11.865

Population

In [10]:
```
population = pd.read_excel(r"D:\Program Files (x86)\Project\PythonProject\Python Ti
population = population.iloc[[42], :].T.iloc[3:,].reset_index()
population.columns = ['time', 'Population']
population["Population"] = population["Population"] * 1000
population.head()
```

Out[10]:

	time	Population
0	1947Q1	143143000
1	1947Q2	143790000
2	1947Q3	144449000
3	1947Q4	145122000
4	1948Q1	145709000

Combine Multi-Sheet EXCEL

In [11]:
```
names = [gdp, private_consumption, government_expenditure, government_revenue, priv
multi_sheet_combined_data = pd.DataFrame({'time': gdp['time']})
for i in names:
    multi_sheet_combined_data = pd.merge(multi_sheet_combined_data, i, on='time', h
multi_sheet_combined_data.head()
```

Out[11]:

	time	GDP	private_consumption	government_expenditure	governmen
0	1947Q1	243164000000	156161000000	40274000000	547
1	1947Q2	245968000000	160031000000	40138000000	541
2	1947Q3	249585000000	163543000000	39361000000	537
3	1947Q4	259745000000	167672000000	39534000000	563
4	1948Q1	265742000000	170372000000	40875000000	571

Combine ALL the data

In [12]:
```
mydata = pd.merge(one_sheet_combined_data, multi_sheet_combined_data, left_on='obse
mydata.rename(columns={'FEDFUNDS': 'Interest Rate', 'COMPRNFB': 'Real Wages', "ADJR
mydata.drop(columns=['observation_date'], inplace=True)
mydata.set_index("time", inplace=True)
mydata.dropna(inplace=True)
mydata.head()
```

Out[12]:

time	Interest Rate	Real Wages	Adjusted Reserves	PPIC	GDP	private_consumption	gover
1954Q3	1.03	42.832	8.191	31.4	390996000000	240303000000	
1954Q4	0.99	43.388	8.362	31.1	399734000000	245093000000	
1955Q1	1.34	43.629	8.339	30.9	413073000000	251398000000	
1955Q2	1.50	44.054	8.358	30.6	421532000000	256466000000	
1955Q3	1.94	44.802	8.320	30.3	430221000000	260651000000	

Save as CSV

In [13]:
```
mydata.to_csv(r"D:\Program Files (x86)\Project\PythonProject\Python Time Series\Rep
```