

# mutual-fund-project

April 18, 2023

```
[ ]: import pandas as pd
import numpy as np
from random import sample
```

## 1 Reading Data

```
[ ]: df = pd.read_excel('./MUTUL_FUND_DATA_ FINAL_CODONS_new.xlsx')
df
```

## 2 Displaying Row

```
[ ]: def displayrow(df, var):
    df1 = df.loc[df['Category'] == var]

    #Extracting Row numbers to a list
    a = []
    for i in df1.index:
        a.append(i)

    #Extracting random 3 rows from dataframe
    b = sample(a,3)

    #Creating a new data frame with row present in the list
    df11 = df.iloc[b]

    return df11
```

## 3 Displaying Row with Rating Parameter

```
[ ]: def displayrowwithrating(df, var, star=''):
    df1 = df.loc[(df['Category'] == var) & (df['Fund_Rating'] == float(star))]

    #Extracting Row numbers to a list
    a = []
```

```

for i in df1.index:
    a.append(i)

#Extracting random 3 rows from dataframe
b = sample(a,3)

#Creating a new data frame with row present in the list
df11 = df.iloc[b]

return df11

```

## 4 Fetching Row Numbers

tolist() function can be used instead of this...

```

[ ]: def convertintorow(var):
    a = []
    for i in var:
        a.append(i)
    return a

```

## 5 Taking Input

```

[ ]: def takeint(var):
    while(True):
        print('\nPlease Choose a Fund from above list and Enter the Code Number,
↳Written Before AMCName')
        choice = input('Please Enter Your Choice: ')
        if choice.isnumeric():
            choice = int(choice)
            if choice in var:
                return choice
            else:
                print('\nInvalid Input')
        else:
            print('\nPlease Enter Correct Number')

```

## 6 Options Function

```

[ ]: def options(df, category):
    #displaying 3 rows randomly for dataframe using displayrow function
    dfequity = displayrow(df, category)
    rows = convertintorow(dfequity.index)
    display(dfequity)

```

```

#Other Options
while(True):
    x = input('Type "y" if you want new options \nType "a" if you have a
↳specific AMC in mind.. \nType "n" to continue...')
    if x.lower() == 'y':
        #displaying 3 rows randomly for dataframe using displayrow function
        dfequity = displayrow(df, category)
        rows = convertingtorow(dfequity.index)
        display(dfequity)

    elif x.lower() == 'a':
        print(df.AMC_Name.unique())
        y = input("Please Enter your Choice: ")
        if y in df.AMC_Name.unique():
            print("Amc of Choice: ",y)
            dftemp = df.loc[(df['AMC_Name']==y) &
↳(df['Category']==category)]
            rows = convertingtorow(dftemp.index)
            display(dftemp)
            break
        else:
            print("Invalid AMC")
    elif x.lower() == 'n':
        break
    else:
        print("Please Enter Appropriate Response!")

#getting user's choice
temp = takeint(rows)
print('\n',df.iloc[temp])
return temp

```

## 7 Options Function with Rating Parameter

```

[ ]: def optionswithrating(df, category, star=''):
    star = float(star)

    #displaying 3 rows randomly for dataframe using displayrow function
    dfequity = displayrowwithrating(df, category, star)
    rows = convertingtorow(dfequity.index)
    display(dfequity)

    #Other Options
    while(True):

```

```

x = input('Type "y" if you want new options \nType "a" if you have a
↳specific AMC in mind.. \nType "n" to continue...')
if x.lower() == 'y':
    #displaying 3 rows randomly for dataframe using displayrow function
    dfequity = displayrowwithrating(df, category, star)
    rows = convertingtorow(dfequity.index)
    display(dfequity)

elif x.lower() == 'a':
    print(df.AMC_Name.unique())
    y = input("Please Enter your Choice: ")
    if y in df.AMC_Name.unique():
        print("Amc of Choice: ",y)
        dftemp = df.loc[(df['AMC_Name']==y) &
↳(df['Category']==category)]
        rows = convertingtorow(dftemp.index)
        display(dftemp)
        break
    else:
        print("Invalid AMC")
elif x.lower() == 'n':
    break
else:
    print("Please Enter Appropriate Response!")

#getting user's choice
temp = takeint(rows)
print('\n',df.iloc[temp])
return temp

```

## 8 Calculator Program

```

[ ]: def calculator(investment, r, years):
    #Returns Calculator

    months = years * 12

    #Compound Interest Formula
    i = r/100/12

    #Seperate Calculation for FV
    a = (1+i)**months-1
    b = (1+i)/i

    #Final Future Value Calculation

```

```

FV = investment * a * b
#print(FV)

return FV

```

## 9 Calculation Function

```

[ ]: def calculate(df, amount, age, row):
    print("\nFollowing are the Funds you have chosen: ")
    display(df.loc[row])

    if amount <= 5000:
        if age <= 18:
            equity = amount

            a = df.loc[row]
            rate = a.Return_1Yr
            equity1 = calculator(equity, rate, 1)

            rate = a.Return_3Yr
            equity3 = calculator(equity, rate, 3)

            rate = a.Return_5Yr
            equity5 = calculator(equity, rate, 5)

            #Calculating Returns

            ## for 1 year
            print("Returns for 1 Year \n")
            totalinv = amount * 12
            print('Total Amount Invested = {value:.2f}'.format(value=totalinv))

            equity1 = float(equity1)
            print('Total Portfolio Value = {value:.2f}'.format(value=equity1))

            totalgain = equity1 - totalinv
            print('Net Profit = {value:.2f}'.format(value=totalgain))

            gainper = totalgain/totalinv*100
            print('Percentage Gained = {value:.2f}'.format(value=gainper))

            ## for 3 years
            print('\nReturns for 3 Years\n')
            totalinv = amount * 36
            print('Total Amount Invested = {value:.2f}'.format(value=totalinv))

```

```

equity3 = float(equity3)
print('Total Portfolio Value = {value:.2f}'.format(value=equity3))

totalgain = equity3 - totalinv
print('Net Profit = {value:.2f}'.format(value=totalgain))

gainper = totalgain/totalinv*100
print('Percentage Gained = {value:.2f}'.format(value=gainper))

## for 5 years
print('\nReturns for 5 Years\n')
totalinv = amount * 60
print('Total Amount Invested = {value:.2f}'.format(value=totalinv))

equity5 = float(equity5)
print('Total Portfolio Value = {value:.2f}'.format(value=equity5))

totalgain = equity5 - totalinv
print('Net Profit = {value:.2f}'.format(value=totalgain))

gainper = totalgain/totalinv*100
print('Percentage Gained = {value:.2f}'.format(value=gainper))

else:
    var = 100 - age
    per = var/100
    equity = amount * per
    hybrid = amount - equity

    #For Equity
    a = df.loc[row[0]]
    rate = a.Return_1Yr
    equity1 = calculator(equity, rate, 1)

    rate = a.Return_3Yr
    equity3 = calculator(equity, rate, 3)

    rate = a.Return_5Yr
    equity5 = calculator(equity, rate, 5)

    #For Hybrid
    a = df.loc[row[1]]
    rate = a.Return_1Yr
    hybrid1 = calculator(hybrid, rate, 1)

    rate = a.Return_3Yr
    hybrid3 = calculator(hybrid, rate, 3)

```

```

rate = a.Return_5Yr
hybrid5 = calculator(hybrid, rate, 5)

#Calculating Returns

## for 1 year
print("Returns for 1 Year \n")
totalinv = amount * 12
print('Total Amount Invested = {value:.2f}'.format(value=totalinv))

Year1 = float(equity1) + float(hybrid1)
print('Total Portfolio Value = {value:.2f}'.format(value=Year1))

totalgain = Year1 - totalinv
print('Net Profit = {value:.2f}'.format(value=totalgain))

gainper = totalgain/totalinv*100
print('Percentage Gained = {value:.2f}'.format(value=gainper))

## for 3 years
print('\nReturns for 3 Years\n')
totalinv = amount * 36
print('Total Amount Invested = {value:.2f}'.format(value=totalinv))

Year3 = float(equity3) + float(hybrid3)
print('Total Portfolio Value = {value:.2f}'.format(value=Year3))

totalgain = Year3 - totalinv
print('Net Profit = {value:.2f}'.format(value=totalgain))

gainper = totalgain/totalinv*100
print('Percentage Gained = {value:.2f}'.format(value=gainper))

## for 5 years
print('\nReturns for 5 Years\n')
totalinv = amount * 60
print('Total Amount Invested = {value:.2f}'.format(value=totalinv))

Year5 = float(equity5) + float(hybrid5)
print('Total Portfolio Value = {value:.2f}'.format(value=Year5))

totalgain = Year5 - totalinv
print('Net Profit = {value:.2f}'.format(value=totalgain))

gainper = totalgain/totalinv*100
print('Percentage Gained = {value:.2f}'.format(value=gainper))

```

```

elif amount > 5000 and amount <= 10000:
    if age <= 18:
        amount1 = amount/2
        amount2 = amount/2

        a = df.loc[row[0]]
        rate = a.Return_1Yr
        equity1 = calculator(amount1, rate, 1)

        rate = a.Return_3Yr
        equity3 = calculator(amount1, rate, 3)

        rate = a.Return_5Yr
        equity5 = calculator(amount1, rate, 5)

        b = df.loc[row[1]]
        rate = b.Return_1Yr
        equity21 = calculator(amount2, rate, 1)

        rate = b.Return_3Yr
        equity23 = calculator(amount2, rate, 3)

        rate = b.Return_5Yr
        equity25 = calculator(amount2, rate, 5)

#Calculating Returns

## for 1 year
print("Returns for 1 Year \n")
totalinv = amount * 12
print('Total Amount Invested = {value:.2f}'.format(value=totalinv))

Year1 = equity1 + equity21
print('Total Portfolio Value = {value:.2f}'.format(value=Year1))

totalgain = Year1 - totalinv
print('Net Profit = {value:.2f}'.format(value=totalgain))

gainper = totalgain/totalinv*100
print('Percentage Gained = {value:.2f}'.format(value=gainper))

## for 3 years
print('\nReturns for 3 Years\n')
totalinv = amount * 36
print('Total Amount Invested = {value:.2f}'.format(value=totalinv))

```



```

Year3 = equity3 + equity23
print('Total Portfolio Value = {value:.2f}'.format(value=Year3))

totalgain = Year3 - totalinv
print('Net Profit = {value:.2f}'.format(value=totalgain))

gainper = totalgain/totalinv*100
print('Percentage Gained = {value:.2f}'.format(value=gainper))

## for 5 years
print('\nReturns for 5 Years\n')
totalinv = amount * 60
print('Total Amount Invested = {value:.2f}'.format(value=totalinv))

Year5 = equity5 + equity25
print('Total Portfolio Value = {value:.2f}'.format(value=Year5))

totalgain = Year5 - totalinv
print('Net Profit = {value:.2f}'.format(value=totalgain))

gainper = totalgain/totalinv*100
print('Percentage Gained = {value:.2f}'.format(value=gainper))

else:
    var = 100 - age
    per = var/100
    equity = amount * per
    var2 = amount - equity
    hybrid = var2 / 2
    debt = var2 / 2

    #for equity funds
    fund1 = equity/2
    fund2 = equity/2

    #For Equity Fund 1
    a = df.loc[row[0]]
    rate = a.Return_1Yr
    equity1 = calculator(fund1, rate, 1)

    rate = a.Return_3Yr
    equity3 = calculator(fund1, rate, 3)

    rate = a.Return_5Yr
    equity5 = calculator(fund1, rate, 5)

```

```

#For Equity Fund 2
a = df.loc[row[1]]
rate = a.Return_1Yr
equity21 = calculator(fund2, rate, 1)

rate = a.Return_3Yr
equity23 = calculator(fund2, rate, 3)

rate = a.Return_5Yr
equity25 = calculator(fund2, rate, 5)

#For Hybrid
a = df.loc[row[2]]
rate = a.Return_1Yr
hybrid1 = calculator(hybrid, rate, 1)

rate = a.Return_3Yr
hybrid3 = calculator(hybrid, rate, 3)

rate = a.Return_5Yr
hybrid5 = calculator(hybrid, rate, 5)

#For Debt
a = df.loc[row[3]]
rate = a.Return_1Yr
debt1 = calculator(debt, rate, 1)

rate = a.Return_3Yr
debt3 = calculator(debt, rate, 3)

rate = a.Return_5Yr
debt5 = calculator(debt, rate, 5)

#Calculating Returns

## for 1 year
print("Returns for 1 Year \n")
totalinv = amount * 12
print('Total Amount Invested = {value:.2f}'.format(value=totalinv))

Year1 = equity1 + equity21 + hybrid1 + debt1
print('Total Portfolio Value = {value:.2f}'.format(value=Year1))

totalgain = Year1 - totalinv
print('Net Profit = {value:.2f}'.format(value=totalgain))

```

```

gainper = totalgain/totalinv*100
print('Percentage Gained = {value:.2f}'.format(value=gainper))

## for 3 years
print('\nReturns for 3 Years\n')
totalinv = amount * 36
print('Total Amount Invested = {value:.2f}'.format(value=totalinv))

Year3 = equity3 + equity23 + hybrid3 + debt3
print('Total Portfolio Value = {value:.2f}'.format(value=Year3))

totalgain = Year3 - totalinv
print('Net Profit = {value:.2f}'.format(value=totalgain))

gainper = totalgain/totalinv*100
print('Percentage Gained = {value:.2f}'.format(value=gainper))

## for 5 years
print('\nReturns for 5 Years\n')
totalinv = amount * 60
print('Total Amount Invested = {value:.2f}'.format(value=totalinv))

Year5 = equity5 + equity25 + hybrid5 + debt5
print('Total Portfolio Value = {value:.2f}'.format(value=Year5))

totalgain = Year5 - totalinv
print('Net Profit = {value:.2f}'.format(value=totalgain))

gainper = totalgain/totalinv*100
print('Percentage Gained = {value:.2f}'.format(value=gainper))

elif amount > 10000:
    if age <= 18:
        amount1 = amount/3
        amount2 = amount/3
        amount3 = amount/3

        a = df.loc[row[0]]
        rate = a.Return_1Yr
        equity1 = calculator(amount1, rate, 1)

        rate = a.Return_3Yr
        equity3 = calculator(amount1, rate, 3)

        rate = a.Return_5Yr
        equity5 = calculator(amount1, rate, 5)

```

```

b = df.loc[row[1]]
rate = b.Return_1Yr
equity21 = calculator(amount2, rate, 1)

rate = b.Return_3Yr
equity23 = calculator(amount2, rate, 3)

rate = b.Return_5Yr
equity25 = calculator(amount2, rate, 5)

c = df.loc[row[2]]
rate = c.Return_1Yr
equity31 = calculator(amount3, rate, 1)

rate = c.Return_3Yr
equity33 = calculator(amount3, rate, 3)

rate = c.Return_5Yr
equity35 = calculator(amount3, rate, 5)

#Calculating Returns

## for 1 year
print("Returns for 1 Year \n")
totalinv = amount * 12
print('Total Amount Invested = {value:.2f}'.format(value=totalinv))

Year1 = equity1 + equity21 + equity31
print('Total Portfolio Value = {value:.2f}'.format(value=Year1))

totalgain = Year1 - totalinv
print('Net Profit = {value:.2f}'.format(value=totalgain))

gainper = totalgain/totalinv*100
print('Percentage Gained = {value:.2f}'.format(value=gainper))

## for 3 years
print('\nReturns for 3 Years\n')
totalinv = amount * 36
print('Total Amount Invested = {value:.2f}'.format(value=totalinv))

Year3 = equity3 + equity23 + equity33
print('Total Portfolio Value = {value:.2f}'.format(value=Year3))

totalgain = Year3 - totalinv

```

```

print('Net Profit = {value:.2f}'.format(value=totalgain))

gainper = totalgain/totalinv*100
print('Percentage Gained = {value:.2f}'.format(value=gainper))

## for 5 years
print('\nReturns for 5 Years\n')
totalinv = amount * 60
print('Total Amount Invested = {value:.2f}'.format(value=totalinv))

Year5 = equity5 + equity25 + equity35
print('Total Portfolio Value = {value:.2f}'.format(value=Year5))

totalgain = Year5 - totalinv
print('Net Profit = {value:.2f}'.format(value=totalgain))

gainper = totalgain/totalinv*100
print('Percentage Gained = {value:.2f}'.format(value=gainper))

else:
    var = 100 - age
    per = var/100
    equity = amount * per
    var2 = amount - equity
    hybrid = var2 / 2
    debt = var2 / 2

    #for equity funds
    fund1 = equity/3
    fund2 = equity/3
    fund3 = equity/3

    #For Equity Fund 1
    a = df.loc[row[0]]
    rate = a.Return_1Yr
    equity1 = calculator(fund1, rate, 1)

    rate = a.Return_3Yr
    equity3 = calculator(fund1, rate, 3)

    rate = a.Return_5Yr
    equity5 = calculator(fund1, rate, 5)

    #For Equity Fund 2
    a = df.loc[row[1]]
    rate = a.Return_1Yr
    equity21 = calculator(fund2, rate, 1)

```

```

rate = a.Return_3Yr
equity23 = calculator(fund2, rate, 3)

rate = a.Return_5Yr
equity25 = calculator(fund2, rate, 5)

#For Equity Fund 3
c = df.loc[row[2]]
rate = c.Return_1Yr
equity31 = calculator(fund3, rate, 1)

rate = c.Return_3Yr
equity33 = calculator(fund3, rate, 3)

rate = c.Return_5Yr
equity35 = calculator(fund3, rate, 5)

#For Hybrid
a = df.loc[row[3]]
rate = a.Return_1Yr
hybrid1 = calculator(hybrid, rate, 1)

rate = a.Return_3Yr
hybrid3 = calculator(hybrid, rate, 3)

rate = a.Return_5Yr
hybrid5 = calculator(hybrid, rate, 5)

#For Debt
a = df.loc[row[4]]
rate = a.Return_1Yr
debt1 = calculator(debt, rate, 1)

rate = a.Return_3Yr
debt3 = calculator(debt, rate, 3)

rate = a.Return_5Yr
debt5 = calculator(debt, rate, 5)

#Calculating Returns

## for 1 year
print("Returns for 1 Year \n")
totalinv = amount * 12

```

```

print('Total Amount Invested = {value:.2f}'.format(value=totalinv))

Year1 = (equity1 + equity21 + equity31 + hybrid1 + debt1)
print('Total Portfolio Value = {value:.2f}'.format(value=Year1))

totalgain = Year1 - totalinv
print('Net Profit = {value:.2f}'.format(value=totalgain))

gainper = totalgain/totalinv*100
print('Percentage Gained = {value:.2f}'.format(value=gainper))

## for 3 years
print('\nReturns for 3 Years\n')
totalinv = amount * 36
print('Total Amount Invested = {value:.2f}'.format(value=totalinv))

Year3 = equity3 + equity23 + equity33 + hybrid3 + debt3
print('Total Portfolio Value = {value:.2f}'.format(value=Year3))

totalgain = Year3 - totalinv
print('Net Profit = {value:.2f}'.format(value=totalgain))

gainper = totalgain/totalinv*100
print('Percentage Gained = {value:.2f}'.format(value=gainper))

## for 5 years
print('\nReturns for 5 Years\n')
totalinv = amount * 60
print('Total Amount Invested = {value:.2f}'.format(value=totalinv))

Year5 = float(equity5) + float(equity25) + float(equity35) + ↪float(hybrid5) + float(debt5)
print('Total Portfolio Value = {value:.2f}'.format(value=Year5))

totalgain = Year5 - totalinv
print('Net Profit = {value:.2f}'.format(value=totalgain))

gainper = totalgain/totalinv*100
print('Percentage Gained = {value:.2f}'.format(value=gainper))

```

## 10 Dashboard

```

[ ]: while(True):
    age = input('Please Enter your age: ')
    if age.isnumeric():
        age = int(age)

```

```

    if age > 100:
        print('Please Enter Appropriate Age.\n')
    elif age <= 18:
        per1 = 1
        break
    else:
        per = 100-age
        per1 = per/100
        #print('\nper: {value:.2f}'.format(value = int(per)))
        break
else:
    print('Please Enter appropriate Age.\n')

while(True):
    while(True):
        amount = input('\nPlease Enter Amount you wish to Invest: ')
        if amount.isnumeric():
            amount = int(amount)
            break
        else:
            print('\nPlease Enter Appropriate Amount.')

    if amount < 1000:
        print('\nMinimum Amount should be atleast Rs.1000')

    elif amount <= 5000:
        equity = amount * per1
        hybrid = amount - equity
        if hybrid == 0.00:
            print('\nAllocated Amount: \nEquity: {value:.2f}'.format(value = equity))
            rowno = []
            print('\nPlease Choose an Equity Fund: ')
            rowno.append(options(df, 'Equity'))

            xyz = df.loc[rowno]
            xyz.to_csv('Mutual_Funds.csv')
            calculate(df, amount, age, rowno)
            break
        else:
            print('\nAllocated Amount: \nEquity: {value:.2f} \nHybrid: {value2:.2f}'.format(value = equity, value2 = hybrid))
            rowno = []
            print('\nPlease Choose an Equity Fund: ')
            rowno.append(options(df, 'Equity'))

            print('\nPlease Choose a Hybrid Fund')

```



```

        rowno.append(options(df, 'Hybrid'))

        xyz = df.loc[rowno]
        xyz.to_csv('Mutual_Funds.csv')

        calculate(df, amount, age, rowno)
        break

    elif amount > 5000 and amount <= 10000:
        equity = amount * per1
        var = amount - equity
        hybrid = var / 2
        debt = var / 2

        if hybrid == 0.00 and debt == 0.00:
            print('\nAllocated Amount: \nEquity: {value:.2f} \nHybrid: {value2:.2f} \nDebt: {value3:.2f}'.format(value = equity, value2 = hybrid, value3 = debt))
            rowno = []
            print('\nPlease Choose Your First Equity Fund: ')
            rowno.append(optionwithrating(df, 'Equity', 5))

            print('\nPlease Choose Your Second Equity Fund: ')
            rowno.append(optionwithrating(df, 'Equity', 4))

            xyz = df.loc[rowno]
            xyz.to_csv('Mutual_Funds.csv')

            calculate(df, amount, age, rowno)
            break
        else:
            print('\nAllocated Amount: \nEquity: {value:.2f} \nHybrid: {value2:.2f} \nDebt: {value3:.2f}'.format(value = equity, value2 = hybrid, value3 = debt))
            rowno = []
            print('\nPlease Choose Your First Equity Fund: ')
            rowno.append(optionwithrating(df, 'Equity', 5))

            print('\nPlease Choose Your Second Equity Fund: ')
            rowno.append(optionwithrating(df, 'Equity', 4))

            print('\nPlease Choose a Hybrid Fund')
            rowno.append(options(df, 'Hybrid'))

            print('\nPlease Choose a Debt Fund')
            rowno.append(options(df, 'Debt'))

```

```

xyz = df.loc[rowno]
xyz.to_csv('Mutual_Funds.csv')

calculate(df, amount, age, rowno)
break

elif amount > 10000:
    equity = amount * per1
    var = amount - equity
    hybrid = var / 2
    debt = var / 2

    print('\nAllocated Amount: \nEquity: {value:.2f} \nHybrid: {value2:.2f} \nDebt: {value3:.2f}'.format(value = equity, value2 = hybrid, value3 = debt))
    if hybrid == 0.00 and debt == 0.00:
        print('\nAllocated Amount: \nEquity: {value:.2f} \nHybrid: {value2:.2f} \nDebt: {value3:.2f}'.format(value = equity, value2 = hybrid, value3 = debt))

    rowno = []
    print('\nPlease Choose Your First Equity Fund: ')
    rowno.append(optionswithrating(df, 'Equity', 5))

    print('\nPlease Choose Your Second Equity Fund: ')
    rowno.append(optionswithrating(df, 'Equity', 4))

    print('\nPlease Choose Your Third Equity Fund: ')
    rowno.append(optionswithrating(df, 'Equity', 3))

    xyz = df.loc[rowno]
    xyz.to_csv('Mutual_Funds.csv')

    calculate(df, amount, age, rowno)
    break
else:
    print('\nAllocated Amount: \nEquity: {value:.2f} \nHybrid: {value2:.2f} \nDebt: {value3:.2f}'.format(value = equity, value2 = hybrid, value3 = debt))

    rowno = []
    print('\nPlease Choose Your First Equity Fund: ')
    rowno.append(optionswithrating(df, 'Equity', 5))

    print('\nPlease Choose Your Second Equity Fund: ')
    rowno.append(optionswithrating(df, 'Equity', 4))

    print('\nPlease Choose Your Third Equity Fund: ')
    rowno.append(optionswithrating(df, 'Equity', 3))

```

```
print('\nPlease Choose a Hybrid Fund')
rowno.append(options(df, 'Hybrid'))

print('\nPlease Choose a Debt Fund')
rowno.append(options(df, 'Debt'))

xyz = df.loc[rowno]
xyz.to_csv('Mutual_Funds.csv')

calculate(df, amount, age, rowno)
break

else:
    print('Invalid Input')
```

```
[ ]:
```