

*DESARROLLO DE ENTORNO WEB  
CLIENTE*

**UD3**

# **ACTIVIDADES JAVASCRIPT**

A large yellow square containing the letters 'JS' in a bold, dark grey, sans-serif font, representing JavaScript.

DAW 2W

**ALBERT PÉREZ BALEYTO**

# ÍNDICE

ÍNDICE.....	2
ENUNCIADO.....	3
PREPARACIÓN.....	4
REALIZACIÓN.....	5
TRAZA.....	11
FIN DEL SCRIPT.....	19

# ENUNCIADO

ACTIVIDAD 4: Depuración			
<b>Código:</b> UD3.3	<b>CE:</b> RA2.h	<b>Módulo:</b> DWES	<b>Estimación:</b> 20 min
<p>Haz una traza de ejecución de alguno de los programas de los apartados anteriores. Haz capturas de pantalla donde se muestre las tareas de depuración que has realizado, explicándolas. Has de incluir:</p> <ul style="list-style-type: none"><li>• Puntos de ruptura</li><li>• Cambio de valor de las variables</li><li>• Inspección de variables/expresiones (seguimiento en Chrome)</li><li>• Mostrar la ejecución paso a paso</li></ul>			
<b>Enlaces y otras cuestiones:</b>			

# PREPARACIÓN

Procedo a realizar la traza del ejercicio 3.3.1 en el que se nos pide a través de dos ventanas prompt la cantidad de columnas y filas de una tabla y que una vez introducidos procede a imprimirla en pantalla.

El código JS del ejercicio es el siguiente:

```
let filas=0;
let columnas=0;
filas=parseInt(prompt("Vamos a montar una tabla. Cuantas filas quieres que tenga?"));
columnas=parseInt(prompt("Cuantas columnas quieres que tenga?"));

document.write("<table style='border-collapse: collapse; width: 75%;'>");
for(let i = 1; i <= filas ; i++){
    document.write("<tr>");

    for(let j = 1; j <= columnas ; j++){
        document.write("<td style='border: 2px solid black; padding: 12px; text-align: center;'>");
        document.write("CELDA "+i+": "+j);
        document.write("</td>");
    }
    document.write("</tr>");
}

document.write("</table>");
```

# REALIZACIÓN

Procedo a cargar el html en el navegador y abro el depurador dentro de las herramientas de desarrollador escogiendo el código javascript.

The screenshot shows a web browser window with the title "Actividades Cliente". The address bar shows the URL: `b%3A/Cerbell%20Beto/GSDAW/EntornoCliente/UD3Javascript/EjerciciosUD3JS/JSUD3...`. The page content displays the heading "Bienvenidos a la pagina de actividades" and "Desarrollo Aplicaciones Web Cliente". Below this, a paragraph states: "Vamos a realizar varios ejercicios de la unidad 3 sobre **JavaScript** o JS para los amigos. Esta es la sección donde haremos las pruebas. Hay que ir desbloqueando los bloques en el archivo JS."

CELDA 1:1	CELDA 1:2	CELDA 1:3	CELDA 1:4
CELDA 2:1	CELDA 2:2	CELDA 2:3	CELDA 2:4
CELDA 3:1	CELDA 3:2	CELDA 3:3	CELDA 3:4
CELDA 4:1	CELDA 4:2	CELDA 4:3	CELDA 4:4

Below the table, the browser's developer tools are open, showing the "Fuentes" (Sources) tab. The file `actividades_cliente.js` is selected in the file explorer. The code editor shows the following JavaScript code:

```
84 }
85 */
86
87
88
89
90 // 3.3.1 Realiza un programa en javascript donde se le pregunte al usu
91 // filas y columnas indicadas por el usuario. En cada celda deberá mos
92
93 let filas=0;
94 let columnas=0;
95 filas=parseInt(prompt("Vamos a montar una tabla. Cuantas filas quieres
96 columnas=parseInt(prompt("Cuantas columnas quieres que tenga?"));
97
98 document.write("<table style='border-collapse: collapse; width: 75%;
99 for(let i = 1; i <= filas ; i++){
100     document.write("<tr>");
101
102     for(let j = 1; j <= columnas ; j++){
103         document.write("<td style='border: 2px solid black; padding: 1
104             document.write("CELDA "+i+":"+j);
105             document.write("</td>");
106         }
107     document.write("</tr>");
108 }
109
110 document.write("</table>");
111
112
113
114
115 /*
```

The status bar at the bottom indicates "Línea 99, columna 13" and "Cobertura: no disponible".

Decido poner puntos de ruptura en los dos for que contiene el código y en la impresión de las coordenadas en la celda y añado en expresiones vigiladas o seguimiento las variables filas y columnas e i y j de los for. A posterior procedo a recargar la web para iniciar la depuración. Introduzco en las ventanas prompt que quiero una tabla con dos filas y cuatro columnas y luego ya se para en el primer punto de ruptura.

**Bienvenidos a la pagina de actividades**

## Desarrollo Aplicaciones Web Cliente

Vamos a realizar varios ejercicios de la unidad 3 sobre **JavaScript** o JS para los amigos.

Esta es la sección donde haremos las pruebas.  
Hay que ir desbloqueando los bloques en el archivo JS.

```
83 document.write("<p><strong>'Ha introducido mal su elección. Vuelva a intentarlo.'</strong></p>");
84 }
85 */
86
87
88
89
90 // 3.3.1 Realiza un programa en javascript donde se le pregunte al usuario
91 // filas y columnas indicadas por el usuario. En cada celda deberá mostrar
92
93 let filas=0;
94 let columnas=0;
95 filas=parseInt(prompt("Vamos a montar una tabla. Cuantas filas quieres que tenga?"));
96 columnas=parseInt(prompt("Cuantas columnas quieres que tenga?"));
97
98 document.write("<table style='border-collapse: collapse; width: 75%; text-align: center;'>");
99 for(let i = 1; i <= filas ; i++){
100 document.write("<tr>");
101
102     for(let j = 1; j <= columnas ; j++){
103         document.write("<td style='border: 2px solid black; padding: 10px;'>");
104         document.write("CELDA "+i+"-"+j);
105         document.write("</td>");
106     }
107     document.write("<tr>");
108 }
109
110 document.write("</table>");
111
112
113
114
115 /*
116 // 3.3.2 Copia el código anterior y realiza una modificación donde se
117 // usuario si quiere continuar pintando la tabla o si abandona la ejecución
118
119 let filas=0;
120 let columnas=0;
121 filas=parseInt(prompt("Vamos a montar una tabla. Cuantas filas quieres que tenga?"));
122 columnas=parseInt(prompt("Cuantas columnas quieres que tenga?"));
123
124 // 3.3.2 Creo un nuevo juego de variables y prompts para saber las coordenadas
125 let stopfilas=0;
126 let stopcolumnas=0;
127 stopfilas=parseInt(prompt("En que fila quieres que hagamos la pausa de ejecución?"));
128 stopcolumnas=parseInt(prompt("Y en que columna?"));
```

**Pausado en el punto de interrupción**

**Seguimiento**

- filas: 2
- columnas: 4
- "i ": <no disponible>
- "j ": <no disponible>

**Puntos de interrupción**

- ☐ Pausar en las excepciones no detectadas
- ☐ Pausar en las excepciones detectadas

**actividades\_cliente.js**

- ☒ for(let i = 1; i <= f... 99
- ☒ for(let j = 1; j <= ... 102
- ☒ document.write("<CELDA... 104

**Alcance**

- Bloqueo**
  - i: <valor no disponible>
- Secuencia de comandos**
  - columnas: 4
  - filas: 2
- Global Window

**Pila de llamadas**

- (anónimo) actividades\_cliente.js:99
- Puntos de interrupción de XHR/obten...
- Puntos de interrupción DOM
- Procesadores globales
- Puntos de interrupción del procesador
- Puntos de interrupción de infracción C

Línea 99, columna 13 Cobertura: no disponible

Como podemos ver en el seguimiento de las variables filas tiene un valor de 2 y columnas de 4. Como hemos puesto el primer punto de ruptura en la línea de el primer for la realización del código se para al principio de la línea antes del for con lo que no se ha inicializado y el valor de i sigue sin definirse.

Procedo a adelantar un paso, f9, e i recibe el valor 1 determinado por el for. La lectura se para ahora en la línea 100. Le doy a salir función actual, f11, y saltamos a la línea 102 que es donde tenemos el siguiente punto de ruptura.

**Bienvenidos a la pagina de actividades**

## Desarrollo Aplicaciones Web Cliente

Vamos a realizar varios ejercicios de la unidad 3 sobre **JavaScript** o JS para los amigos.

Esta es la sección donde haremos las pruebas.  
Hay que ir desbloqueando los bloques en el archivo JS.

```
82 default:
83   document.write("<p><strong>'Ha introducido mal su elección. V
84   }
85   */
86
87
88
89
90 // 3.3.1 Realiza un programa en javascript donde se le pregunte al usu
91 // filas y columnas indicadas por el usuario. En cada celda deberá mos
92
93 let filas=0;
94 let columnas=0;
95 filas=parseInt(prompt("Vamos a montar una tabla. Cuantas filas quieres
96 columnas=parseInt(prompt("Cuantas columnas quieres que tenga?"));
97
98 document.write("<table style='border-collapse: collapse; width: 75%;'
99 for(let i = 1; i <= filas ; i++){
100   document.write("<tr>");
101
102   for(let j = 1; j <= columnas ; j++){
103     document.write("<td style='border: 2px solid black; padding: 1
104     document.write("CELDA "+i+"-"+j);
105     document.write("</td>");
106   }
107   document.write("<tr>");
108 }
109
110 document.write("</table>");
111
112
113
114
115 /*
116 // 3.3.2 Copia el código anterior y realiza una modificación donde se
117 // usuario si quiere continuar pintando la tabla o si abandona la ejec
118
119 let filas=0;
120 let columnas=0;
121 filas=parseInt(prompt("Vamos a montar una tabla. Cuantas filas quieres
122 columnas=parseInt(prompt("Cuantas columnas quieres que tenga?"));
123
124 // 3.3.2 Creo un nuevo juego de variables y prompts para saber las coor
125 let stopfilas=0;
126 let stopcolumnas=0;
127 stopfilas=parseInt(prompt("En que fila quieres que hagamos la pausa de
```

**Debugger**

**Pausado en el punto de interrupción**

**Seguimiento**

- filas: 2
- columnas: 4
- i: 1
- j: <no disponible>

**Puntos de interrupción**

- ☐ Pausar en las excepciones no dete...
- ☐ Pausar en las excepciones detecta...

**actividades\_cliente.js**

- ☒ for(let i = 1; i <= f... 99
- ☒ for(let j = 1; j <= ... 102
- ☒ document.write("CELD... 104

**Alcance**

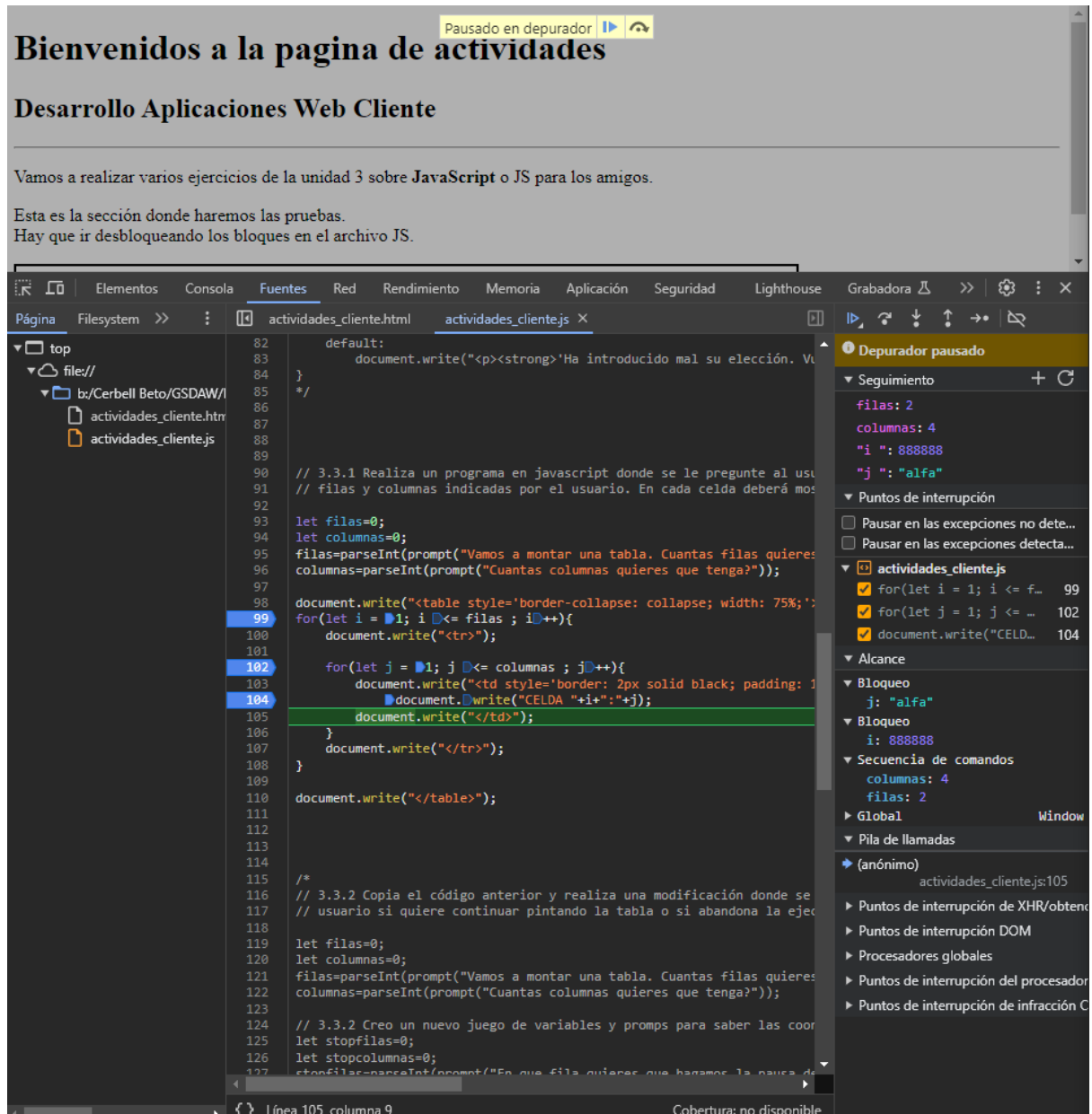
- Bloqueo**
  - j: <valor no disponible>
- Bloqueo**
  - i: 1
- Secuencia de comandos**
  - columnas: 4
  - filas: 2
- Global** Window
- Pila de llamadas**
  - (anónimo) actividades\_cliente.js:102
- Puntos de interrupción de XHR/obten...**
- Puntos de interrupción DOM**
- Procesadores globales**
- Puntos de interrupción del procesador**
- Puntos de interrupción de infracción C**

Línea 102, columna 17

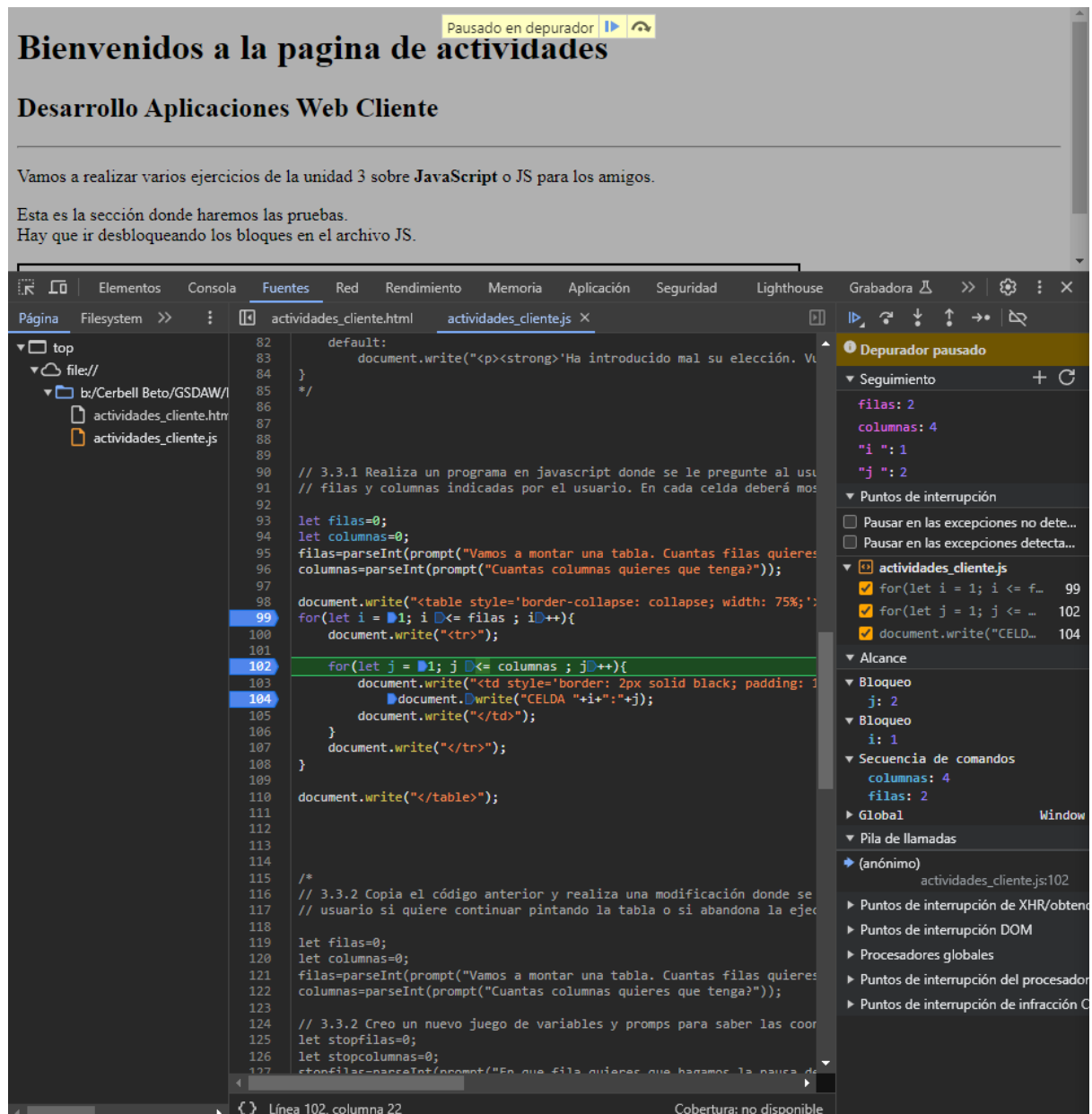
Cobertura: no disponible

Volvemos a saltar hasta el siguiente punto de ruptura que es el que imprime las coordenadas dentro de la celda de la tabla y procedo a modificar en alcance el valor de las variables  $j$  e  $i$  para que la impresión de las coordenadas de la primera casilla sea diferente.

Le doy a f9, un paso, y vuelvo a modificar los valores de j e i a 1 y vuelvo a mover un paso para que cargue nuevamente las variables.







Desconecto los puntos de ruptura y le doy a f8 para que reanude todo el código para poder ver las modificaciones realizadas.

# Bienvenidos a la pagina de actividades

## Desarrollo Aplicaciones Web Cliente

Vamos a realizar varios ejercicios de la unidad 3 sobre **JavaScript** o JS para los amigos.

Esta es la sección donde haremos las pruebas.

Hay que ir desbloqueando los bloques en el archivo JS.

CELDA 888888:alfa	CELDA 1:2	CELDA 1:3	CELDA 1:4
CELDA 2:1	CELDA 2:2	CELDA 2:3	CELDA 2:4

```
82 default:
83     document.write("<p><strong>'Ha introducido mal su elección. V...
84 }
85 */
86
87
88
89
90 // 3.3.1 Realiza un programa en javascript donde se le pregunte al usu
91 // filas y columnas indicadas por el usuario. En cada celda deberá mos
92
93 let filas=0;
94 let columnas=0;
95 filas=parseInt(prompt("Vamos a montar una tabla. Cuantas filas quieres
96 columnas=parseInt(prompt("Cuantas columnas quieres que tenga?"));
97
98 document.write("<table style='border-collapse: collapse; width: 75%;
99 for(let i = 1; i <= filas ; i++){
100     document.write("<tr>");
101
102     for(let j = 1; j <= columnas ; j++){
103         document.write("<td style='border: 2px solid black; padding: 1
104             document.write("CELDA "+i+"-"+j);
105         document.write("</td>");
106     }
107     document.write("</tr>");
108 }
109 document.write("</table>");
110
111
112
113
114
115 /*
116 // 3.3.2 Copia el código anterior y realiza una modificación donde se
117 // usuario si quiere continuar pintando la tabla o si abandona la eje
118
119 let filas=0;
120 let columnas=0;
```

Como podemos ver en la primera casilla de la tabla aparecen las modificaciones de las variables de las coordenadas realizadas y el resto impreso correctamente.

# TRAZA

Esta aplicación en Java Script crea un bloque de código html para representar una tabla de líneas y columnas determinadas por el usuario al momento de cargar el script a través de ventanas prompt.

Voy a proceder a detallar la ejecución paso a paso con una traza y con alguna captura significativa. Para no alargar la traza más de lo necesario daré valor fila 1 y columna 2.

Utilizaré el siguiente patrón:

Paso:

Código:

Explicación:

Variables:

filas    columnas    i        j

código html/Final:

-----

Lanzo script

Paso: 1

Código:        let filas=0;

Explicación: Inicializo la variable filas con valor 0

Variables:

filas    columnas    i        j

0

código html/Final:

-----

Paso: 2

Código:        let columnas=0;

Explicación: Inicializo la variable columnas con valor 0

Variables:

filas    columnas    i        j

0        0

código html/Final:

-----

Paso: 3

Código: `filas=parseInt(prompt("Vamos a montar una tabla. Cuantas filas quieres que tenga?"));`

Explicación: Le doy valor a filas, tras convertir lo dado a integer, de lo que devuelve una ventana prompt con un texto. Doy valor 1

Variables:

filas	columnas	i	j
1	0		

código html/Final:

---

Paso: 4

Código: `columnas=parseInt(prompt("Cuantas columnas quieres que tenga?"));`

Explicación: Le doy valor a columnas, tras convertir lo dado a integer, de lo que devuelve una ventana prompt con un texto. Doy valor 2

Variables:

filas	columnas	i	j
1	2		

código html/Final:

---

Paso: 5

Código: `document.write("<table style='border-collapse: collapse; width: 75%;'>");`

Explicación: Escribo la primera línea de html para la creación de la tabla.

Variables:

filas	columnas	i	j
1	2		

código html/Final:

```
<table style='border-collapse: collapse; width: 75%;'>
```

---

Paso:6

Código: `for(let i = 1; i <= filas ; i++){`

Explicación: Entramos en el primer for que inicializa i a 1, seguirá dando vueltas mientras i sea igual o menor a variable filas y al acabar las ordenes incrementara en uno a i.

Variables:

filas	columnas	i	j
1	2	1	

código html/Final:

```
<table style='border-collapse: collapse; width: 75%;'>
```

---

Paso:7

Código: `document.write("<tr>");`

Explicación: añade al html la etiqueta de apertura tr para la fila

Variables:

filas	columnas	i	j
1	2	1	

código html/Final:

```
<table style='border-collapse: collapse; width: 75%;'>
<tr>
```

Paso:8

Código: `for(let j = 1; j <= columnas ; j++){`

Explicación: Entramos en el for anidado que inicializa j a 1, seguirá dando vueltas mientras j sea igual o menor a variable columna y al acabar las ordenes incrementara en uno a j.

Variables:

filas	columnas	i	j
1	2	1	1

código html/Final:

```
<table style='border-collapse: collapse; width: 75%;'>
<tr>
```

Paso:9

Código: `document.write("<td style='border: 2px solid black; padding: 12px; text-align: center;'>");`

Explicación: Añade al html la etiqueta td para columna con style.

Variables:

filas	columnas	i	j
1	2	1	1

código html/Final:

```
<table style='border-collapse: collapse; width: 75%;'>
<tr>
<td style='border: 2px solid black; padding: 12px; text-align:
center;'>
```

Paso: 10

Código: `document.write("CELDA "+i+"":"+j");`

Explicación: Añade al html el contenido de la celda dependiendo de los valores de las variables i y j .

Variables:

filas	columnas	i	j
1	2	1	1

código html/Final:

```
<table style='border-collapse: collapse; width: 75%;'>
<tr>
<td style='border: 2px solid black; padding: 12px; text-align:
center;'>
CELDA 1:1
```

---

Paso: 11

Código: `document.write("</td>");`

Explicación: Añade al html la etiqueta de cierre de la td.

Variables:

filas	columnas	i	j
1	2	1	1

código html/Final:

```
<table style='border-collapse: collapse; width: 75%;'>
<tr>
<td style='border: 2px solid black; padding: 12px; text-align:
center;'>
CELDA 1:1
</td>
```

---

Paso: 12

Código: `}`

Explicación: Es el cierre del for anidado con lo que procedemos al último paso que es aumentar j en 1.

Variables:

filas	columnas	i	j
1	2	1	2

código html/Final:

```
<table style='border-collapse: collapse; width: 75%;'>
<tr>
<td style='border: 2px solid black; padding: 12px; text-align:
center;'>
CELDA 1:1
</td>
```

---

## Paso: 13

Código: `for(let j = 1; j <= columnas ; j++){`

Explicación: Volvemos a revisar el for y j sigue siendo menor o igual a columnas con lo que volvemos a entrar en el for.

Variables:

filas	columnas	i	j
1	2	1	2

código html/Final:

```
<table style='border-collapse: collapse; width: 75%;'>
<tr>
<td style='border: 2px solid black; padding: 12px; text-align:
center;'>
CELDA 1:1
</td>
```

## Paso:14

Código: `document.write("<td style='border: 2px solid black; padding: 12px; text-align: center;'>");`

Explicación: Añade al html la etiqueta td para columna con style.

Variables:

filas	columnas	i	j
1	2	1	2

```
<table style='border-collapse: collapse; width: 75%;'>
<tr>
<td style='border: 2px solid black; padding: 12px; text-align:
center;'>
CELDA 1:1
</td>
<td style='border: 2px solid black; padding: 12px; text-align:
center;'>
```

## Paso: 15

Código: `document.write("CELDA "+i+" ":"+j);`

Explicación: Añade al html el contenido de la celda dependiendo de los valores de las variables i y j .

Variables:

código html/Final:

1	2	1	2
---	---	---	---

```
<table style='border-collapse: collapse; width: 75%;'>
```

```
<tr>
<td style='border: 2px solid black; padding: 12px; text-align:
center;'>
CELDA 1:1
</td>
<td style='border: 2px solid black; padding: 12px; text-align:
center;'>
CELDA 1:2
```

---

Paso: 16

Código: `document.write("</td>");`

Explicación: Añade al html la etiqueta de cierre de la td.

Variables:

filas	columnas	i	j
1	2	1	2

código html/Final:

```
<table style='border-collapse: collapse; width: 75%;'>
<tr>
<td style='border: 2px solid black; padding: 12px; text-align:
center;'>
CELDA 1:1
</td>
<td style='border: 2px solid black; padding: 12px; text-align:
center;'>
CELDA 1:2
</td>
```

---

Paso: 17

Código: `}`

Explicación: Es el cierre del for anidado con lo que procedemos al último paso que es aumentar j en 1.

Variables:

filas	columnas	i	j
1	2	1	3

código html/Final:

```
<table style='border-collapse: collapse; width: 75%;'>
<tr>
<td style='border: 2px solid black; padding: 12px; text-align:
center;'>
CELDA 1:1
</td>
```



```
<td style='border: 2px solid black; padding: 12px; text-align:
center;'>
CELDA 1:2
</td>
```

---

Paso: 18

Código: `for(let j = 1; j <= columnas ; j++){`

Explicación: Volvemos a revisar el for j es mayor que la variable columnas con lo que nos saltamos el for.

Variables:

filas	columnas	i	j
1	2	1	3

código html/Final:

```
<table style='border-collapse: collapse; width: 75%;'>
<tr>
<td style='border: 2px solid black; padding: 12px; text-align:
center;'>
CELDA 1:1
</td>
<td style='border: 2px solid black; padding: 12px; text-align:
center;'>
CELDA 1:2
</td>
```

---

Paso: 19

Código: `document.write("</tr>");`

Explicación: Añadimos al html la etiqueta cierre de la fila.

Variables:

filas	columnas	i	j
1	2	1	3

código html/Final:

```
<table style='border-collapse: collapse; width: 75%;'>
<tr>
<td style='border: 2px solid black; padding: 12px; text-align:
center;'>
CELDA 1:1
</td>
<td style='border: 2px solid black; padding: 12px; text-align:
center;'>
CELDA 1:2
</td>
</tr>
```

---

Paso: 20

Código: }

Explicación: Es el cierre del for inicial con lo que procedemos al último paso que es aumentar i en 1.

Variables:

filas	columnas	i	j
1	2	2	3

código html/Final:

```
<table style='border-collapse: collapse; width: 75%;'>
<tr>
<td style='border: 2px solid black; padding: 12px; text-align:
center;'>
CELDA 1:1
</td>
<td style='border: 2px solid black; padding: 12px; text-align:
center;'>
CELDA 1:2
</td>
</tr>
```

---

Paso: 21

Código: `for(let i = 1; i <= filas ; i++){`

Explicación: Volvemos a revisar el for i es mayor que la variable filas con lo que nos saltamos el for.

Variables:

filas	columnas	i	j
1	2	2	3

código html/Final:

```
<table style='border-collapse: collapse; width: 75%;'>
<tr>
<td style='border: 2px solid black; padding: 12px; text-align:
center;'>
CELDA 1:1
</td>
<td style='border: 2px solid black; padding: 12px; text-align:
center;'>
CELDA 1:2
</td>
</tr>
```

---

Paso: 22

Código: `document.write("</table>");`

Explicación: Añadimos al html la etiqueta cierre de la tabla.

Variables:

filas	columnas	i	j
1	2	2	3

código html/Final:

```
<table style='border-collapse: collapse; width: 75%;'>
<tr>
<td style='border: 2px solid black; padding: 12px; text-align:
center;'>
CELDA 1:1
</td>
<td style='border: 2px solid black; padding: 12px; text-align:
center;'>
CELDA 1:2
</td>
</tr>
</table>
```

---

## FIN DEL SCRIPT

El código que nos devuelve el script introduciendo los valores 1 en filas y 2 en columnas.

```
<table style='border-collapse: collapse; width: 75%;'>
<tr>
<td style='border: 2px solid black; padding: 12px; text-align:
center;'>
CELDA 1:1
</td>
<td style='border: 2px solid black; padding: 12px; text-align:
center;'>
CELDA 1:2
</td>
</tr>
</table>
```