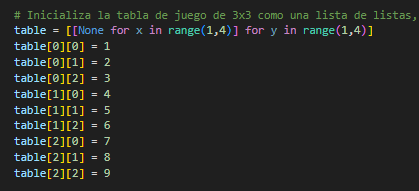
Juego de Tic Tac Toe con Python

El programa desarrollado en Python simula el clásico juego de Tic-Tac-Toe. El flujo del juego se basa en turnos alternos entre el jugador y una IA básica. El juego finaliza cuando uno de los dos jugadores logra completar una fila, columna o diagonal con tres símbolos iguales. Otra forma en la que concluye el juego es cuando no quedan movimientos disponibles en la tabla.

**Estructura y funciones del programa:**

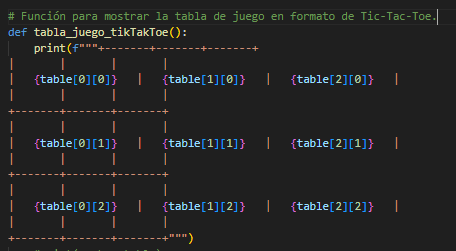
**Creación de la tabla:** para poder crear la tabla del juego se realizó una lista de listas, donde cada lista solo va tener tres elementos, durante la creación de cada lista cada elemento comienza con un dato nulo, luego en la parte externa de la lista se le asigna un único dato que va del 1 al 9.



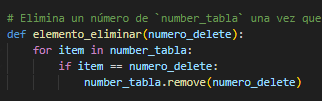
**Creación de una lista:** La siguiente lista denominada como “number\_tabla” cuenta con todos los elementos de la tabla, es decir cuenta con los elementos del 1 al 9, además que también se le ha agregado un 0 adicional. la función de esta tabla es que por cada turno siempre se va ir eliminando un dato, los datos que se van a eliminar van a ser solo los elementos de la tabla por lo que el 0 siempre va permanecer en la lista.



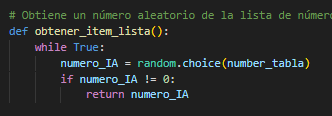
**Función de la tabla de tic tac toe:** Esta función cuenta con la finalidad de mostrar la tabla del juego en la consola, para la cual también se le ha acomodado un elemento del 1 al 9, esta misma tabla también mostrara las actualizaciones que se realicen por cada jugada.



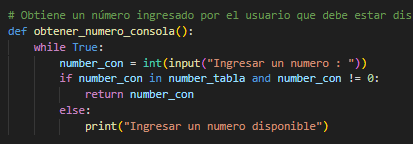
**Función elemento eliminar:** Cuenta con la finalidad de eliminar un elemento de la lista de “number\_tabla” según sea seleccionado por el usuario o por la IA básica.



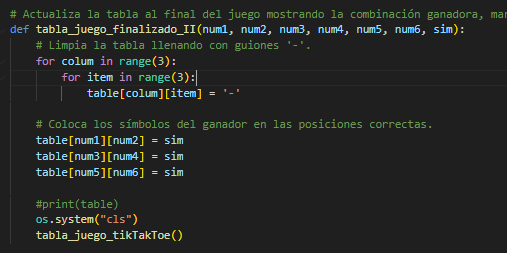
**Función obtener ítem de la lista:** Está programada para ser utilizada únicamente por la IA básica, inicia un bucle infinito y a través de la importación “random choice” se va seleccionar un numero aleatorio existente en la lista, menos 0, finalmente retorna el valor seleccionado.



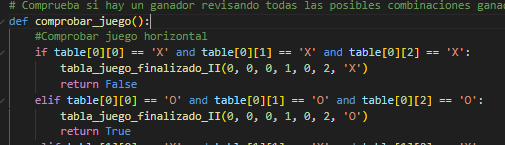
**Función obtener número de consola:** La función inicia con un bucle infinito y solicita el ingreso de dato por consola donde solo se puede seleccionar un numero existente en la tabla. finaliza retornando el dato seleccionado.



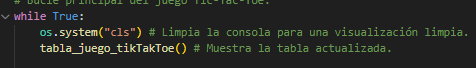
**Función de tabla del juego finalizado:** Esta función trabaja con 7 argumentos, donde 6 son datos numéricos de ubicaciones de la tabla y uno es el símbolo ganador. La función comienza reconstruyendo la tabla cambiando todos los elementos por el carácter “-”, seguidamente, se seleccionan las 3 ubicaciones diferentes de la tabla donde se cambia con el carácter del símbolo ganador, finaliza limpiando y mostrando la nueva tabla en la consola.



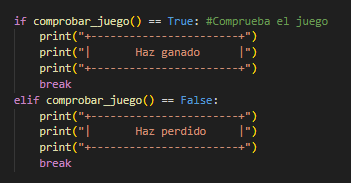
**Función comprobar juego:** Esta función cuenta con la finalidad de comprobar el estado del juego, para conocer si gano el usuario o la IA, tras seleccionar un ganador mostrara una tabla modificada y retornara un valor booleano donde “True” gano el usuario y “False” gano la IA.



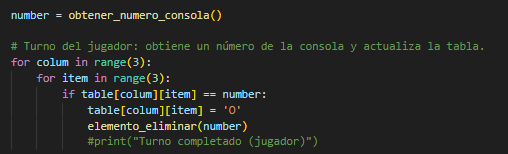
**limpieza y mostrar la consola (Bucle principal):** El programa comienza realizando una limpieza de la consola, donde seguidamente mostrara la tabla con todos los elementos disponibles.



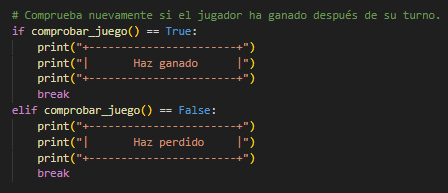
**primera comprobación** **(Bucle principal):** La sentencia condición utiliza a la función “comprobar juego” donde a través de esta obtendrá el valor booleano True y False y según eso mandará el mensaje de ganaste o perdiste. Al no obtener ningún de ambos valores se tomará como None y la sentencia se ignorará y se continuará con el programa.



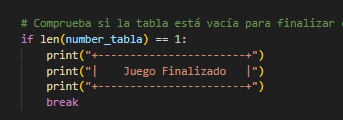
**Comprobar jugada del usuario** **(Bucle principal)**: Atreves de una variable se llamará a la función de “obtener número de consola”, tras obtener el numero comenzara un bucle el cual va buscar el numero seleccionado en la tabla, luego de obtenerlo cambiara la posición del número seleccionado por el carácter “O” y finaliza por eliminar el número de la lista de “number\_tabla”.



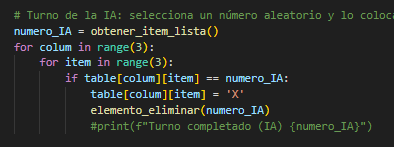
**Segunda comprobación (Bucle principal)**: comprueba el estado del juego y verifica en caso de que exista un ganador. La comprobación se realiza cada vez que se realiza una jugada ya sea por el usuario o por la IA.



**Verifica el estado de la lista (principal):** la sentencia condicional verifica si la longitud de la silla es igual a 1, de ser así el programa terminará porque ya no habrá más turnos para jugar.



**Comprobar jugada de la IA básica:** En una variable se va almacenar el numero aleatorio que se va elegir de la lista “number\_tabla”, luego entrara en un bucle donde va a buscar la posición del número seleccionado, al encontrarlo cambiara el número de la posición seleccionada por “X”, finalmente, se va a eliminar el numero elegido por la IA de la lista de “number\_tabla”



**Resumen:**

* **Creación de la tabla:** se crea una lista de listas de forma 3x3. Se agregan los datos individuales
* **Creación de lista:** lista que almacenara cada valor de la tabla.
* **Función tabla tic tac toe:** muestra la tabla en la consola.
* **Función eliminar elemento:** Elimina un elemento de la lista de “number\_tabla”.
* **Función obtener ítem de la lista:** obtiene un elemento aleatorio de la lista “number\_tabla”, solo utilizado por la IA.
* **Función obtener dato de consola:** Obtiene un elemento por consola.
* **Función tabla finalizada:** muestra la tabla con la combinación ganadora.
* **Función comprobar juego:** comprueba si hay una combinación ganadora.
* **Limpia y muestra tabla:** limpia y muestra la tabla en la consola.
* **primera comprobación:** comprueba estado del juego.
* **Comprobar jugada del usuario:** obtiene un dato por el usuario.
* **Segunda comprobación:** comprueba estado del juego.
* **Verifica estado de la lista:** comprueba si hay más de un valor en la lista.
* **Comprobar jugada de la IA básica:** La IA hace una jugada en el tablero