

Proyecto fin de curso

“Programación Python” UDE - 2022

EL AHORCADO

```
+-----+
|       |
|       O
|      /|\
|      / \
|
=====
```

Integrantes:

Francisco del Rio - 4.900.889-3

Matías Tournier - 5.087.161-9

Adrian Parrilla - 4.484.649-0

***Fecha de Entrega:** Lunes 05 de Septiembre de 2022*

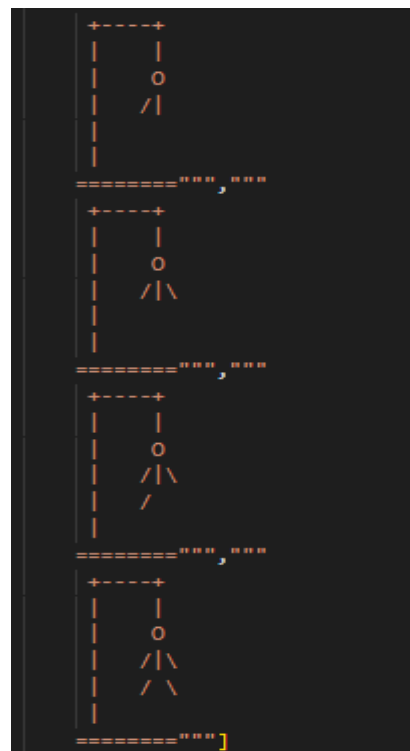
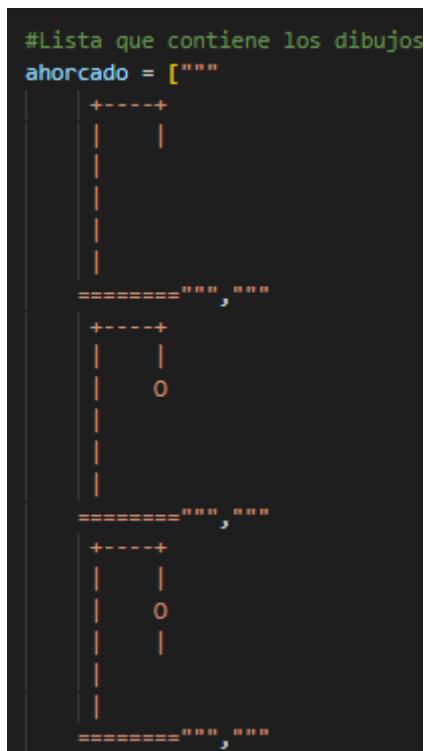
Desarrollo:

Objetivos:

Aplicar los conocimientos del curso con el juego del ahorcado.

Algoritmos usados:

- 1- Se crea una lista la cual contiene el “Dibujo de el Ahorcado” en todas sus etapas.



- 2- Al comienzo del juego "La máquina" elige una palabra secreta aleatoriamente, que fue preestablecida con antelación en el código realizado.

```
#Lista que contiene palabras a ser adivinadas por el jugador.
diccionario = ["curso", "python", "computadora", "programa", "juego", "grupo", "estudio", "heladera", "castaña", "fortaleza", "migrña", "aspirina"]
```

```
#Con random, se selecciona una palabra aleatoria del "diccionario" y se almacena en variable.
palabraRandom = random.choice(diccionario)
palabraRandom = palabraRandom.upper()
```

- 3- Acto seguido, "La máquina" dibuja el ahorcado e inserta la misma cantidad de líneas que letras contenga la palabra secreta.

```
#Lista de guiones bajos [_, _, _, ...] del largo de la palabra aleatoria.  
palabraOculto = ["_" for _ in range(largoPalabra)]
```

```
while contadorVidas < 6: #Son 6 intentos por las 6 partes del cuerpo. Comienza solo la "piola".  
    print(ahorcado[contadorVidas]) #Imprimimos el tanteador inicial.  
    letra = ingreseLetra(intento) #Utiliza la función para asegurarse que sea una letra y no repetida.  
    intento += letra #Agrega la letra a los intentos ya ingresados.  
    acierto = False  
    print("\n=====")
```

- 4- "El Jugador" ingresa una letra.

```
#FUNCIONES  
def ingreseLetra(intento):  
    # Devuelve la letra que el jugador ingreso y filtra que se ingrese una letra y no cualquier otra cosa.  
    while True:  
        letra = input("\nIngresa una letra: ")  
        letra = letra.upper()  
        if len(letra) != 1:  
            print ("\nERROR: Ingresa una sola letra.")  
        elif letra in intento:  
            print ("\nERROR: Ya has ingresado esa letra, intenta nuevamente ")  
        elif letra not in abecedario:  
            print ('\nERROR: Tiene que ser una letra.')  
        else:  
            return letra
```

- 5- "La máquina" comprueba la letra que "El Jugador" ingresó y recorre la palabra secreta.
- 6- Si la letra está en la palabra "La máquina" escribe la letra sobre las líneas correspondientes, de caso contrario la escribe como letra utilizada y dibuja una parte del ahorcado.
- 7- "El Jugador" cuenta con un total de 6 chances para poder adivinar la palabra secreta.

```
for i in range(largoPalabra): #Recorre desde el índice 0 hasta la última letra de la palabra.  
    if listaLetras[i] == letra: #Evalúa que la letra dada esté en la lista de la palabra aleatoria.  
        palabraOculto[i] = letra.upper() #Reemplaza el guión bajo por la letra dada, modifica la lista.  
        contador += 1 #Aumenta el contador, una letra de la palabra adivinada.  
        acierto = True  
if not acierto: #Si la letra no está en la palabra, se pierde una vida.  
    contadorVidas += 1 #Tendremos 6 vidas.  
imprimir(palabraOculto)  
if contador == largoPalabra: #Se adivinaron todas la letras  
    ganador = True  
    break
```

- 8- Si se completó la palabra el juego termina y el jugador gana.
- 9- Si se completó el dibujo del ahorcado el juego termina y el jugador pierde.
- 10- Si no se completó ni el dibujo ni la palabra se vuelve al paso 3

```
if ganador:
    print("\nHAS GANADO!\n")
else:
    print(ahorcado[6]) #Imprime el último muñeco, el ahorcado final.
    print(f"\nHAS PERDIDO! \n\nLa palabra era: ",palabraRandom,"\n")
```

Conclusiones:

Las conclusiones que pudimos tomar en este pequeño proyecto son que:

- Al ser Python un lenguaje bastante simple uno puede llegar con fluidez al objetivo que se plantea.
- Otra de las conclusiones que pudimos llegar a sacar es que si bien el curso fue corto para poder aprender todos los campos que puede abarcar Python, con una base un poco sólida se pueden realizar proyectos ambiciosos.
- Nos hubiese gustado poder agregar un manejo de archivos con nuestras palabras pre guardadas, lo que haría al proyecto un poco más completo, ya que utiliza la gran mayoría de temas abarcados en el curso.

Bibliografía:

- Material del curso
- <https://www.w3schools.com/python/default.asp>
- <https://diveintopython3.net/>