Midterm-1 Program

# Author = Darren Isaacson

# This program is designed to cakcukate and display the information for a one way flight

# to New Zealand.

print("Welcome to D Airline agent")

print("This trip is going to New Zealand today! \n---------------------------------------------")

# Global Variables

ticketPrice = 1500

suitcasePrice = 40

taxandFees = 500

singleAgent = 75

# Collects user quanities

getticketQuanity = int(input("How many tickets will you be needing?"))

getsuitcaseQuanity = int(input("How many suitcases are you bringing?"))

#Calculates the quanities by the cost of tickets, suitcases, fees, and agent

suitcaseCost = suitcasePrice \* getsuitcaseQuanity

ticketCost = (ticketPrice + taxandFees) \* getticketQuanity

total = suitcaseCost + ticketCost + singleAgent

# Prints output

print("Cost per ticket: $%.2f" % ticketPrice)

print("Cost per suitcase: $%.2f" % suitcasePrice)

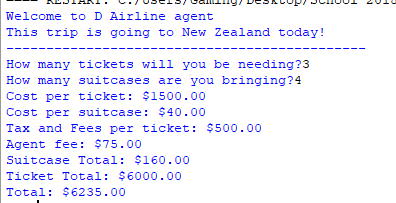
print("Tax and Fees per ticket: $%.2f" % taxandFees)

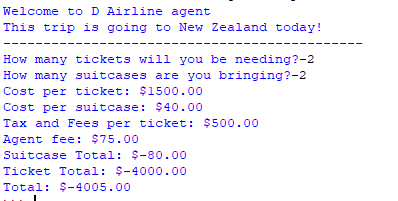
print("Agent fee: $%.2f" % singleAgent)

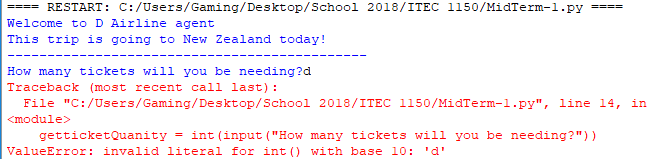
print("Suitcase Total: $%.2f" % suitcaseCost)

print("Ticket Total: $%.2f" % ticketCost)

print("Total: $%.2f" % total)







MidTerm – 2 Program

# Author = Darren Isaacson

# This program is designed to show and example of a taxi service pricing and mileage calculations

def main ():

try:

people,miles = getQuanity() # Collects user quanity

milePrice, miles = getDiscount(miles) # Collects discount based off miles

totalPeople, totalMiles, totalCost = getCalculation(people, miles, milePrice) # Calculates the collected data

printOutput(totalPeople,totalMiles,totalCost) # Prints output

# General Exception

except:

print("There was an error! You cannot have more than 4 people in 1 car.")

def getQuanity():

people = int(input("How many people are you planning to bring in a taxi. Max = 4")) #Collects user people amount

# Data Validation

if people < 0:

people = 0

elif people < 4:

people = people

else:

people = 4

print("You cannot have more than 4 people in a taxi. You will have to call another taxi."

"\nI have automatically set the max to 4 for you so you dont have to go back. \nApologize for the inconvience")

miles = float(input("How far in miles are you planning to drive?")) # Collects user miles

if miles < 0:

miles = 0

else:

miles = miles

return people,miles #Returns values

def getDiscount(miles):

# Sets base variable

priceperMile = 0

# Condition loop based off amount of miles and tells user their cost per mile

if miles < 10:

priceperMile = 2.0

print("Your price per mile is $2.00")

elif miles < 20:

priceperMile = 1.75

print("Your price per mile is $1.75")

elif miles < 50:

priceperMile = 1.50

print("Your price per mile is $1.50")

elif miles < 100:

priceperMile = 1.00

print("Your price per mile is $1.00")

else:

priceperMile = .75

print("Your price per mile is $0.75")

return priceperMile, miles

def getCalculation(people, miles, milePrice):

peoplePrice = 2 #Taxi entrance fee

totalPeople = people \* peoplePrice # Cost of people

totalMile = miles \* milePrice # Cost of miles with discount

totalCost = totalPeople + totalMile # total of all values

return totalPeople,totalMile,totalCost

def printOutput(totalPeople,totalMiles,totalCost):

#Prints output

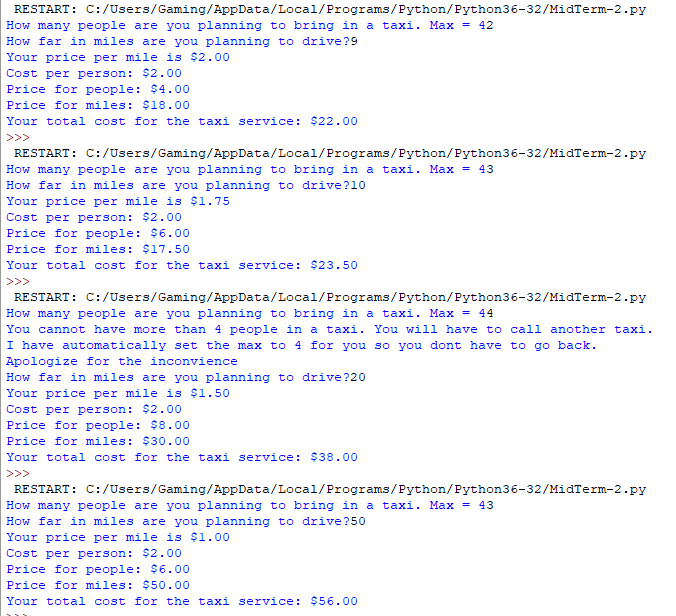
print("Cost per person: $2.00")

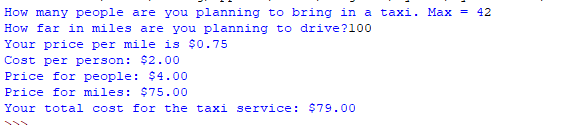
print("Price for people: $%.2f" % totalPeople)

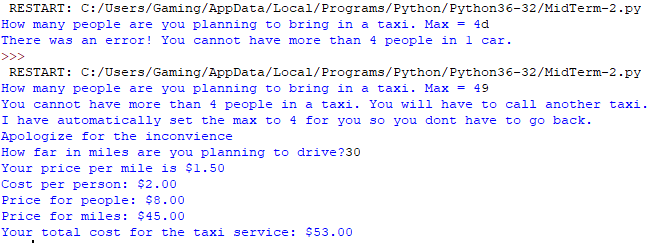
print("Price for miles: $%.2f" % totalMiles)

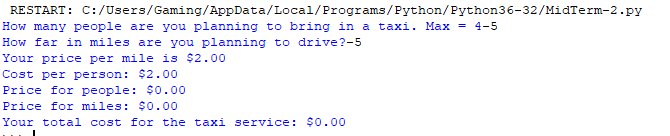
print("Your total cost for the taxi service: $%.2f" % totalCost)

main()









MidTerm-3 program

# Author = Darren Isaacson

# This program is designed to help create list and calculate the values.

def main():

try:

values = getValues()# Gathers list values

count,list,total = listOutput(values) # Accumulates the values

printOutput(list,total) #Prints output

# General Exceptions

except:

print("There was an error")

def getValues():

valueList = [] #Base list

go = "y" # Base variable

num = 0 # Base value accumulator

while go == 'y': # While loop for input value

values = float(input("What is the pricing for values #%d" %(num + 1)))

while values <=0:

values = float(input("Values must be greater that 0. What is the pricing for values #%d" %(num + 1)))

valueList.append(values) # Collects values inputed

num += 1

go = input("Do you want to enter another value? Press y to enter another value or any other key to stop") # Loop break

return valueList

def listOutput(values):

total = 0

for count in range(len(values)): # Takes each value in list

total = total + values[count] # List accumulator

return count, values, total

def printOutput(list,total):

for x in list:

print("Value numbers in your list are %.2f" % x) # Displays each value in list

print("The total of your list is: $%.2f" % total) # Displays total

main()

