Project Report- Shravan Srinivasan

Project Specification- My project will be based on a website creation in python using flash. For further specification I will focus on writing details for my website about a topic called "Number Game". The purpose of this application is to educate people on how to do magic based on mathematics where you pick a random number between 10 to 99 and to guess what number they have. People always want something unique especially in magic of math.

The format will be divided into four sections. One is the part where you pick your number in text and about the formula to take note of. To get to the second one you must click the x button for the calculator. The second one is the app of the calculator of the number which is #no from 10 to 99 times 90 times 20 times 50 and finally divide by 11.25. ask them the result first. Then you secretly divide the first three numbers by 8 in your head and you get the result being their chosen number. The final sections will be about how I came into terms to making this trick.

Solution Design-

Vision:

Second to further enhance the process of magic using math.

Plan: Make an educational website to specify the discovery.

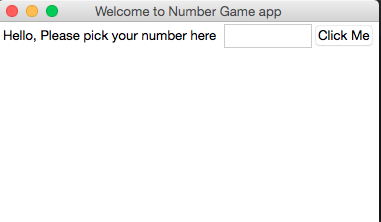
Design: Make an educational website using knowledge of Python programming.

Discovery: First found New ways to read someone’s mind just by picking a number

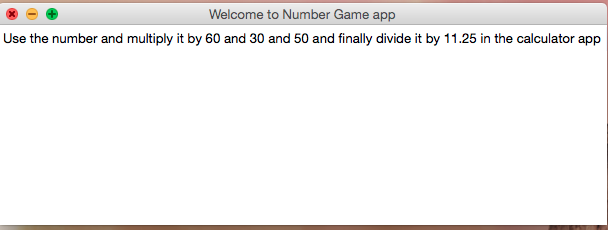
Result: Will be displayed in a web browser with 1.2.5.0 domain.

Develop: Use the design to further teach people new math magic tricks.

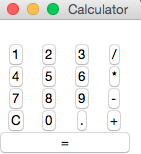
Discussion of what was implemented and how it works-

Fronthand evidence:

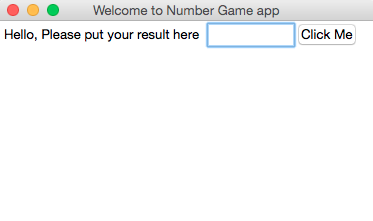
This will be the first screen that will appear.

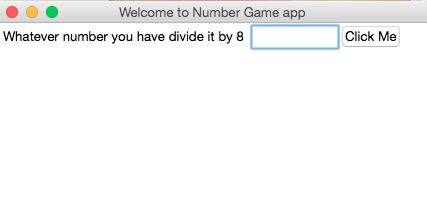


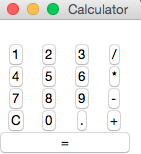
This is the second screen when to click on “click me”



Once you take note of the formula you have on the second screen click exit button and this app will appear for you to calculate.

Once your done click exit and input the result from your calculator app.

 From here you input the result and divide it by 8 by filling the following formula format in here to confirm before going back to the calculator app: Result of no picked/8.



You will be taken back to the calculator to divided the result by 8 one last time.

Backhand codes:

from tkinter import \* # I use tkinker code

import tkinter# This is where I install thinker

window = Tk() # this sets up the apps

window.title("Welcome to Number Game app") # This tells the name of the app

window.geometry('350x200')# The size of it

lbl = Label(window, text="Hello, Please pick your number here") # this is the words in the app

lbl.grid(column=0, row=0) #the empty box

txt = Entry(window,width=10)# The empty box size

txt.grid(column=1, row=0) # The length of the empty text.

def clicked():# a default for the function “clicked”

res = "Use the number and multiply it by 60 and 30 and 50 and finally divide it by 11.25 in the calculator app"

lbl.configure(text= res) #Make sure that the text would respond

btn = Button(window, text="Click Me", command=clicked)# This creates a button for the resp text

btn.grid(column=2, row=0) # this creates a blank space.

window.mainloop() #This is to make sure the app is working

root = tkinter.Tk()

root.title("Calculator")

expression = ""

# Create functions of the calculator

def add(value): # addition function

global expression

expression += value

label\_result.config(text=expression)

def clear():# clear function

global expression

expression = ""

label\_result.config(text=expression)

def calculate(): #calculation function

global expression

result = "" # I use the variable result

if expression != "": # I use if , try, and except statement

try:

result = eval(expression)

except:

result = "error"

expression = ""

label\_result.config(text=result)

expression = str(result)

# Create key bindings of the calculator

def key\_handler(event):

global expression

if event.keysym in ("1", "2", "3", "4", "5", "6", "7", "8", "9", "0"):

add(event.keysym)

elif event.keysym == "plus":

add("+")

elif event.keysym == "minus":

add("-")

elif event.keysym == "asterisk":

add("\*")

elif event.keysym == "slash":

add("/")

elif event.keysym in ("c", "C"):

clear()

elif event.keysym == "period":

add(".")

elif event.keysym in ("Return", "equal"):

calculate()

elif event.keysym == "BackSpace":

expression = expression[0:len(expression)-1]

label\_result.config(text=expression)

root.bind("<Key>", key\_handler)

label\_result = tkinter.Label(root, text="")

label\_result.grid(row=0, column=0, columnspan=4)

button\_1 = tkinter.Button(root, text="1", command=lambda: add("1"))

button\_1.grid(row=1, column=0)

button\_2 = tkinter.Button(root, text="2", command=lambda: add("2"))

button\_2.grid(row=1, column=1)

button\_3 = tkinter.Button(root, text="3", command=lambda: add("3"))

button\_3.grid(row=1, column=2)

button\_divide = tkinter.Button(root, text="/", command=lambda: add("/"))

button\_divide.grid(row=1, column=3)

button\_4 = tkinter.Button(root, text="4", command=lambda: add("4"))

button\_4.grid(row=2, column=0)

button\_5 = tkinter.Button(root, text="5", command=lambda: add("5"))

button\_5.grid(row=2, column=1)

button\_6 = tkinter.Button(root, text="6", command=lambda: add("6"))

button\_6.grid(row=2, column=2)

button\_multiply = tkinter.Button(root, text="\*", command=lambda: add("\*"))

button\_multiply.grid(row=2, column=3)

button\_7 = tkinter.Button(root, text="7", command=lambda: add("7"))

button\_7.grid(row=3, column=0)

button\_8 = tkinter.Button(root, text="8", command=lambda: add("8"))

button\_8.grid(row=3, column=1)

button\_9 = tkinter.Button(root, text="9", command=lambda: add("9"))

button\_9.grid(row=3, column=2)

button\_subtract = tkinter.Button(root, text="-", command=lambda: add("-"))

button\_subtract.grid(row=3, column=3)

button\_clear = tkinter.Button(root, text="C", command=lambda: clear())

button\_clear.grid(row=4, column=0)

button\_0 = tkinter.Button(root, text="0", command=lambda: add("0"))

button\_0.grid(row=4, column=1)

button\_dot = tkinter.Button(root, text=".", command=lambda: add("."))

button\_dot.grid(row=4, column=2)

button\_add = tkinter.Button(root, text="+", command=lambda: add("+"))

button\_add.grid(row=4, column=3)

button\_equals = tkinter.Button(root, text="=", width=16, command=lambda: calculate())

button\_equals.grid(row=5, column=0, columnspan=4)

root.mainloop()

window = Tk()

window.title("Welcome to Number Game app")

window.geometry('350x200')

lbl = Label(window, text="Hello, Please put your result here")

lbl.grid(column=0, row=0)

txt = Entry(window,width=10)

txt.grid(column=1, row=0)

def clicked():

res = "Whatever number you have divide it by 8"

lbl.configure(text= res)

btn = Button(window, text="Click Me", command=clicked)

btn.grid(column=2, row=0)

root = tkinter.Tk()

root.title("Calculator")

expression = ""

def add(value):

global expression

expression += value

label\_result.config(text=expression)

def clear():

global expression

expression = ""

label\_result.config(text=expression)

def calculate():

global expression

result = ""

if expression != "":

try:

result = eval(expression)

except:

result = "error"

expression = ""

label\_result.config(text=result)

expression = str(result)

def key\_handler(event):# these are the formatting of what the expression button would look like.

global expression

if event.keysym in ("1", "2", "3", "4", "5", "6", "7", "8", "9", "0"):

add(event.keysym)

elif event.keysym == "plus":

add("+")

elif event.keysym == "minus":

add("-")

elif event.keysym == "asterisk":

add("\*")

elif event.keysym == "slash":

add("/")

elif event.keysym in ("c", "C"):

clear()

elif event.keysym == "period":

add(".")

elif event.keysym in ("Return", "equal"):

calculate()

elif event.keysym == "BackSpace":

expression = expression[0:len(expression)-1]

label\_result.config(text=expression)

root.bind("<Key>", key\_handler)

# Create GUI

label\_result = tkinter.Label(root, text="")

label\_result.grid(row=0, column=0, columnspan=4)

# These are button on the gui calculator

button\_1 = tkinter.Button(root, text="1", command=lambda: add("1"))

button\_1.grid(row=1, column=0)

button\_2 = tkinter.Button(root, text="2", command=lambda: add("2"))

button\_2.grid(row=1, column=1)

button\_3 = tkinter.Button(root, text="3", command=lambda: add("3"))

button\_3.grid(row=1, column=2)

button\_divide = tkinter.Button(root, text="/", command=lambda: add("/"))

button\_divide.grid(row=1, column=3)

button\_4 = tkinter.Button(root, text="4", command=lambda: add("4"))

button\_4.grid(row=2, column=0)

button\_5 = tkinter.Button(root, text="5", command=lambda: add("5"))

button\_5.grid(row=2, column=1)

button\_6 = tkinter.Button(root, text="6", command=lambda: add("6"))

button\_6.grid(row=2, column=2)

button\_multiply = tkinter.Button(root, text="\*", command=lambda: add("\*"))

button\_multiply.grid(row=2, column=3)

button\_7 = tkinter.Button(root, text="7", command=lambda: add("7"))

button\_7.grid(row=3, column=0)

button\_8 = tkinter.Button(root, text="8", command=lambda: add("8"))

button\_8.grid(row=3, column=1)

button\_9 = tkinter.Button(root, text="9", command=lambda: add("9"))

button\_9.grid(row=3, column=2)

button\_subtract = tkinter.Button(root, text="-", command=lambda: add("-"))

button\_subtract.grid(row=3, column=3)

button\_clear = tkinter.Button(root, text="C", command=lambda: clear())

button\_clear.grid(row=4, column=0)

button\_0 = tkinter.Button(root, text="0", command=lambda: add("0"))

button\_0.grid(row=4, column=1)

button\_dot = tkinter.Button(root, text=".", command=lambda: add("."))

button\_dot.grid(row=4, column=2)

button\_add = tkinter.Button(root, text="+", command=lambda: add("+"))

button\_add.grid(row=4, column=3)

button\_equals = tkinter.Button(root, text="=", width=16, command=lambda: calculate())

button\_equals.grid(row=5, column=0, columnspan=4)

root.mainloop() # Run the application

References:

1. <https://flask.palletsprojects.com/en/1.1.x/>
2. <https://www.fullstackpython.com/flask.html>
3. <https://realpython.com/tutorials/flask/>
4. <https://code-maven.com/hello-world-with-flask-and-python>
5. <https://www.freecodecamp.org/news/how-to-build-a-web-application-using-flask-and-deploy-it-to-the-cloud-3551c985e492/>
6. <https://www.thoughtco.com/math-tricks-that-will-blow-your-mind-4154742>
7. <https://www.learn-with-math-games.com/math-number-tricks.html>