



Recorrer una Matriz

DigitalHouse>



**Certified Tech
Developer**

The Ultimate Degree

Índice

1. [La matriz de ejemplo](#)
2. [Un ciclo for](#)
3. [Dos ciclos for](#)
4. [Recorrer por fila](#)
5. [Recorrer por columna](#)

1 | La matriz de ejemplo

La matriz de ejemplo

Definición de una matriz

Vimos cómo definir una matriz y qué obtenemos cuando definimos una.

```
{  
  let butacasCine = [['ocupado', 'libre', 'libre', 'libre'],  
    ['libre', 'libre', 'libre', 'libre'],  
    ['libre', 'libre', 'ocupado', 'ocupado'],  
    ['ocupado', 'ocupado', 'ocupado', 'ocupado'],  
    ['libre', 'libre', 'libre', 'libre']];  
}
```

Hemos definido una matriz que representa el estado de las butacas de un cine.

La matriz de ejemplo: qué obtenemos

Como sabemos, los cines tienen filas y, a su vez, las filas se dividen en posiciones en la fila. En una matriz tenemos filas y columnas.

Filas	ocupado	libre	libre	libre
	libre	libre	libre	libre
	libre	libre	ocupado	ocupado
	ocupado	ocupado	ocupado	ocupado
	libre	libre	libre	libre
	Columnas			

La matriz de ejemplo: qué estamos representando

Cuando indiquemos una fila y columna, lo que vamos a obtener es el estado de una butaca, si está ocupada o no. Esto ocurre en cualquier matriz, al indicar una posición obtenemos el valor de esa posición.

Vamos a ver a ahora qué podemos hacer con esta matriz.



2 | Un ciclo for

Un ciclo for

Para efectuar un ciclo for debemos saber cuántas veces debe iterar. Si bien para nuestro ejemplo podemos poner directamente los valores, es mejor que la matriz “se mida a sí misma”. Podemos hacerlo de dos formas.

```
{}
```

```
let cantidadFilas = butacasCine.length;
```

Medimos la cantidad de filas.

```
let cantidadColumnas = butacasCine[0].length;
```

Medimos la cantidad de columnas de la fila 0.

Si bien medimos solo la fila 0, todas las columnas tienen la misma cantidad de filas.

Un ciclo for

Recorrer una fila

Ahora vamos a recorrer la matriz. Por ejemplo, queremos saber si en la fila 1 hay butacas libres.

```
{  
  {}  
    let libres= false;  
    for (let i=0;i<butacasCine[1].length;i++)  
    {  
      if (butacas[1][i]=='libre')  
        libres= true;  
    }  
}
```

Utilizamos un solo for, podemos fijar la fila y solo recorreremos la fila indicada, preguntando por cada elemento —columna— de la fila.

Un ciclo for

Recorrer una columna

Si quisiéramos comprobar que la columna central —las butacas del medio— están ocupadas, entonces deberíamos recorrer por columna.

```
let ocupadas= true;
for (let i=0;i<butacasCine.length;i++)
{
    if (butacas[i][1]== 'libre')
        ocupadas= false;
}
```

En este caso, también utilizamos **un único for**, pero fijamos la columna, por tener cantidad de elementos par —0,1,2,3— elegimos 1 como elemento central, también podría haber sido 2.

Un ciclo for

¿Qué podemos hacer utilizando **solo un ciclo for** en una matriz?

Recorrer una fila

Si en el **for** fijamos la **fila**, estaremos recorriendo todos los elementos de ella.

Para medir cuántas iteraciones debe hacer el for debemos medir la longitud de la fila.

Recorrer una columna

Si en el **for** fijamos la **columna**, estaremos recorriendo todos los elementos de ella.

Para medir cuántas iteraciones debe hacer el for debemos medir la longitud de la columna.

3 | Dos ciclos for

Dos ciclos for

Si utilizamos dos ciclos for podemos recorrer la matriz posición a posición, es decir, podemos recorrerla completa, no solo una fila o una columna determinada.

Pero vamos a tener que decidir si recorreremos primero las filas o las columnas.

Esto quiere decir, definir si queremos comprobar algo que ocurre en las filas o queremos comprobar algo que pasa en las columnas.

4 | Recorrer por fila

Recorrer por fila

Para nuestra matriz de ejemplo queremos ver las filas que tengan butacas libre. Veamos primero qué quiere decir esto.

ocupado	libre	libre	libre	Fila 0
libre	libre	libre	libre	Fila 1
libre	libre	ocupado	ocupado	Fila 2
ocupado	ocupado	ocupado	ocupado	Fila 3
libre	libre	libre	libre	Fila 4

Vamos a tener que comprobar **para cada fila** si tiene butacas libres.

El resultado debería ser fila 0, fila 1, fila 2, fila 4.

Recorrer por fila

Recorremos la matriz con dos for, el primero recorre cada fila y el segundo cada elemento de la fila.

```
let libres= false;
for (let fila=0;fila<butacasCine.length;fila++){
  libres=false;
  for (let col=0;col<butacasCine[fila].length, col++){
    if (butacas[fila][col]=='libre')
      libres= true;
  }
  if (libres==true)
    console.log('En la fila' + fila + 'hay butacas libres');
}
```


Recorrer por fila

Analicemos el código

```
let libres= false;
for (let fila=0;fila<butacasCine.length;fila++){ //Este for
  libres=false; //itera por cada fila y mide cuántas filas hay
  for (let col=0;col<butacasCine[fila].length, col++){
    if (butacas[fila][col]=='libre') //este segundo for
      libres= true; //comprueba cada elemento de
  } //la fila, o sea cada columna
  if (libres==true)
    console.log('En la fila' + fila + 'hay butacas libres');
} //Luego de recorrer una fila muestra el resultado
```

Recorrer por fila



Para recorrer por filas debemos posicionarnos en cada fila y recorrerla completa, para saber cuántas filas tenemos medimos **matriz.length**.

5 | Recorrer por columna

Recorrer por columna

Ahora necesitamos ver si hay columnas completamente libres.

ocupado	libre	libre	libre
libre	libre	libre	libre
libre	libre	ocupado	ocupado
ocupado	ocupado	ocupado	ocupado
libre	libre	libre	libre

columna 0

columna 1

columna 2

columna 3

Vamos a tener que comprobar **para cada columna** si todas sus butacas están libres.

Recorrer por columna

Recorremos la matriz con dos for: el primero recorre cada columna y el segundo cada elemento de la columna.

```
let ocupado= true;
for (let col=0;col<butacasCine[0].length;col++){
  ocupado=true;
  for (fila=0;fila<butacasCine.length, fila++){
    if (butacas[fila][col]=='ocupado')
      ocupado= false;
  }
  if (ocupado==false)
    console.log('En la ' + col + 'hay butacas libres');
}
```

Recorrer por columna

Analicemos el código

```
let ocupado= true;
for (let col=0;columna<butacasCine[0].length;col++){
    ocupado=true; //Recorremos cada columna, para saber cuántas
                  // veces, medimos cuántos elementos hay en una fila
    for (let fila=0;fila<butacasCine.length, fila++){// y por
{}      if (butacas[fila][col]=='ocupado') // cada columna,
        ocupado= false;                //recorremos todas sus filas
    }
    if (ocupado==false)
        console.log('En la ' + col + 'hay butacas libres');
}
```

Recorrer por columna



Para recorrer por columnas debemos posicionarnos en cada columna y recorrerla completa. Para saber cuántas columnas tenemos, medimos alguna fila de la matriz, por ejemplo la fila 0 **`matriz[0].length`** o **`matriz[i].length`**, si `i` tiene el valor de una fila.

Recorrer una matriz

Recorrer una matriz

En los ejemplos llamamos fila y col a las variables de los for, pero podrían ser i, j. Lo importante es recordar que el primer subíndice de la matriz está nombrando la fila y, el segundo, la columna.

DigitalHouse>