```
# Project:      Lab03 (VoMikeLab03SecHY02Ver01.py)
# Name:         Mike Vo
# Date:         01/23/15
# Description:  This program consists of 2 parts:
#
#   Part 1: The Konditorei coffee shop sells coffee at $10.50 a pound
#           plus the cost of shipping. Each order ships for $.86 per
#           pound + $1.50 fixed cost for overhead. This part calculates
#           the cost of an order.
#
#   Part 2: Asks a user for their age and the balance of their savings
#           account. Then calculates how much they need to contribute
#           to that savings account, on a monthly basis, so that they
#           can retire at age 68 with $1,125,000.

# Part 1:
def MainA():

    # Print part 1 header to shell
    print( "\n*************** Part 1 ***************" )
    print( "\nKONDITOREI COFFEE SHOP" )
    print( "Price: $10.50/pound  |  Shipping: $0.86/pound + $1.50
overhead (fixed)" )
    print( "\nEnter your order:")

    # Prompt user for order info
    fltCoffeeAmountPounds = float( input( "Coffee Amount (pounds)>>> " )
)

    # Calculate coffee total price, shipping price (including fixed
overhead),
    # sub-total (without tax)
    fltCoffeeTotalPrice = fltCoffeeAmountPounds * 10.50
    fltShippingPriceWithOverhead = fltCoffeeAmountPounds * .86 + 1.50
    fltSubTotal = fltCoffeeTotalPrice + fltShippingPriceWithOverhead

    # Tax Rate: 9%
    fltTaxRate = .09

    # Calculate Total (with tax)
    fltTotal = fltSubTotal * (1 + fltTaxRate)

    # Print output
    print( "\nCoffee                  :     ${0:0.2f}".format(
fltCoffeeTotalPrice ) )
    print( "Shipping                :     ${0:0.2f}".format(
fltShippingPriceWithOverhead - 1.50 ) )
    print( "Overhead fee (fixed)    :     $1.50" )
    print( "----------------------------------------" )
    print( "Sub-total               :     ${0:0.2f}".format( fltSubTotal
) )
    print( "Tax                     :      9%" )
    print( "----------------------------------------" )
    print( "Total                   :     ${0:0.2f}".format( fltTotal ) )
```

```
    print( "\nThank you for shopping at Konditorei Coffee Shop :-)" )

# Part 2:
def MainB():

    # Print part 2 header to shell
    print( "\n*************** Part 2 ***************" )
    print( "\nBANK OF 'MURRICA" )
    print( "Saving account calculator. Goal: Retire at age 68 with
$1,125,000" )
    print( "\nEnter your info:" )

    # Prompt user for age and savings account balance
    intAge = int( input( "How old are you? (I'm sorry)>>> " ) )
    intMonthsUntilNextBirthday = int( input( "How many month(s) from now
until next birthday?>>> " ) )
    fltCurrentSavingsBalance = float( input( "Savings Account Balance>>>
$" ) )

    # Calculate time until aged 68 (in months, then years)
    intYearsUntilAged68 = 68 - intAge
    intMonthsUntilAged68 = intYearsUntilAged68 * 12 - (12 -
intMonthsUntilNextBirthday)

    # Calculate the amount of money to save (total, then monthly)
    fltMoneyToSave = 1125000.00 - fltCurrentSavingsBalance
    fltMoneyToSavePerMonth = fltMoneyToSave / intMonthsUntilAged68

    # Print output
    print( "\nYou have to save ${0:0.2f} per month to retire at age 68
with $1,125,000".format( fltMoneyToSavePerMonth ) )


# Main program:
def main():

    # Print global header to shell
    print( "########################################" )
    print( "#     Mike Vo - CSC110 HY02 Lab03      #" )
    print( "#               Ver. 01                #" )
    print( "########################################" )

    # Evoke part 1 and part 2
    MainA()
    MainB()

# Evoke main program
main()

"""

TEST DATA (Created on MS Excel)

Part 1:
```

Coffee (lbs)	Price	Shipping	Overhead	Sub-Total	Tax Rate	Total
1	10.5	0.86	1.5	12.86	0.09	14.0174
1.5	15.75	1.29	1.5	18.54	0.09	20.2086
10	105	8.6	1.5	115.1	0.09	125.459
17.25	181.125	14.835	1.5	197.46	0.09	215.2314
120	1260	103.2	1.5	1364.7	0.09	1487.523
101.52	1065.96	87.3072	1.5	1154.7672	0.09	1258.696248
1000	10500	860	1.5	11361.5	0.09	12384.035
1234.56	12962.88	1061.7216	1.5	14026.1016	0.09	15288.45074

Part 2:

Age	MTB*	Balance	Monthly Saving
20	3	5700	1974.074074
45	0	50000	4071.969697
1	11	100000	1276.463263
67	1	0	1125000
67	11	102500	92954.54545
68	0	1125000	0
68	0	0	-93750
75	0	1126000	10.41666667
73	-60	1126000	7.575757576
75	0	0	-11718.75
73	-60	0	-8522.727273
0	0	0	1399.253731
0	0	1250000	-155.4726368
0	0	2000000	-1088.308458

*MTB: Months till birthday

"""