



## Práctica 1.

### Parte 1: Otros entornos de pruebas

**1.- Codepen.** Créate una cuenta en la web [codepen.io](https://codepen.io) para trabajar en clase. Pruébala introduciendo algo de código HTML y CSS. En la parte de JavaScript coloca el script visto en la prueba inicial del módulo:

```
var x, respuesta, contador;
contador =4;
for (x=1;x<=4;x++){
    respuesta=prompt("te caigo bien");
    if (respuesta=="si") {
        contador = contador+2;
    }
    if (respuesta=="no") {
        contador = contador-1;
    }
}
document.write(contador);
```

**Comprueba que el script funciona correctamente.**

Haz dos capturas de pantalla en las que se vean dos puntuaciones diferentes obtenidas al ejecutar el script.



Introduce "si" o "no" a la pregunta mostrada: 12

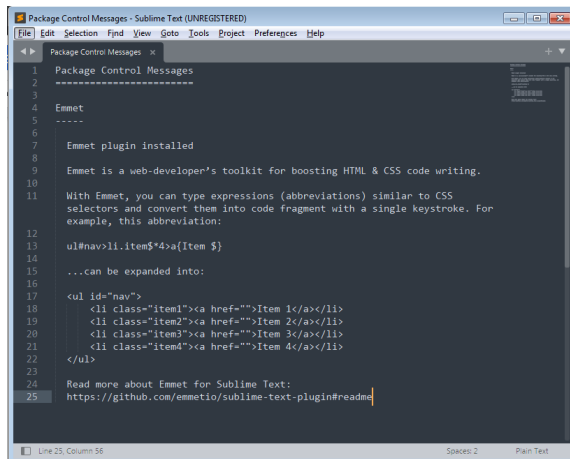


Introduce "si" o "no" a la pregunta mostrada: 0



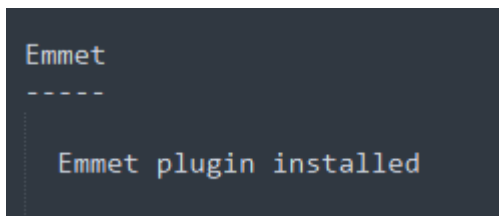
**2.- Sublime Text.** Para utilizar Sublime Text como entorno de pruebas realiza las siguientes acciones:

- Instala el editor de código Sublime Text.

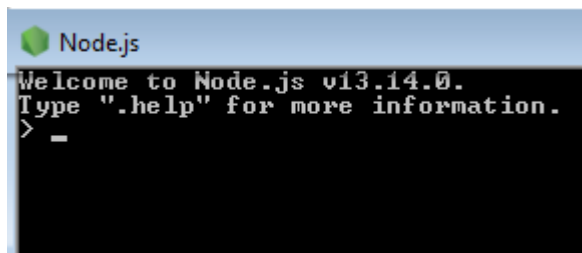


- Configurar Sublime Text instalando al menos el paquete Emmet. Puedes seguir las instrucciones que encontrarás en:

<https://www.nobledesktop.com/downloads/goodies/sublime-enhancements-edition1.2.pdf>



- Instala Node.js (<https://nodejs.org/es/download/>).



- Instalar browser-sync ejecutando la terminal de node con el comando:
  - o `npm install -g browser-sync`



```
C:\Users\alumno>npm install -g browser-sync@2.26.7
npm WARN deprecated urix@0.1.0: Please see https://github.com/lydell/urix#depre
ated
npm WARN deprecated resolve-url@0.2.1: https://github.com/lydell/resolve-url#depre
recated
npm WARN deprecated chokidar@2.1.8: Chokidar 2 will break on node v14+. Upgrade
to chokidar 3 with 15x less dependencies.
npm WARN deprecated debug@4.1.1: Debug versions >=3.2.0 <3.2.7 !! >=4 <4.3.1 hav
e a low-severity ReDos regression when used in a Node.js environment. It is reco
mmended you upgrade to 3.2.7 or 4.3.1. (https://github.com/visionmedia/debug/iss
ues/797)
npm WARN deprecated axios@0.19.0: Critical security vulnerability fixed in v0.21
.1. For more information, see https://github.com/axios/axios/pull/3410
added 334 packages, and audited 335 packages in 2m
3 packages are looking for funding
  run `npm fund` for details
13 vulnerabilities (2 low, 2 moderate, 8 high, 1 critical)
To address issues that do not require attention, run:
  npm audit fix
To address all issues (including breaking changes), run:
  npm audit fix --force
Run `npm audit` for details.
```

He tenido que instalar el comando añadiendo al final @2.26.7, para poder instalarlo.  
Este comando instala esa versión.

Por lo que el comando quedaría como (npm install -g browser-sync@2.26.7)

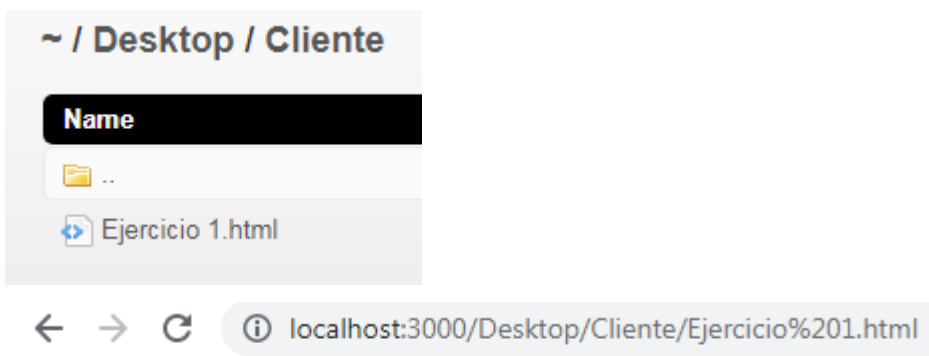
- Utiliza browser-sync para visualizar en tu navegador Chrome la web diseñada en el ejercicio 1. A tener en cuenta:

Para lanzar browser-sync utiliza el comando:

- o browser-sync start --server --directory --files "\*"

```
C:\Users\alumno>browser-sync start --server --directory --files "*"
[Browsersync] Access URLs:
-----
    Local: http://localhost:3000
  External: http://10.0.2.15:3000
-----
    UI: http://localhost:3001
  UI External: http://localhost:3001
-----
```

Si tu navegador por defecto no es Chrome necesitarás especificar también la opción  
--browser google-chrome



Introduce "si" o "no" a la pregunta mostrada: 12



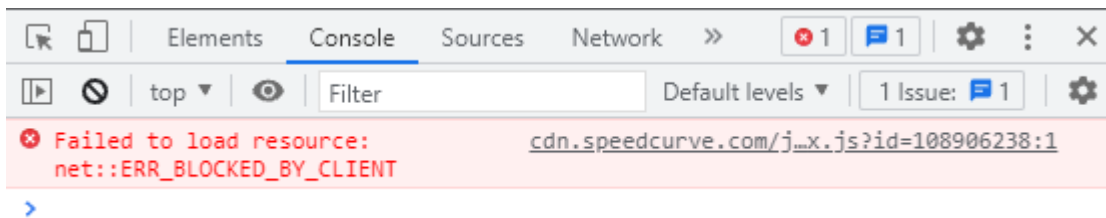
← → ↻ ⓘ localhost:3000/Desktop/Cliente/Ejercicio%201.html

Introduce "si" o "no" a la pregunta mostrada: 0

**Parte 2: Busca información en Internet y responde a las siguientes cuestiones:**

3. Indica cómo depurar el código usando Chrome o Firefox.

Para depurar código de Chrome o Firefox, debemos abrir el debugger que viene con el navegador web, para ello debemos presionar las teclas *Ctrl+Shift+J*, con este conjunto de teclas abriremos el debugger donde aparecerán los errores que tengamos.

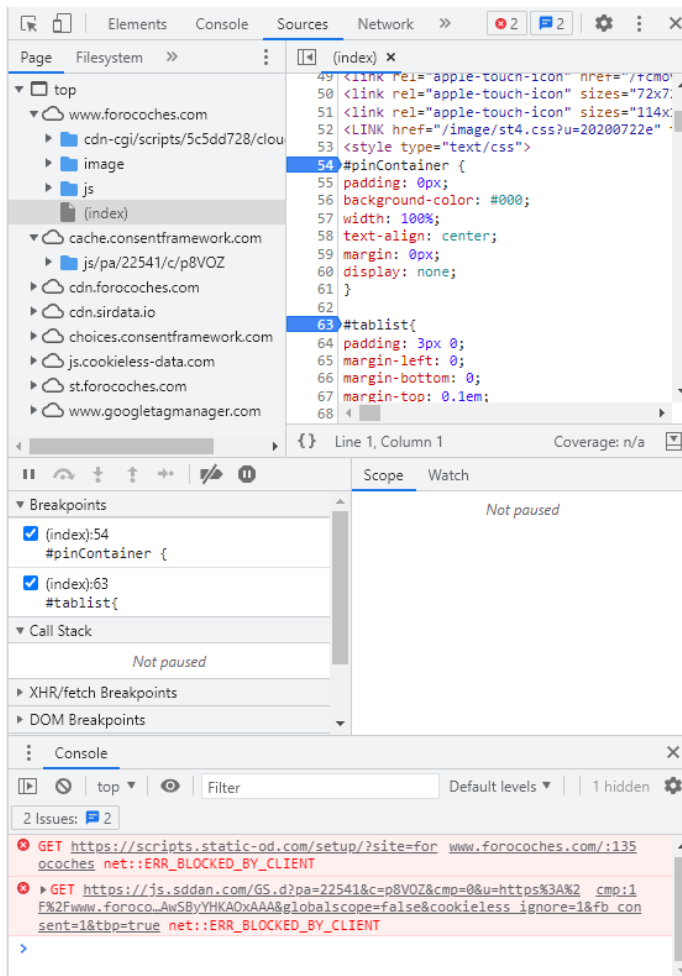


En esa foto vemos el error, para acceder ahora a los *Breakpoints* debemos clicar sobre *Sources* y luego accedemos a la página en cuestión, nos aparecerá una pantalla a su derecha en el que podemos asignar tantos *Breakpoints* como queramos, clicando sobre el número de línea que queramos.



**CFGS DESARROLLO DE APLICACIONES WEB.**  
**Desarrollo Web en Entorno Cliente**

**Pág.5de8**



**4. Define los siguientes acrónimos:**

- W3C:** El World Wide Web Consortium (W3C) es la principal organización internacional de normas para la World Wide Web.
- DHTML:** Es una colección de tecnologías utilizadas para crear sitios web interactivos y animados utilizando una combinación de un lenguaje de markup language, como HTML, un lenguaje client-side scripting, como JavaScript, un lenguaje de definición de presentación (como CSS), y el Document Object Model (DOM).
- WYSIWYG:** Es un acrónimo de What You See Is What You Get, es un sistema en el que el software de edición permite editar contenido en una forma que se asemeja a su apariencia cuando se imprime o se muestra como un producto terminado, como si fuese un documento impreso, página web, o presentación de diapositivas.
- API:** Es un tipo de interfaz de software, que ofrece un servicio a otras piezas de software. Un documento o estándar que describe cómo construir dicha conexión o interfaz se llama



una especificación de API.

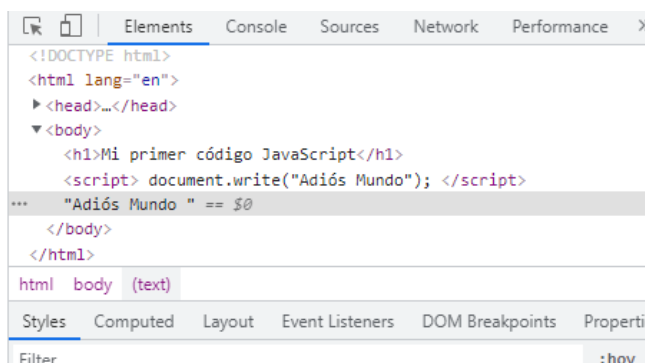
- e) **IDE:** Es una aplicación de software que proporciona instalaciones completas a los programadores informáticos para el desarrollo de software. Un IDE normalmente consiste en al menos un editor de código fuente, herramientas de automatización de compilación y un depurador.
- f) **CRM:** Es un proceso en el que una empresa u otra organización administra sus interacciones con los clientes, normalmente utilizando el análisis de datos para estudiar grandes cantidades de información.
- g) **WHATWG:** Es una comunidad de personas interesadas en la evolución de HTML y tecnologías relacionadas. Se formó en respuesta al lento desarrollo de los estándares web de W3C y la decisión de W3C de abandonar el HTML en favor de las tecnologías basadas en XML.

5. En esta actividad deberás abrir con Google Chrome las páginas web propuestas y utilizar la Consola Web para realizar pruebas básicas de funcionamiento.

- a) En primer lugar abriremos la página web adjunta (“HolaMundo.html”). Deberás documentar dos pruebas en el fichero “holaMundo.html”:
  1. Conseguir modificar un valor de algún elemento HTML y comprobar que el cambio se aplica en tiempo real.

**Mi primer  
código  
JavaScript**

Adiós Mundo

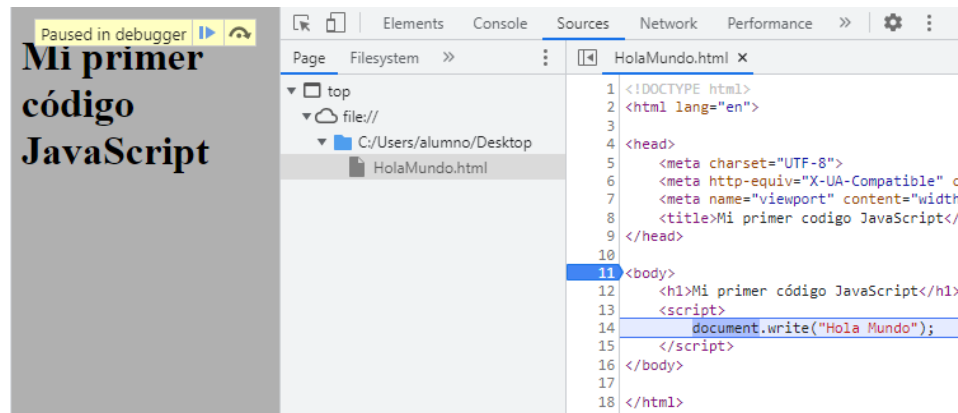


2. Establecer un “breakpoint” (punto de ruptura) del código Javascript de la página antes de que se muestre el mensaje “Hola mundo” y cuando comprobemos que se detenga, hacer que prosiga su ejecución.



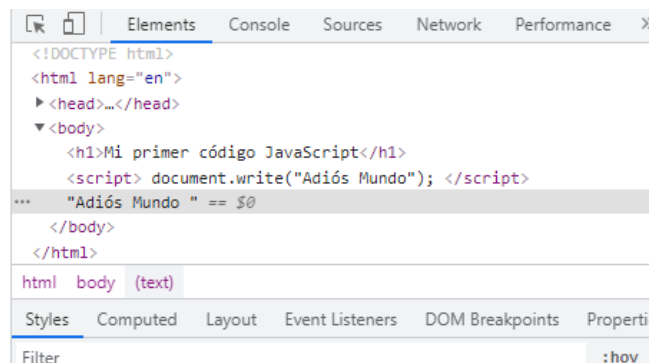
**CFGS DESARROLLO DE APLICACIONES WEB.**  
**Desarrollo Web en Entorno Cliente**

**Pág.7de8**



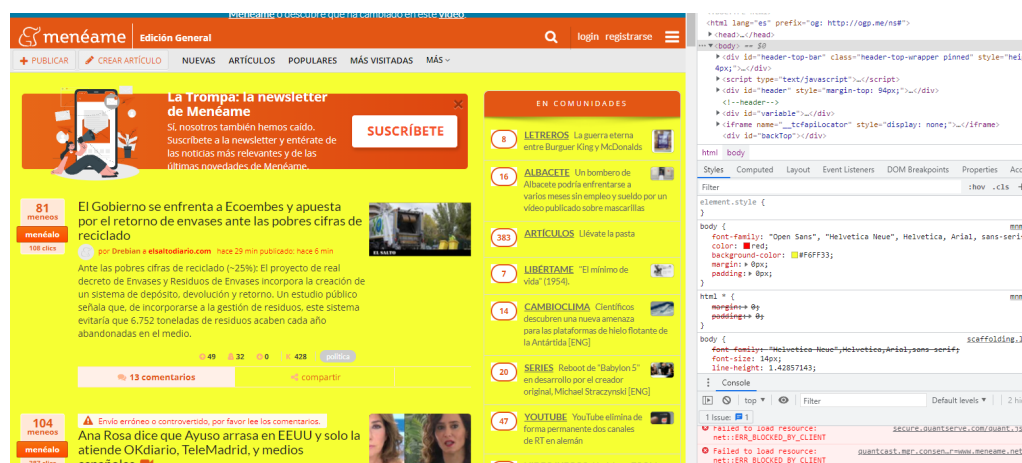
# Mi primer código JavaScript

Adiós Mundo



b) En segundo lugar abriremos <https://www.meneame.net/> y en ella deberás documentar dos pruebas:

1. Conseguir modificar algún elemento CSS.

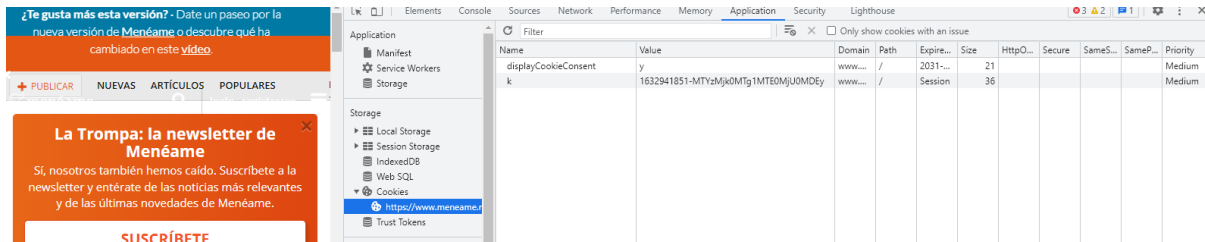


2. Inspeccionar las cookies existentes del sitio "meneame.net".



## CFGS DESARROLLO DE APLICACIONES WEB. Desarrollo Web en Entorno Cliente

Pág.8de8



HolaMundo.html

```
<html>
<head><title>Mi primer codigo JavaScript</title></head>
<body>
<h1>Mi primer código JavaScript</h1>
<script>
  document.write("Hola Mundo");
</script>
</body>
</html>
```

### CALIFICACIÓN

- El valor de esta práctica es de un **15% de la nota de prácticas de la evaluación.**
- Apartado bien: 2. Mal o regular: 0. Para cada punto deberán entregarse las capturas de pantalla que demuestren que se han realizado las acciones requeridas.
- Entregar con nombre: **U1\_T1\_Arq\_y\_herram\_nombreyapellidos.pdf.**
- Fecha límite de entrega: **30 de septiembre de 2021 a las 23:59 en AEducAR.**