

Como se hace una menú horizontal, cuando pincha rutas del menú pues se guía hasta rutas

- Menú Horizontal: El menú está en la cabecera y utiliza enlaces que llaman a la función `showSection()` con el ID de la sección correspondiente.
- Mostrar/Ocultar Secciones: La función `showSection()` oculta todas las secciones y luego muestra la sección seleccionada.

Como se crea un tablero para el típico juego de conecta4 utilizando el javascript?

```
function crearTablero() {  
    var tablero = document.getElementById('game-board');  
    tablero.innerHTML = '';  
    for (var i = 0; i < nfilas; i++) {  
        var fila = document.createElement('tr');  
        for (var j = 0; j < ncol; j++) {  
            var celda = document.createElement('td');  
            celda.dataset.col = j;  
            celda.addEventListener('click', mover);  
            fila.appendChild(celda);  
        }  
        tablero.appendChild(fila);  
    }  
}
```

Para verificar las celdas ganador, tanto horizontal, vertical como diagonalmente, usando la función `pintarGanador`:

```
function comprobar() {  
    // Comprueba las condiciones de victoria (horizontal, vertical y  
    diagonal)  
    for (var i = 0; i < nfilas; i++) {  
        for (var j = 0; j < ncol; j++) {  
            if (matriz[i][j] !== 0) {  
                // Horizontal  
                if (j + 3 < ncol && matriz[i][j] === matriz[i][j + 1]  
&& matriz[i][j] === matriz[i][j + 2] && matriz[i][j] === matriz[i][j +  
3]) {  
                    pintarGanador([[i, j], [i, j + 1], [i, j + 2], [i,  
j + 3]]);  
                    return true;  
                }  
                // Vertical  
                if (i + 3 < nfilas && matriz[i][j] === matriz[i + 1][j]  
&& matriz[i][j] === matriz[i + 2][j] && matriz[i][j] === matriz[i +  
3][j]) {  
                    pintarGanador([[i, j], [i + 1, j], [i + 2, j], [i +  
3, j]]);  
                }  
            }  
        }  
    }  
}
```

```

        return true;
    }
    // Diagonal \
    if (i + 3 < nfilas && j + 3 < ncol && matriz[i][j] ===
matriz[i + 1][j + 1] && matriz[i][j] === matriz[i + 2][j + 2] &&
matriz[i][j] === matriz[i + 3][j + 3]) {
        pintarGanador([[i, j], [i + 1, j + 1], [i + 2, j +
2], [i + 3, j + 3]]);
        return true;
    }
    // Diagonal /
    if (i + 3 < nfilas && j - 3 >= 0 && matriz[i][j] ===
matriz[i + 1][j - 1] && matriz[i][j] === matriz[i + 2][j - 2] &&
matriz[i][j] === matriz[i + 3][j - 3]) {
        pintarGanador([[i, j], [i + 1, j - 1], [i + 2, j -
2], [i + 3, j - 3]]);
        return true;
    }
}
return false; // No hay ganador
}

```

No me pinta el celda ganador, sino que muestra la alerta de ganador y se actualiza, pero no pinta la celda amarilla, ¿por qué puede ser?

El problema ocurre porque el tablero se reinicia inmediatamente después de mostrar el mensaje de ganador. Para solucionarlo, se debe retrasar el reinicio del tablero para que las celdas ganadoras se pinten antes de que el juego se reinicie.

```

setTimeout(() => {
    alert('¡Jugador ' + (turno === "rojo" ? "rojo" :
"verde") + ' ha ganado!');
    reiniciar();
}, 100); // Retrasa el reinicio para que se vean las
celdas ganadoras

```

Como puedo mostrar el historial de este tipo, que es devuelto por la URL:

```
{"title":"Ranking Jugadores","description":"Muestra el ranking de
jugadores","UltimaModificacion":{"@type":"ISODate","value":"2023-03-09T09:00:00.594Z"},"
posiciones":{"primero":{"nombre":"Luis","valor":"15","fecha":"2023-03-08T09:00:00.594Z"},"segundo":{"nombre":"Ana","valor":"16","fecha":"2023-03-08T09:01:00.594Z"},"tercero":{"nombre":"Juan","valor":"18","fecha":"2023-03-08T09:00:00.594Z"},"cuarto":{"nombre":"Margarita","
```

```
valor": "18", "fecha": "2023-03-08T09:02:00.594Z"}, "quinto": {"nombre": "Lucia", "valor": "20", "fecha": "2023-03-08T09:00:00.594Z"}, "sexta": {"nombre": "Javier", "valor": "20", "fecha": "2023-03-08T09:01:00.594Z"}, "septimo": {"nombre": "Antonio", "valor": "20", "fecha": "2023-03-08T09:02:00.594Z"}, "octavo": {"nombre": "Mario", "valor": "21", "fecha": "2023-03-08T09:00:00.594Z"}, "noveno": {"nombre": "Julia", "valor": "22", "fecha": "2023-03-08T09:20:00.594Z"}, "decimo": {"nombre": "Macarena", "valor": "22", "fecha": "2023-03-08T09:30:00.594Z"}}}
```

```
.then(data => {
    // Mostrar título, descripción y última modificación
    const infoDiv = document.getElementById('info');
    infoDiv.innerHTML =
        <h3>${data.title}</h3>
        <p>${data.description}</p>
        <p><strong>Última Modificación:</strong> ${new Date(data.UltimaModificacion.value).toLocaleString()}</p>
    ;
}

// Mostrar el ranking de jugadores
const posiciones = data.posiciones;
if (posiciones) {
    const scoresDiv =
document.getElementById('scores');
    scoresDiv.innerHTML = '<ul>' +
        Object.keys(posiciones).map(key => {
            const jugador = posiciones[key];
            return `<li><strong>${key}:</strong>
${jugador.nombre} - ${jugador.valor} puntos (Fecha: ${new Date(jugador.fecha).toLocaleDateString()})</li>`;
        }).join('') +
        '</ul>';
} else {
    throw new Error('Datos no válidos recibidos de la API');
}
})
```