

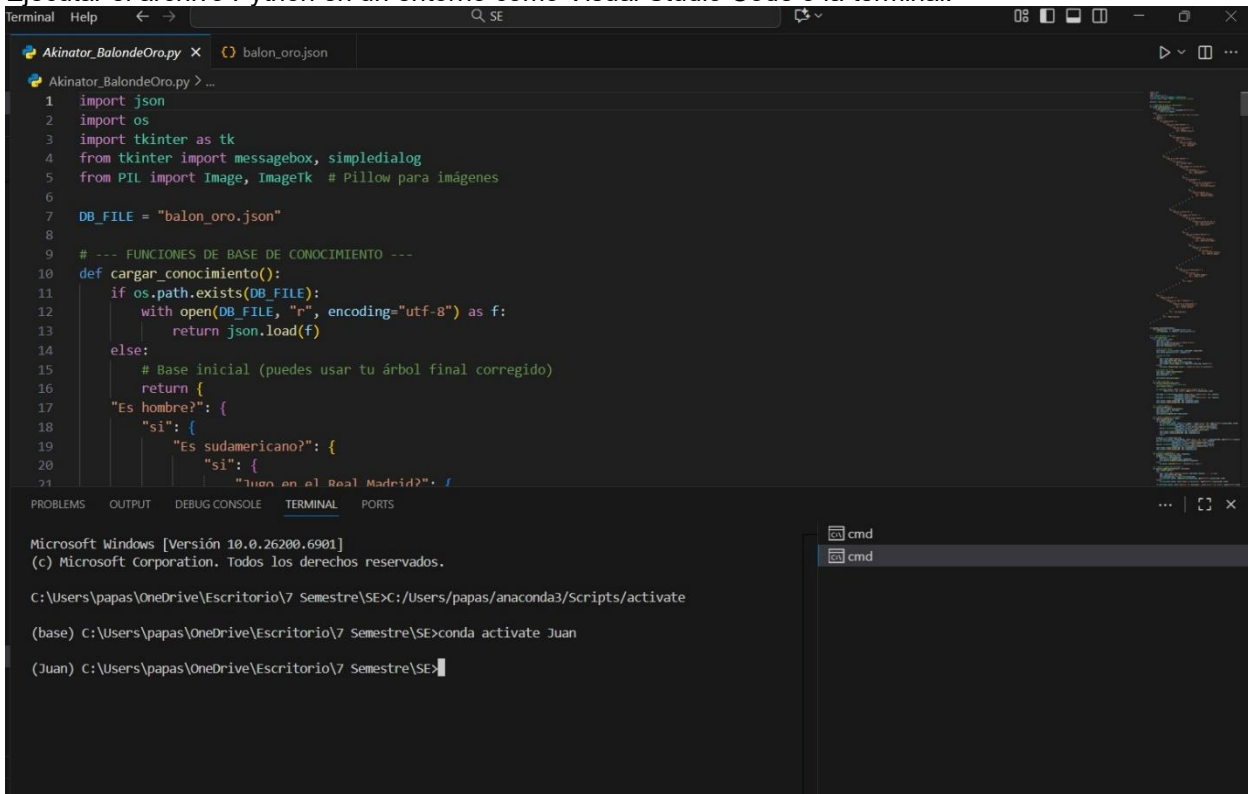
# Manual de Usuario – Adivina Quién: Ganadores del Balón de Oro (1990–2025)

## 1. Descripción general del programa

Este programa implementa un sistema experto en Python que simula el juego 'Adivina Quién', adaptado para los ganadores y ganadoras del Balón de Oro entre 1990 y 2025. El sistema utiliza un árbol de decisiones basado en reglas que guían al usuario a través de preguntas de tipo 'sí' o 'no', hasta identificar correctamente a la persona pensada. Si el sistema no conoce al personaje, aprende automáticamente agregando nueva información al archivo JSON.

## 2. Instrucciones de uso

- 1 Ejecutar el archivo Python en un entorno como Visual Studio Code o la terminal.



The screenshot shows the Visual Studio Code editor with the file `Akinator_BalondeOro.py` open. The code includes imports for `json`, `os`, `tkinter`, `messagebox`, `simpledialog`, and `PIL`. It defines a database file `balon_oro.json` and a function `cargar_conocimiento()` that loads the database or initializes it with a default structure. The terminal window shows the command prompt with the following commands and output:

```
Microsoft Windows [Versión 10.0.26200.6901]
(c) Microsoft Corporation. Todos los derechos reservados.

C:\Users\papas\OneDrive\Escritorio\7 Semestre\SE>C:/Users/papas/anaconda3/Scripts/activate

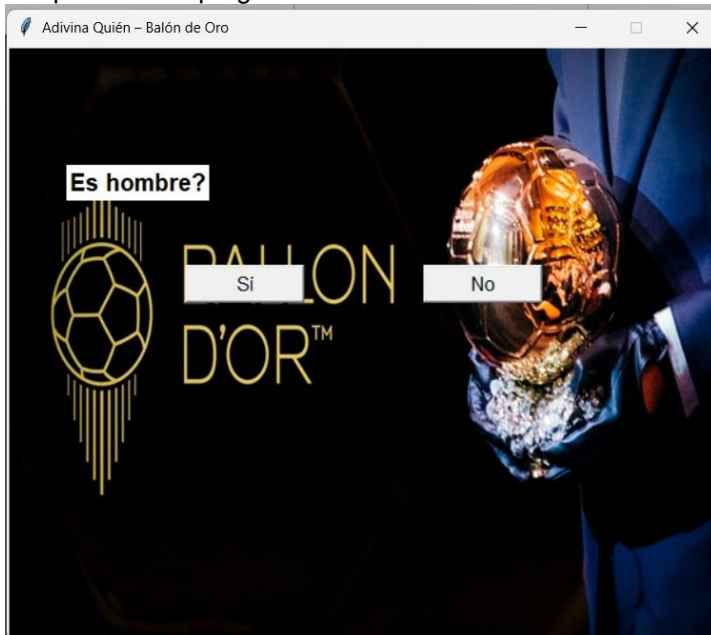
(base) C:\Users\papas\OneDrive\Escritorio\7 Semestre\SE>conda activate Juan

(Juan) C:\Users\papas\OneDrive\Escritorio\7 Semestre\SE>
```

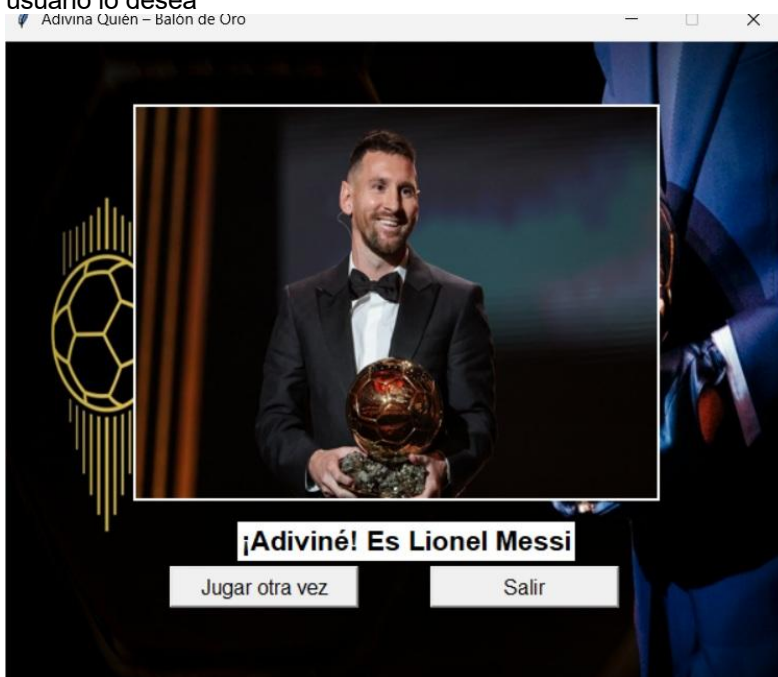
- 2 El programa mostrará un mensaje de bienvenida y pedirá que pienses en un ganador o ganadora del Balón de Oro.



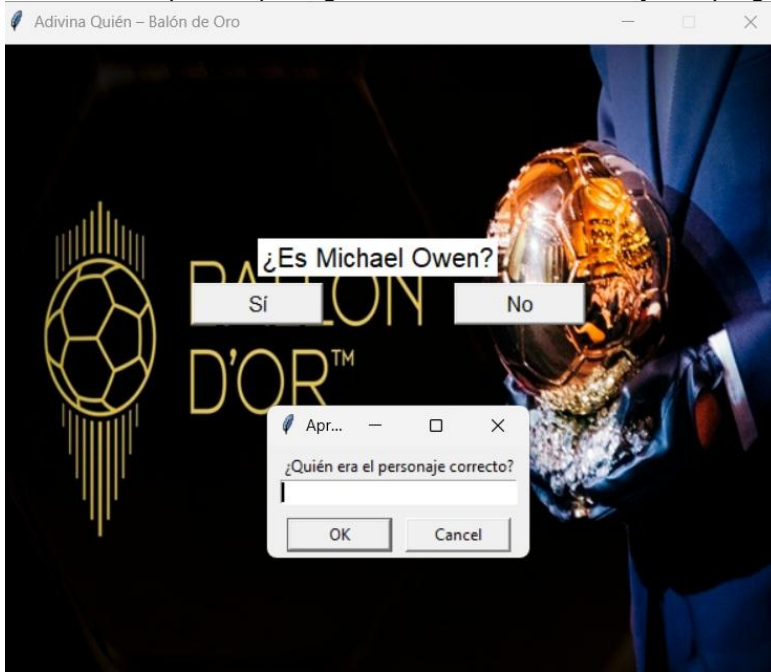
3 Responde a las preguntas con 'sí' o 'no'.



4 Si el programa adivina correctamente, mostrará un mensaje de éxito, también dará la opción de volver a jugar si el usuario lo desea



- 5 Si no acierta, pedirá que ingreses el nombre correcto y una pregunta nueva para aprenderlo.



Te pedirá el nombre del personaje que pensaste y que el programa no adivino, además te pedirá una pregunta para poder distinguir a tu ganador del balón de oro, de igual forma la respuesta a esa pregunta (sí o no)

- 6 La información se guarda automáticamente en el archivo 'balon\_oro.json' y una vez presionando jugar o volver a jugar, se actualizaran los datos y saldrá tu ganador del balón de oro que proporcionaste

```
balon_oro.json > {} Es hombre? > {} si > {} Es sudamericano? > {} si > {} Jugo en el Real Madrid? > {} no > {} Es Argentino? > {}  
1 {  
2   "Es hombre?": {  
3     "si": {  
4       "Es sudamericano?": {  
5         "si": {  
6           "Jugo en el Real Madrid?": {  
7             "si": {  
8               "Jugo en el AC Milan?": {  
9                 "si": "Kaka",  
10                "no": "Ronaldo Nazario"  
11              }  
12            },  
13            "no": {  
14              "Es Argentino?": {  
15                "si": "Lionel Messi",  
16                "no": {  
17                  "Jugo en el PSG?": {  
18                    "si": "Ronaldinho",  
19                    "no": "Rivaldo"  
20                  }  
21                }  
22              }  
23            }  
24          }  
25        },  
26        "no": {  
27          "Jugo en el Real Madrid?": {  
28            "si": {  
29              "Es ingles?": {  
30                "si": "Michael Owen",  
31                "no": {  
32                  "Jugo tambien en la Serie A?": {  
33                    "si": {  
34                      "Es Italiano?": {  
35                        "si": "Fabio Cannavaro",  
36                        "no": "Luka Modric"  
37                      }  
38                    }  
39                  }  
40                }  
41              }  
42            }  
43          }  
44        }  
45      }  
46    }  
47  }  
48 }
```

### 3. Teoría del Funcionamiento

#### 3.1 Reglas

Las **reglas** son declaraciones de tipo 'si... entonces...', utilizadas para tomar decisiones. En este programa, cada pregunta representa una condición ('si la respuesta es sí, entonces sigue por esta rama del árbol; de lo contrario, por la otra'). Por ejemplo: '¿Es sudamericano?' es una regla que guía el flujo hacia diferentes jugadores.

#### 3.2 Casos

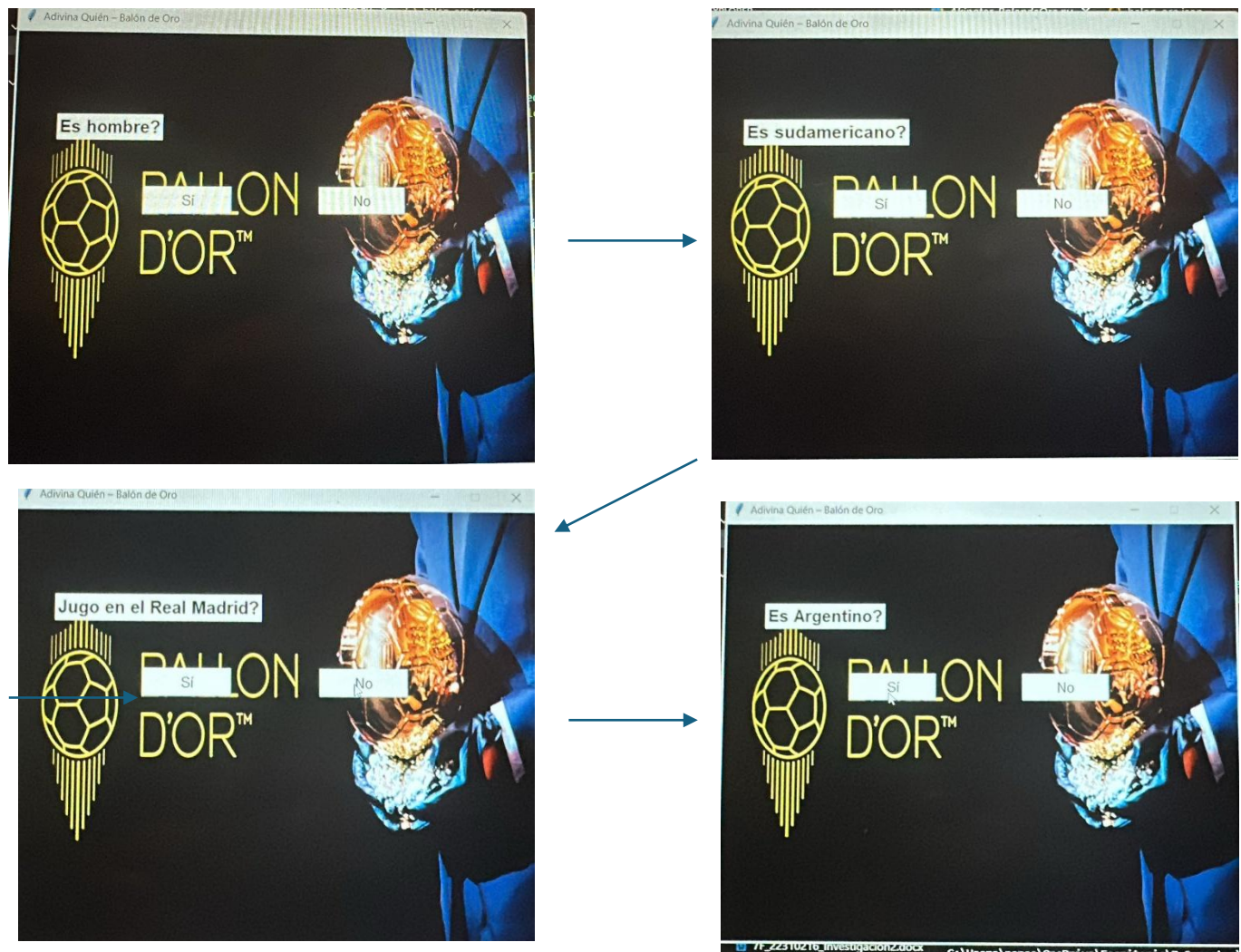
Cada **caso** corresponde a un nodo terminal del árbol, es decir, a un posible jugador o jugadora. Por ejemplo, si las respuestas fueron 'sí' → 'sí' → 'no', el sistema podría llegar al caso 'Lionel Messi'. Los casos son almacenados en formato JSON para permitir que el sistema recuerde el conocimiento adquirido.

#### 3.3 Encadenamiento hacia adelante

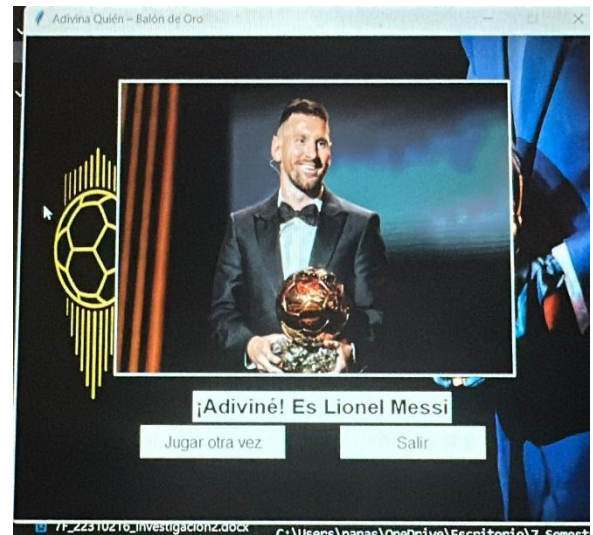
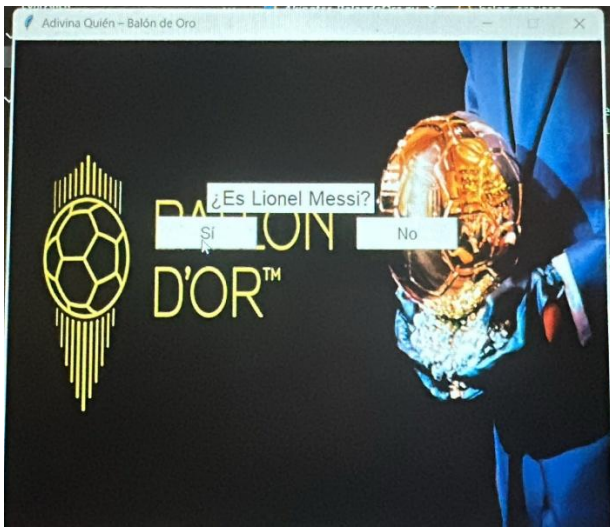
El **encadenamiento hacia adelante** es una técnica de inferencia en sistemas expertos donde el razonamiento comienza con los hechos conocidos y aplica reglas para derivar nueva información. En este caso, el programa empieza con las respuestas del usuario (hechos) y avanza aplicando las reglas del árbol hasta llegar a una conclusión (el jugador o jugadora adivinada). Si no hay coincidencia, el sistema aprende una nueva regla.

### 4. Ejemplo de ejecución

Ejemplo de interacción con el sistema: Sistema: ¿Es hombre? Usuario: sí Sistema: ¿Es sudamericano? Usuario: sí Sistema: ¿Jugó en la Liga de España? Usuario: sí Sistema: ¿Jugó en el Real Madrid? Usuario: no Sistema: ¿Es argentino? Usuario: sí Sistema: ¿Es Lionel Messi? Usuario: sí Sistema: ¡Adiviné! Era Lionel Messi.







## CONCLUSION

El sistema experto implementado demuestra cómo un modelo simple de reglas y aprendizaje puede simular razonamiento humano. Utiliza estructuras de decisión tipo árbol y técnicas de encadenamiento hacia adelante para llegar a una conclusión basada en las respuestas del usuario. Su capacidad de aprendizaje incremental convierte este programa en un ejemplo básico de inteligencia artificial simbólica aplicada a un dominio concreto