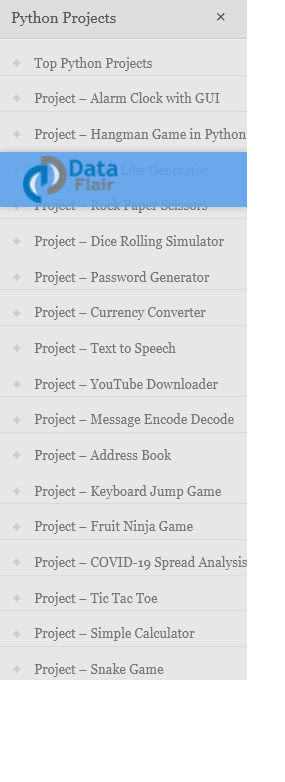
*“We connect with code and learn from it and change with it. I code for a chance to turn into a change.”* -Dr. Afonso Pena (AVVA4LLLL)

This document is the main Read Me file but each of the 10 projects named respectively have their own ReadMe\_ in their folder: GUI Alarm Clock, Password Generator, Mad-libs Generator, Dice Rolling Generator, Currency Converter, Encode and Decode Message, Address Book, A Simple Calculator, Snake Game. The projects were taken from Data Flair, Python Projects Folder, click the plus sign, then pick and choose what you want to build and learn from.



*Picture 1.0 - Project Folder*



*Picture 1.01 - Project Varieties*

This projects were developed using Python 3.9 as the programming language, PyCharm 2022.3.2 (Community Edition) as the main IDE and Thonny as a code tester, because it is simple to use. The version number of each project means: first digit is for the years of existence; second digit is used if any permanent update was made; the third digit is used as the project ID. Please take note that this is not a tutorial.

Developing them felt like “my back is killing me...aahhh...” but is was worth it, because I got to do them, now I can share my mistakes, future ideas for them and be proud for aiming at 10, and get the 10 I aimed at.

For no good reason what does Python mean to you?

For me it means that I can do silly things with it, like solving problems and making cool games or write something that does not make sense but runs without an error. I confused about the spaces (indentation), when to use concatenation and a comma, python is fun.

I made a forloop script that needs fixing, it’s silly but please solve it if you can or if you have the time to waste, you just need to remove/include a *break, a “\n”* or a *\n.*

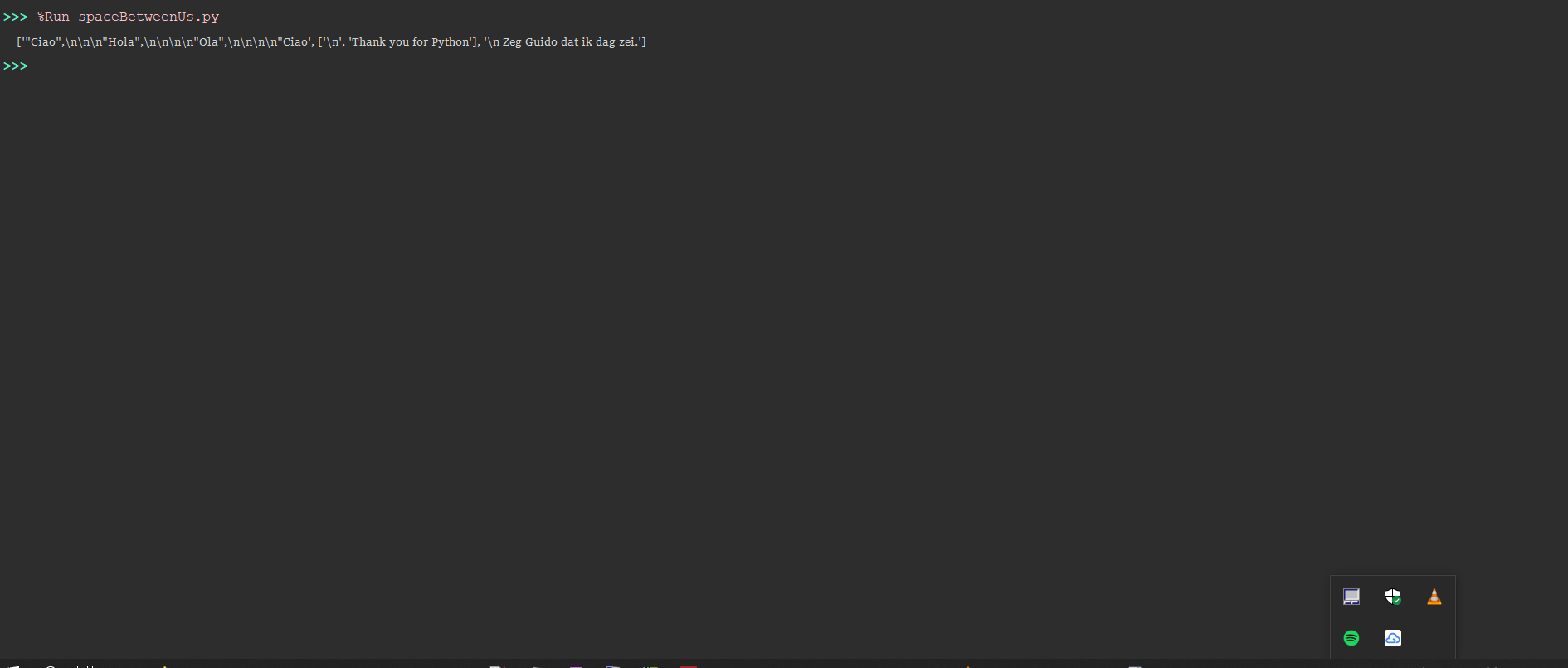
Please answer riddle.py if you want. Else click.

**No *\n* between us:**

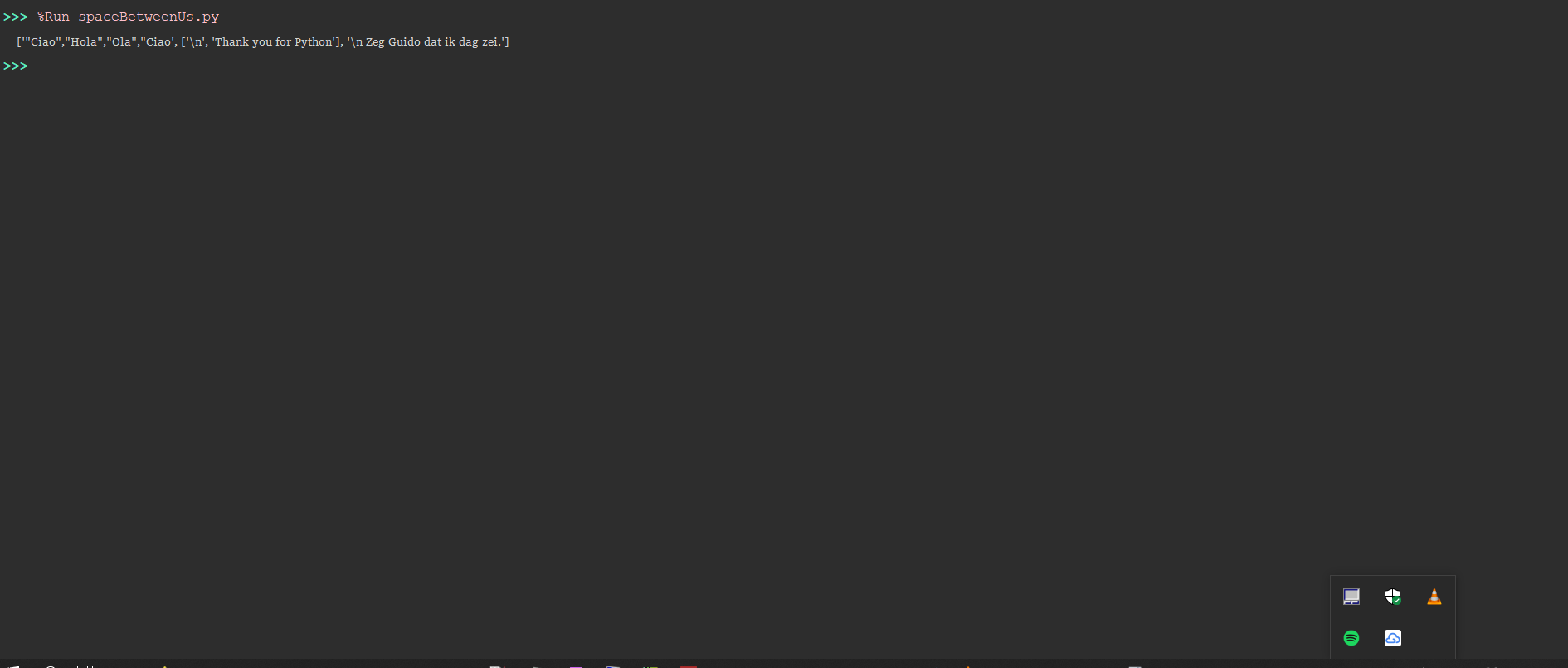
Have a *break* before you *\n* , to run you need an IDE and a f*orloop* but you also need to *break* the number of “\n’s” present in the sentence. Please dont *break* your system. If your computer crashes blame it on the *break* not given not on the “\n” taken from the sentence.

**NOTE: DONT RUN WITHOUT TRYING IT IN YOUR HEAD. BEACUSE THE OUT COME IS A PERFECT** *break.*

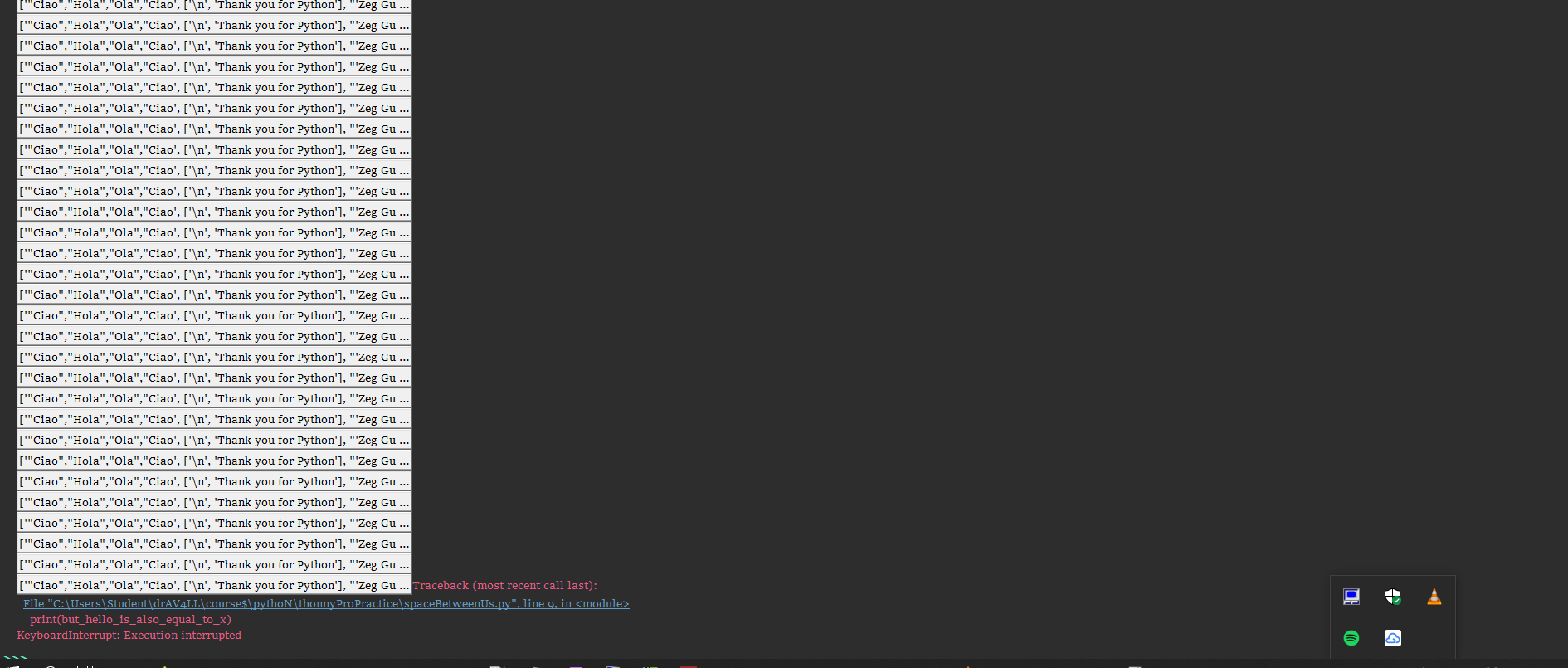
This is **NOT** what we want:



This in **NOT** what we want:



This defenetally is what we **DO NOT** want, so please *break* the *forloop*:



If you want to*break, reduce “\n” than* \nfor real**:**

The outcome needs to be without any breaks “\n”.

Use the code below to start the riddle:

# there is something missing

Guido = ["\n", "Thank you for Python"]

hallo = [Guido, "'Zeg Guido dat ik dag zei.'"]

hello = ["""

"Ciao",

"Hola",

"Ola",

"Ciao"""]

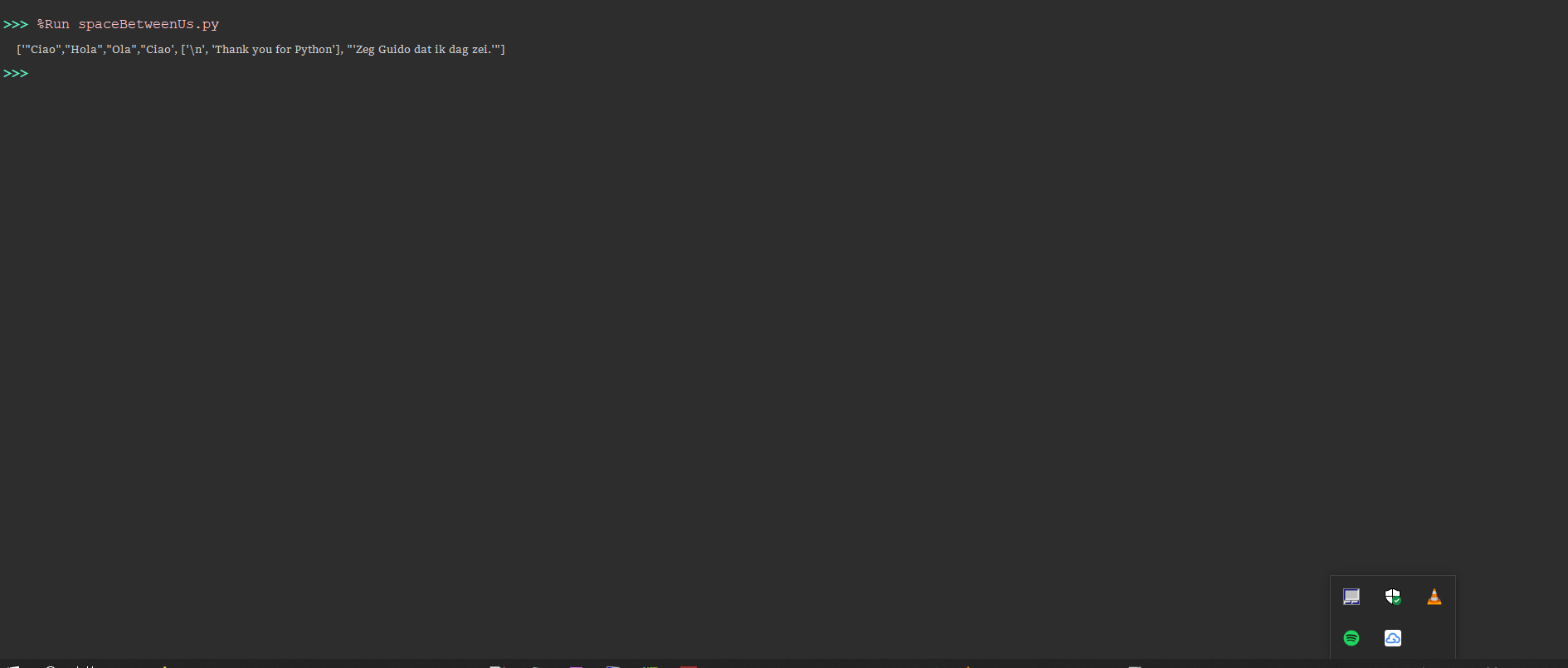
for x in hello:

hello += hallo

but\_hello\_is\_also\_equal\_to\_x = hello

print(but\_hello\_is\_also\_equal\_to\_x)

If you want the correct one try at least to reach to this output:

******

**To “\n” a *forloop* we must***\n* ***with*** *break****.***

This is the answer. A “\n” was left because now we know how to *break* a *forloop* with the right *\n*.

Guido = ["\n", "Thank you for Python"]

hallo = [Guido, "'Zeg Guido dat ik dag zei.'"]

hello = [""""Ciao","Hola","Ola","Ciao"""]

for x in hello:

hello += hallo

but\_hello\_is\_also\_equal\_to\_x = hello

print(but\_hello\_is\_also\_equal\_to\_x)

break

BELOW WE HAVE PROJECTS DEVELOPED IN PYTHON, SO FAR SO GOOD DATE RATE:

**Note:** Make sure to have your libraries installed before starting a new project, this projects are for practising rather than copy and paste. Meaning that the projects are due to change and that typing is hard but very efficient for practising and have the hands on the mud.

Check if a certain library is already installed in your machine:

Eg: pip3 show tk

**# \*\*GUI Alarm Clock v 0.1.0\*\* -(29-12-2022)**

The first project I developed is a GUI Alarm clock that can take user input and rings at the desired time but I failed to play put sound and make it ring, we have all necessary features of an alarm clock but we dont have sound. It is on version 0.1.01 with the size of 3.07 Kb.

**Utility or why Develop:** This project was developed to motivate me to be punctual with my study time, it does not play sound but it sure introduced me to a place where possibilities appear all the time.

**## \*\*Libraries Used:\*\*** TKinter, datatime, time and windsound.

### Program instructions:

"""

GIT: @drafonsopena

+ This objective of this project is to create an Alarm Clock using Python.

| Group:

+-+---------------- 1 ----------------

| Prerequisites:

| Install libraries (eg: pip3 install tk)

| Basic Python skills

| Use of a virtual environment

+---------------- 2 ----------------

| Project File Structure:

| Import all the needed libraries/modules

| Use 'while' loop which takes argument of the time

| Create a dialog box for user input

+---------------- 3 ----------------

| Libraries for the GUI Alarm Clock:

| From tkinter import \*

| Import datetime

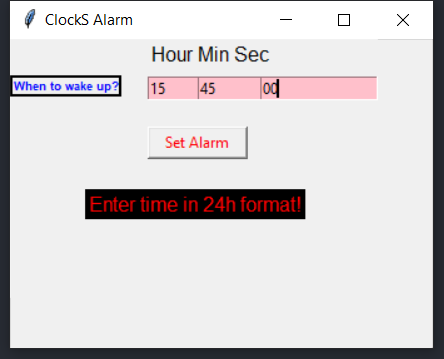
| Import time

| Import winsound

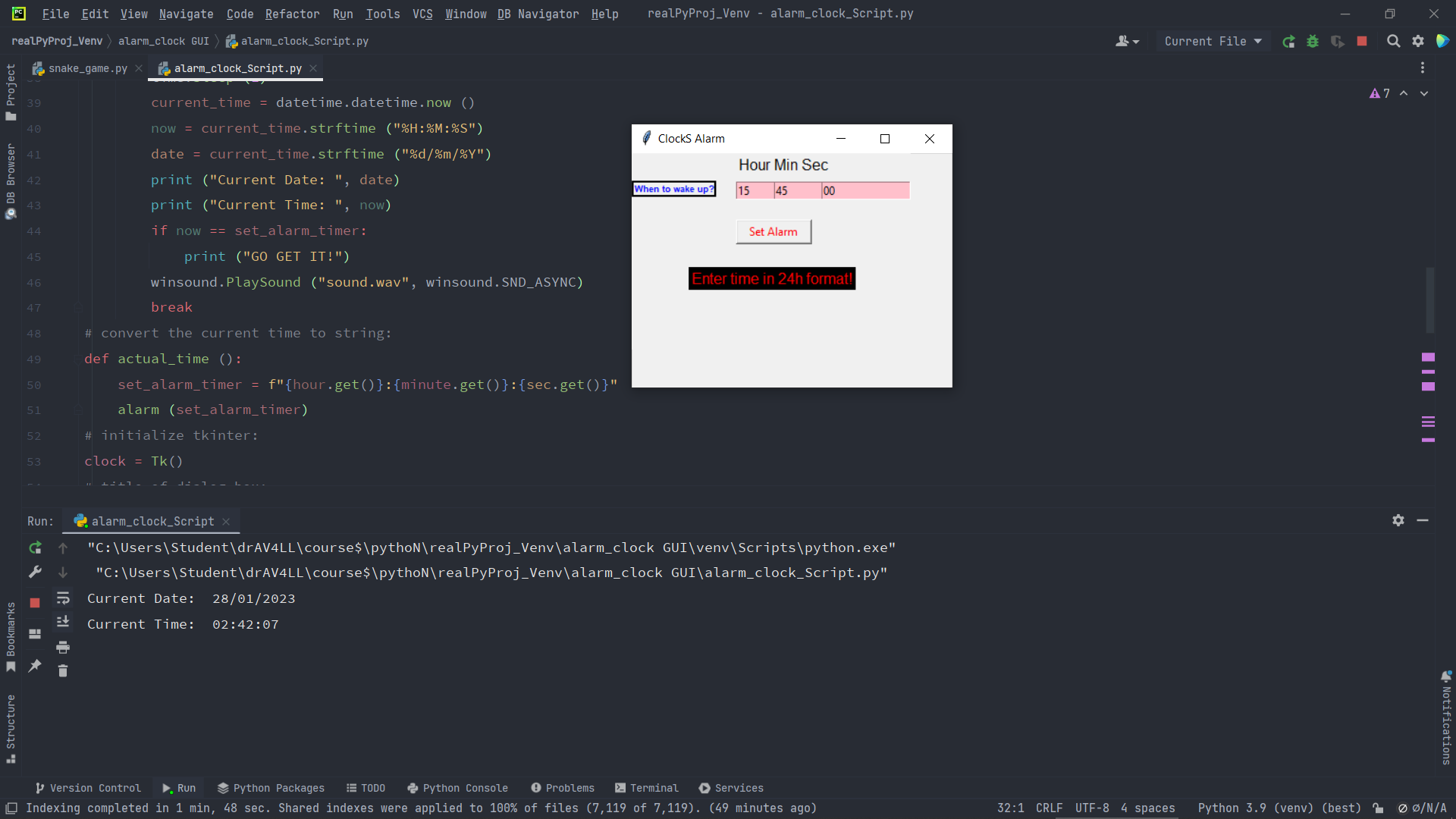
+------------------------------------

"""

**## \*\*Future Changes:\*\*** In a later version we will have sound, pop-up reminders, probably a different front-end and back-end too.



This how the GUI Alarm Clock looks like. And I named ir “ClockS Alarm”.



Last time I run it, outputs this but I dont understand how the countdown of the clock works! For the waking time the user inserts!

**# \*\*Password Generator\*\* -(31-12-2022)**

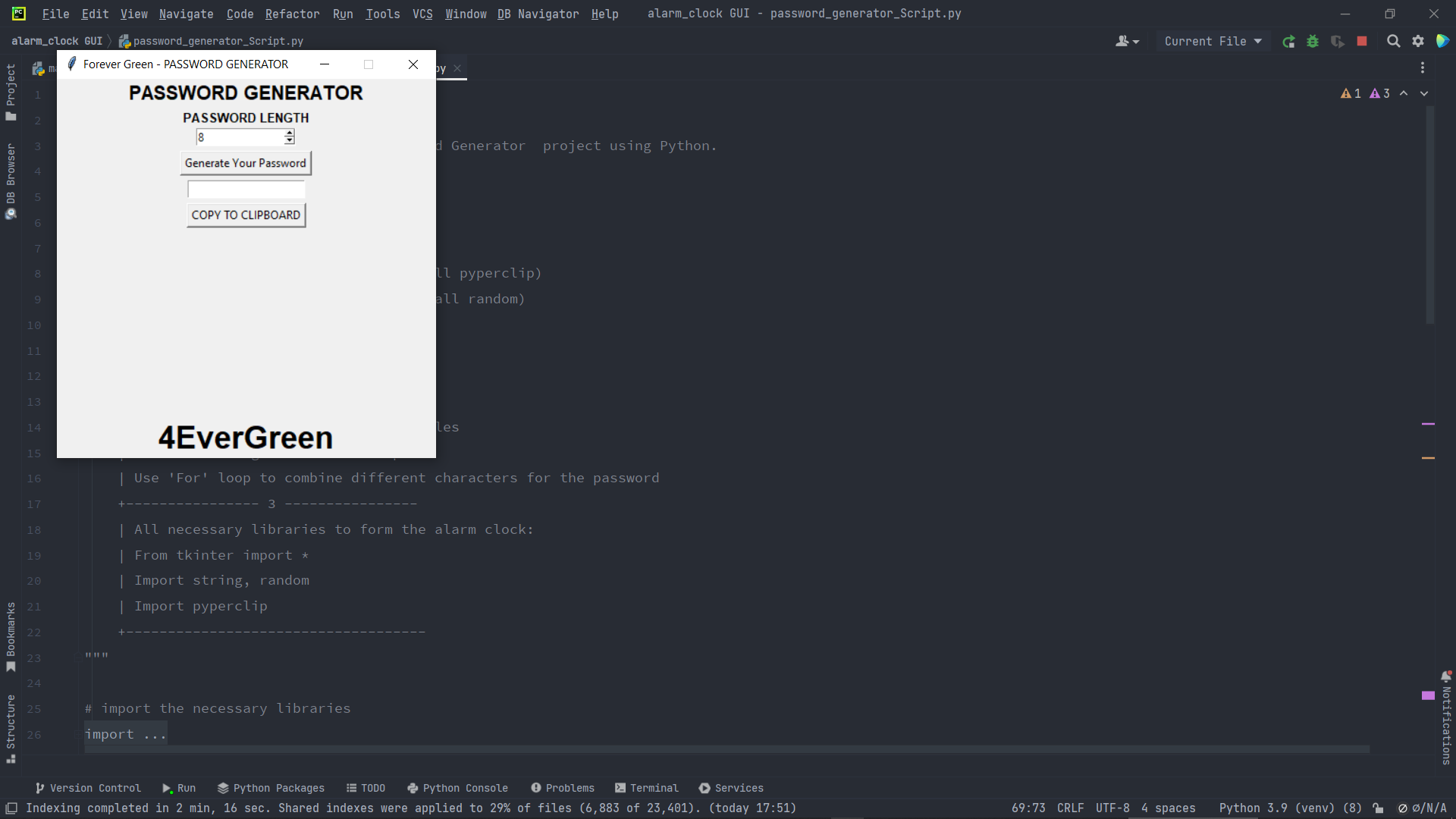
The second project I developed is a Password Generator that can create a random or customized password for the user; to allow the user to copy the generated password with a click; and choose the length of the password. It was hard to understand how to increase the number of characters, currently it goes from eight to thirty two real quick. It is on version 0.1.02 with the size of 2.92 Kb.

**Utility or why Develop:** This project was developed to keep my ‘*password1*’ strange and safer. This project can be used for security measures and protection.

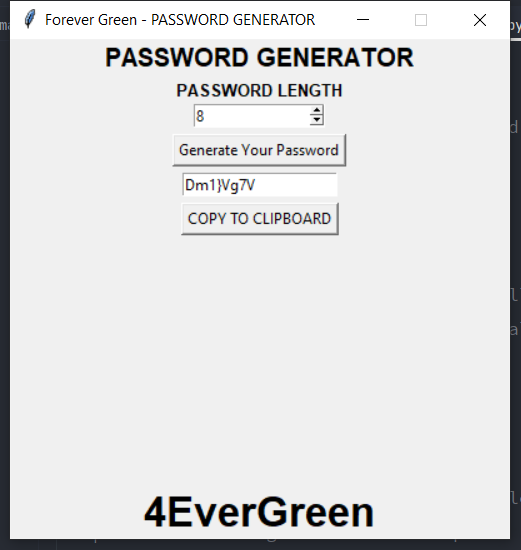
**Libraries and Prerequisites:** TKinter, datatime, time and windsound. Make sure to have your libraries installed before starting a new project, this document is more practical than copy and paste. Meaning that the projects are due to change and that typing is hard but very efficient for practising and have the hands on the mud. Worry not the source is in the folder GUI Alarm Clock and so on.

"""  
 GIT: @drafonsopena  
 + The objective of this project is to create a Password Generator using Python.  
 | Group:  
 +-+---------------- 1 ----------------  
 | Prerequisites:  
 | Install libraries (eg: pip3 install tk)  
 | Basic Python skills  
 | Use a virtual environment  
 +---------------- 2 ----------------  
 | Project File Structure:  
 | Import all the needed libraries/modules  
 | Create a dialog box for user input  
 | Use 'For' loop to combine different characters for the password  
 +---------------- 3 ----------------  
 | All necessary libraries for the Password Generator:  
 | From tkinter import \*  
 | Import string, random  
 | Import pyperclip  
 +------------------------------------  
"""

\*\*Screen-shot of the Password Generator: initial menu\*\*



\*\*Screen-shot of the Password Generator: with result.\*\*



**## \*\*Future Changes:\*\*** In a later version I am thinking to integrate it with the Encode and Decode Message, as in we encode a message but we can only decode/read it if we have the right password. Maybe a bot that helps you encode or decode (translate) morse messages or binary with a password generator. Any other idea on how to use it for other things? If so, please share.

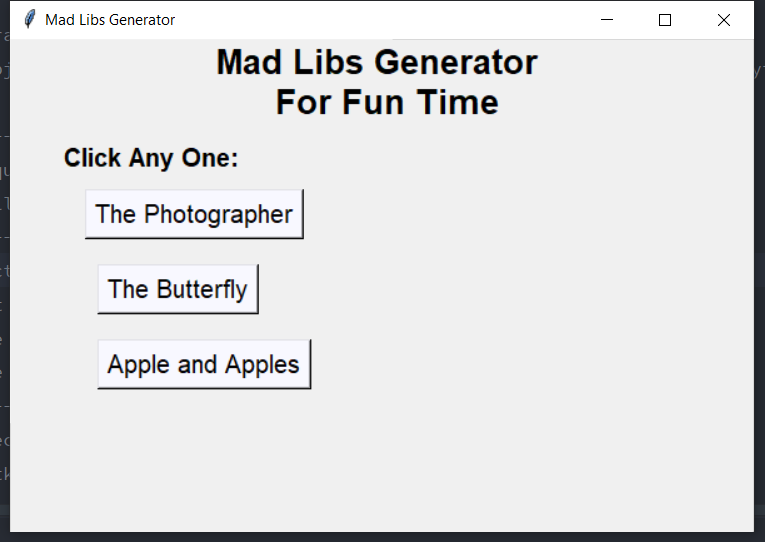
**Madlibs Generator (09-01-2023)**

The third project I developed is a Madlibs Generator that can take user input and add that input to a story at a random word sequencing, it also allows the user to select one of the three stories. It was hard to understand write all these lines and understand how concatenation works. It is on version 0.1.03 with the size of 4.58 Kb.

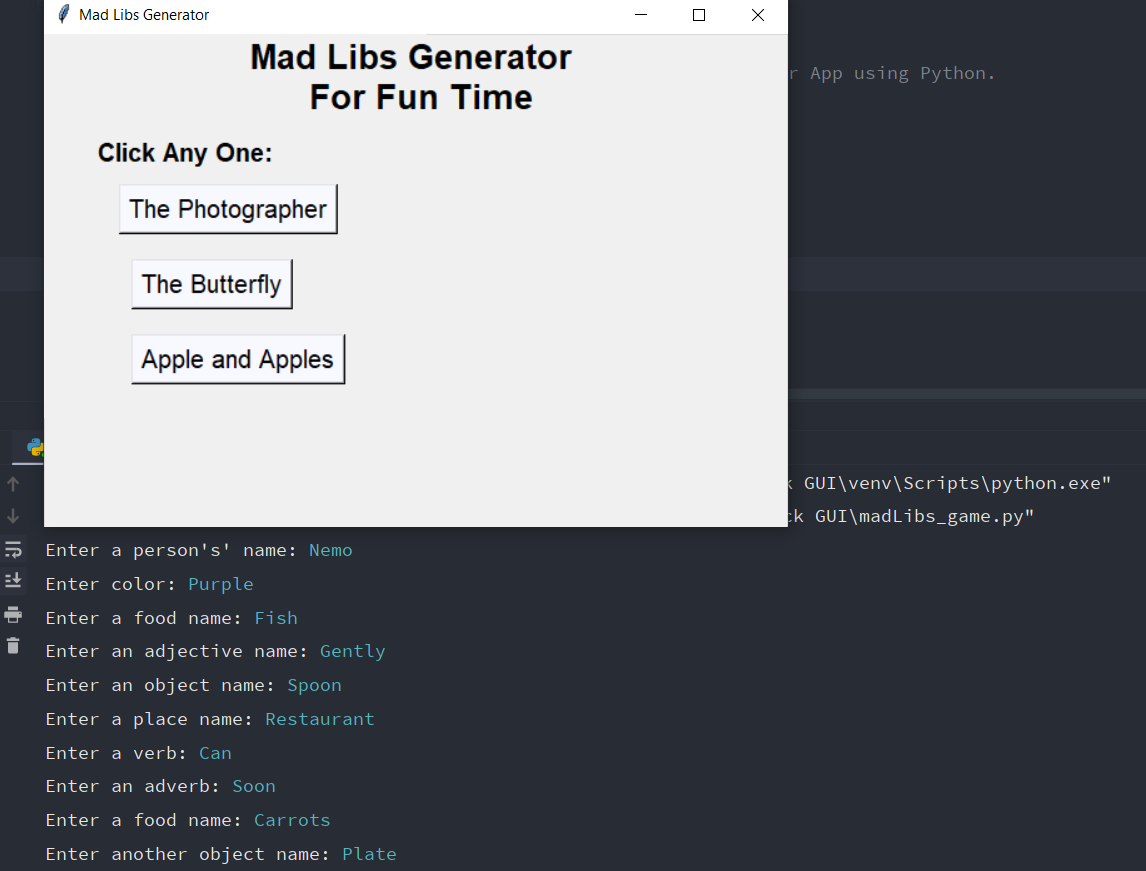
**Utility or why Develop:** This project was developed to keep my password1 strange and safer. This project can be used for security measures and protection.

**Libraries and Prerequisites:** For this project we only need TKinter.

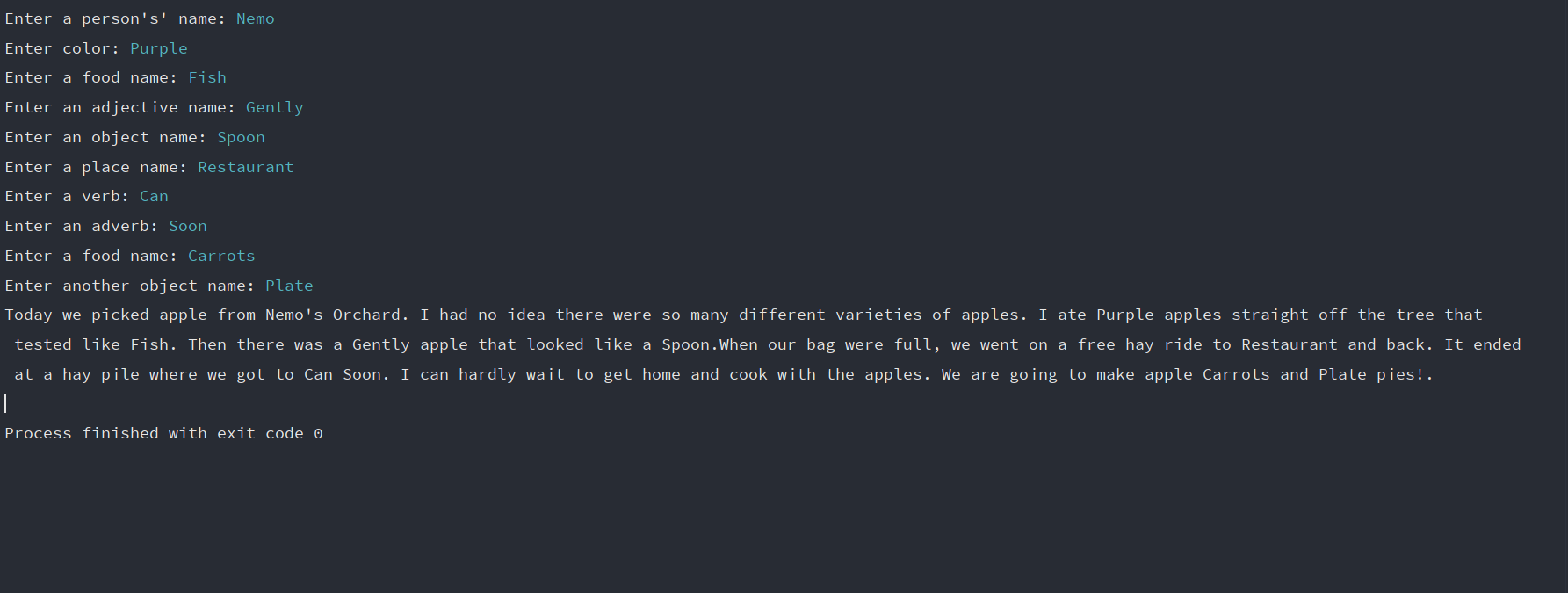
"""  
 GIT: @drafonsopena  
 + The objective of this project is to create a Madlibs Generator using Python.  
 | Group:  
 +-+---------------- 1 ----------------  
 | Prerequisites:  
 | Install libraries (eg: pip3 install tk)  
 +---------------- 2 ----------------  
 | Project File Structure:  
 | Import all the needed libraries/modules  
 | Create display window  
 | Create functions and input variables  
 +---------------- 3 ----------------  
 | All necessary libraries for the Madlibs Generator:  
 | From tkinter import \*  
 +------------------------------------  
"""



Madlibs main window



Madlibs story selection (Apple and Apples)



The output of the story after user input

**## \*\*Future Changes:\*\*** In a later version I am thinking to add new stories but this time extracted from a PDF file. Maybe make a whole story book based on the Madlibs generated from the app.

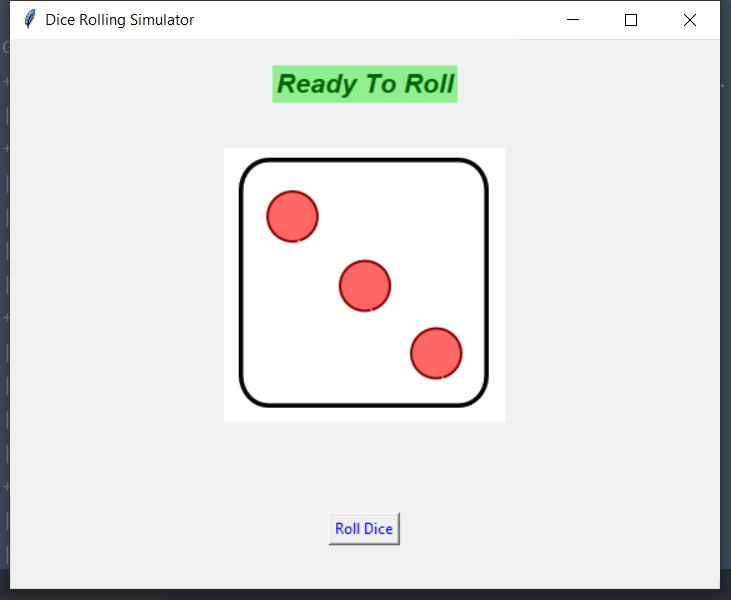
**Die Rolling Simulator (10-01-2023)**

The forth project I developed is a Die Rolling Simulator that can display random dice numbers using images. This project is quite simple. It is on version 0.1.04 with the size of 2.34 Kb.

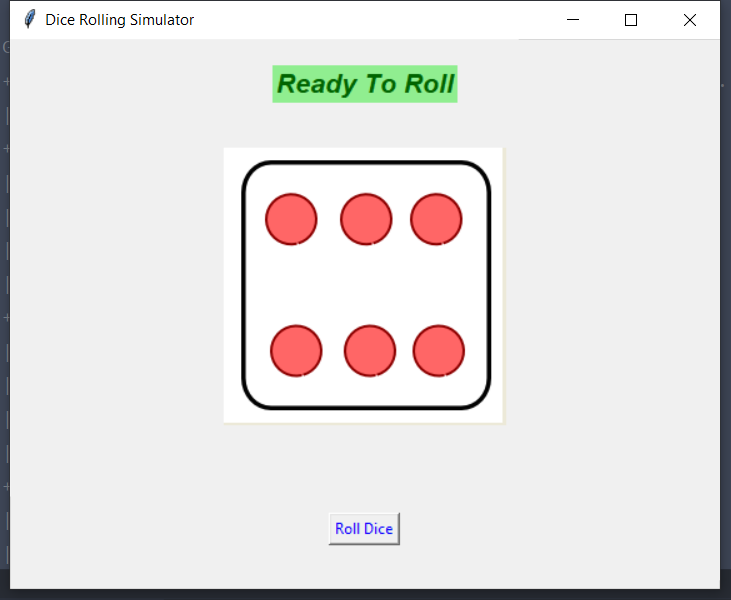
**Utility or why Develop:** This project was developed because my die are always getting lost. This project came to substitute the my previous die, and this way I cant lose my die so easily.

"""  
 GIT: @drafonsopena  
 + The objective is to create a Die Rolling Simulator using Python.  
 | Group:  
 +-+---------------- 1 ----------------  
 | Prerequisites:  
 | Install libraries (eg: pip3 install tk)  
 | Basic Python skills  
 | Use a virtual environment  
 +---------------- 2 ----------------  
 | Project File Structure:  
 | Import all the needed libraries/modules  
 | Form the images representing the dice parts  
 | Create labels, functions and buttons  
 +---------------- 3 ----------------  
 | All necessary libraries for the Die Rolling Simulator:  
 | From tkinter import \*  
 | from PIL import Image, ImageTk  
 | import random  
 +------------------------------------  
"""

**Libraries and Prerequisites:** For the Die Rolling Simulator we need TKintern, PIL is the python imaging library and random. The images used are in the project folder.



First Output



Second Output

**## \*\*Future Changes:\*\*** In a later version I am thinking about changing the dice aspect, instead of images we can use a 2D or 3D display.

**Currency Converter (13-01-2023)**

The fifth project I developed is a Currency Converter that can provide the value of all major currencies in the globe (no cripto currencies yet) with the aid of an API (https://api.exchangerate-api.com/v4/latest/USD). This project is quite useful and fun to make, you can feel internationally rich. It is on version 0.1.05 with the size of 5.92 Kb.

**Utility or why Develop:** This project was developed because before travelling it is a must to know about the customs and what monetary system is being used locally. True we can search with Google of any new AI search engine but why not make and have your own?

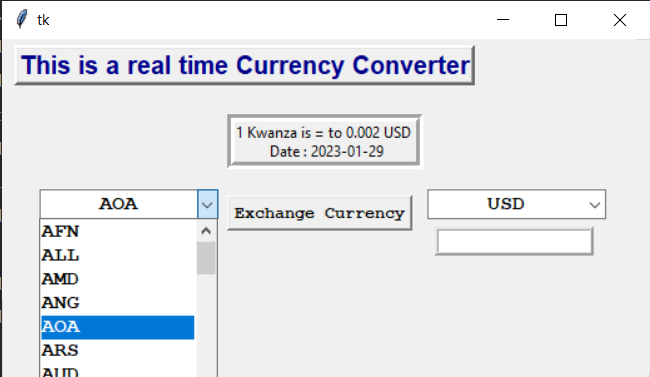
"""  
 GIT: @drafonsopena  
 + The objective is to create a Currency Converter using Python.  
 | Group:  
 +-+---------------- 1 ----------------  
 | Prerequisites:  
 | Install libraries (eg: pip3 install tk)  
 | Basic Python skills  
 | Use a virtual environment  
 | API (url = 'https://api.exchangerate-api.com/v4/latest/USD')  
 +---------------- 2 ----------------  
 | Project File Structure:  
 | Import all the needed libraries/modules  
 | Form the images representing the dice parts  
 | Create labels, functions and buttons  
 +---------------- 3 ----------------  
 | All necessary libraries for the Currency Converter:

| import re  
 | From tkinter import \*  
 | import tkinter as tk  
 | from tkinter import ttk  
 | import requests  
 +------------------------------------  
"""

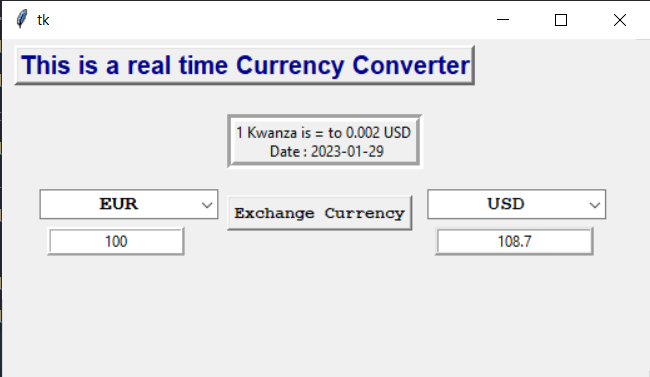
**Libraries and Prerequisites:** For the Currency Converter we used TKinter, re which provides regular expression matching and requests. The exchange rate is done at real time with the help of an API (https://api.exchangerate-api.com/v4/latest/USD)

****

Currency Converter: Initial Window

****

Currency Converter: currency menu

****

Currency Converter: 100 Euro is equal to 108.7 USD

**## \*\*Future Changes:\*\*** In a later version I am thinking to add cripto exchange-rate or integrate a calculator to it. Like this we can exchange the currency and make our diner budget within one application.

**Text-to-Speech Converter (16-01-2023)**

The sixth project I developed is a Text to Speech that can convert any English text to speech/sound. After the text is converted to speech we save a mp3 file (the copy of the text we converted). It is on version 0.1.06 with the size of 2.7 Kb.

**Utility or why Develop:** This project was complicated to make because currently it does not accept large sums of text and had some difficulties finding the recommended modules, that aside I say that it is useful to teach people how to read or write.

"""  
 GIT: @drafonsopena  
 + The objective is to create a Text-to-Speech Converter using Python.  
 | Group:  
 +-+---------------- 1 ----------------  
 | Prerequisites:  
 | Install libraries (eg: pip3 install tk)  
 | Basic Python skills  
 | Use a virtual environment  
 +---------------- 2 ----------------  
 | Project File Structure:  
 | Import all the needed libraries/modules  
 | Create display window  
 | Create labels, functions and buttons  
 +---------------- 3 ----------------  
 | All necessary libraries for the Text-to-Speech Converter:  
 | import tkinter as tk  
 | from gtts import gTTS  
 | from playsound import playsound  
 +------------------------------------  
"""

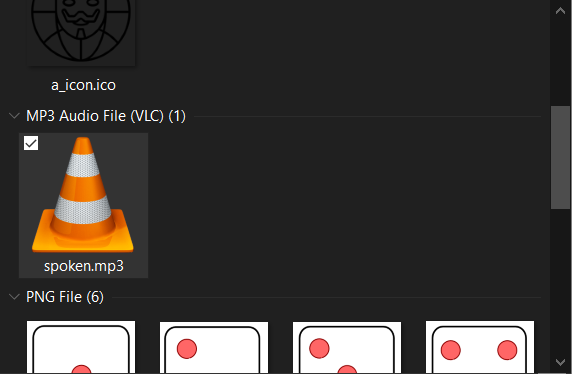
**Libraries and Prerequisites:** For the Text to Speech we used TKinter, gtts a library made by Google to aid in the text to speech conversion, and playsound. This project had to be developed used python 3.10 because 3.9 was giving errors regarding the library gtts, and the funny thing is that while the program is running we can not convert another text to speech (we have to close it and run it again) because our speech is saved as an mp3 file (we cannot have files with the same name but we can override it), so make sure to open, convert, close, open again and convert.



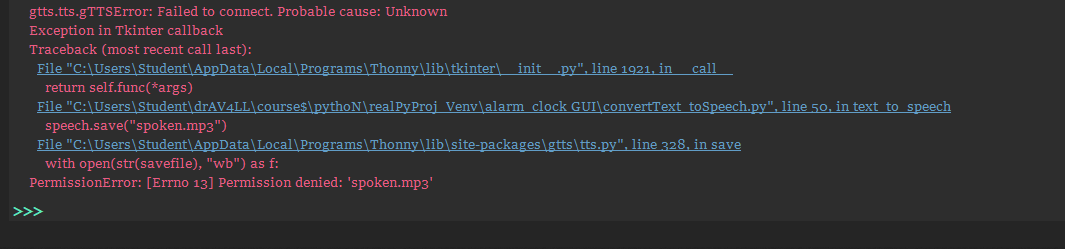
Text to Speech Converter: Initial Window



Text to Speech: reading user input



Text to Speech: Output audio in project’s folder



Text\_to\_Speech: the errors we get during the runtime of the program

**Error explanation:**

gtts.tts.gTTSError: Failed to connect. Probable cause: Unknown

This means that it failed to connect to the internet, this is the probable cause.

speech.save("spoken.mp3")

This is the name of the file each time we press the “Play” button.

with open(str(savefile), "wb") as f:

PermissionError: [Errno 13] Permission denied: 'spoken.mp3'

This is the error we get when we try to convert a text again without closing the program.

**## \*\*Future Changes:\*\*** In a later version I am thinking like “PDF PDF PDF READER”, which is a good idea specially when it comes to books and other long articles. But before the PDF idea to happen we need to increase the number of characters the program can scan through; allow the user to convert text to speech without having to close the program; add more languages... then “PDF PDF PDF READER”.

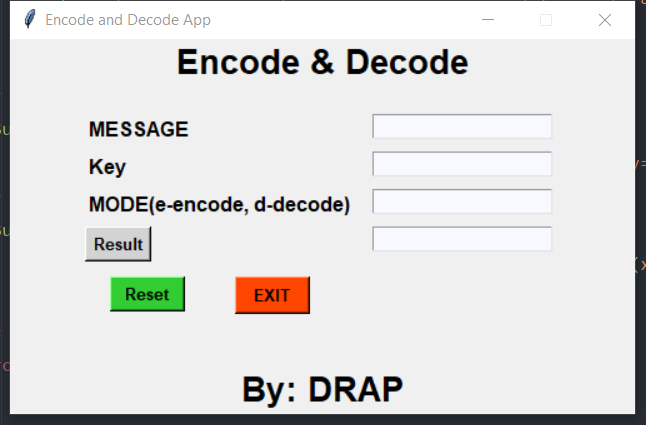
**Encode and Decode Message (18-01-2023)**

The seventh project I developed is a Encode and Decode Message that can encode any English text to random characters, I thought it would be easy but it was not, because the program is only encoding and not decoding. It is on version 0.1.07 with the size of 3.75 Kb.

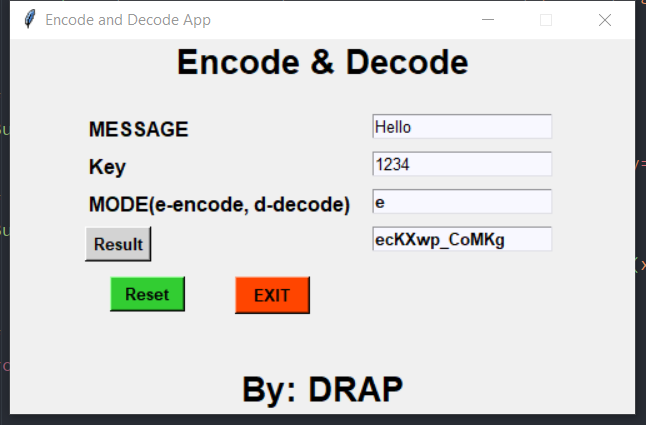
**Utility or why Develop:** This program can be used to encode and decode messages using base64 data encoding. The user has to enter a message and choose between encode (e) and decode (d); the same passkey must be used in the encoding and decoding of the message

"""  
 GIT: @drafonsopena  
 + The objective is to create a base64 Encode and Decode Message using Python.  
 | Group:  
 +-+---------------- 1 ----------------  
 | Prerequisites:  
 | Install libraries (eg: pip3 install tk)  
 | Basic Python skills  
 | Use a virtual environment  
 +---------------- 2 ----------------  
 | Project File Structure:  
 | Import all the needed libraries/modules  
 | Create display window  
 | Create labels, functions and buttons  
 +---------------- 3 ----------------  
 | All necessary libraries for the Encode and Decode Message:  
 | import tkinter as tk  
 | import base64  
 +------------------------------------  
"""

**Libraries and Prerequisites:** For the Encode and Decode Message program we used TKinter, and base64 a library used to encode data. This project was complicated to make because currently it does not decode the messages after encoding it nor it recognises the passkey (it basically does nothing if I may say), that aside I still say that it is useful.



Encode and Decode Message: Initial screen



Encode and Decode Message: Message being encoded

**NOTE:** It does not decode as of yet.

**## \*\*Future Changes:\*\*** This program is mainly being developed to create a chat app and change the encoding level of the messages later on.

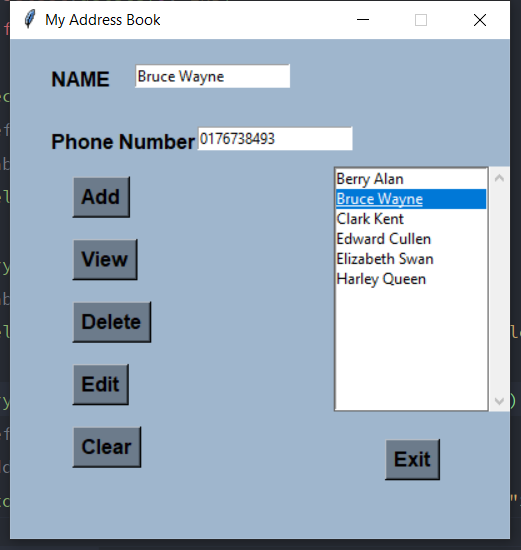
**Address Book (20-01-2023)**

The eighth project I developed is an Address Book that allows the user to add, edit and delete a contact. This project does not make the use of a database as of yet so the contacts are volatile. It is on version 0.1.08 with the size of 3.72 Kb.

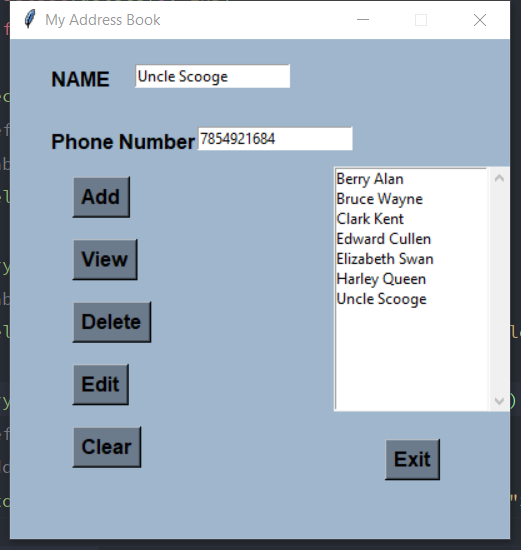
**Utility or why Develop:** This aim of this project was to develop an Address Book using python, and we pretty much did it. This Address Book as all necessary functions operational.

"""  
 GIT: @drafonsopena  
 + The objective is to create an Address Book using Python.  
 | Group:  
 +-+---------------- 1 ----------------  
 | Prerequisites:  
 | Install libraries (eg: pip3 install tk)  
 | Basic Python skills  
 | Use a virtual environment  
 +---------------- 2 ----------------  
 | Project File Structure:  
 | Import all the needed libraries/modules  
 | Create display window  
 | Create labels, functions and buttons  
 +---------------- 3 ----------------  
 | All necessary libraries for the Address Book:  
 | from tkinter import \*  
 +------------------------------------  
"""

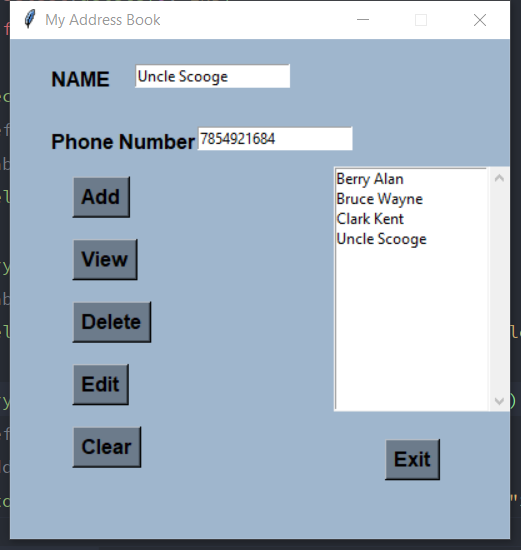
**Libraries and Prerequisites:** For the Address Book program we only used TKinter. This project was not so complicated to make because currently it does not make the used of a database or registers pictures and emails.



Address Book: Contact selection from list



Address Book: Adding “Uncle Scrooge” to contact list



Address Book: Deleting contacts from the list.

NOTE: The list is in alphabetic order.

**## \*\*Future Changes:\*\*** This program is an integration of the chat app and the improvements we’ll make start from the database, layout and allow email address as well.

**Simple Calculator (22-01-2023)**

The seventh project I developed is a Encode and Decode Message that can encode any English text to random characters, I thought it would be easy but it was not, because the program is only encoding and not decoding. It is on version 0.1.07 with the size of 4.23 Kb.

**Utility or why Develop:** This program can be used to encode and decode messages using base64 data encoding. The user has to enter a message and choose between encode (e) and decode (d); the same passkey must be used in the encoding and decoding of the message

**Libraries and Prerequisites:** For the Encode and Decode Message program we used TKinter, and base64 a library used to encode data. This project was complicated to make because currently it does not decode the messages after encoding it nor it recognises the passkey (it basically does nothing if I may say), that aside I still say that it is useful.

"""  
 GIT: @drafonsopena  
 + The objective is to create a Text-to-Speech Converter using Python.  
 | Group:  
 +-+---------------- 1 ----------------  
 | Prerequisites:  
 | Install libraries (eg: pip3 install tk)  
 | Basic Python skills  
 | Use a virtual environment  
 +---------------- 2 ----------------  
 | Project File Structure:  
 | Import all the needed libraries/modules  
 | Create display window  
 | Create labels, functions and buttons  
 +---------------- 3 ----------------  
 | All necessary libraries for the Text-to-Speech Converter:  
 | import tkinter as tk  
 | from gtts import gTTS  
 | from playsound import playsound  
 +------------------------------------  
"""

**## \*\*Future Changes:\*\*** This program is mainly being developed to create a chat app and change the encoding level of the messages later on.

**Snake Game (24-01-2023)**

The seventh project I developed is a Encode and Decode Message that can encode any English text to random characters, I thought it would be easy but it was not, because the program is only encoding and not decoding. It is on version 0.1.07 with the size of 4.23 Kb.

**Utility or why Develop:** This program can be used to encode and decode messages using base64 data encoding. The user has to enter a message and choose between encode (e) and decode (d); the same passkey must be used in the encoding and decoding of the message

**Libraries and Prerequisites:** For the Encode and Decode Message program we used TKinter, and base64 a library used to encode data. This project was complicated to make because currently it does not decode the messages after encoding it nor it recognises the passkey (it basically does nothing if I may say), that aside I still say that it is useful.

"""  
 GIT: @drafonsopena  
 + The objective is to create a Text-to-Speech Converter using Python.  
 | Group:  
 +-+---------------- 1 ----------------  
 | Prerequisites:  
 | Install libraries (eg: pip3 install tk)  
 | Basic Python skills  
 | Use a virtual environment  
 +---------------- 2 ----------------  
 | Project File Structure:  
 | Import all the needed libraries/modules  
 | Create display window  
 | Create labels, functions and buttons  
 +---------------- 3 ----------------  
 | All necessary libraries for the Text-to-Speech Converter:  
 | import tkinter as tk  
 | from gtts import gTTS  
 | from playsound import playsound  
 +------------------------------------  
"""

**## \*\*Future Changes:\*\*** This program is mainly being developed to create a chat app and change the encoding level of the messages later on.