

ADIVINA EL NÚMERO

EQUIPO 7

LOS OBJETIVOS

Nuestro objetivo es realizar un programa cuyo propósito sea el que el usuario pueda adivinar el número que la computadora elija al azar. Donde se le dan indicaciones, de si el número es mayor o menor al número que eligió el usuario.



FASES DEL PROYECTO

1º ETAPA

Luego de definir la variable de predicción, se tiene que comenzar una aparte repetitiva del proceso por eso se usa un ciclo “while”

2º ETAPA

para que el proceso siga es que la predicción no sea igual que el número aleatorio, si es igual el usuario ya predijo el número aleatorio correctamente y el juego termina.

3º ETAPA

primero se le tiene que pedir al usuario que adivine un número entre 1 y x, se necesita usar la función “input” es una función muy útil que permite hacer los programas interactivos (mostrar un mensaje al usuario y recibir un valor a cambio), funciona con solo escribir la palabra “input” y llamar a la función,

4º ETAPA

se puede escribir una cadena de caracteres dentro de los paréntesis, ese será el mensaje que se le va a mostrar al usuario es adivina un número entre 1 y el valor x, se deben agregar dos puntos y un espacio para que este mejor presentado

5º ETAPA

```
10 número_aleatorio=random.randint(1,x)
# Número aleatorio entre 1 y x
11
12 predicción=0
13
14 while predicción != número aleatorio:
15     predicción=input ("Adivina un número
entre 1 y {x}: ")
```

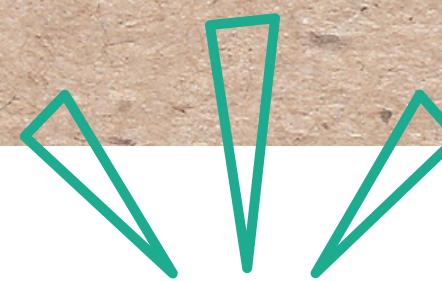
DEFINAMOS EL PROGRAMA

Empezamos definiendo una función, llamada "adivina el número" con un parámetro llamado (x)

```
[ ] def adivina_el_número(x):
```

Con la función **print**, ya que esta nos permite mostrar un mensaje, el cuál será "**bienvenido al juego**" y poner una meta la cuál es "**Tu meta es adivinar el número generado**"

```
print("¡Bienvenido al juego!")
print("el objetivo es adivinar el número generado")
```



1. Generar el número aleatorio: definimos una variable
"número_aleatorio"

```
número_aleatorio
```

Para esto utilizamos la sentencia "import" importaremos
el módulo "random"

```
[ ] import random
```

En la variable "numero_aleatorio" ponemos "numero_aleatorio = random". Estos nos permite trabajar con un número aleatorio

```
número_aleatorio = random.
```

Para generar lo escribimos en nombre del módulo "random" seguido de la función "randint". Quedando "random.randint ()" que significa que vamos a generar un número entero aleatorio.

```
número_aleatorio = random.randint(1, x)
```

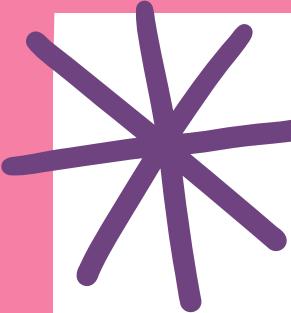
IMPLEMENTEMOS LA LÓGICA PRINCIPAL DEL JUEGO.

Así, una vez teniendo nuestro número aleatorio, tenemos que preguntarle al usuario que número va a predecir, que es el número aleatorio de la computadora, es por eso que vamos a crear una variable que se va a llamar “predicción”, de la cual su valor inicial va a ser 0 para que no haya ninguna posibilidad de que coincida inicialmente con el número aleatorio.

```
predicción = 0
```

Tenemos que crear una parte repetitiva del proceso, y para ello vamos a utilizar un ciclo “while” ya que necesitamos repetir una secuencia de instrucciones, un número no específico de veces, pues nosotros no sabemos cuantas veces tendremos que pedirle al usuario que de una predicción del número, porque el usuario puede adivinar el número a la primera vez o quizá se necesitan 500 repeticiones para llegar al número aleatorio, dependiendo también del tamaño del intervalo.

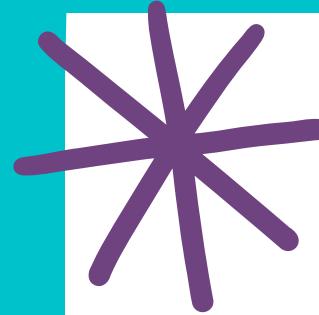
```
while predicción != número_aleatorio:
```



Primero tenemos que pedirle al usuario que adivine un número entre 1 y x. Para obtener un valor del usuario necesitamos usar la **función input**, que nos permite interactuar con el usuario, es decir, mostrar un mensaje en el que solicitas algo y recibir algo a cambio.

```
predicción = int(input(f"Adivina un número entre 1 y {x}: "))
```

Una vez que tenemos la predicción, si la predicción es correcta el ciclo se va a detener, por la condición va a ser falsa. Entonces el jugador gana el juego.



Ahora bien, si la predicción es menor que el número aleatorio, entonces mostramos el mensaje ¡fallaste, intenta con un número más grande!.

```
print("fallaste, intenta con un número más grande")
```

En cambio, si la predicción es mayor que el número aleatorio, entonces mostramos el mensaje ¡fallaste, intenta con un número más pequeño!

```
print("fallaste, intenta con un número más pequeño")
```

Finalmente, como se dijo en (2) si la predicción es igual que el número aleatorio, el ciclo se va a detener inmediatamente, cuando esto pase, mostraremos el siguiente mensaje ¡Muy bien! haz hallado el numero númeroaleatorio.Felicidades.

```
print(f"¡Muy bien! haz hallado el numero {número_aleatorio}. Felicidades")
```

FUNCIIONES

1º

La función randint toma dos parámetros (a,b) retorna un entero aleatorio N tal que ese N es mayor o igual que a y menor o igual que b ,

2º

Si los valores están entre 1 y x porque el valor debe ser mayor o igual a 1 por eso 1 debe ser el límite inferior de ese intervalo y el límite superior de ese intervalo sea x, el número aleatorio que se va generar debe ser menor o igual a x

3º

x es el parámetro cuando se pase a la función cuando se llame. Se agrega un comentario que diga “Número aleatorio entre 1 y x.”

CICLO WHILE

```
10 número_aleatorio=random.randint(1,x) # Número aleatorio entre 1 y x
11, 12 predicción=0
13,14 while predicción != número aleatorio:
15 # El usuario ingresa un número
16 predicción=int(input (f'Adivina un número entre 1 y {x}: '))
17, 18 if predicción < número aleatorio
19 print("Intenta otra vez. Este número es muy pequeño.")
20 elif predicción > número aleatorio
21 print("Intenta otra vez. Este número es muy grande.")
```

F-STRING

El proyecto

se usa un tipo específico de caracteres en Python llamada f-string, este tipo de cadena de caracteres permite reemplazar el valor de una variable o de una expresión en la cadena de forma más concisa. Esto hace que el código parezca mucho más fácil de leer, para crear una f-string se necesita agregar la letra f antes de la cadena, antes de abrir las comillas dobles o simples

después dentro de la cadena donde se quiera reemplazar el valor de la variable o de la expresión simplemente se rodea con llaves y ahí se reemplaza el valor de la variable.

```
10 número_aleatorio=random.randint(1,x) # Número aleatorio entre 1 y x  
11  
12 predicción=0 13 14 while predicción != número aleatorio:  
15 predicción=input (f'Adivina un número entre 1 y {x}:') # f-string
```

PUNTOS IMPORTANTES

Se implementa la lógica principal del juego, una vez que se tiene el número aleatorio se debe preguntar al usuario que número va a adivinar o predecir que es el número aleatorio de la computadora, se crea una variable que se va a llamar predicción: su valor inicial será 0 para que no coincida y que no haya ninguna posibilidad de que coincida con el número aleatorio, pues se está generando un número aleatorio entre 1 y x

```
10 número_aleatorio=random.randint(1,x) # Número aleatorio entre 1 y x  
11  
12 predicción=0
```

CONCLUSIONES

1º Conclusión

La función input tiene un valor muy importante, cualquier valor que ingrese el usuario se va a representar como una cadena de caracteres automáticamente y se va a asignar a la variable

2º Conclusión

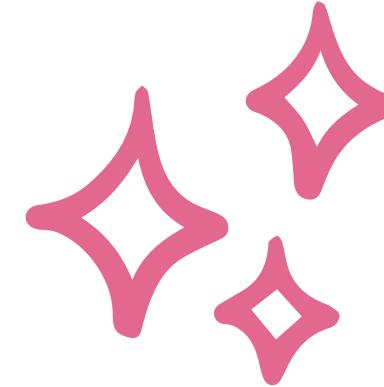
se quiere trabajar con numero entre entre 1 y x, pero se hará una cadena de caracteres, para cambiar el tipo de dato en el programa, se puede rodear con paréntesis a la llamada input y escribir la palabra “int”

3º Conclusión
número_aleatorio=random.randint(1,x) # Número aleatorio entre 1 y x
predicción=0
13
14 while predicción != número aleatorio:
15 predicción=int(input(f"Adivina un número entre 1 y {x}: ")) # f-string

GRACIAS POR
SU ATENCION



REFERENCIAS



1º Referencia:

Navone, E. C. (5 de julio de 2021). free code camp. Obtenido de 6 Proyectos de Python Básicos – Curso Completo Paso a Paso:
<https://youtu.be/tWnyBD2src0>