```
# Project:      HW02(VoMikeHW02SecHY02Ver02.py)
# Name:         Mike Vo
# Date:         01/26/17
# Description:  This program determines the cost of 3 items shipped with
tax
#
#   User input (for each of those 3 items)
#    > Description
#    > Price
#    > Quantity
#    > Weight per item
#
#   Output:
#    > Item purchased
#    > Sub-total
#    > Shipping + Handling costs
#    > Tax
#    > Total

def main():

    ### Display main header ###
    print( "########################################" )
    print( "#     Mike Vo - CSC110 HY02 HW02     #" )
    print( "#               Ver. 02              #" )
    print( "########################################" )

    ### Initialize data bank ###
    strItemDescriptionSummary = "\t"
    fltPriceSubTotal = 0
    fltTotalShippingAndHandlingCost = 0

    ### Initialize constants ###
    fltConstShippingCostPerLb = .25
    fltConstHandlingCostPerOrder = 5.0
    fltConstTaxRate = .09
    intNumberOfOrders = 3

    # try-except structure to catch bad input
    try:

        ### User input and data processing ###
        print( "\nPlease enter your purchase information" )
        print( "--------------------------------------" )

        # Input loop
        for intIndex in range( intNumberOfOrders ):

            # Item header
            print( "Item " + str( intIndex + 1 ) + ":" )

            # Item data input
            strItemDescription = str( input( " > Description       : " )
)
```

```
            fltItemUnitPrice = float( input( " > Price             : $" )
)
            intItemQuantity = int( input( " > Quantity          : " ) )
            fltWeightPerItemLbs = float( input( " > Weight/Item (lbs) : "
) )

            # Add description to data bank
            strItemDescriptionSummary += str( intIndex + 1 ) + ". " +
strItemDescription
            if intIndex != (intNumberOfOrders - 1):
                strItemDescriptionSummary += "\n\t"

            # Calculate Shipping cost and item Price; then add them to
data bank (not including Handling cost)
            fltTotalShippingAndHandlingCost += fltWeightPerItemLbs *
intItemQuantity * fltConstShippingCostPerLb
            fltPriceSubTotal += fltItemUnitPrice * intItemQuantity

        # Finalize data bank
        fltTotalShippingAndHandlingCost += fltConstHandlingCostPerOrder
        fltPriceSubTotal += fltTotalShippingAndHandlingCost
        fltPriceTotal = fltPriceSubTotal * (1 + fltConstTaxRate)

        ### Display output ###
        print( "\n\nYou have purchased:\n" + strItemDescriptionSummary )
        print( "--------------------------------------" )
        print( "Sub-total                    : ${0:0.2f}".format(
fltPriceSubTotal ) )
        print( "Shipping and Handling costs : ${0:0.2f}".format(
fltTotalShippingAndHandlingCost ) )
        print( "Tax (9%)                     : ${0:0.2f}".format(
(fltPriceSubTotal * fltConstTaxRate) ) )
        print( "--------------------------------------" )
        print( "Total                        : ${0:0.2f}".format(
fltPriceTotal ) )

    # Print a message when ecounter ValueError (user enter the wrong data
type)
    except ValueError:
        print( "Error: Bad input. Program terminated." )

main()

"""

TEST DATA
Generated using MS Excel

Item #           Description                          Price Quantity
     Weight/Item (lb)
1          Gibson Hummingbird guitar M2017-Standard     3,349 1
     2.28
2          DAddario EJ16 Phosphor Bronze, Light, 12-53   6.49  3
     0.03
```

```
3           Musicians Gear Deluxe Dreadnought Case B      79.99 1
      3.11
-------------------------------------------------------------------------
--------------------------------
Sub-Total                                                       3454.805

S+H*                                                            6.355

Tax                                                             310.93245

Total                                                           3765.73745

*S+H: Shipping and Handling costs

"""
```