CEIS150

Course Project

By Glenn Delostrico

Table of Contents
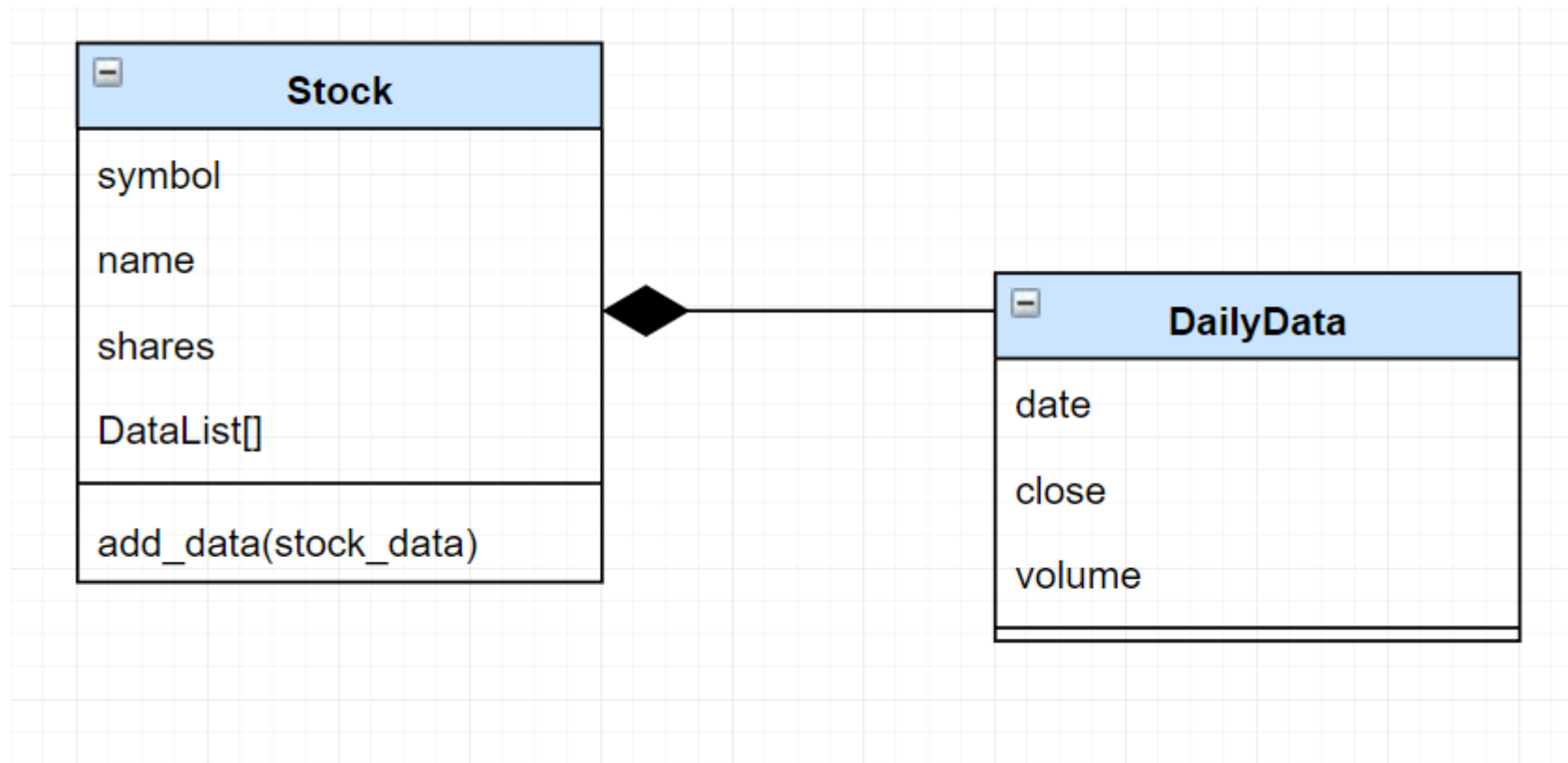
Creating the Classes

Class Diagrams

Class Coding

Unit Testing

Class Diagram

Paste your Visio Class Diagram

Class Code

Screen Shot of your stock_class.py file.

```python
class Stock:
    def __init__(self, symbol, name, shares):
        self.symbol = symbol
        self.name = name
        self.shares = shares
        self.DataList = [] # list of proce and volume history

    def add_data(self, stock_data):
        self.DataList.append(stock_data)

class DailyData:
    def __init__(self, date, close, volume):
        self.date = date
        self.close = close
        self.volume = volume
```

Unit Test

Screen Shot of your successful unit test.

```
Unit Testing Starting---
Testing Add Stock...Successful!
Test Change Symbol...Successful!
Test Change Name...Successful!
Successful!
Creating daily stock data...Successful!
Congratulations - All Tests Passed
Goodbye
```

Creating the Console Interface

Adding a Stock

Listing 3 Stocks

Daily Data

Adding a Stock

Paste a screen shot of your working Stock program.

```
Enter Menu Option: 1
Enter Ticker Symbol: amzn
Enter Company Name: Amazon
Enter Number of Shares: 8
Stock Added - Enter to Add Another Stock or 0 to Stop:
Enter Ticker Symbol: abnb
Enter Company Name: Airbnb
Enter Number of Shares: 3
Stock Added - Enter to Add Another Stock or 0 to Stop:
Enter Ticker Symbol: aapl
Enter Company Name: Apple
Enter Number of Shares: 4
Stock Added - Enter to Add Another Stock or 0 to Stop: 0
```

Listing 3 Stocks

Paste a screen shot of your working Stock program.

```
Enter Menu Option: 3
Stock List ----
Symbol              Name              Shares

=================================================
AMZN                Amazon            8
ABNB                Airbnb            3
AAPL                Apple             4


Press Enter to Continue ***
```

Daily Data

Paste a screen shot of your working Stock program.

```
Enter Menu Option: 4
Add Daily Stock Data ----
Stock List: [AMZN  ABNB  AAPL  ]
Which stock do you want to use?: amzn
Ready to add data for:  AMZN
Enter Data Separated by Commas - Do Not use Spaces
Enter a Blank Line to Quit
Enter Date,Price,Volume
Example: 8/28/20,47.85,10550
Enter Date,Price,Volume: 8/28/20,47.85,10550
Enter Date, Price, Volume:
Data Entry Complete
Stock Analyzer ---
```

Inheritance

Inherited Classes
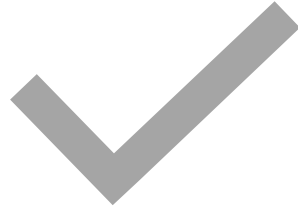
Unit Tests

Stock Menu

Inherited classes

Paste a screen shot of your classes

```python
class Retirement_Account:
    def __init__(self, balance, number):
        self.balance = balance
        self.number = number


class Traditional(Retirement_Account):
    def __init__(self, balance, number):
        Retirement_Account.__init__(self, balance, number)
        self.Stock_List = []

    def add_stock(self, stock_data):
        self.Stock_List.append(stock_data)


class Robo(Retirement_Account):
    def __init__(self, balance, number, years):
        Retirement_Account.__init__(self, balance, number)
        self.years = years

    def investment_return(self):
        return (self.years*self.balance*1.05)
```

Unit Tests

Paste a screen shot of your unit tests successfully completed

```
PS E:\My Drive\Education\DeVry\2022\Sem 4\CEIS 150 - Programmin with Objects\Coarse Project> & C:/
roject/account_class.py"
Unit Testing Starting---
Testing Add Retirement Account...Successful!
Testing Add Traditional Account...Successful!
Test Change Balance...Successful!
Unit Testing Starting---
Testing Add Retirement Account...Successful!
Testing Add Traditional Account...Successful!
Test Change Balance...Successful!
Test Change Number...Successful!
Testing Add Robo Account...Successful!
Test Change Balance...Successful!
Test investment return...Successful!
Congratulations - All Tests Passed
Goodbye
PS E:\My Drive\Education\DeVry\2022\Sem 4\CEIS 150 - Programmin with Objects\Coarse Project> 
```

Stock menu program

Paste a screen shot of your classes in the main program

```
Stock Analyzer ---
1 - Add Stock
2 - Delete Stock
3 - List stocks
4 - Add Daily Stock Data (Date, Price, Volume)
5 - Show Chart
6 - Investor Type
7 - Load Data
0 - Exit Program
Enter Menu Option: 6
Investment Account ---
What is your initial balance: 40
What is your account number: 1234
Do you want a Traditional (t) or Robo (r) account: t
Bought  40.0 of APL
Stock List: [APL  MSFT  ]
Which stock do you want to purchase, 0 to quit: 0
Stock Analyzer ---
1 - Add Stock
2 - Delete Stock
3 - List stocks
4 - Add Daily Stock Data (Date, Price, Volume)
5 - Show Chart
6 - Investor Type
7 - Load Data
0 - Exit Program
Enter Menu Option: 6
Investment Account ---
What is your initial balance: 30
What is your account number: 12345
Do you want a Traditional (t) or Robo (r) account: r
How many years until retirement: 20
Your investment return is  630.0
```

Creating a Chart
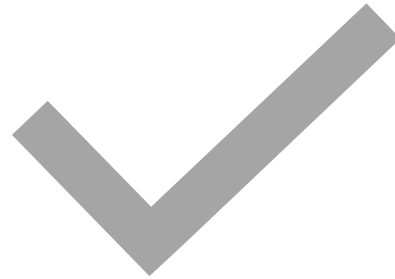
Chart

Chart

Paste a screen shot of your stock chart.
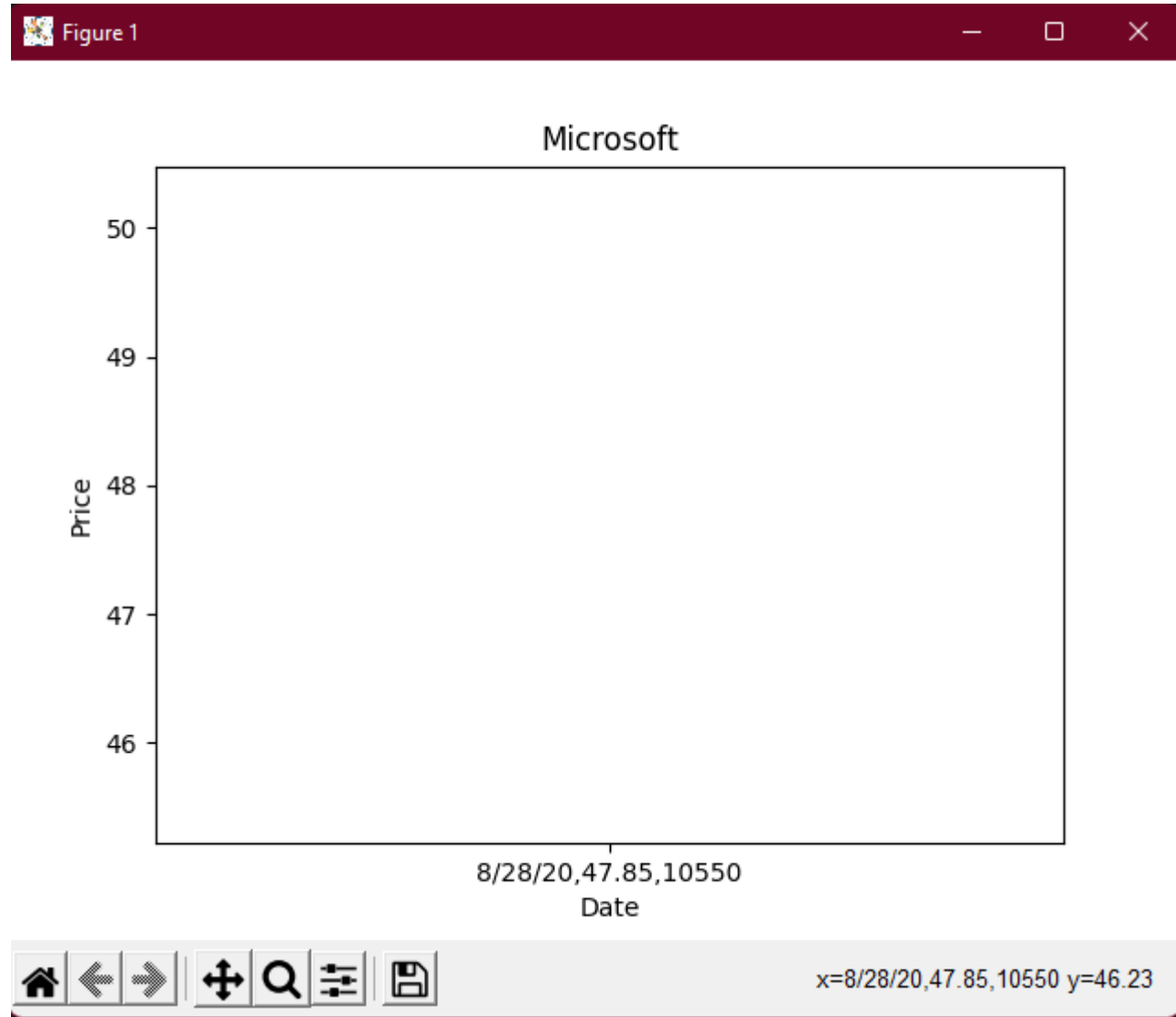
Inheritance

File

Importing Data

File

Paste a screen shot of the file downloaded from Yahoo finance

.git	8/9/2022 9:40 PM	File folder	
.vs	7/29/2022 6:08 PM	File folder	
__pycache__	8/1/2022 8:37 AM	File folder	
css	7/29/2022 5:54 PM	File folder	
.gitignore	7/18/2022 2:51 PM	Git Ignore Source ...	1 KB
AAPL	8/8/2022 3:16 PM	Microsoft Excel C...	19 KB
account_class	7/30/2022 12:10 PM	Python Source File	5 KB
CEIS150 Project Guide Module 3 v2	7/17/2022 8:39 PM	Microsoft Word D...	330 KB
CEIS150 Project Template Module Deliverable Week 3 v2	7/18/2022 2:41 PM	Microsoft PowerP...	87 KB
stock_class	7/30/2022 12:11 PM	Python Source File	4 KB
stock_menu	8/9/2022 9:41 PM	Python Source File	11 KB

Importing data

Screenshot of the historical data import

Creating a GUI

Stocks in GUI

History

Report

Stocks in GUI

Paste a screen shot of your GUI working

History Tab

Paste a screen shot of your History tab with import working.

aapl
msft

Manage **History** **Report**

| – Date –	– Price –	– Volume –
===
2021-08-09	$146.09	48908700.0
2021-08-10	$145.60	69023100.0
2021-08-11	$145.86	48493500.0
2021-08-12	$148.89	72282600.0
2021-08-13	$149.10	59375000.0
2021-08-16	$151.12	103296000.0
2021-08-17	$150.19	92229700.0
2021-08-18	$146.36	86326000.0
2021-08-19	$146.70	86960300.0
2021-08-20	$148.19	60549600.0
2021-08-23	$149.71	60131800.0
2021-08-24	$149.62	48606400.0
2021-08-25	$148.36	58991300.0
2021-08-26	$147.54	48597200.0
2021-08-27	$148.60	55802400.0
2021-08-30	$153.12	90956700.0
2021-08-31	$151.83	86453100.0
2021-09-01	$152.51	80313700.0
2021-09-02	$153.65	71115500.0
2021-09-03	$154.30	57808700.0
2021-09-07	$156.69	82278300.0

Report Complete

Paste a screen shot of your Report tab

aapl
msft

Manage History Report

```
Summary Data--

Low Price: $130.06
High Price: $182.01
Average Price: $156.94

Low Volume: 999999999999
High Volume: 195432700.0
Average Volume: $88,331,163.82
Profit/Loss: $-779.25
```

Career Skills Developed

Object Oriented
Programming with Python
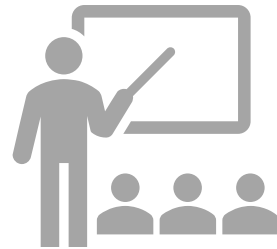
Classes and Inheritance

GUI Programming

Conclusion

This course builds on structured programming and introduces object-oriented and functional programming concepts. Students design, code, test and document business-oriented solutions using complex algorithms. Advanced topics include the use of libraries for data manipulation and visualization.