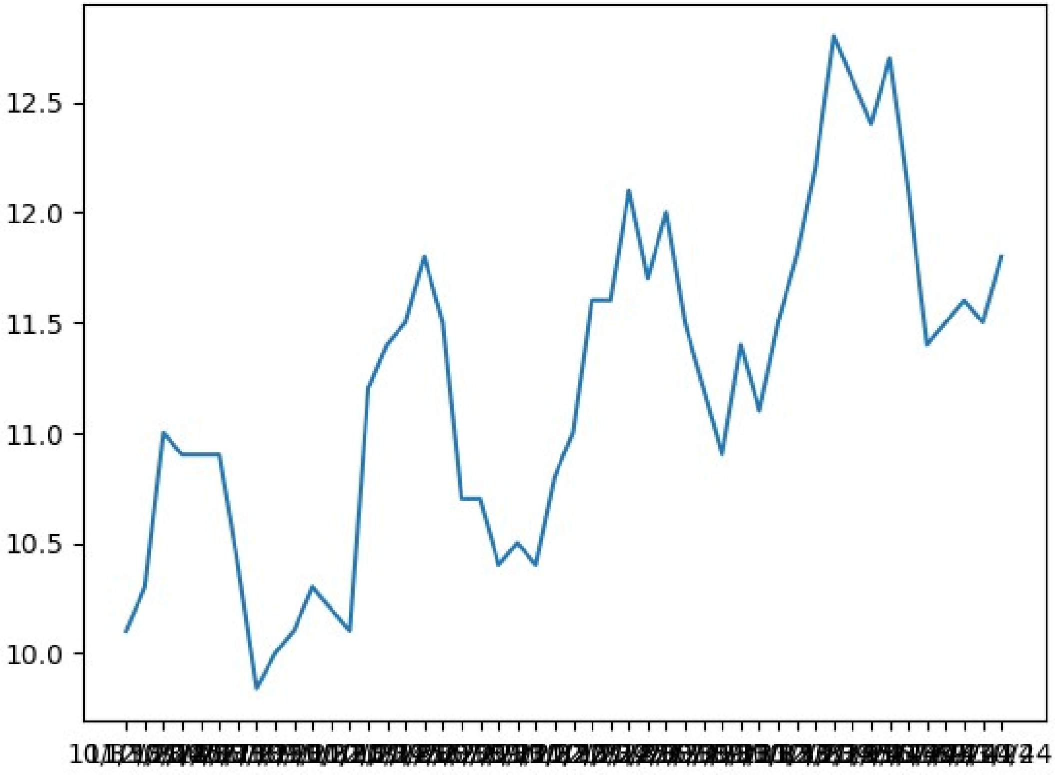
```python
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import datetime as dt
from sklearn.linear_model import LinearRegression
from datetime import datetime, timedelta

natgas_df = pd.read_csv('Nat_Gas.csv')
natgas_df
```

```
       Dates  Prices
0   10/31/20   10.10
1   11/30/20   10.30
2   12/31/20   11.00
3    1/31/21   10.90
4    2/28/21   10.90
5    3/31/21   10.90
6    4/30/21   10.40
7    5/31/21    9.84
8    6/30/21   10.00
9    7/31/21   10.10
10   8/31/21   10.30
11   9/30/21   10.20
12  10/31/21   10.10
13  11/30/21   11.20
14  12/31/21   11.40
15   1/31/22   11.50
16   2/28/22   11.80
17   3/31/22   11.50
18   4/30/22   10.70
19   5/31/22   10.70
20   6/30/22   10.40
21   7/31/22   10.50
22   8/31/22   10.40
23   9/30/22   10.80
24  10/31/22   11.00
25  11/30/22   11.60
26  12/31/22   11.60
27   1/31/23   12.10
28   2/28/23   11.70
29   3/31/23   12.00
30   4/30/23   11.50
31   5/31/23   11.20
32   6/30/23   10.90
33   7/31/23   11.40
34   8/31/23   11.10
35   9/30/23   11.50
36  10/31/23   11.80
37  11/30/23   12.20
38  12/31/23   12.80
```

```
39   1/31/24   12.60
40   2/29/24   12.40
41   3/31/24   12.70
42   4/30/24   12.10
43   5/31/24   11.40
44   6/30/24   11.50
45   7/31/24   11.60
46   8/31/24   11.50
47   9/30/24   11.80
```

```python
plt.plot(natgas_df['Dates'],natgas_df['Prices'])
```

```
[<matplotlib.lines.Line2D at 0x1bdf5fe98e0>]
```



```python
#Using only the max of each year, we have a linear method
natgas_df['Dates'] = pd.to_datetime(natgas_df['Dates'],
format="%m/%d/%y")

natgas_df['Year'] = natgas_df['Dates'].dt.year
natgas_df['Month'] = natgas_df['Dates'].dt.month

natgas_jan = natgas_df[natgas_df['Month'] == 1]
natgas_jan
```

```
        Dates  Prices  Year  Month
3  2021-01-31    10.9  2021      1
15 2022-01-31    11.5  2022      1
27 2023-01-31    12.1  2023      1
39 2024-01-31    12.6  2024      1
```

```
from sklearn.linear_model import LinearRegression

X = np.array(natgas_df[natgas_df['Month'] == 1]['Year'] ).reshape(-
1,1)
Y = np.array(natgas_df[natgas_df['Month'] == 1]['Prices'])

reg = LinearRegression().fit(X,Y)

round(reg.predict([[2025]])[0], 2)
```

```
13.2
```

```
#extra year extrapolation

from sklearn.linear_model import LinearRegression

def next_year_price(next_year):
    price_list = []
    for i in np.arange(12):
        X = np.array(natgas_df[natgas_df['Month'] == 1+i]
['Year'] ).reshape(-1,1)
        Y = np.array(natgas_df[natgas_df['Month'] == 1+i]['Prices'])
        reg = LinearRegression().fit(X,Y)
        price = reg.predict([[next_year]])
        price_list.append(round(price[0], 2))

    return price_list

gas_price25 = next_year_price(2025)
np.array(gas_price25)
```

```
array([13.2 , 12.8 , 13.25, 12.65, 12.08, 11.95, 12.25, 11.9 , 12.45,
       12.85, 13.46, 13.66])
```

```
def get_last_of_each_month(year):
    dates_array = []
    current_date = datetime(year, 12, 31)
    while current_date.year == year:

        dates_array.append(current_date.strftime('%Y-%m-%d'))
        month = current_date.month
        year = current_date.year

        #Take the first day of the previous month
        current_date = current_date.replace(year=year, month=month,
day=1)

        #Move back one day to get the last day of the current month
        current_date -= timedelta(days=1)

    return dates_array[::-1]

dates_2025 = get_last_of_each_month(2025)
dates_2025
```

```
['2025-01-31',
 '2025-02-28',
 '2025-03-31',
 '2025-04-30',
 '2025-05-31',
 '2025-06-30',
 '2025-07-31',
 '2025-08-31',
 '2025-09-30',
 '2025-10-31',
 '2025-11-30',
 '2025-12-31']
```

```
#2025 data (dates & prices)
projected_gas_prices25_df = pd.DataFrame({'Dates': dates_2025,
'Prices' : gas_price25})
projected_gas_prices25_df

projected_gas_prices25_df['Dates'] =
pd.to_datetime(projected_gas_prices25_df['Dates'])

projected_gas_prices25_df['Year'] =
projected_gas_prices25_df['Dates'].dt.year
projected_gas_prices25_df['Month'] =
projected_gas_prices25_df['Dates'].dt.month
projected_gas_prices25_df
```

```
        Dates  Prices  Year  Month
0  2025-01-31   13.20  2025      1
1  2025-02-28   12.80  2025      2
2  2025-03-31   13.25  2025      3
3  2025-04-30   12.65  2025      4
4  2025-05-31   12.08  2025      5
5  2025-06-30   11.95  2025      6
6  2025-07-31   12.25  2025      7
7  2025-08-31   11.90  2025      8
8  2025-09-30   12.45  2025      9
9  2025-10-31   12.85  2025     10
10 2025-11-30   13.46  2025     11
11 2025-12-31   13.66  2025     12
```

```
gas_df = pd.concat([natgas_df, projected_gas_prices25_df],
ignore_index=True)
gas_df
```

```
        Dates  Prices  Year  Month
0  2020-10-31   10.10  2020     10
1  2020-11-30   10.30  2020     11
2  2020-12-31   11.00  2020     12
3  2021-01-31   10.90  2021      1
4  2021-02-28   10.90  2021      2
5  2021-03-31   10.90  2021      3
6  2021-04-30   10.40  2021      4
7  2021-05-31    9.84  2021      5
8  2021-06-30   10.00  2021      6
9  2021-07-31   10.10  2021      7
10 2021-08-31   10.30  2021      8
11 2021-09-30   10.20  2021      9
12 2021-10-31   10.10  2021     10
13 2021-11-30   11.20  2021     11
14 2021-12-31   11.40  2021     12
15 2022-01-31   11.50  2022      1
16 2022-02-28   11.80  2022      2
17 2022-03-31   11.50  2022      3
18 2022-04-30   10.70  2022      4
19 2022-05-31   10.70  2022      5
20 2022-06-30   10.40  2022      6
21 2022-07-31   10.50  2022      7
22 2022-08-31   10.40  2022      8
23 2022-09-30   10.80  2022      9
24 2022-10-31   11.00  2022     10
25 2022-11-30   11.60  2022     11
26 2022-12-31   11.60  2022     12
27 2023-01-31   12.10  2023      1
28 2023-02-28   11.70  2023      2
29 2023-03-31   12.00  2023      3
30 2023-04-30   11.50  2023      4
31 2023-05-31   11.20  2023      5
32 2023-06-30   10.90  2023      6
33 2023-07-31   11.40  2023      7
34 2023-08-31   11.10  2023      8
35 2023-09-30   11.50  2023      9
36 2023-10-31   11.80  2023     10
37 2023-11-30   12.20  2023     11
38 2023-12-31   12.80  2023     12
39 2024-01-31   12.60  2024      1
40 2024-02-29   12.40  2024      2
41 2024-03-31   12.70  2024      3
42 2024-04-30   12.10  2024      4
43 2024-05-31   11.40  2024      5
44 2024-06-30   11.50  2024      6
```

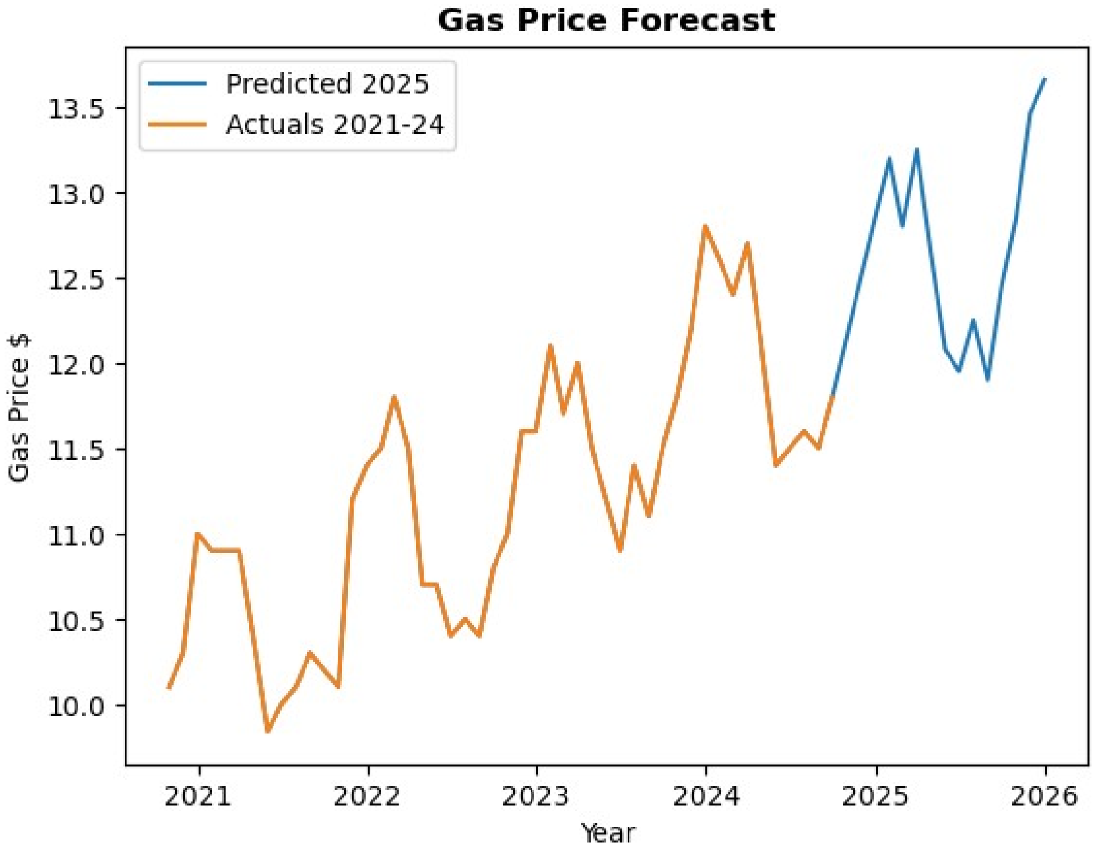
```
45 2024-07-31   11.60  2024      7
46 2024-08-31   11.50  2024      8
47 2024-09-30   11.80  2024      9
48 2025-01-31   13.20  2025      1
49 2025-02-28   12.80  2025      2
50 2025-03-31   13.25  2025      3
51 2025-04-30   12.65  2025      4
52 2025-05-31   12.08  2025      5
53 2025-06-30   11.95  2025      6
54 2025-07-31   12.25  2025      7
55 2025-08-31   11.90  2025      8
56 2025-09-30   12.45  2025      9
57 2025-10-31   12.85  2025     10
58 2025-11-30   13.46  2025     11
59 2025-12-31   13.66  2025     12
```

```
def get_gas_price(month, year):
    #to get predicted or historic gas price
    print(gas_df[(gas_df['Year'] == year) & (gas_df['Month'] ==
month)]['Prices'])

get_gas_price(10,2025)
```

```
57    12.85
Name: Prices, dtype: float64
```

```
#Analysis
plt.plot(gas_df['Dates'],gas_df['Prices'],label = 'Predicted 2025')
plt.plot(natgas_df['Dates'],natgas_df['Prices'], label = 'Actuals
2021-24')
plt.ylabel('Gas Price $')
plt.xlabel('Year')
plt.title('Gas Price Forecast', fontweight = 'bold')
plt.legend()
```

```
<matplotlib.legend.Legend at 0x1bdf6107080>
```


Gas Price Forecast
Predicted 2025
Actuals 2021-24
Gas Price $
13.5
13.0
12.5
12.0
11.5
11.0
10.5
10.0
Year
2021
2022
2023
2024
2025
2026