

# Homework1

October 26, 2021

## 0.1 Software for a Financial Institution to Model their Clients' Portfolio

```
[90]: import random

class Portfolio:

    def __init__(self):

        self.cash = 0
        self.stock_dict = {}
        self.fund_dict = {}
        self.History=[]
        self.History_stock=[]
        self.total_MF = {}
        self.History_MF=[]

    def __str__(self):
        return 'Current Portfolio: \n Net cash in (USD) : {self.cash} \n
↪ Stocks history:, {self.History_stock} \n Mutual Funds history: {self.
↪ History_MF}'.format(self=self)

    def addcash(self, amount):
        self.cash = self.cash + amount
        p=print('Total amount after cash deposit of {fname} '.format(fname =
↪ amount))
        self.History.append(self.cash )
        return self.cash

    def withdrawCash(self, amount):
        self.cash = self.cash - amount
        q= print('Total amount after cash withdrawal of {fname} '.format(fname =
↪ amount))
        self.History.append(self.cash )
        return self.cash

    def new_stock(self, price , symbol):
```

```

        self.stock_dict[symbol] = price
        # return self.stock_dict

def buy_stock(self, num, stk_symbol):

    for symbol, price in self.stock_dict.items():
        if symbol == stk_symbol:
            #newval= list(self.stock_dict.values())

            stockvalue = (num* price)
            self.cash = self.cash - stockvalue
            r= print('Total amount after purchasing {fname} stocks of {s}_'
→with price ${p} per stock ' .format(fname = num, s = stk_symbol ,p=price))
            self.History.append(self.cash )
            num+= num
            self.History_stock.append(num)
            self.History_stock.append(symbol)
            return self.cash

def sell_stock(self, num , stk_symbol):

    for symbol, price in self.stock_dict.items():
        if symbol == stk_symbol:
            #newval= list(self.stock_dict.values())

            selling_price= (random.uniform(0.5*price, 1.5*price)) ##_
→selling uniform RV BW[0.5X,1.5X]
            stockvalue = (num* selling_price)
            self.cash = self.cash + stockvalue
            s= print('Total amount after selling {fname} stocks of {s} ' ' .
→format(fname = num, s = stk_symbol))
            num+= num
            num= -1*num
            self.History_stock.append(num)
            self.History_stock.append(symbol)
            self.History.append(self.cash )
            return self.cash

def new_MFUND(self, price , symbol):
    self.fund_dict[symbol] = price
    #return self.fund_dict

```

```

def buy_MFUND(self, num, stk_symbol):

    for symbol, price in self.fund_dict.items():
        if symbol == stk_symbol:
            #newval= list(self.stock_dict.values())

            stockvalue = (num* price)
            self.cash = self.cash - stockvalue
            t= print('Total amount after purchasing {fname} mutual funds of_
→{s} of worth $ {p} per MF ' .format(fname = num, s = stk_symbol, p=price))
            self.History.append(self.cash )
            num+= num
            self.History_MF.append(num)
            self.History_MF.append(symbol)

            return self.cash

def sell_MFUND(self, num , stk_symbol):

    for symbol, price in self.fund_dict.items():
        if symbol == stk_symbol:
            #newval= list(self.stock_dict.values())

            selling_price= (random.uniform(0.9, 1.2)) ## selling unform RV_
→btw[0.9,1.2]
            stockvalue = (num* selling_price)
            self.cash = self.cash + stockvalue
            u= print('Total amount after selling {fname} mutual fund of {s}_
→' .format(fname = num, s = stk_symbol))
            self.History.append(self.cash )
            num+= num
            num= -1*num
            self.History_MF.append(num)
            self.History_MF.append(symbol)
            return self.cash

def history(self):
    # for x in range(len(self.History)):
    #     print( self.History[x])
    print("Cash amount(in USD)) ordered by time",str(self.History) )
    print(' \n ')
    print('Transaction history:\n ')
    print("Net cash available in in (USD):", self.History[-1] )
    print("Stocks history:", self.History_stock )
    print("Mutual Funds history:", self.History_MF )

```

[ ]:

```
[91]: # Test1 portfolio
portfolio=Portfolio()
print(portfolio.addcash(300.50))
print(portfolio.new_stock(20,'HFH'))
print(portfolio.buy_stock(5,'HFH'))

print(portfolio.new_MFUND(1,'BRT'))
print(portfolio.new_MFUND(1,'GHT'))
print(portfolio.buy_MFUND(10.3,'BRT'))
print(portfolio.buy_MFUND(2,'GHT'))

print(portfolio.sell_stock(1,'HFH'))
print(portfolio.sell_MFUND(3,'BRT'))
print(portfolio.withdrawCash(50))

print(portfolio.history() )

#print(port1.portfolio() )

print(' \n ')
print(portfolio )
```

Total amount after cash deposit of 300.5

300.5

None

Total amount after purchasing 5 stocks of HFH with price \$20 per stock

200.5

None

None

Total amount after purchasing 10.3 mutual funds of BRT of worth \$ 1 per MF

190.2

Total amount after purchasing 2 mutual funds of GHT of worth \$ 1 per MF

188.2

Total amount after selling 1 stocks of HFH

208.0471224718932

Total amount after selling 3 mutual fund of BRT

210.95226103210234

Total amount after cash withdrawal of 50

160.95226103210234

Cash amount(in (USD)) ordered by time [300.5, 200.5, 190.2, 188.2,

208.0471224718932, 210.95226103210234, 160.95226103210234]

Transaction history:

Net cash available in in (USD): 160.95226103210234

Stocks history: [5, 'HFH', -1, 'HFH']

Mutual Funds history: [10.3, 'BRT', 2, 'GHT', -3, 'BRT']

None

Current Portfolio:

Net cash in (USD) : 160.95226103210234

Stocks history:, [5, 'HFH', -1, 'HFH']

Mutual Funds history: [10.3, 'BRT', 2, 'GHT', -3, 'BRT']

```
[93]: port2=Portfolio()
print(port2.addcash(1000))

print(port2.new_stock(30,'Tesla'))

print(port2.buy_stock(3,'Tesla'))

print(port2.new_stock(50,'FB'))

print(port2.buy_stock(2,'FB'))

print(port2.addcash(500))

print(port2.new_stock(80,'GUCCI'))

print(port2.buy_stock(2,'GUCCI'))

print(port2.withdrawCash(200))

print(port2.sell_stock(2,'FB'))

print(port2.sell_stock(5,'FB'))

print(port2.new_MFUND(1,'GHT'))

print(port2.withdrawCash(300))
```

```

print(port2.new_MFUND(1, 'BRT'))

print(port2.buy_MFUND(20.5, 'GHT'))

print(port2.sell_MFUND(10.3, 'BRT'))

print(port2.buy_MFUND(12.6, 'GHT'))


print(port2.sell_MFUND(11.5, 'GHT'))


print(' \n ')
print(port2.history() )


print(' \n ')
print(portfolio )

```

Total amount after cash deposit of 1000

1000

None

Total amount after purchasing 3 stocks of Tesla with price \$30 per stock

910

None

Total amount after purchasing 2 stocks of FB with price \$50 per stock

810

Total amount after cash deposit of 500

1310

None

Total amount after purchasing 2 stocks of GUCCI with price \$80 per stock

1150

Total amount after cash withdrawal of 200

950

Total amount after selling 2 stocks of FB

1003.966332213174

Total amount after selling 5 stocks of FB

1211.3624019199826

None

Total amount after cash withdrawal of 300

911.3624019199826

None

Total amount after purchasing 20.5 mutual funds of GHT of worth \$ 1 per MF

890.8624019199826

Total amount after selling 10.3 mutual fund of BRT

900.351210648416

Total amount after purchasing 12.6 mutual funds of GHT of worth \$ 1 per MF

887.751210648416

Total amount after selling 11.5 mutual fund of GHT  
901.2823728513298

Cash amount(in (USD)) ordered by time [1000, 910, 810, 1310, 1150, 950,  
1003.966332213174, 1211.3624019199826, 911.3624019199826, 890.8624019199826,  
900.351210648416, 887.751210648416, 901.2823728513298]

Transaction history:

Net cash available in in (USD): 901.2823728513298

Stocks history: [3, 'Tesla', 2, 'FB', 2, 'GUCCI', -2, 'FB', -5, 'FB']

Mutual Funds history: [20.5, 'GHT', -10.3, 'BRT', 12.6, 'GHT', -11.5, 'GHT']

None

Current Portfolio:

Net cash in (USD) : 160.95226103210234

Stocks history:, [5, 'HFH', -1, 'HFH']

Mutual Funds history: [10.3, 'BRT', 2, 'GHT', -3, 'BRT']

[ ]:

[ ]: