```
"""
Blackbelt Challenge - Changemaker GUI
mthorman_blackbelt_changemakergui.py
Michael Thorman
6/13/2017
CSCI 23000
Create a program that calculates change after a purchase. Add a GUI.
"""
from tkinter import *

class changeMaker(Tk):      #Creates GUI for determining change
      def __init__(self):
            Tk.__init__(self)
            self.headerFont = ("Helvetica", "12")

            self.title("Changemaker")
            self.setInfo()

      def setInfo(self):     #Method for creating labels and button
            Label(self, text = "Change Calculator", fg="purple",
                        font = self.headerFont).grid(columnspan = 4)

                #Label for user input price
            Label(self, text = "Price of item:", fg="purple", font=("Helvetica",
12)).grid(row = 1, column = 0)
            self.textPriceItem = Entry(self, font=("Helvetica", 12))
            self.textPriceItem.grid(row = 1, column = 1 )

            #Label for user input money tendered
            Label(self, text = "Money given:", fg="purple", font=("Helvetica",
12)).grid(row = 2, column = 0)
            self.textMoneyGiven = Entry(self, font=("Helvetica", 12))
            self.textMoneyGiven.grid(row = 2, column = 1)


            #Button which initiates the calculation of change and money
distribution
            self.buttonCalc = Button(self, text = "Calculate Change", fg="purple",
relief="raised", font=("Helvetica", 12), bg ="light blue")
            self.buttonCalc.grid(row = 3, columnspan = 2)
            self.buttonCalc["command"] = self.calcChange
            #Change output label
            Label(self, text = "Change:", fg="purple", font=("Helvetica",
12)).grid(row = 4, column = 0)
            self.labelChange = Label(self, bg = "#fff", anchor = "w", relief =
"ridge", font=("Helvetica", 12))
            self.labelChange.grid(row = 4, column = 1, sticky = "we")

            #Output labels for change distribution
            Label(self, text = "Twenties:", fg="purple", font=("Helvetica",
12)).grid(row = 6, column = 0)
            self.labelTwenties = Label(self, bg = "#fff", anchor = "w", relief =
"ridge", font=("Helvetica", 12))
            self.labelTwenties.grid(row = 6, column = 1, sticky = "we")

            Label(self, text = "Tens:", fg="purple", font=("Helvetica",
12)).grid(row = 7, column = 0)
            self.labelTens = Label(self, bg = "#fff", anchor = "w", relief =
"ridge", font=("Helvetica", 12))
```

```
            self.labelTens.grid(row = 7, column = 1, sticky = "we")

            Label(self, text = "Fives:", fg="purple", font=("Helvetica",
12)).grid(row = 8, column = 0)
            self.labelFives = Label(self, bg = "#fff", anchor = "w", relief =
"ridge", font=("Helvetica", 12))
            self.labelFives.grid(row = 8, column = 1, sticky = "we")

            Label(self, text = "Ones:", fg="purple", font=("Helvetica",
12)).grid(row = 9, column = 0)
            self.labelOnes = Label(self, bg = "#fff", anchor = "w", relief =
"ridge", font=("Helvetica", 12))
            self.labelOnes.grid(row = 9, column = 1, sticky = "we")

            Label(self, text = "Quarters:", fg="purple", font=("Helvetica",
12)).grid(row = 10, column = 0)
            self.labelQuarters = Label(self, bg = "#fff", anchor = "w", relief =
"ridge", font=("Helvetica", 12))
            self.labelQuarters.grid(row = 10, column = 1, sticky = "we")

            Label(self, text = "Dimes:", fg="purple", font=("Helvetica",
12)).grid(row = 11, column = 0)
            self.labelDimes = Label(self, bg = "#fff", anchor = "w", relief =
"ridge", font=("Helvetica", 12))
            self.labelDimes.grid(row = 11, column = 1, sticky = "we")

            Label(self, text = "Nickels:", fg="purple", font=("Helvetica",
12)).grid(row = 12, column = 0)
            self.labelNickels = Label(self, bg = "#fff", anchor = "w", relief =
"ridge", font=("Helvetica", 12))
            self.labelNickels.grid(row = 12, column = 1, sticky = "we")

            Label(self, text = "Pennies:", fg="purple", font=("Helvetica",
12)).grid(row = 13, column = 0)
            self.labelPennies = Label(self, bg = "#fff", anchor = "w", relief =
"ridge", font=("Helvetica", 12))
            self.labelPennies.grid(row = 13, column = 1, sticky = "we")



      def calcChange(self): #Method for determining change
            price = float(self.textPriceItem.get()) #Obtains information from user
- price as a float value
            tendered = float(self.textMoneyGiven.get()) #Obtains information from
user = money tendered as a float value
            change = tendered - price #Formula used to find change
            penniesChange = (tendered - price)*100 # Change is converted to pennies

            twenties = 2000
            twentiesChange =  penniesChange % twenties
            newchange1 = (penniesChange - twentiesChange) // twenties

            tens = 1000
            tensChange = twentiesChange % tens
            newchange2 = (twentiesChange - tensChange) // tens

            fives = 500
            fivesChange = tensChange % fives
            newchange3 = (tensChange - fivesChange) // fives
            ones = 100
            onesChange = fivesChange % ones
            newchange4 = (fivesChange - onesChange) // ones

            quarters = 25
            quartersChange = onesChange % quarters
            newchange5 = (onesChange - quartersChange) // quarters

            dimes = 10
            dimesChange = quartersChange % dimes
            newchange6 = (quartersChange - dimesChange) // dimes

            nickels = 5
            nickelsChange = dimesChange % nickels
            newchange7 = (dimesChange - nickelsChange) // nickels

            pennies = 1
            penniesChange = nickelsChange % pennies
            newchange8 = (nickelsChange - penniesChange) // pennies


                #Output for each label
            self.labelChange["text"] = "%.2f" % change
            self.labelTwenties["text"] = newchange1
            self.labelTens["text"] = newchange2
            self.labelFives["text"] = newchange3
            self.labelOnes["text"] = newchange4
            self.labelQuarters["text"] = newchange5
            self.labelDimes["text"] = newchange6
            self.labelNickels["text"] = newchange7
            self.labelPennies["text"] = newchange8







def main():   #Main function
      a = changeMaker()
      a.mainloop()


if __name__ == "__main__":
      main()
```