In [2]:
```
import pandas as pd
import matplotlib.pyplot as plt
import numpy as np
```

In [3]:
```
# Load the Excel file
excel_file = pd.ExcelFile('D:\Derivatives Trading\TAIEX derivatives trading record

# Get the sheet you want to read
sheet_name = 'ForPython' # Replace with the name of the sheet you want to read
df = excel_file.parse(sheet_name)
```

In [4]:
```
# Output data information
print(df.head())
```

```
        Date   PnL Index     TAIEX    VIX   Returns  Unnamed: 5  Unnamed: 6  \
0 2022-07-01  100.000000  14343.08  27.01  0.000000         NaN         NaN
1 2022-07-04   95.577858  14217.06  27.56 -0.044221         NaN         NaN
2 2022-07-05   93.953178  14349.20  27.18 -0.016998         NaN         NaN
3 2022-07-06   92.057052  13985.51  29.40 -0.020182         NaN         NaN
4 2022-07-07   92.698962  14335.27  28.26  0.006973         NaN         NaN

    Base
0  100.0
1    NaN
2    NaN
3    NaN
4    NaN
```

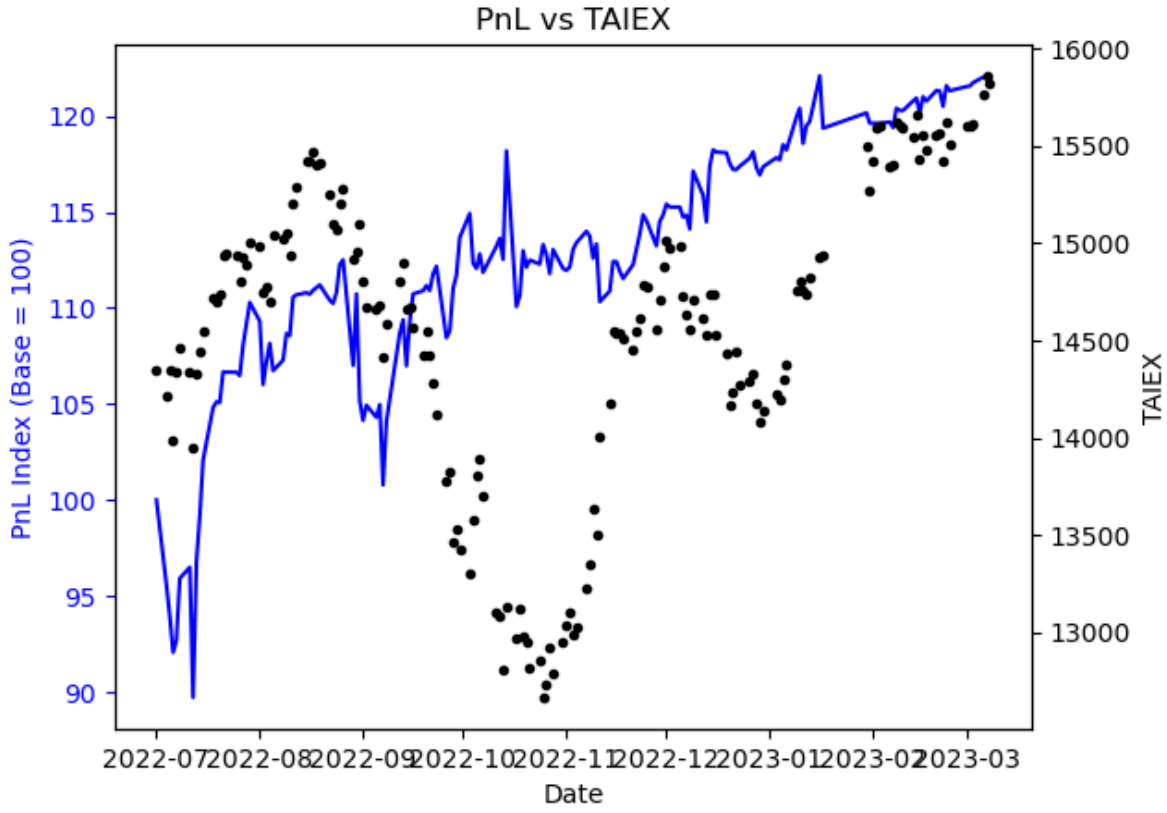
In [5]:
```
#******Plotting setup*****#
# Generate some data
Date = df["Date"]
Date
y1 =df["PnL Index"]
y1
y2 = df["TAIEX"]
y2
```

Out[5]:
```
0      14343.08
1      14217.06
2      14349.20
3      13985.51
4      14335.27
         ...
161    15598.72
162    15608.42
163    15763.51
164    15857.89
165    15818.20
Name: TAIEX, Length: 166, dtype: float64
```

In [6]:
```
# Create the plot and set the first y-axis (left)
fig, ax1 = plt.subplots()
ax1.plot(Date, y1, 'b-')
ax1.set_xlabel('Date')
ax1.set_ylabel('PnL Index (Base = 100)', color='b')
ax1.tick_params('y', colors='b')

# Set the second y-axis (right)
ax2 = ax1.twinx()
ax2.plot(Date, y2, 'k.')
ax2.set_ylabel('TAIEX', color='k')
ax2.tick_params('y', colors='k')
```

```
# Show the plot
plt.title('PnL vs TAIEX')
plt.show()
```

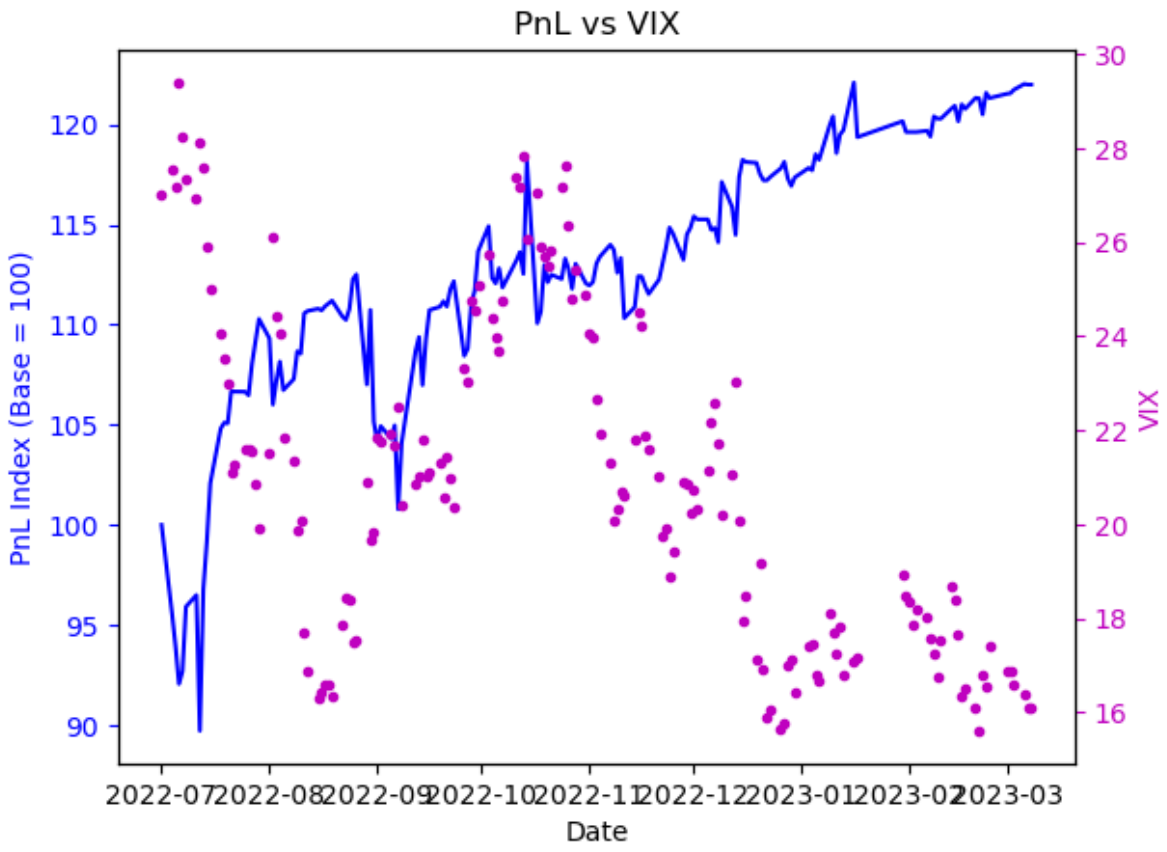


In [7]:
```
#Pnl vs VIX
y3 = df["VIX"]
y3

# Create the plot and set the first y-axis (left)
fig, ax1 = plt.subplots()
ax1.plot(Date, y1, 'b-')
ax1.set_xlabel('Date')
ax1.set_ylabel('PnL Index (Base = 100)', color='b')
ax1.tick_params('y', colors='b')

# Set the second y-axis (right)
ax3 = ax1.twinx()
ax3.plot(Date, y3, 'm.')
ax3.set_ylabel('VIX', color='m')
ax3.tick_params('y', colors='m')

# Show the plot
plt.title('PnL vs VIX')
plt.show()
```



In [8]:
```
#Sharpe ratio
# Read in the portfolio returns data from a CSV file
R_first=df["PnL Index"].iloc[0,]
R_first
R_last=df["PnL Index"].iloc[165,]  #Always excel's actual row-2
R_last
```

Out[8]:
```
121.98400800102736
```

In [10]:
```
portfolio_returns=(R_last-R_first)/R_first
portfolio_returns
```

Out[10]:
```
0.21984008001027364
```

In [11]:
```
daily_returns=df["Returns"]
daily_returns
```

Out[11]:
```
0      0.000000
1     -0.044221
2     -0.016998
3     -0.020182
4      0.006973
         ...
161    0.000206
162    0.001462
163    0.002357
164   -0.000266
165    0.000000
Name: Returns, Length: 166, dtype: float64
```

In [14]:
```
# Calculate the excess returns and standard deviation
risk_free_rate = 0.0145  # Taiwan savings rate
excess_returns = portfolio_returns - risk_free_rate
std_dev = np.std(daily_returns)
print("Standard Deviation of Daily Return:", std_dev)
```

```
Standard Deviation of Daily Return: 0.017094435505975857
```

In [13]:
```python
# Calculate the Sharpe ratio
sharpe_ratio = excess_returns.mean() / std_dev

print("Sharpe Ratio:", sharpe_ratio)
```

```
Sharpe Ratio: 12.012100659216914
```

In [ ]:
```python

```