# 1.) Pull in Data and Convert ot Monthly

In [1]:
```
import yfinance as yf
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
```

In [2]:
```
apple_data = yf.download('AAPL')
df = apple_data.resample("M").last()[["Adj Close"]]
```

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

# 2.) Create columns.

- Current Stock Price, Difference in stock price, Whether it went up or down over the next month, option premium

In [3]:
```
df

# difference in stockprice
df["Diff"] = df["Adj Close"].diff().shift(-1)

# Target up or down
df["Target"] = np.sign(df["Diff"])

# Option Premium
df["Premium"] = 0.08 * df["Adj Close"]
```

In [4]:
```
df.head()
```

Out[4]:

| Date | Adj Close | Diff | Target | Premium |
|---|---|---|---|---|
| **1980-12-31** | 0.117887 | -0.020296 | -1.0 | 0.009431 |
| **1981-01-31** | 0.097591 | -0.006045 | -1.0 | 0.007807 |
| **1981-02-28** | 0.091546 | -0.006909 | -1.0 | 0.007324 |
| **1981-03-31** | 0.084637 | 0.013386 | 1.0 | 0.006771 |
| **1981-04-30** | 0.098023 | 0.016409 | 1.0 | 0.007842 |

# 3.) Pull in X data, normalize and build a LogReg on column 2

In [5]:
```
import numpy as np
import pandas as pd
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LogisticRegression
from sklearn import metrics
```

In [6]:
```
X = pd.read_csv("Xdata.csv", index_col="Date", parse_dates=["Date"])
```

In [7]:
```
y = df.loc[:"2023-09-30","Target"].copy()

df = df.loc[:"2023-09-30",:].copy()
```

In [8]:
```
logreg = LogisticRegression()

logreg.fit(X,y)

y_pred = logreg.predict(X)
```

In [9]:
```
df
```

Out[9]:

| Date | Adj Close | Diff | Target | Premium |
|---|---|---|---|---|
| **1980-12-31** | 0.117887 | -0.020296 | -1.0 | 0.009431 |
| **1981-01-31** | 0.097591 | -0.006045 | -1.0 | 0.007807 |
| **1981-02-28** | 0.091546 | -0.006909 | -1.0 | 0.007324 |
| **1981-03-31** | 0.084637 | 0.013386 | 1.0 | 0.006771 |
| **1981-04-30** | 0.098023 | 0.016409 | 1.0 | 0.007842 |
| **...** | ... | ... | ... | ... |
| **2023-05-31** | 176.778061 | 16.675507 | 1.0 | 14.142245 |
| **2023-06-30** | 193.453568 | 2.473389 | 1.0 | 15.476285 |
| **2023-07-31** | 195.926956 | -8.304138 | -1.0 | 15.674156 |
| **2023-08-31** | 187.622818 | -16.638077 | -1.0 | 15.009825 |
| **2023-09-30** | 170.984741 | -0.439423 | -1.0 | 13.678779 |

514 rows × 4 columns

# 4.) Add columns, prediction and profits.

In [10]:
```
df["Predictions"] = y_pred
```

In [17]:
```
df["Profits"] = 0

# True Positives
df.loc[(df["Predictions"] == 1) & (df["Target"] == 1),"Profits"] = df["Pr

# False Positives
df.loc[(df["Predictions"] == 1) & (df["Target"] == -1),"Profits"] = 100 *

# True Negatives
df.loc[(df["Predictions"] == -1) & (df["Target"] == 1),"Profits"] = 100 *

# False Negatives
df.loc[(df["Predictions"] == -1) & (df["Target"] == -1),"Profits"] = df["
```

In [18]:
```
df
```

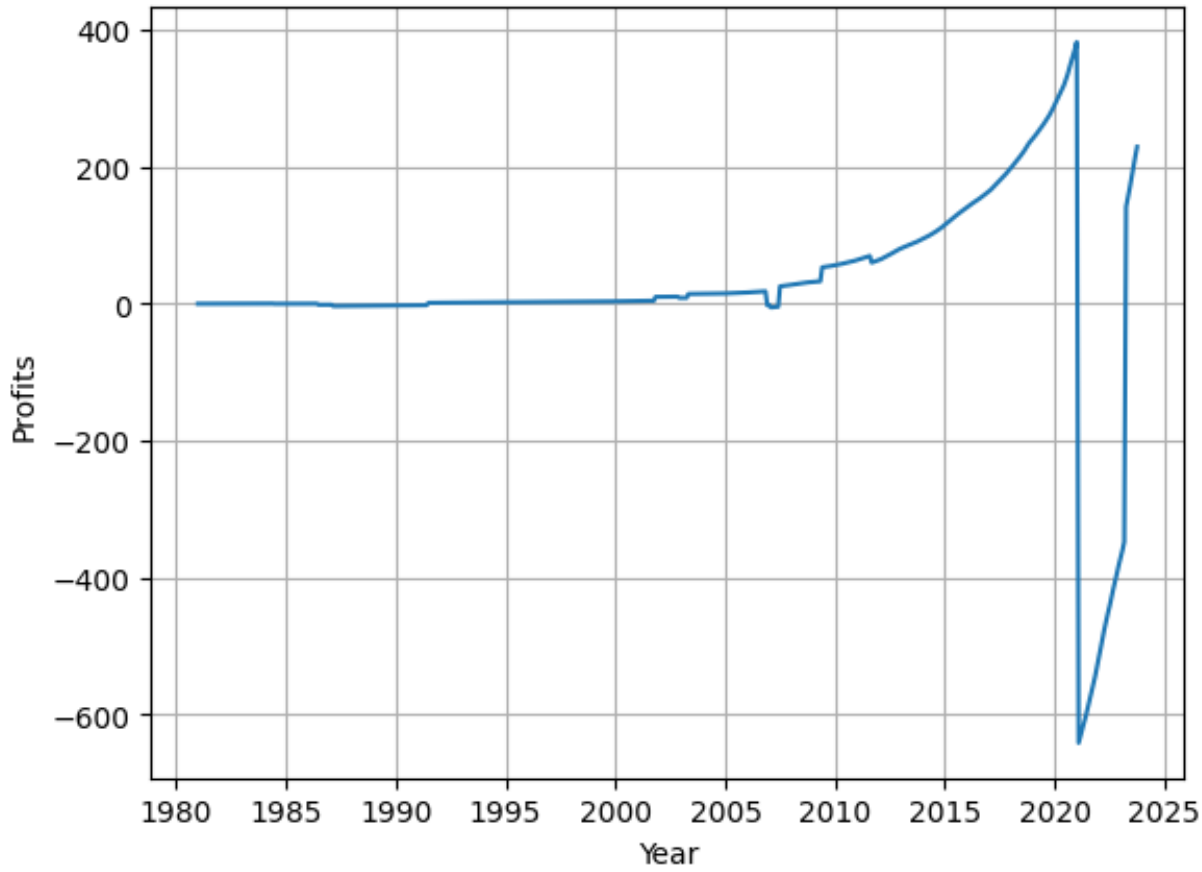
Out[18]:

| Date | Adj Close | Diff | Target | Premium | Predictions | Profits |
|---|---|---|---|---|---|---|
| **1980-12-31** | 0.117887 | -0.020296 | -1.0 | 0.009431 | -1.0 | 0.009431 |
| **1981-01-31** | 0.097591 | -0.006045 | -1.0 | 0.007807 | -1.0 | 0.007807 |
| **1981-02-28** | 0.091546 | -0.006909 | -1.0 | 0.007324 | -1.0 | 0.007324 |
| **1981-03-31** | 0.084637 | 0.013386 | 1.0 | 0.006771 | 1.0 | 0.006771 |
| **1981-04-30** | 0.098023 | 0.016409 | 1.0 | 0.007842 | 1.0 | 0.007842 |
| **...** | ... | ... | ... | ... | ... | ... |
| **2023-05-31** | 176.778061 | 16.675507 | 1.0 | 14.142245 | 1.0 | 14.142245 |
| **2023-06-30** | 193.453568 | 2.473389 | 1.0 | 15.476285 | 1.0 | 15.476285 |
| **2023-07-31** | 195.926956 | -8.304138 | -1.0 | 15.674156 | -1.0 | 15.674156 |
| **2023-08-31** | 187.622818 | -16.638077 | -1.0 | 15.009825 | -1.0 | 15.009825 |
| **2023-09-30** | 170.984741 | -0.439423 | -1.0 | 13.678779 | -1.0 | 13.678779 |

514 rows × 6 columns

# 5.) Plot profits over time

In [22]:
```
plt.plot(np.cumsum(df["Profits"]))
plt.xlabel("Year")
plt.ylabel("Profits")
plt.grid()
plt.show()
```




## 5.5 your skills from the mqe to help Mr.Lui's Ventures?

My capabilities include comprehensive data analysis, which encompasses the collection, processing, and interpretation of complex datasets. Moreover, I possess the proficiency to comprehend and convey the outcomes of my analyses, ensuring that I can provide clear explanations and actionable insights to clients based on the data-driven evidence. This dual expertise in data handling and interpretation is crucial for making informed decisions in today's data-centric business environment.

# 6.) Create a loop that stores total profits over time

In [ ]:

# 7.) What is the optimal threshold and plot the total profits for this model.

In [ ]: