

PIEDRA, PAPEL O TIJERA

PLAY

Proyecto Lógica de Programación

¿De qué trata el proyecto y sus objetivos?

1

EL PROYECTO CONSISTE EN CREAR
EL JUEGO PIEDRA, PAPEL O TIJERA
USANDO PYTHON.

2

EL USUARIO JUEGA CONTRA LA
COMPUTADORA.

3

EL JUEGO SIGUE LAS REGLAS
TRADICIONALES.

4

APLICAR LÓGICA DE PROGRAMACIÓN.

5

USAR LISTAS Y TUPLAS.

6

UTILIZAR FUNCIONES PARA
ORDENAR EL CÓDIGO.

7

EVITAR ERRORES CON
VALIDACIONES.

ESTRUCTURA DEL JUEGO

Listas y Tuplas

Lógica para
determinar el
Ganador

Validación de
entrada del
Usuario

Control de Rondas
y Puntos

Eleción de la
Computadora y
Temporizador

Opción de Volver a
Jugar

1

LA LISTA CONTIENE LAS OPCIONES VÁLIDAS DEL JUEGO.

LA TUPLA DEFINE LAS REGLAS PARA SABER CUÁNDO GANA EL USUARIO.

ESTO EVITA USAR MUCHOS IF REPETIDOS Y ORDENA EL CÓDIGO.

```
def obtener_opcion_usuario():
    while True:
        opcion = input("Elige piedra, papel o tijera: ").strip().lower()

        if opcion.isnumeric():
            print("No se permiten numeros.")
            continue

        if opcion in OPCIONES:
            return opcion
        else:
            print("Opcion no valida.")
```

OPCIONES = ["piedra", "papel", "tijera"]

```
REGLAS = (
    ("piedra", "tijera"),
    ("papel", "piedra"),
    ("tijera", "papel")
)
```

2

EL PROGRAMA NO ACEPTA NÚMEROS.

SOLO PERMITE OPCIONES VÁLIDAS.

SI EL USUARIO SE EQUIVOCΑ, VUELVE A PREGUNTAR.

EL JUEGO NO SE CIERRA POR ERRORES.

3

LA COMPUTADORA ELIGE UNA OPCIÓN AL AZAR.

SE USA UN TEMPORIZADOR PARA QUE EL JUEGO SEA MÁS CLARO VISUALMENTE.

```
def obtener_opcion_computadora():
    print("La computadora esta eligiendo...")
    time.sleep(1)
    return random.choice(OPCIONES)
```

```
def determinar_ganador(usuario, computadora):
    if usuario == computadora:
        return "empate"
    elif (usuario, computadora) in REGLAS:
        return "usuario"
    else:
        return "computadora"
```

```
def obtener_opcion_computadora():
    print("La computadora esta eligiendo...")
    time.sleep(1)
    return random.choice(OPCIONES)
```

SE COMPARAN LAS ELECCIONES.

SE VERIFICA SI HAY EMPATE.

SE USA LA TUPLA DE REGLAS PARA DECIDIR EL GANADOR.

4

5

EL JUEGO SE EJECUTA POR RONDAS.

SE LLEVA UN CONTEO DE PUNTOS.

AL FINAL SE MUESTRAN LOS RESULTADOS.

```
puntos_usuario = 0  
puntos_computadora = 0  
empates = 0  
ronda = 1  
MAX_RONDAS = 3
```

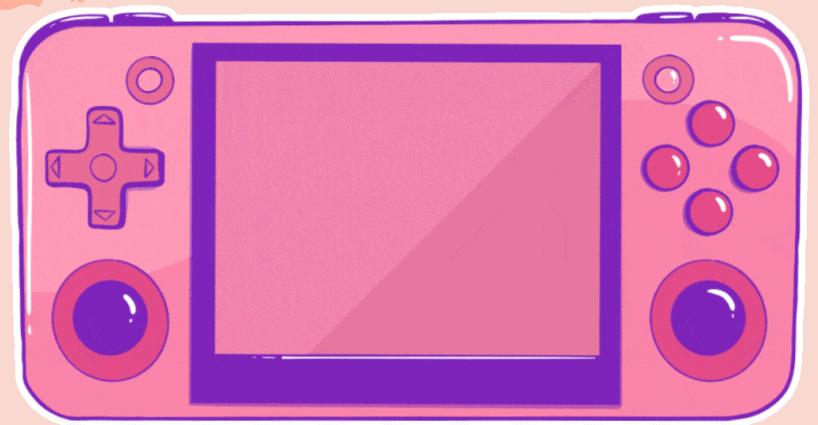
```
if not preguntar_reinicio():  
    print("Fin del juego.")  
    break
```

6

EL USUARIO DECIDE SI DESEA JUGAR OTRA VEZ.

EL PROGRAMA TERMINA DE FORMA CORRECTA.

VIDEO DE PRUEBA



START

