WifeyAlpha - Research Drop - 50 Indicators

Sir, this is houskeeping, we need to check the keys and make sure our vacuum cleaners are working correctly

In [1]:

```
# Install a conda package in the current Jupyter kernel
# This way you can solve any "ModuleNotFoundError: No module named 'ipympl'" Error by
# putting the module name here where you found ipympl
import sys

#!conda install --yes --prefix {sys.prefix} libgcc
#!conda install --yes --prefix {sys.prefix} fredapi
#!conda update --yes -n base conda
# amazing interactive charts # https://towardsdatascience.com/get-interactive-plots-dire
!conda install --yes --prefix {sys.prefix} -c patrikhlobil pandas-bokeh
!pip install multitasking
# Yahoo Finance
!pip install lxml
# Error in YF
!pip install pandas==1.3.5
!pip install -i https://pypi.anaconda.org/ranaroussi/simple yfinance==0.1.62
# Nasdaq Data Link
!pip install nasdaq-data-link
!pip install matplotlib
#!pip install fredapi
!conda install --yes -c conda-forge fredapi
!pip install requests
```

```
Collecting package metadata (current_repodata.json): done
Solving environment: done

## Package Plan ##

  environment location: /home/p/anaconda3

  added / updated specs:
    - pandas-bokeh


The following packages will be UPDATED:

  ca-certificates    conda-forge::ca-certificates-2022.6.1~ --> pkgs/main::ca-certificat
es-2022.07.19-h06a4308_0

The following packages will be SUPERSEDED by a higher-priority channel:

  conda              conda-forge::conda-4.14.0-py38h578d9b~ --> pkgs/main::conda-4.14.0-
py38h06a4308_0


Preparing transaction: done
Verifying transaction: done
Executing transaction: done
Retrieving notices: ...working... done
Requirement already satisfied: multitasking in /home/p/anaconda3/lib/python3.8/site-pack
ages (0.0.11)
Requirement already satisfied: lxml in /home/p/anaconda3/lib/python3.8/site-packages (4.
9.1)
Requirement already satisfied: pandas==1.3.5 in /home/p/anaconda3/lib/python3.8/site-pac
kages (1.3.5)
Requirement already satisfied: pytz>=2017.3 in /home/p/anaconda3/lib/python3.8/site-pack
```

```
ages (from pandas==1.3.5) (2022.2.1)
Requirement already satisfied: numpy>=1.17.3 in /home/p/anaconda3/lib/python3.8/site-pac
kages (from pandas==1.3.5) (1.23.2)
Requirement already satisfied: python-dateutil>=2.7.3 in /home/p/anaconda3/lib/python3.
8/site-packages (from pandas==1.3.5) (2.8.2)
Requirement already satisfied: six>=1.5 in /home/p/anaconda3/lib/python3.8/site-packages
(from python-dateutil>=2.7.3->pandas==1.3.5) (1.16.0)
Looking in indexes: https://pypi.anaconda.org/ranaroussi/simple
Requirement already satisfied: yfinance==0.1.62 in /home/p/anaconda3/lib/python3.8/site-
packages (0.1.62)
Requirement already satisfied: requests>=2.20 in /home/p/anaconda3/lib/python3.8/site-pa
ckages (from yfinance==0.1.62) (2.28.1)
Requirement already satisfied: lxml>=4.5.1 in /home/p/anaconda3/lib/python3.8/site-packa
ges (from yfinance==0.1.62) (4.9.1)
Requirement already satisfied: pandas>=0.24 in /home/p/anaconda3/lib/python3.8/site-pack
ages (from yfinance==0.1.62) (1.3.5)
Requirement already satisfied: multitasking>=0.0.7 in /home/p/anaconda3/lib/python3.8/si
te-packages (from yfinance==0.1.62) (0.0.11)
Requirement already satisfied: numpy>=1.15 in /home/p/anaconda3/lib/python3.8/site-packa
ges (from yfinance==0.1.62) (1.23.2)
Requirement already satisfied: pytz>=2017.3 in /home/p/anaconda3/lib/python3.8/site-pack
ages (from pandas>=0.24->yfinance==0.1.62) (2022.2.1)
Requirement already satisfied: python-dateutil>=2.7.3 in /home/p/anaconda3/lib/python3.
8/site-packages (from pandas>=0.24->yfinance==0.1.62) (2.8.2)
Requirement already satisfied: urllib3<1.27,>=1.21.1 in /home/p/anaconda3/lib/python3.8/
site-packages (from requests>=2.20->yfinance==0.1.62) (1.26.12)
Requirement already satisfied: idna<4,>=2.5 in /home/p/anaconda3/lib/python3.8/site-pack
ages (from requests>=2.20->yfinance==0.1.62) (3.3)
Requirement already satisfied: certifi>=2017.4.17 in /home/p/anaconda3/lib/python3.8/sit
e-packages (from requests>=2.20->yfinance==0.1.62) (2022.6.15)
Requirement already satisfied: charset-normalizer<3,>=2 in /home/p/anaconda3/lib/python
3.8/site-packages (from requests>=2.20->yfinance==0.1.62) (2.1.1)
Requirement already satisfied: six>=1.5 in /home/p/anaconda3/lib/python3.8/site-packages
(from python-dateutil>=2.7.3->pandas>=0.24->yfinance==0.1.62) (1.16.0)
Requirement already satisfied: nasdaq-data-link in /home/p/anaconda3/lib/python3.8/site-
packages (1.0.4)
Requirement already satisfied: numpy>=1.8 in /home/p/anaconda3/lib/python3.8/site-packag
es (from nasdaq-data-link) (1.23.2)
Requirement already satisfied: inflection>=0.3.1 in /home/p/anaconda3/lib/python3.8/site
-packages (from nasdaq-data-link) (0.5.1)
Requirement already satisfied: pandas>=0.14 in /home/p/anaconda3/lib/python3.8/site-pack
ages (from nasdaq-data-link) (1.3.5)
Requirement already satisfied: more-itertools in /home/p/anaconda3/lib/python3.8/site-pa
ckages (from nasdaq-data-link) (8.14.0)
Requirement already satisfied: python-dateutil in /home/p/anaconda3/lib/python3.8/site-p
ackages (from nasdaq-data-link) (2.8.2)
Requirement already satisfied: six in /home/p/anaconda3/lib/python3.8/site-packages (fro
m nasdaq-data-link) (1.16.0)
Requirement already satisfied: requests>=2.7.0 in /home/p/anaconda3/lib/python3.8/site-p
ackages (from nasdaq-data-link) (2.28.1)
Requirement already satisfied: pytz>=2017.3 in /home/p/anaconda3/lib/python3.8/site-pack
ages (from pandas>=0.14->nasdaq-data-link) (2022.2.1)
Requirement already satisfied: charset-normalizer<3,>=2 in /home/p/anaconda3/lib/python
3.8/site-packages (from requests>=2.7.0->nasdaq-data-link) (2.1.1)
Requirement already satisfied: certifi>=2017.4.17 in /home/p/anaconda3/lib/python3.8/sit
e-packages (from requests>=2.7.0->nasdaq-data-link) (2022.6.15)
Requirement already satisfied: urllib3<1.27,>=1.21.1 in /home/p/anaconda3/lib/python3.8/
site-packages (from requests>=2.7.0->nasdaq-data-link) (1.26.12)
Requirement already satisfied: idna<4,>=2.5 in /home/p/anaconda3/lib/python3.8/site-pack
ages (from requests>=2.7.0->nasdaq-data-link) (3.3)
Requirement already satisfied: matplotlib in /home/p/anaconda3/lib/python3.8/site-packag
es (3.5.3)
Requirement already satisfied: pillow>=6.2.0 in /home/p/anaconda3/lib/python3.8/site-pac
kages (from matplotlib) (9.2.0)
Requirement already satisfied: pyparsing>=2.2.1 in /home/p/anaconda3/lib/python3.8/site-
packages (from matplotlib) (3.0.9)
```

```
Requirement already satisfied: python-dateutil>=2.7 in /home/p/anaconda3/lib/python3.8/s
ite-packages (from matplotlib) (2.8.2)
Requirement already satisfied: numpy>=1.17 in /home/p/anaconda3/lib/python3.8/site-packa
ges (from matplotlib) (1.23.2)
Requirement already satisfied: cycler>=0.10 in /home/p/anaconda3/lib/python3.8/site-pack
ages (from matplotlib) (0.11.0)
Requirement already satisfied: fonttools>=4.22.0 in /home/p/anaconda3/lib/python3.8/site
-packages (from matplotlib) (4.37.1)
Requirement already satisfied: packaging>=20.0 in /home/p/anaconda3/lib/python3.8/site-p
ackages (from matplotlib) (21.3)
Requirement already satisfied: kiwisolver>=1.0.1 in /home/p/anaconda3/lib/python3.8/site
-packages (from matplotlib) (1.4.4)
Requirement already satisfied: six>=1.5 in /home/p/anaconda3/lib/python3.8/site-packages
(from python-dateutil>=2.7->matplotlib) (1.16.0)
Collecting package metadata (current_repodata.json): done
Solving environment: done

## Package Plan ##

  environment location: /home/p/anaconda3

  added / updated specs:
    - fredapi


The following packages will be SUPERSEDED by a higher-priority channel:

  ca-certificates    pkgs/main::ca-certificates-2022.07.19~ --> conda-forge::ca-certific
ates-2022.6.15-ha878542_0
  conda              pkgs/main::conda-4.14.0-py38h06a4308_0 --> conda-forge::conda-4.14.
0-py38h578d9bd_0



Preparing transaction: done
Verifying transaction: done
Executing transaction: done
Retrieving notices: ...working... done
Requirement already satisfied: requests in /home/p/anaconda3/lib/python3.8/site-packages
(2.28.1)
Requirement already satisfied: charset-normalizer<3,>=2 in /home/p/anaconda3/lib/python
3.8/site-packages (from requests) (2.1.1)
Requirement already satisfied: urllib3<1.27,>=1.21.1 in /home/p/anaconda3/lib/python3.8/
site-packages (from requests) (1.26.12)
Requirement already satisfied: idna<4,>=2.5 in /home/p/anaconda3/lib/python3.8/site-pack
ages (from requests) (3.3)
Requirement already satisfied: certifi>=2017.4.17 in /home/p/anaconda3/lib/python3.8/sit
e-packages (from requests) (2022.6.15)
```

In [2]:

```python
from fredapi import Fred

# importing matplotlib module
import pandas as pd
# for interactive plots
import pandas_bokeh
# for static plots
import matplotlib.pyplot as plt
# initalize bohek interactive
pandas_bokeh.output_notebook()

# personal use only! Yahoo finance
import yfinance as yf
# nasdaq data link! Todo: Check for usages under commercial
import nasdaqdatalink
import datetime
```

```
# here are the tweets
# Part 1:
# https://twitter.com/wifeyalpha/status/1563608271310622720?s=21&t=oLpOLjdycNyKnM_c6kcJW
# Part 2:
# https://twitter.com/wifeyalpha/status/1565058274294104066?s=21&t=oLpOLjdycNyKnM_c6kcJW
# Part 3:
# https://twitter.com/wifeyalpha/status/1566093228142743554?s=21&t=oLpOLjdycNyKnM_c6kcJW
```

BokehJS 2.4.3 successfully loaded.

In [3]:

```
import requests
imf_url = 'http://dataservices.imf.org/REST/SDMX_JSON.svc/'
```

In [4]:

```
# Make an account with fred and get a free key!
fred = Fred(api_key='')
```

Wifey Data Download

In [5]:

```
# Wifey Download
# get data from fred
# 1. GDP deflator
gdp_deflator=fred.get_series('GDPDEF')
# 2. Gold price
data_gold_futures=yf.download('GC=F', period='max')
# 3. Misery Index
# Misery Index is based on two timeseries which you need to add
us_unemployment=fred.get_series('UNRATE')
# this series is not posted as delta
us_cpi=fred.get_series('CPIAUCSL')
us_median_cpi_change=fred.get_series('MEDCPIM159SFRBCLE')
# 4. PPI
us_ppi=fred.get_series('PPIACO')
```

```
[*********************100%***********************]  1 of 1 completed
```

In [6]:

```
# 5. Retail investment activity
# No direct link given, here is an example
# https://data.nasdaq.com/databases/RTAT/documentation?anchor=product-overview
# Premium + Sample
# nasdaq_top_10_retail_flow=nasdaqdatalink.get_table('NDAQ/RTAT10', date='2022-09-02', t
#weekly
us_retail_money_market_funds=fred.get_series('RMFNS')
# weekly
us_assets_on_margin_with_brokers=fred.get_series('BOGZ1FL663067003Q')
#monthly
#us_assets_on_margin_with_brokers=fred.get_series('RMFNS')
```

In [7]:

```
# Here is the link to the data https://www.theia.org/industry-data/fund-statistics/retai
# 6. Credit spreads
# here some data, divide and conquer
y30tips=fred.get_series('DFII30')
y20tips=fred.get_series('DFII20')
y10tips=fred.get_series('DFII10')
y7tips=fred.get_series('DFII7')
y5tips=fred.get_series('DFII5')
y30=fred.get_series('DGS30')
y20=fred.get_series('DGS20')
y10=fred.get_series('DGS10')
y7=fred.get_series('DGS7')
y5=fred.get_series('DGS5')
y3=fred.get_series('DGS3')
y2=fred.get_series('DGS2')
```

```
y1=fred.get_series('DGS1')
m6=fred.get_series('DGS6MO')
m3=fred.get_series('DGS3MO')
m1=fred.get_series('DGS1MO')
fedfund=fred.get_series('FEDFUNDS')
aaa_credit_effective=fred.get_series('BAMLC0A1CAAAEY')
bbb_credit_effective=fred.get_series('BAMLC0A4CBBBEY')
ccc_credit_effective=fred.get_series('BAMLH0A3HYCEY')
# 7. TED Spread
# you could go libor vs 10y or you take the fed funds rate
```

In [8]:

```
# 8. Texas Zombie Bank Ratio
# api specs https://banks.data.fdic.gov/docs/#/Historical/getHistorical
# For simplicity, we just download the xlsx and parse it
# not working any more xlrd stopped support
# texas_zombi_bank_ratio=pd.read_excel("https://www.fdic.gov/analysis/quarterly-banking-

# 9. TIPS spread
# y10tips - fedfund

# 10. VIX Index
data_vix_total=yf.download("^VIX", period="max")

# 11. VIXEN Index - Hot waitresses - or maybe nasdaq 100 vol?
data_vixen_total=yf.download("^VXN", period="max")
```

```
[*********************100%***********************]  1 of 1 completed
[*********************100%***********************]  1 of 1 completed
```

In [9]:

```
# Round 2 Wifey
# GDP = C + I + G + NX
# CONSUMPTION (C)
# 12. Automobile sales
# Wifey goes to a dead link (for me) here are total vehicle sales from fred, sir
auto_sales=fred.get_series('TOTALSA')

# 13. Chain Store Sales
# Redbook - I am sorry sir, this data is premium
#nasdaq_redbook=nasdaqdatalink.get('SGE/USARDBK')

# 14. Consumer Sentiment
umcsics=fred.get_series('UMCSENT')

# 15. Existing Home Sales
existing_home_sales=fred.get_series('EXHOSLUSM495S')
existing_home_sales_inventory=fred.get_series('HOSINVUSM495N')

# 16. Underemployment
slack=fred.get_series('LNS12032195')

# INVESTMENT (I)
# 17. Book-To-Bill
# premium

# 18. Copper
copper_comex=yf.download("HG=F", period="max")
# lme is premium copper_lme https://www.lme.com/en/metals/non-ferrous/lme-copper#Trading
copper_lme_spot_usd=nasdaqdatalink.get('ODA/PCOPP_USD') # should be free

imf_key_copper = ''
# TODO: IMF Calls
#copper_imf=(requests.get(f'{imf_url}{imf_key_copper}').json()['CompactData']['DataSet']

# 19. Durable Goods Orders
durable_goods_orders=fred.get_series('DGORDER')
```

```
# 20. Housing Permits and Starts
building_permits=fred.get_series('PERMIT')
housing_starts=fred.get_series('HOUST')
# 21. Industrial Production and Capacity Utilization
indu_prod=fred.get_series('INDPRO')
capacity_util=fred.get_series('TCU')

# 22. ISM Manufacturing
# This is not allowed to be published and they made this very clear, however there is so
# !!! DATA ONLY TILL 05.2016 !!!
ism_pmi=nasdaqdatalink.get('FRED/NAPM')

# 23. ISM Non Manufacturing
# This is not allowed to be published and they made this very clear, however there is so
# !!! DATA ONLY TILL 05.2016 !!!
ism_nmi=nasdaqdatalink.get('FRED/NMFCI')

# 24. JoC-ECRI Industrial Price Index
# Premium

# 25. London Metal Exchange Inventories
# https://www.lme.com/en/market-data/accessing-market-data/historical-data # Premium
# https://www.cmegroup.com/ftp/bulletin/ # Premium
# https://www.shfe.com.cn/en/products/Copper/

# 26. Personal Saving Rate
personal_savings_rate=fred.get_series('PSAVERT')

# 27. Unit Labor Costs
unit_labor_cost=fred.get_series('ULCNFB')


# GOVERNMENT (G)
# 28. Federal Govenment Budget Deficits and the National Debt
federal_debt=fred.get_series('GFDEBTN')
federal_surplus_deficit=fred.get_series('MTSDS133FMS')


# NET EXPROTS (NX)
# 29. Baltic Dry Index
# maybe possible through https://eodhistoricaldata.com/financial-summary/BDIY.INDX

# 30. Big Mac Index
# https://data.nasdaq.com/data/ECONOMIST-the-economist-big-mac-index # A lot of free ind
big_mac_index_switzerland=nasdaqdatalink.get('ECONOMIST/BIGMAC_CHE')
big_mac_index_china=nasdaqdatalink.get('ECONOMIST/BIGMAC_CHN')

# 31. Current Account Deficit
current_account=fred.get_series('IEABC')
```

```
[*********************100%***********************]  1 of 1 completed
```

In [10]:

```
# 32. Oil Inventories
oil_inventory=pd.read_excel("https://www.eia.gov/dnav/pet/hist_xls/WTTSTUS1w.xls","Data

# 33. Tankan Survey
#https://www.boj.or.jp/en/statistics/tk/index.htm/
# A mother load of data in excel files. See above

# 34. TIC Data
tic_data=pd.read_csv("https://treasury.gov/resource-center/data-chart-center/tic/Documen
```

In [ ]:

In [11]:

```
# Round 3 Wifey
# 35. Beige Book
# goodread: https://www.federalreserve.gov/monetarypolicy/beige-book-default.htm

# 36. Crack Spread
oil_futures=yf.download("CL=F", period="max")
gasoline_futures=yf.download("RB=F", period="max")

# 37. Credit Availability Oscillator
# Prop: https://sec.report/CIK/0001329948

# 38. Federal Funds Rate
fedfund=fred.get_series('FEDFUNDS')

# 39. Fertility Rates
# https://data.nasdaq.com/data/FRED-federal-reserve-economic-data?keyword=fertility%20ra
fertility_ger=nasdaqdatalink.get('FRED/SPADOTFRTDEU')

# 40. Gross Domestic Product per Capita
real_gdp_per_capita=fred.get_series('A939RX0Q048SBEA')

# 41. Libor
libor=fred.get_series('LIOR3M')

# 42. M2 Money Supply
m2=fred.get_series('M2SL')
data_spx_total=yf.download("SPY", start="1955-01-01", end="2023-01-01")
# ^SPX is buggy
#data_spx_total=yf.download("^SPX", start="1923-01-01", end="2023-01-01")

# 43. New Home Sales
median_sales_price_of_houses_sold_for_the_united_states=fred.get_series('MSPUS')

# 44. The Aruoba-Diebold-Scotti Business Conditiona Index
# XLSX is no longer supported
#ads_bc_index=pd.read_excel('https://www.philadelphiafed.org/-/media/frbp/assets/surveys

# 45. Business Outlook Survey
business_index_philly=pd.read_csv('https://www.philadelphiafed.org/-/media/frbp/assets/s

# 46. Real Interest Rates
nominal_rate=y1
cpi_ex_food_energy=fred.get_series('CORESTICKM159SFRBATL')
ppi=fred.get_series('PPIACO')
gdp_pce_deflator=fred.get_series('A191RI1Q225SBEA')
# 47. Short Interest
# premium https://data.nasdaq.com/databases/NSIR/documentation and https://shortsqueeze.

# 48. Russel 2000
rut=yf.download("^RUT", period="max")

# 49. Weekly Leading Index
# Premium

# 50. Yield Curve
# see top rate download
```

```
[*********************100%***********************]  1 of 1 completed
[*********************100%***********************]  1 of 1 completed
[*********************100%***********************]  1 of 1 completed
[*********************100%***********************]  1 of 1 completed
```

In [12]:

```
# Round 3 Wifey
# 35. Beige Book
# goodread: https://www.federalreserve.gov/monetarypolicy/beige-book-default.htm
# 36. Crack Spread
oil_futures=yf.download("CL=F", period="max")
gasoline_futures=yf.download("RB=F", period="max")

# 37. Credit Availability Oscillator
# Prop: https://sec.report/CIK/0001329948

# 38. Federal Funds Rate
fedfund=fred.get_series('FEDFUNDS')

# 39. Fertility Rates
# https://data.nasdaq.com/data/FRED-federal-reserve-economic-data?keyword=fertility%20ra
#### LimitExceededError: (Status 429) (Nasdaq Data Link Error QELx04) You have exceeded
fertility_ger=nasdaqdatalink.get('FRED/SPADOTFRTDEU')

# 40. Gross Domestic Product per Capita
real_gdp_per_capita=fred.get_series('A939RX0Q048SBEA')

# 41. Libor
libor=fred.get_series('LIOR3M')

# 42. M2 Money Supply
m2=fred.get_series('M2SL')
#data_spx_total=yf.download("^SPX", start="1923-01-01", end="2023-01-01")

# 43. New Home Sales
median_sales_price_of_houses_sold_for_the_united_states=fred.get_series('MSPUS')

# 44. The Aruoba-Diebold-Scotti Business Conditiona Index
# XLSX is no longer supported
#ads_bc_index=pd.read_excel('https://www.philadelphiafed.org/-/media/frbp/assets/surveys

# 45. Business Outlook Survey
business_index_philly=pd.read_csv('https://www.philadelphiafed.org/-/media/frbp/assets/s

# 46. Real Interest Rates
nominal_rate=y1
cpi_ex_food_energy=fred.get_series('CORESTICKM159SFRBATL')
ppi=fred.get_series('PPIACO')
gdp_pce_deflator=fred.get_series('A191RI1Q225SBEA')
# 47. Short Interest
# premium https://data.nasdaq.com/databases/NSIR/documentation and https://shortsqueeze.

# 48. Russel 2000
rut=yf.download("^RUT", period="max")

# 49. Weekly Leading Index
# Premium

# 50. Yield Curve
# see top rate download
```

```
[*********************100%***********************]  1 of 1 completed
[*********************100%***********************]  1 of 1 completed
[*********************100%***********************]  1 of 1 completed
```

Wifey Round 1

GDP Deflator

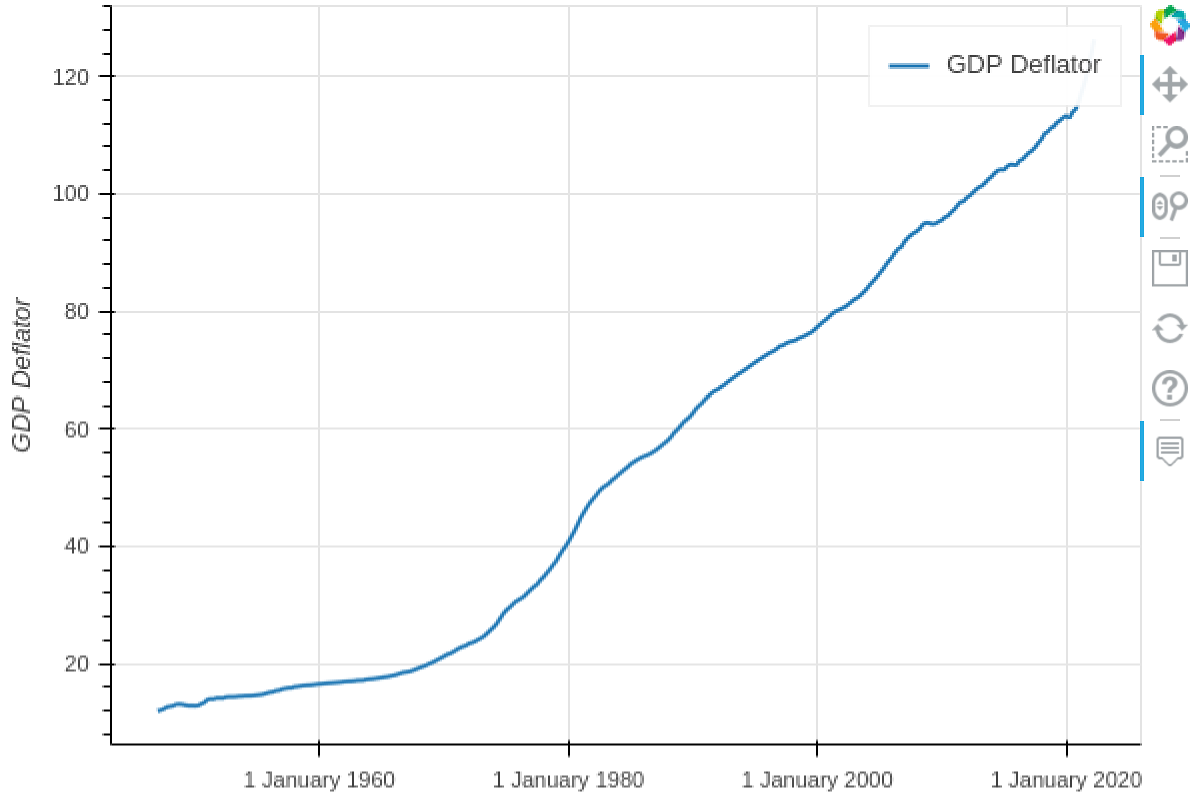
Changes in relative inflation rates between two countries.

In [13]:
```
pd.set_option('plotting.backend', 'pandas_bokeh')
```

```
# now everything must be in one dataframe
```

In [14]:

```
output_gdp_deflator=pd.DataFrame(gdp_deflator)
output_gdp_deflator.columns=['GDP Deflator']
output_gdp_deflator.plot(kind='line')
```

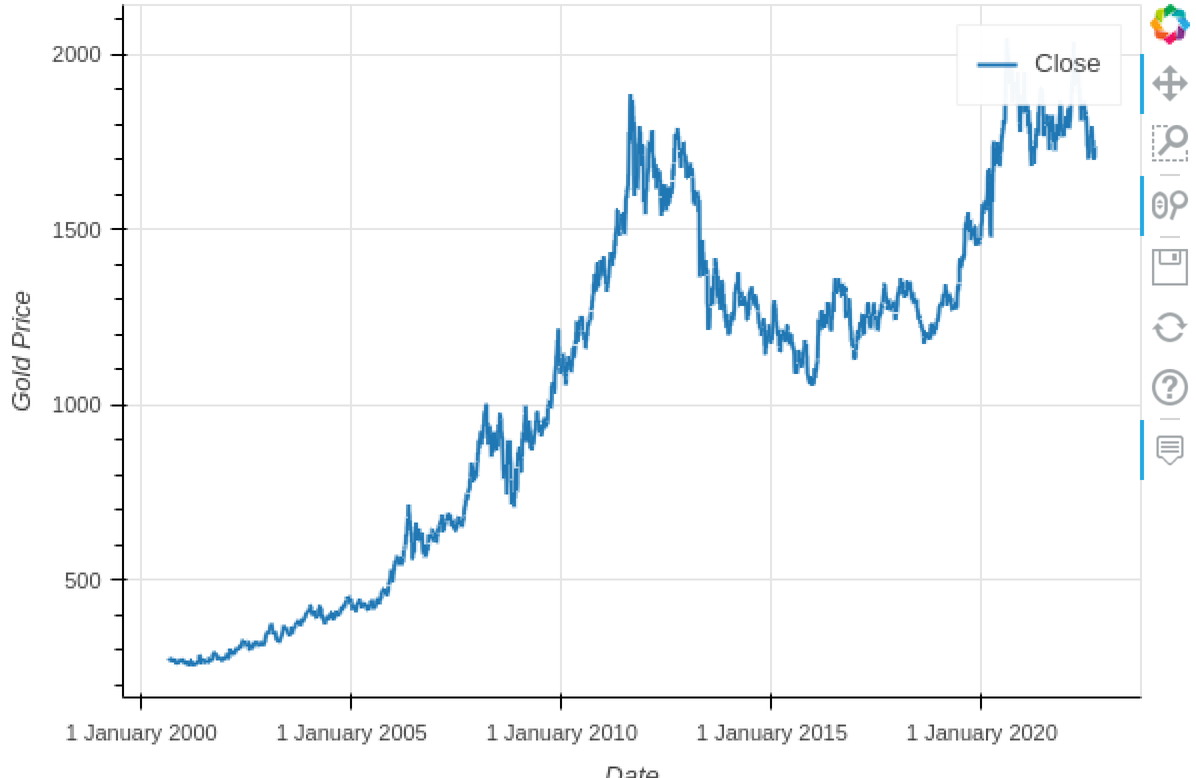


Out[14]: **Figure**(id = '1003', …)

Gold

In [15]:

```
output_gold_futures=data_gold_futures['Close']
output_gold_futures
output_gold_futures.plot(kind="line",ylabel="Gold Price")
```



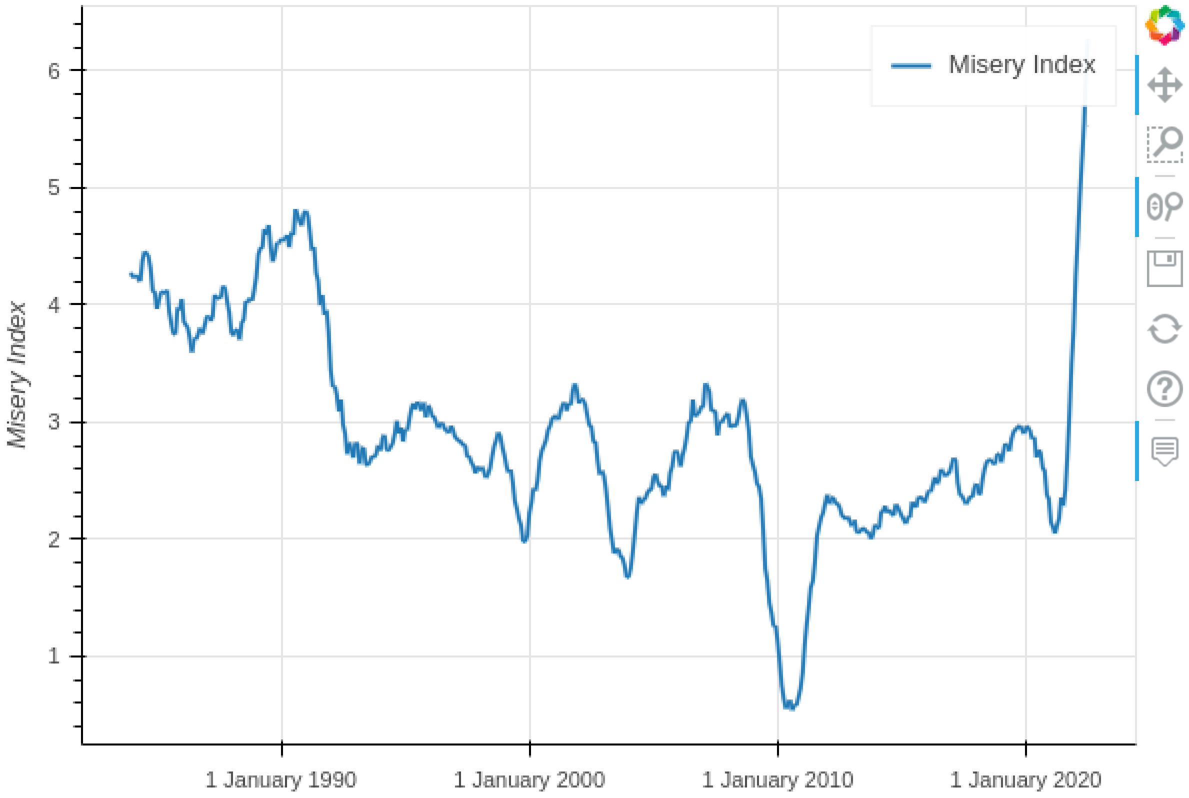
Out[15]: **Figure**(id = '1200', …)

Misery Index

Combining the CPI and the unemployment rate to mark the situation people are in

In [16]:

```
out_misery_index=pd.DataFrame(us_median_cpi_change)# us_unemployment
out_us_unemployment=pd.DataFrame(us_unemployment)
out_misery_index.add(out_us_unemployment)
out_misery_index=out_misery_index.dropna()
out_misery_index.columns=['Misery Index']
out_misery_index.plot(kind='line')
```

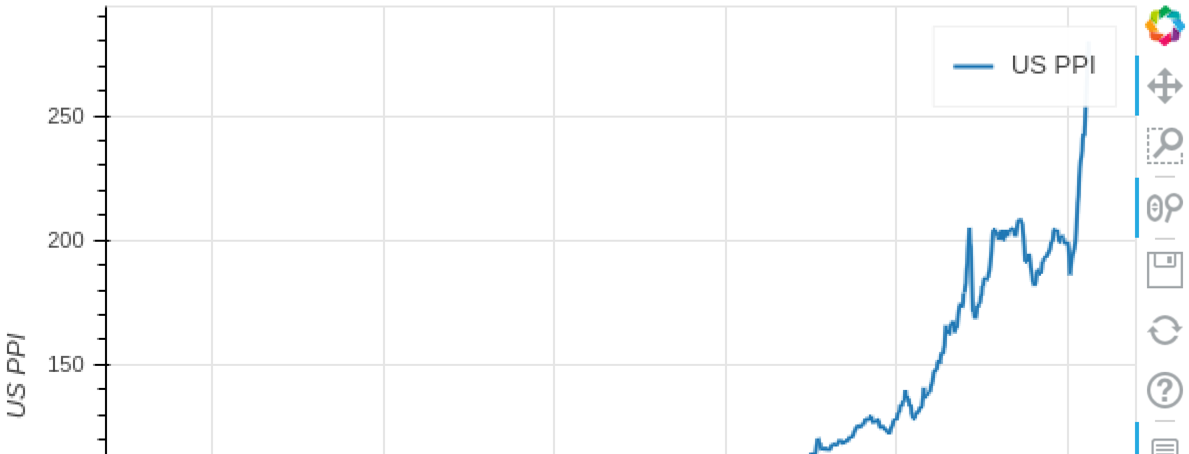


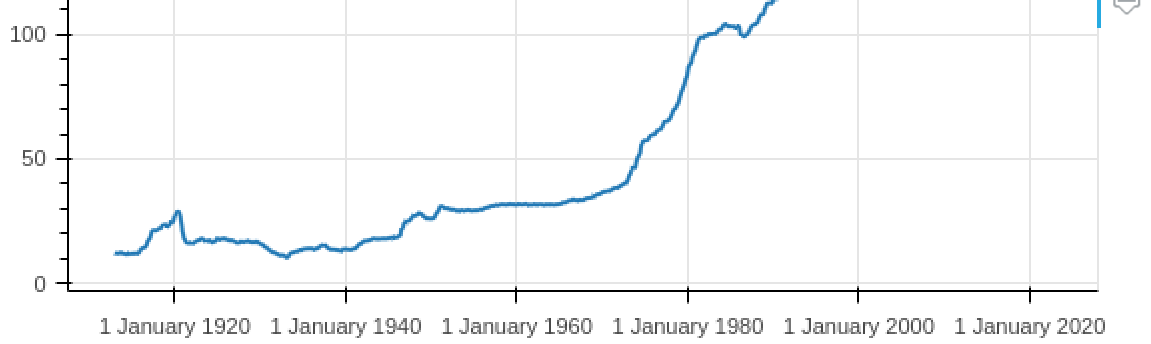
Out[16]: **Figure**(id = '1418', …)

Producer Price Index

In [17]:

```
out_us_ppi=pd.DataFrame(us_ppi)
out_us_ppi.columns=['US PPI']
out_us_ppi.plot(kind='line')
```





Out[17]: **Figure**(id = '1657', …)

Retail Investment Activity

In [18]:

```
out_retail_mm=pd.DataFrame(us_retail_money_market_funds)
out_retail_margin=pd.DataFrame(us_assets_on_margin_with_brokers)
out_retail_investment_activity=out_retail_mm.add(out_retail_margin)
out_retail_margin.dropna()
```

Out[18]:

	0
1945-10-01	1252.0
1946-10-01	594.0
1947-10-01	636.0
1948-10-01	605.0
1949-10-01	969.0
...	...
2021-01-01	573708.0
2021-04-01	595233.0
2021-07-01	571464.0
2021-10-01	574733.0
2022-01-01	582858.0

288 rows × 1 columns

In [19]:

```
y30tipsdf = pd.DataFrame(y30tips)
y20tipsdf = pd.DataFrame(y20tips)
y10tipsdf = pd.DataFrame(y10tips)
y7tipsdf = pd.DataFrame(y7tips)
y5tipsdf = pd.DataFrame(y5tips)
y30tipsdf.columns = ["30y Tips"]
y20tipsdf.columns = ["20y Tips"]
y10tipsdf.columns = ["10y Tips"]
y7tipsdf.columns = ["7y Tips"]
y5tipsdf.columns = ["5y Tips"]
y30df = pd.DataFrame(y30)
y20df = pd.DataFrame(y20)
y10df = pd.DataFrame(y10)
y7df = pd.DataFrame(y7)
y5df = pd.DataFrame(y5)
y3df = pd.DataFrame(y3)
y2df = pd.DataFrame(y2)
y1df = pd.DataFrame(y1)
m6df = pd.DataFrame(m6)
m3df = pd.DataFrame(m3)
```

```
m1df = pd.DataFrame(m1)
fedfunddf = pd.DataFrame(fedfund)
y30df.columns = ["30y"]
y20df.columns = ["20y"]
y10df.columns = ["10y"]
y7df.columns = ["7y"]
y5df.columns = ["5y"]
y3df.columns = ["3y"]
y2df.columns = ["2y"]
y1df.columns = ["1y"]
m6df.columns = ["6m"]
m3df.columns = ["3m"]
m1df.columns = ["1m"]
fedfunddf.columns = ["Fed Fund Rate"]
```

In [20]:

```
out_join_rates=(fedfunddf.join(m1df,how='outer').join(m3df,how='outer').join(m6df,how='o
    .join(y1df,how='outer').join(y2df,how='outer').join(y3df,how='outer').join(y5df,how=
    .join(y7df,how='outer').join(y10df,how='outer').join(y20df,how='outer').join(y30df,h
    .join(y5tipsdf,how='outer').join(y7tipsdf,how='outer').join(y10tipsdf,how='outer').j
    .join(y30tipsdf,how='outer'))
```

In [21]:

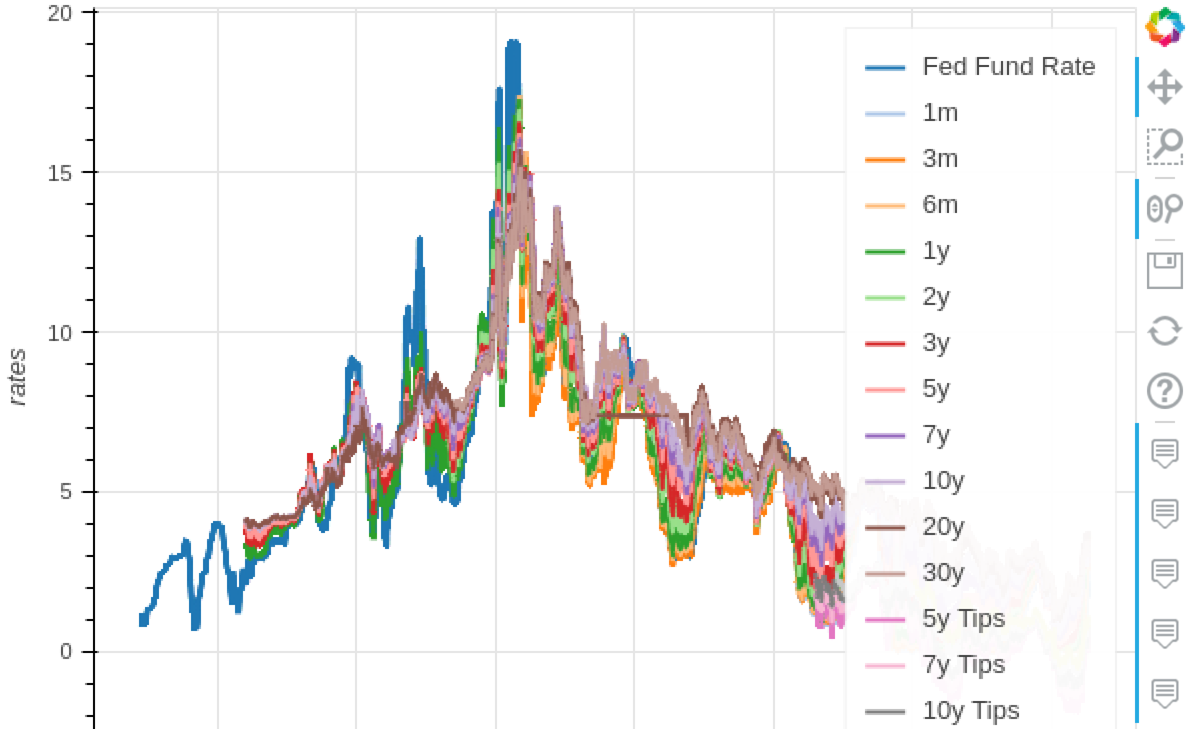
```
out_join_rates=out_join_rates.fillna(method="ffill")
print_out_join_rates=out_join_rates.tail(30)
print_out_join_rates = print_out_join_rates.iloc[::-1]
print_out_join_rates.tail(30)
```

Out[21]:

	Fed Fund Rate	1m	3m	6m	1y	2y	3y	5y	7y	10y	20y	30y	5y Tips	7y Tips	10y Tips	20y Tips	30y Tips
2022-09-06	2.33	2.44	3.04	3.40	3.61	3.50	3.55	3.43	3.41	3.33	3.74	3.49	0.87	0.86	0.85	0.99	1.11
2022-09-05	2.33	2.49	2.94	3.33	3.47	3.40	3.44	3.30	3.29	3.20	3.61	3.35	0.75	0.74	0.73	0.89	1.01
2022-09-02	2.33	2.49	2.94	3.33	3.47	3.40	3.44	3.30	3.29	3.20	3.61	3.35	0.75	0.74	0.73	0.89	1.01
2022-09-01	2.33	2.53	2.97	3.34	3.51	3.51	3.54	3.39	3.36	3.26	3.64	3.37	0.87	0.84	0.81	0.95	1.07
2022-08-31	2.33	2.40	2.96	3.32	3.50	3.45	3.46	3.30	3.25	3.15	3.53	3.27	0.72	0.69	0.67	0.82	0.96
2022-08-30	2.33	2.43	2.97	3.31	3.48	3.46	3.47	3.27	3.22	3.11	3.49	3.23	0.60	0.58	0.56	0.73	0.87
2022-08-29	2.33	2.45	2.97	3.32	3.43	3.42	3.45	3.27	3.21	3.12	3.50	3.25	0.56	0.55	0.54	0.74	0.90
2022-08-26	2.33	2.39	2.89	3.26	3.36	3.37	3.40	3.20	3.14	3.04	3.44	3.21	0.47	0.47	0.47	0.68	0.85
2022-08-25	2.33	2.42	2.88	3.25	3.33	3.35	3.37	3.15	3.11	3.03	3.47	3.25	0.38	0.39	0.43	0.68	0.87
2022-08-24	2.33	2.29	2.82	3.28	3.35	3.36	3.40	3.20	3.20	3.11	3.55	3.32	0.41	0.44	0.49	0.74	0.92
2022-08-23	2.33	2.28	2.80	3.21	3.29	3.29	3.35	3.18	3.14	3.05	3.49	3.26	0.41	0.43	0.47	0.71	0.89
2022-08-22	2.33	2.27	2.82	3.23	3.32	3.32	3.36	3.17	3.12	3.03	3.48	3.24	0.43	0.43	0.46	0.70	0.88
2022-08-19	2.33	2.23	2.74	3.16	3.26	3.25	3.28	3.11	3.06	2.98	3.44	3.22	0.39	0.40	0.43	0.68	0.87
2022-08-18	2.33	2.23	2.71	3.12	3.24	3.22	3.23	3.02	2.97	2.88	3.35	3.14	0.32	0.33	0.36	0.64	0.86

2022-08-17	2.33	2.22	2.68	3.15	3.27	3.28	3.27	3.04	2.99	2.89	3.37	3.15	0.39	0.40	0.43	0.71	0.92
2022-08-16	2.33	2.26	2.70	3.12	3.26	3.25	3.19	2.95	2.90	2.82	3.31	3.11	0.34	0.35	0.38	0.67	0.89
2022-08-15	2.33	2.27	2.72	3.13	3.23	3.20	3.14	2.91	2.86	2.79	3.31	3.10	0.29	0.31	0.35	0.66	0.89
2022-08-12	2.33	2.23	2.63	3.13	3.26	3.25	3.18	2.97	2.92	2.84	3.34	3.12	0.30	0.33	0.37	0.67	0.89
2022-08-11	2.33	2.24	2.62	3.08	3.25	3.23	3.16	2.98	2.94	2.87	3.38	3.15	0.31	0.34	0.40	0.69	0.90
2022-08-10	2.33	2.24	2.65	3.13	3.26	3.23	3.13	2.93	2.86	2.78	3.27	3.04	0.32	0.33	0.35	0.61	0.82
2022-08-09	2.33	2.23	2.67	3.16	3.33	3.28	3.20	2.97	2.89	2.80	3.24	3.01	0.31	0.32	0.33	0.58	0.78
2022-08-08	2.33	2.23	2.65	3.15	3.30	3.21	3.14	2.91	2.85	2.77	3.22	3.00	0.26	0.27	0.29	0.55	0.74
2022-08-05	2.33	2.21	2.58	3.10	3.29	3.24	3.18	2.97	2.91	2.83	3.27	3.06	0.33	0.34	0.37	0.62	0.81
2022-08-04	2.33	2.19	2.50	2.98	3.11	3.03	2.95	2.76	2.73	2.68	3.15	2.97	0.12	0.16	0.22	0.53	0.74
2022-08-03	2.33	2.20	2.52	3.00	3.14	3.10	3.04	2.86	2.81	2.73	3.17	2.96	0.14	0.18	0.23	0.51	0.72
2022-08-02	2.33	2.22	2.56	3.00	3.09	3.06	3.02	2.85	2.82	2.75	3.22	3.00	0.16	0.20	0.27	0.57	0.79
2022-08-01	2.33	2.22	2.56	2.96	2.98	2.90	2.82	2.66	2.64	2.60	3.12	2.92	-0.06	0.00	0.09	0.43	0.67
2022-07-29	1.68	2.22	2.41	2.91	2.98	2.89	2.83	2.70	2.70	2.67	3.20	3.00	-0.03	0.04	0.14	0.47	0.71
2022-07-28	1.68	2.20	2.42	2.90	2.93	2.85	2.81	2.69	2.69	2.68	3.23	3.02	0.02	0.09	0.20	0.53	0.75
2022-07-27	1.68	2.14	2.44	2.93	3.00	2.96	2.93	2.82	2.83	2.78	3.26	3.03	0.21	0.27	0.36	0.62	0.80

```
In [22]: out_join_rates.plot(kind='line',ylabel="rates")
```



Out[22]: **Figure**(id = '1917', …)

The Yield Curve

IG, HY / Junk Spreads over 10 years

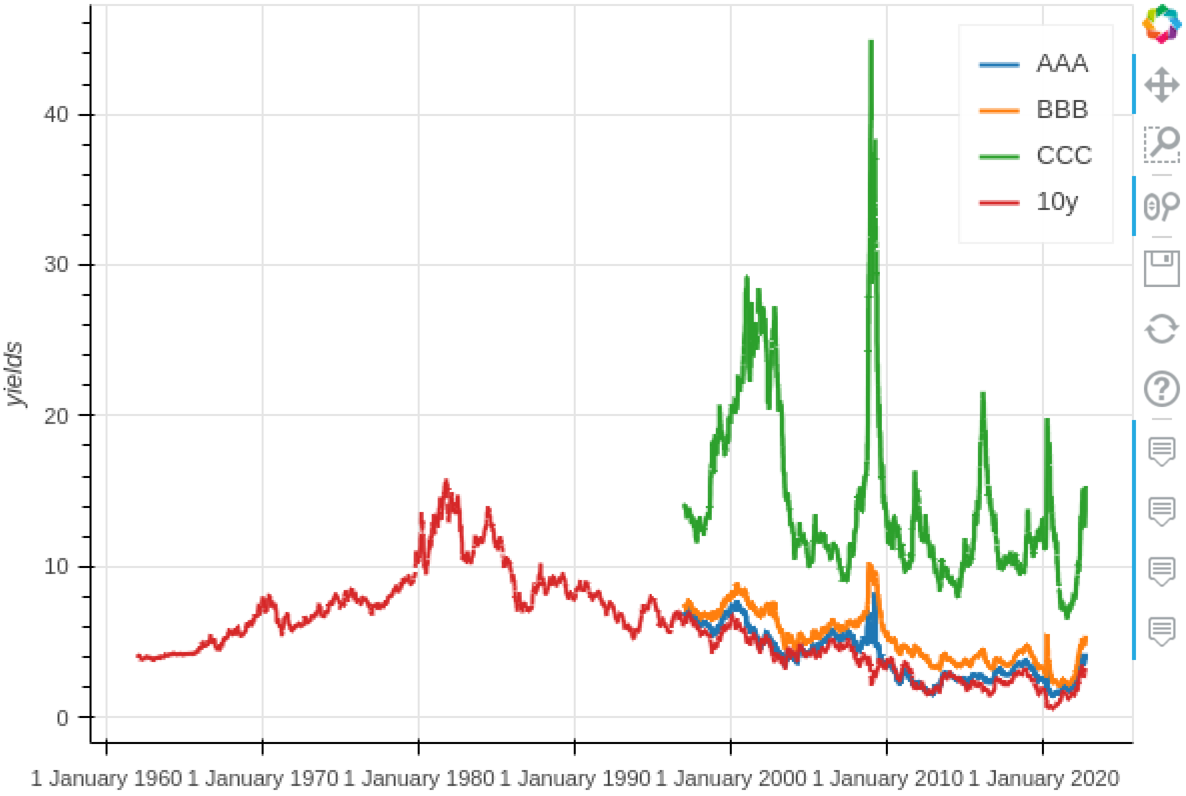
In [23]:

```
aaa_df = pd.DataFrame(aaa_credit_effective)
aaa_df.columns = ["AAA"]
bbb_df = pd.DataFrame(bbb_credit_effective)
bbb_df.columns = ["BBB"]
ccc_df = pd.DataFrame(ccc_credit_effective)
ccc_df.columns = ["CCC"]
y10_df = pd.DataFrame(y10)
y10_df.columns = ["10y"]


out_join_yields=(aaa_df
                 .join(bbb_df,how='outer')
                 .join(ccc_df,how='outer')
                 .join(y10_df,how='outer'))
```

In [24]:

```
out_join_yields.plot(kind='line',ylabel="yields")
```



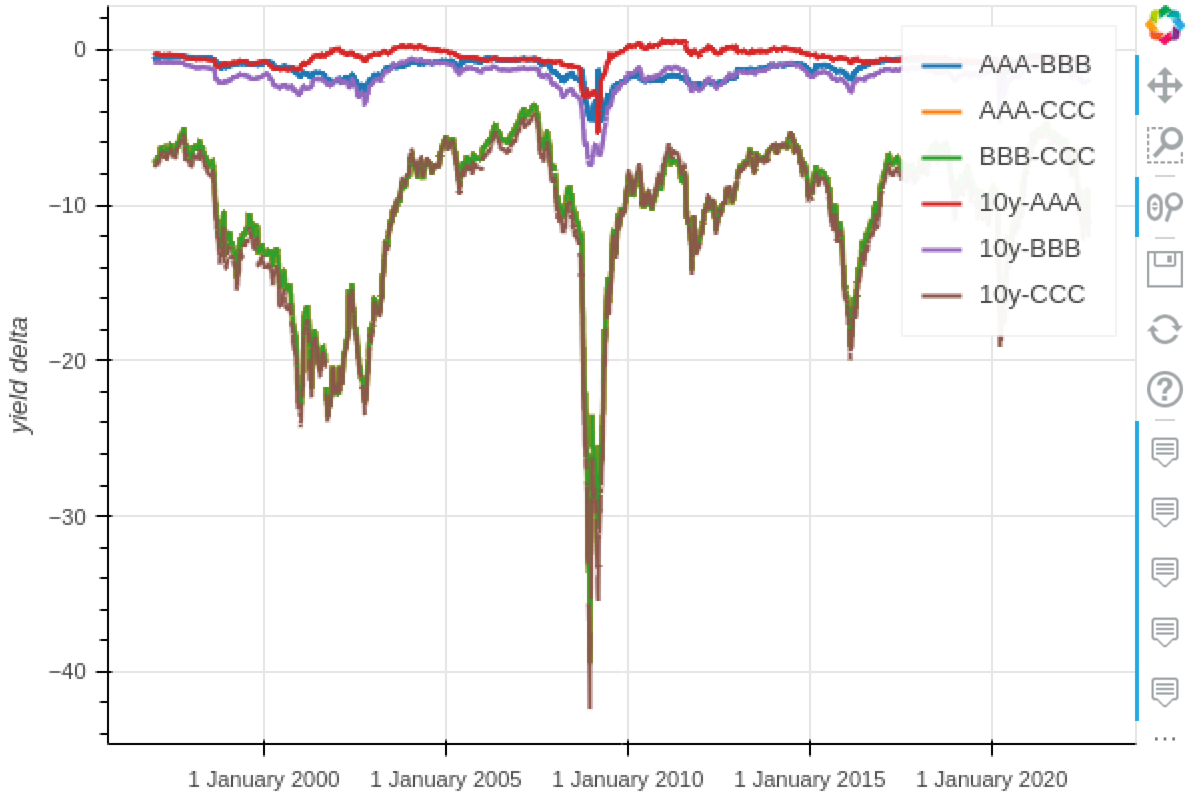
Out[24]: **Figure**(id = '3174', …)

In [25]:

```
# add spreads
out_join_spreads=pd.DataFrame()
out_join_spreads['AAA-BBB']=out_join_yields['AAA']-out_join_yields['BBB']
out_join_spreads['AAA-CCC']=out_join_yields['AAA']-out_join_yields['CCC']
out_join_spreads['BBB-CCC']=out_join_yields['AAA']-out_join_yields['CCC']
out_join_spreads['10y-AAA']=out_join_yields['10y']-out_join_yields['AAA']
out_join_spreads['10y-BBB']=out_join_yields['10y']-out_join_yields['BBB']
```

```
out_join_spreads['10y-CCC']=out_join_yields['10y']-out_join_yields['CCC']
#out_join_yields['AAA-BBB','AAA-CCC','BBB-CCC','10y-AAA','10y-BBB','10y-CCC'].plot(kind=
```

In [26]:
```
out_join_spreads.plot(kind='line',ylabel='yield delta')
```



Out[26]: **Figure**(id = '3652', …)

In [27]:
```
out_join_spreads.tail(30)
```

Out[27]:

	AAA-BBB	AAA-CCC	BBB-CCC	10y-AAA	10y-BBB	10y-CCC
2022-07-28	-1.26	-10.50	-10.50	-0.84	-2.10	-11.34
2022-07-29	-1.26	-10.20	-10.20	-0.80	-2.06	-11.00
2022-07-31	-1.26	-10.24	-10.24	NaN	NaN	NaN
2022-08-01	-1.24	-10.22	-10.22	-0.86	-2.10	-11.08
2022-08-02	-1.23	-9.94	-9.94	-0.85	-2.08	-10.79
2022-08-03	-1.23	-9.77	-9.77	-0.87	-2.10	-10.64
2022-08-04	-1.21	-9.65	-9.65	-0.87	-2.08	-10.52
2022-08-05	-1.20	-9.55	-9.55	-0.88	-2.08	-10.43
2022-08-08	-1.21	-9.46	-9.46	-0.87	-2.08	-10.33
2022-08-09	-1.22	-9.47	-9.47	-0.88	-2.10	-10.35
2022-08-10	-1.19	-9.25	-9.25	-0.87	-2.06	-10.12
2022-08-11	-1.17	-9.07	-9.07	-0.84	-2.01	-9.91
2022-08-12	-1.16	-9.10	-9.10	-0.84	-2.00	-9.94
2022-08-15	-1.13	-8.93	-8.93	-0.86	-1.99	-9.79
2022-08-16	-1.13	-8.94	-8.94	-0.85	-1.98	-9.79
2022-08-17	-1.14	-9.04	-9.04	-0.85	-1.99	-9.89
2022-08-18	-1.14	-9.07	-9.07	-0.84	-1.98	-9.91

2022-08-19	-1.14	-9.15	-9.15	-0.84	-1.98	-9.99
2022-08-22	-1.17	-9.40	-9.40	-0.84	-2.01	-10.24
2022-08-23	-1.16	-9.45	-9.45	-0.83	-1.99	-10.28
2022-08-24	-1.15	-9.34	-9.34	-0.82	-1.97	-10.16
2022-08-25	-1.15	-9.37	-9.37	-0.83	-1.98	-10.20
2022-08-26	-1.15	-9.44	-9.44	-0.83	-1.98	-10.27
2022-08-29	-1.17	-9.68	-9.68	-0.82	-1.99	-10.50
2022-08-30	-1.19	-9.82	-9.82	-0.83	-2.02	-10.65
2022-08-31	-1.18	-11.06	-11.06	-0.85	-2.03	-11.91
2022-09-01	-1.21	-11.19	-11.19	-0.86	-2.07	-12.05
2022-09-02	-1.19	-11.15	-11.15	-0.85	-2.04	-12.00
2022-09-05	-1.19	-11.15	-11.15	NaN	NaN	NaN
2022-09-06	-1.19	-11.05	-11.05	-0.87	-2.06	-11.92

M2 Money Supply

In [28]:

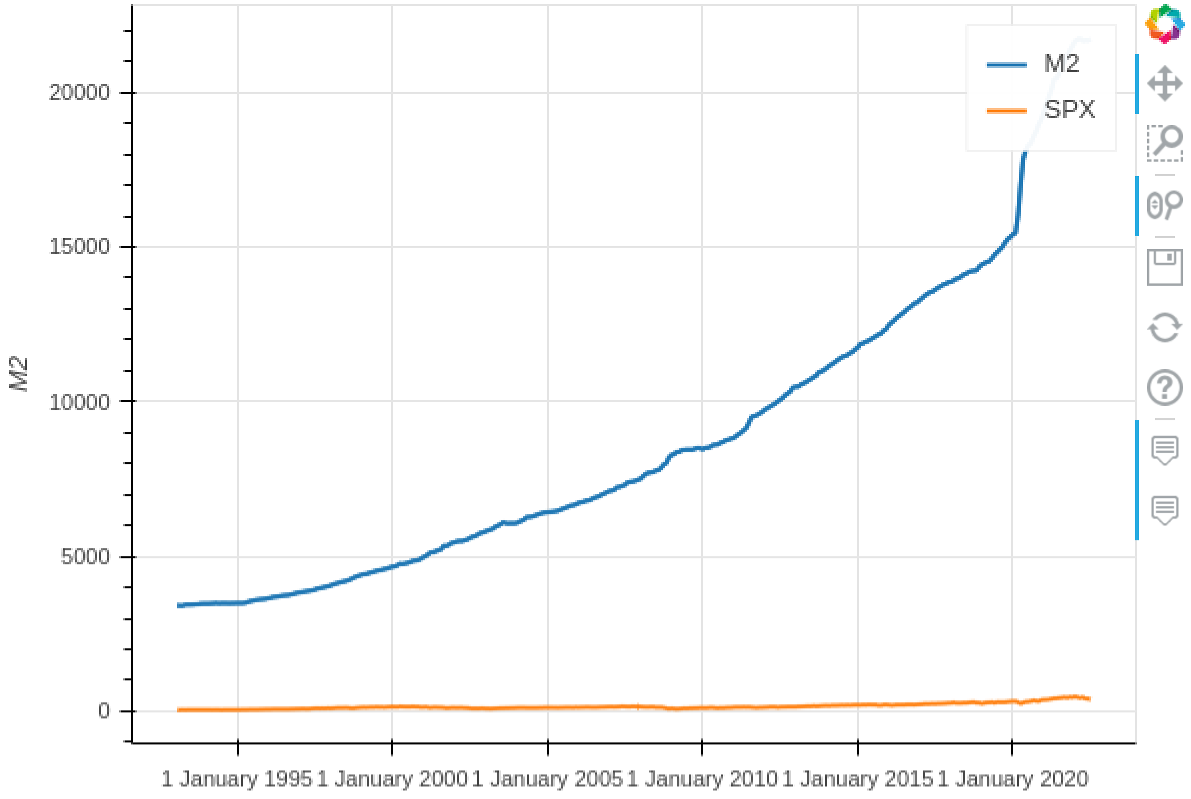
```
m2_df=pd.DataFrame(m2)

out_join_m2=m2_df.join(data_spx_total['Close'],how="outer") #we outer join here, else we

out_join_m2.columns=['M2','SPX']

# we fill up the daily closes vs the monthly data
out_join_m2['SPX']=out_join_m2['SPX'].fillna(method='ffill')

# we remove the datapoints where we don't have a M2 datapoint
out_join_m2=out_join_m2.dropna()
# plotting
out_join_m2.plot(kind='line',ylabel='M2')
```



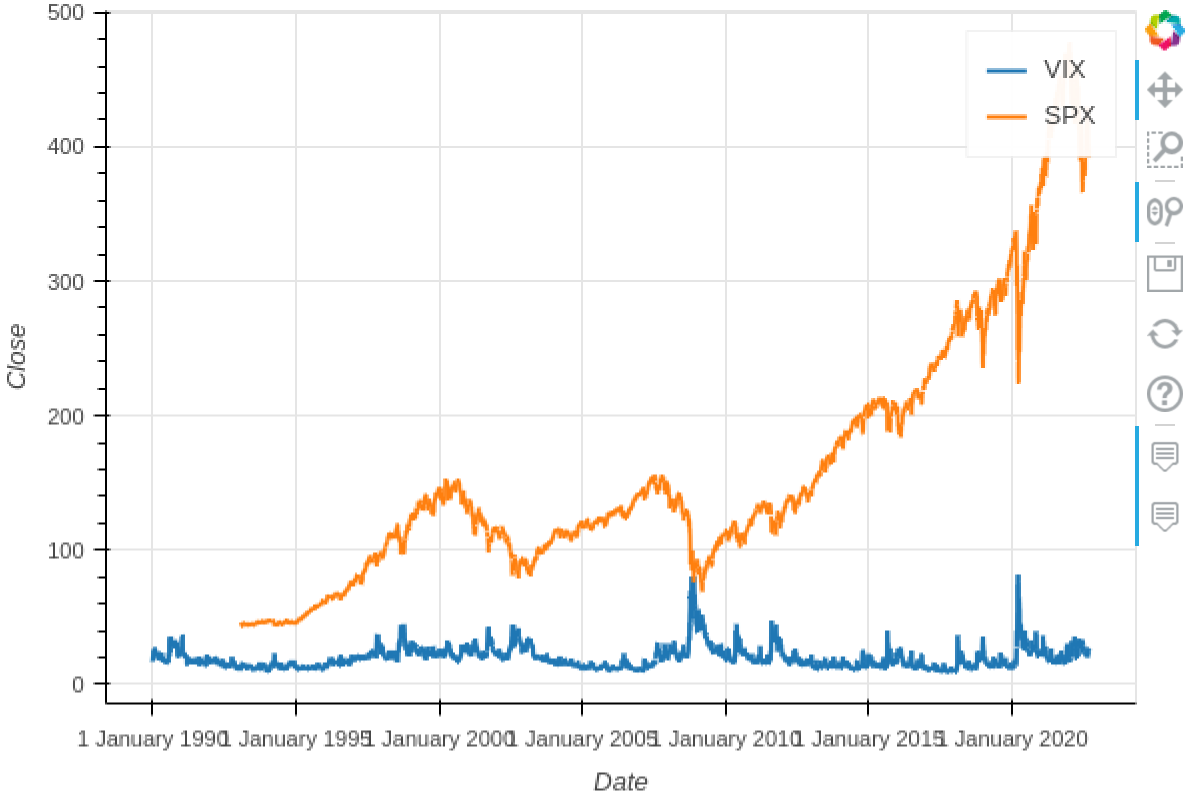
Out[28]: **Figure**(id = '4263', …)

In [ ]:

```
```

In [29]:

```
# Get VIX data - compare to spx
vix=data_vix_total['Close']
vix_df=pd.DataFrame(vix)
vix_df.columns=['VIX']
vix_df.tail()
output_data_vix_total_df=vix_df.join(data_spx_total['Close'],how="outer") #we outer join
output_data_vix_total_df.columns=['VIX','SPX']
output_data_vix_total_df.tail()
output_data_vix_total_df.plot(kind='line',ylabel='Close')
```



Out[29]: **Figure**(id = '4701', …)

Surveys

ISM Manufactoring PMI©

In [30]:

```
# here comes the ism - we need to add manually the data, api requests will only return t
output_ism_pmi=ism_pmi
#ism_pmi_all[datetime.datetime(2000,1,10):]

output_ism_pmi.tail()
```

Out[30]:

	VALUE
DATE	
2016-01-01	48.2
2016-02-01	49.5
2016-03-01	51.8
2016-04-01	50.8
2016-05-01	51.3

In [31]:

```
# here comes the ism - we need to add manually the data, api requests will only return t
output_ism_nmi=ism_nmi
#ism_pmi_all[datetime.datetime(2000,1,10):]

output_ism_nmi.tail()
```

Out[31]:

	VALUE
DATE	
2016-01-01	53.5
2016-02-01	53.4
2016-03-01	54.5
2016-04-01	55.7
2016-05-01	52.9

UMCSI

In [32]:

```
# UMCSI direct data links to csv's
tbmics = pd.read_csv("http://www.sca.isr.umich.edu/files/tbmics.csv")
tbmiccice = pd.read_csv("http://www.sca.isr.umich.edu/files/tbmiccice.csv")
tbmpx1px5 = pd.read_csv("http://www.sca.isr.umich.edu/files/tbmpx1px5.csv")
```

In [33]:

```
tbmics.tail()
```

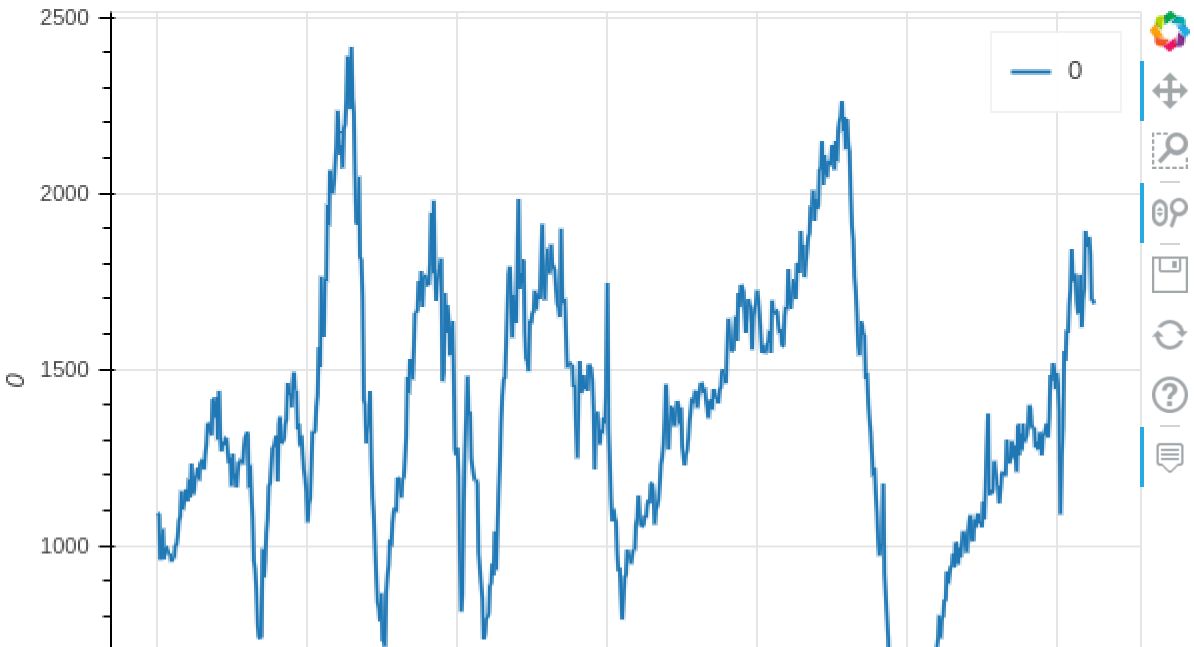
Out[33]:

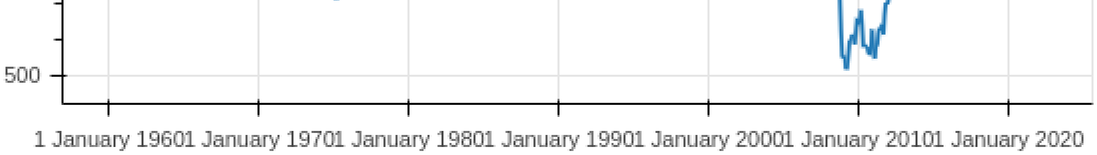
	Month	YYYY	ICS_ALL
623	April	2022	65.2
624	May	2022	58.4
625	June	2022	50.0
626	July	2022	51.5
627	August	2022	58.2

Building Permits

In [34]:

```
building_permits.plot(kind='line')
```





```
Out[34]: Figure(id = '5162', …)
```

.... to be continued

```
In [ ]:
```

```
In [ ]:
```