

Conociendo las herramientas para desarrolladores de los navegadores

HERRAMIENTAS DE DESARROLLADORES EN DIFERENTES
NAVEGADORES

EMILIANO MONTESDEOCA DEL PUERTO

Índice

1. Enunciado.
2. Objetivo.
3. Parte uno.
 - a. Chrome.
 - b. Edge.
 - c. Firefox.
4. Parte dos.
 - a. Chrome.
 - b. Edge.
 - c. Firefox.
 - d. Comparación.
 - e. Opinión personal.

Enunciado

Parte 1

Utilizando los navegadores Firefox, Chrome y Edge, localizar y abrir la herramienta Consola. Dichas consolas probablemente mostrarán muchos avisos relativos a todos los errores que ha ido encontrando el navegador en las páginas que hayas ido visitando desde que lo iniciaste.

Utiliza el botón "Limpiar" para borrar dichos errores y mensajes de aviso; a continuación, recarga alguna de las páginas que estés visitando y comprueba qué nuevos errores han aparecido en la consola de errores. Filtra para mostrar únicamente los errores.

Parte 2

Aquí tienen un programa sencillo que calcula el exponencial de un número.

1) Ejecútala en el navegador Chrome. Proporciona el dígito que te solicita. ¿Que ha ido mal?

Establece un punto de ruptura en el código de la función, recarga la web y vuelve a introducir los dígitos. Monitoriza el valor de las variables a y n del formulario y utiliza la opción adecuada para trazar el programa y ver cómo varían los valores para detectar donde se ha producido el error. Puedes ayudarte del siguiente documento para poder realizar la traza en el navegador Chrome

Ten en cuenta que a medida que se pasa por las llamadas a las funciones recursivas, el subpanel "Call Stack" de la derecha te permite realizar un seguimiento de la pila de llamadas, junto con los parámetros que se pasan en cada llamada a la función.

2) ¿Podrías solucionar el problema que presenta el código que se te ha proporcionado?

3) Repetir la actividad en los navegadores Firefox y Edge. ¿Qué herramienta prefieres? Razona la respuesta, evaluando pros y contras de cada una de ellas.

Objetivo

El objetivo de esta práctica es utilizar las herramientas de desarrolladores de cada para poder ver que no funciona en el código y realizar depuración del código para poder hacer cambios en las funciones y hacer que funcionen.

También sirve para poder ver que ocurre cuando se activa la depuración de los eventos, por ejemplo, cuando se da clic en algún sitio.

Otra utilidad es ver el valor de las variables en caliente y poder cambiarlas o ver en que momento se realiza el error en una función.

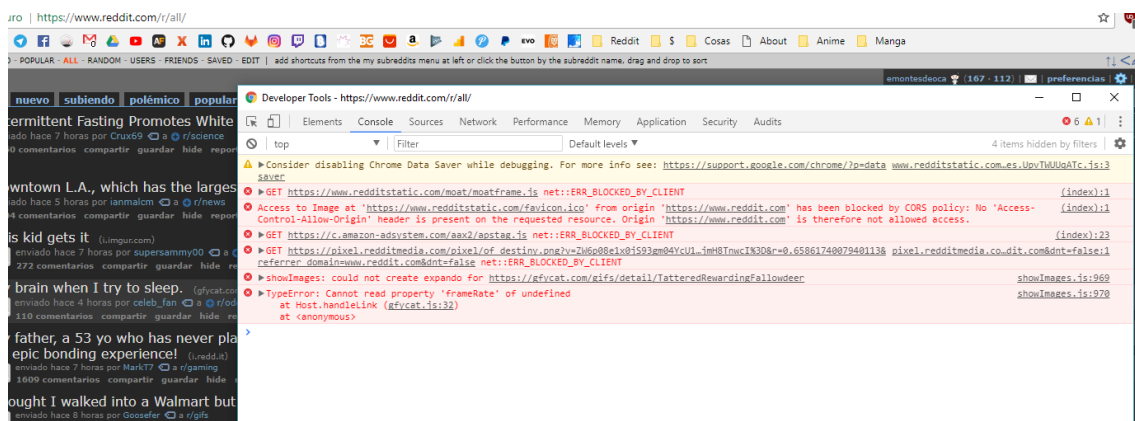
Parte uno

Utilizando los navegadores Firefox, Chrome y Edge, localizar y abrir la herramienta Consola. Dichas consolas probablemente mostrarán muchos avisos relativos a todos los errores que ha ido encontrando el navegador en las páginas que hayas ido visitando desde que lo iniciaste.

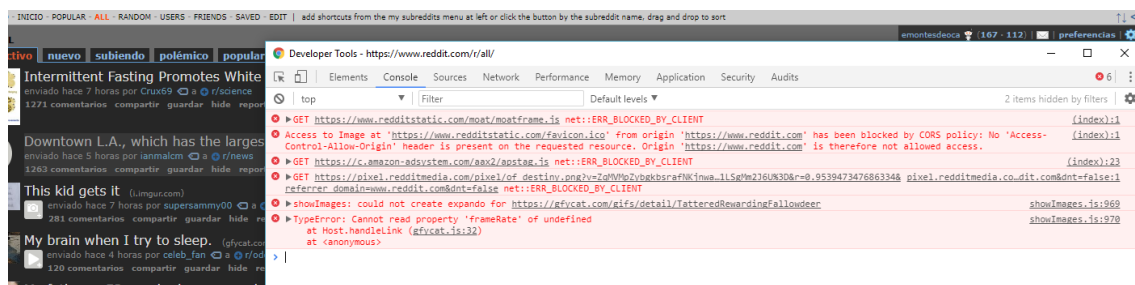
Utiliza el botón "Limpiar" para borrar dichos errores y mensajes de aviso; a continuación, recarga alguna de las páginas que estés visitando y comprueba qué nuevos errores han aparecido en la consola de errores. Filtra para mostrar únicamente los errores.

Chrome

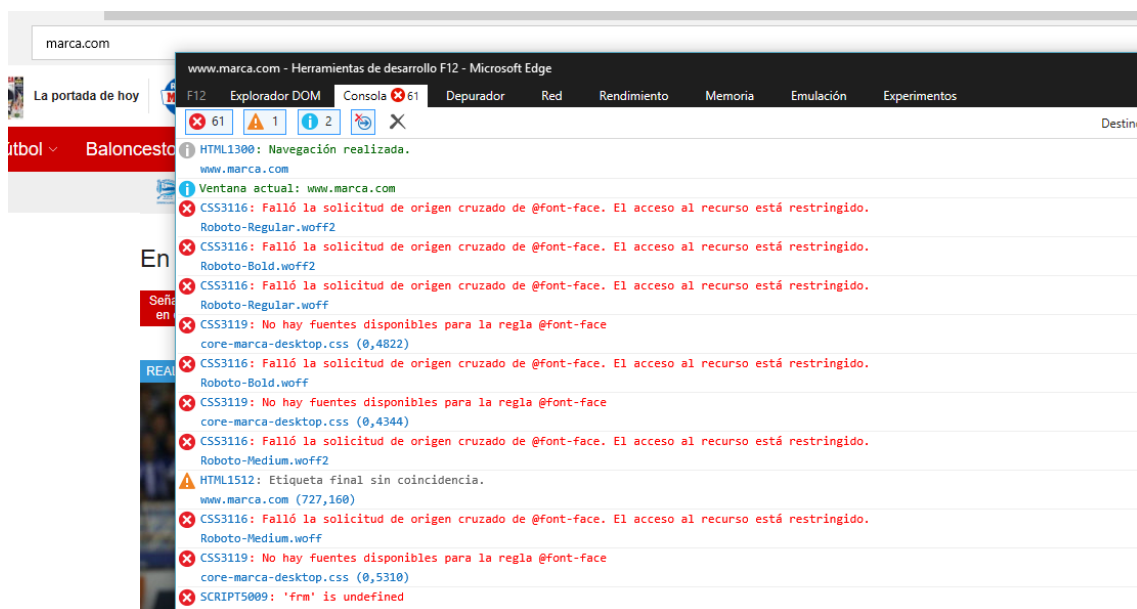
Utilizando la página web Reddit, vemos como se ven algunos errores a la hora de cargar la página, como vemos en la siguiente imagen:



Para Google Chrome, lo más útil para limpiarla consola es utilizar el atajo de teclado **Control + L**, pero si volvemos actualizar la página web y miramos algunos links, se vuelve a rellenar la consola de errores y advertencias:

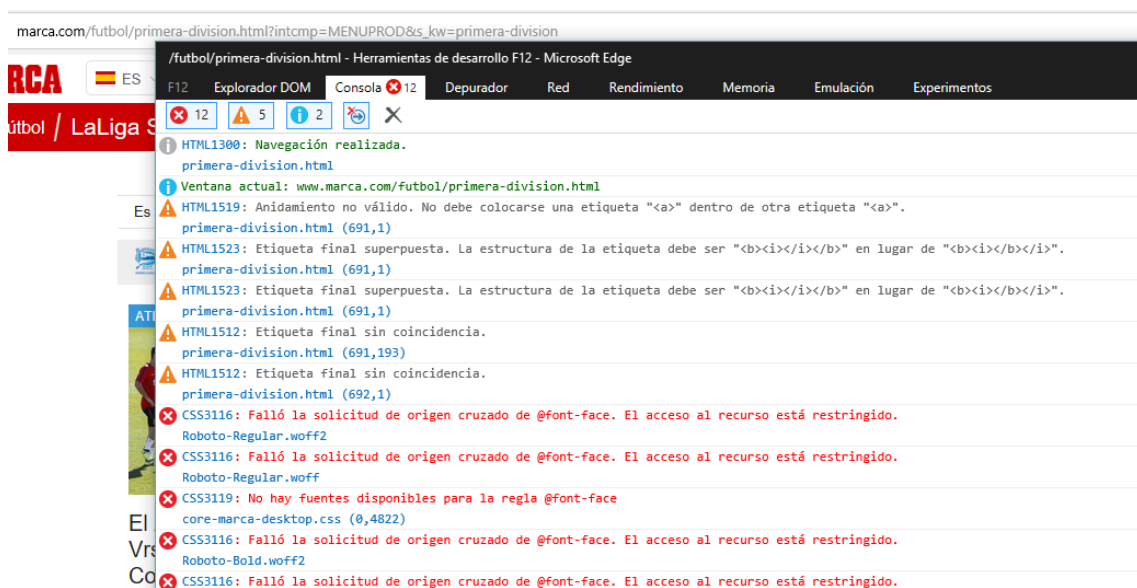


Para la navegación en Edge, utilizaremos la página web de Marca, que siempre tiene errores, y como podemos ver en la consola tiene bastantes:



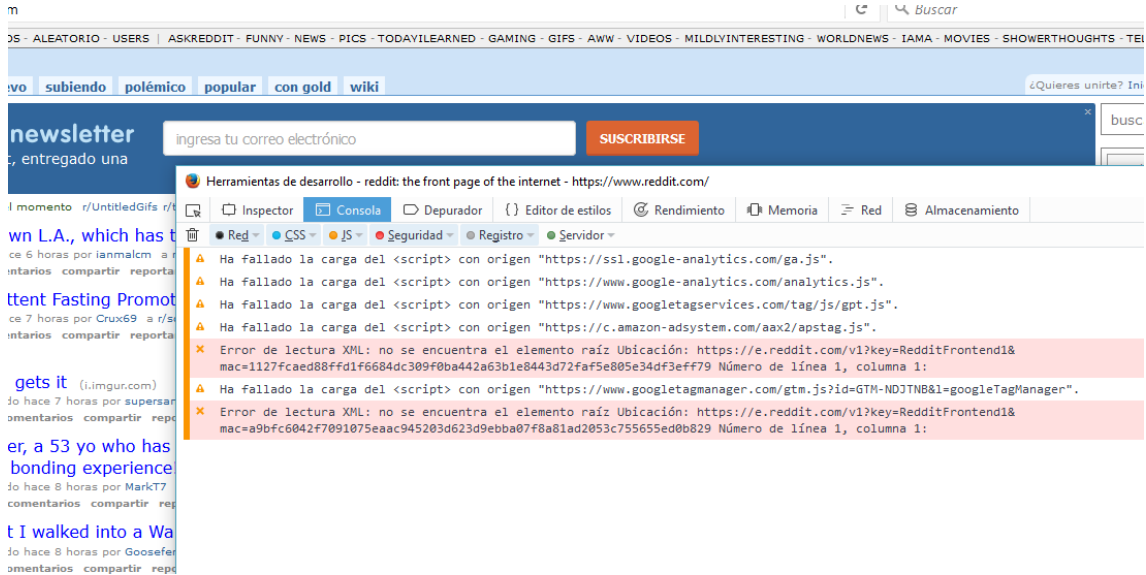
Para realizar una limpieza de la consola, basta como con el Chrome, realizar el atajo de teclado **Control + L**, y listo.

Para ver como se rellenan nuevos errores en la consola, nos cambiamos a otro artículo y vemos como ahora también aparecen otros tipos de errores.



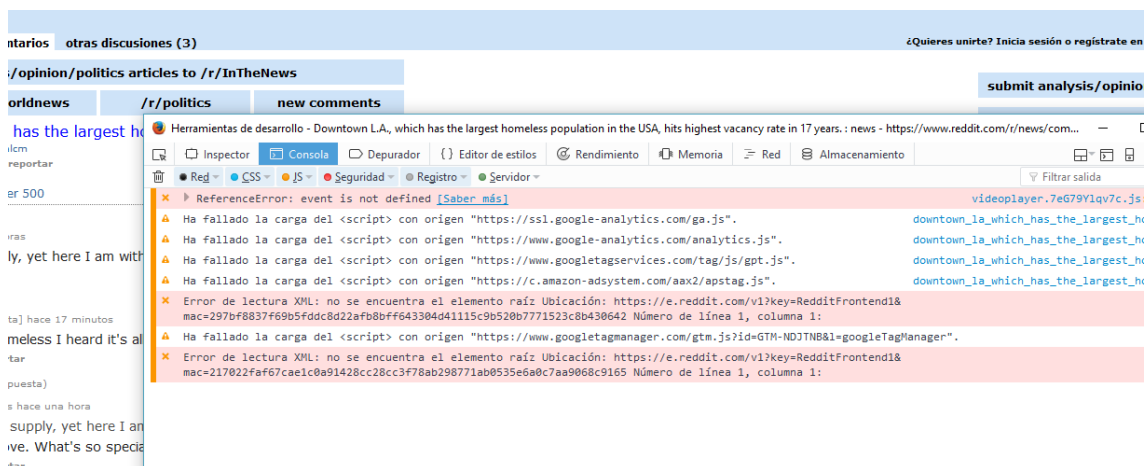
Firefox

Reutilizando la web de Reddit, en Firefox vemos los siguientes mensajes en la consola:



Para poder limpiar la consola, se pulsa en el botón que tiene de icono una papelera, justo a la izquierda de las pestañas.

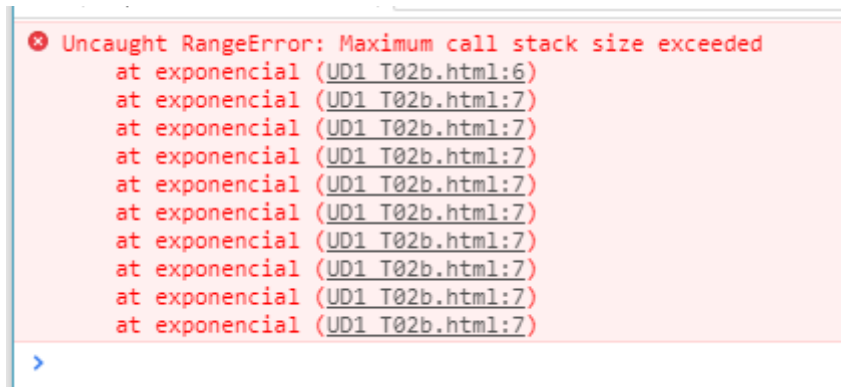
Si volvemos a navegar por la página web, vemos como se vuelven a cargar los mismos errores:



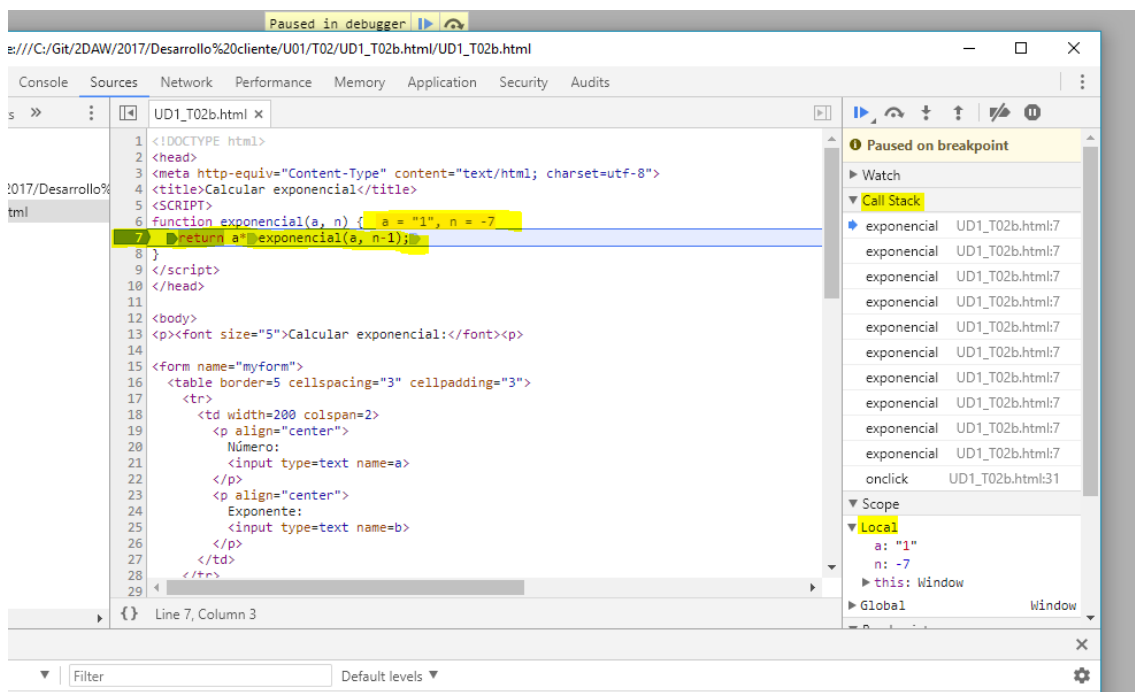
Parte dos

Ejecútala en el navegador Chrome. Proporciona el dígito que te solicita. ¿Que ha ido mal?

Al ejecutar la función exponencial, ocurre que la recursividad no tiene fin y salta una excepción de rango que no es tratada:



Establece un punto de ruptura en el código de la función, recarga la web y vuelve a introducir los dígitos. Monitoriza el valor de las variables *a* y *n* del formulario y utiliza la opción adecuada para trazar el programa y ver cómo varían los valores para detectar donde se ha producido el error.



En esta imagen podemos ver varias cosas:

1. La pila de llamadas: donde vemos que hay muchas llamadas a la misma función “exponencial”.

2. En la parte Local, podemos ver los valores de las variables “a” y “n”, en donde se puede ver el valor de -7.
3. En la función, vemos el punto de interrupción en la función.

Ten en cuenta que a medida que se pasa por las llamadas a las funciones recursivas, el subpanel "Call Stack" de la derecha te permite realizar un seguimiento de la pila de llamadas, junto con los parámetros que se pasan en cada llamada a la función.

En la imagen anterior se ve en todas las llamadas que se ha hecho a la función, y si se hace clic en cada una, se puede ver en ese momento que valor tenían las variables.

exponencial	UD1_T02b.html:7	▼ Call Stack
exponencial	UD1_T02b.html:7	exponencial UD1_T02b.html:7
exponencial	UD1_T02b.html:7	exponencial UD1_T02b.html:7
▶ exponencial	UD1_T02b.html:7	exponencial UD1_T02b.html:7
exponencial	UD1_T02b.html:7	exponencial UD1_T02b.html:7
exponencial	UD1_T02b.html:7	exponencial UD1_T02b.html:7
exponencial	UD1_T02b.html:7	exponencial UD1_T02b.html:7
exponencial	UD1_T02b.html:7	exponencial UD1_T02b.html:7
exponencial	UD1_T02b.html:7	▶ exponencial UD1_T02b.html:7
exponencial	UD1_T02b.html:7	exponencial UD1_T02b.html:7
exponencial	UD1_T02b.html:7	exponencial UD1_T02b.html:7
onclick	UD1_T02b.html:31	onclick UD1_T02b.html:31
▼ Scope		▼ Scope
▼ Local		▼ Local
a: "1"		a: "1"
n: -4		n: 0
▶ this: Window		▶ this: Window

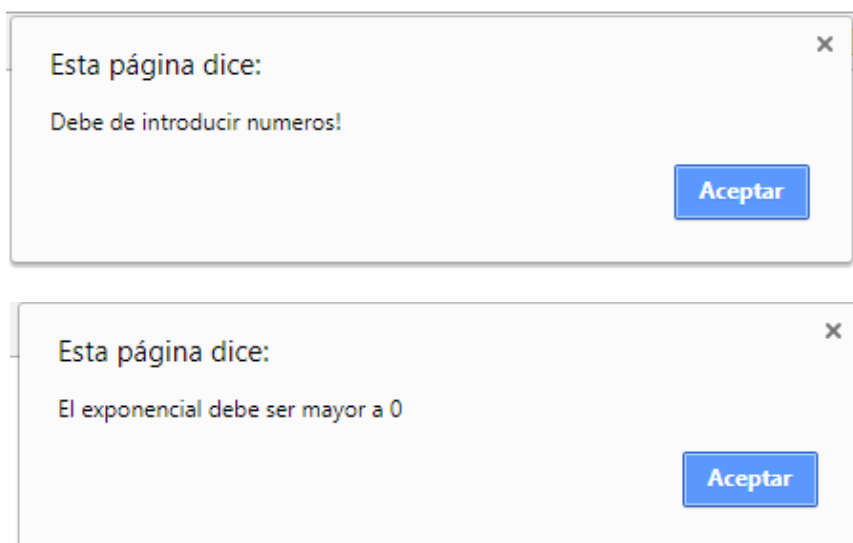
Se puede ver como el valor de “n” va cambiando en la cuarta llamada de -4 a 0 en la octava llamada.

¿Podrías solucionar el problema que presenta el código que se te ha proporcionado?

Para solucionar un problema con la recursividad de esa función, hay que agregar una salida para la recursividad. Una solución sería la siguiente, se le ha agregado tratamiento de excepciones para algunos problemas que pueden encontrarse:

```
function exponencial(a, n) {  
  // Condicion para ver si es numero  
  if (isNaN(a) && isNaN(n)) {  
    alert("Debe de introducir numeros!");  
    return null;  
  }  
  // Condicion para ver si el exponencial es mayor a 0  
  if(n < 0){  
    alert("El exponencial debe ser mayor a 0");  
    return null;  
  }  
  // Condicion para salir del bucle si el exponencial llega a 0  
  if (n == 0) {  
    return a;  
  }  
  // Recursividad  
  else {  
    return a*exponencial(a, n-1);  
  }  
}
```

Con la función modificada, vemos como no solo se realiza el cálculo correctamente pero también nos trata excepciones:



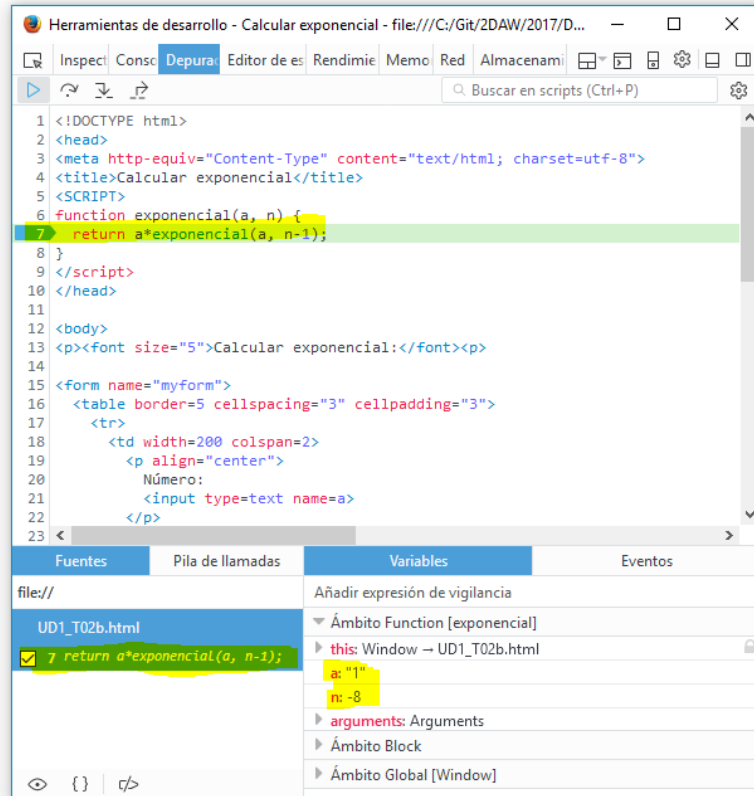
Realizar la práctica con Firefox y Edge

Firefox

En Firefox, es casi complemente igual, ya que las herramientas se parecen bastante, en la siguiente captura se ve las similitudes:

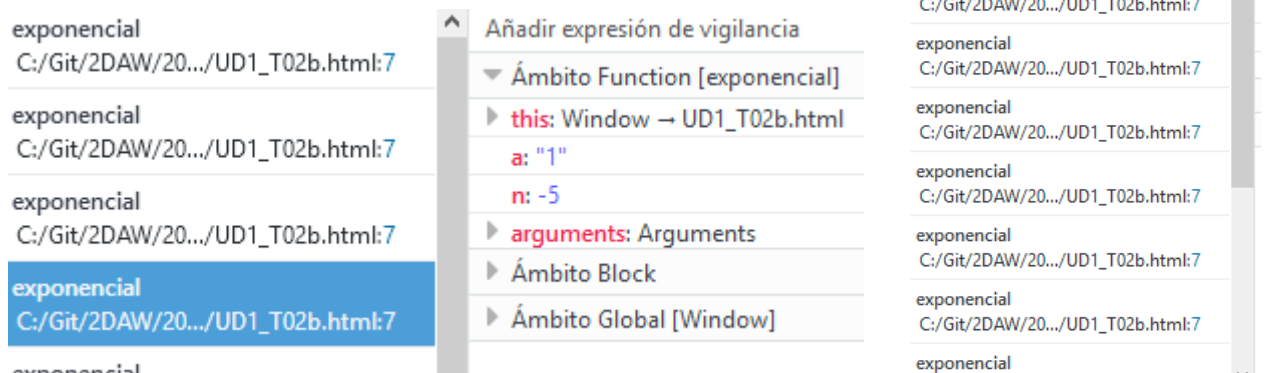
Calcular exponencial:

Número:
<input type="text" value="1"/>
Exponente:
<input type="text" value="2"/>
<input type="button" value="Calcular"/> <input type="button" value="Borrar"/>
Resultado:
<input type="text"/>



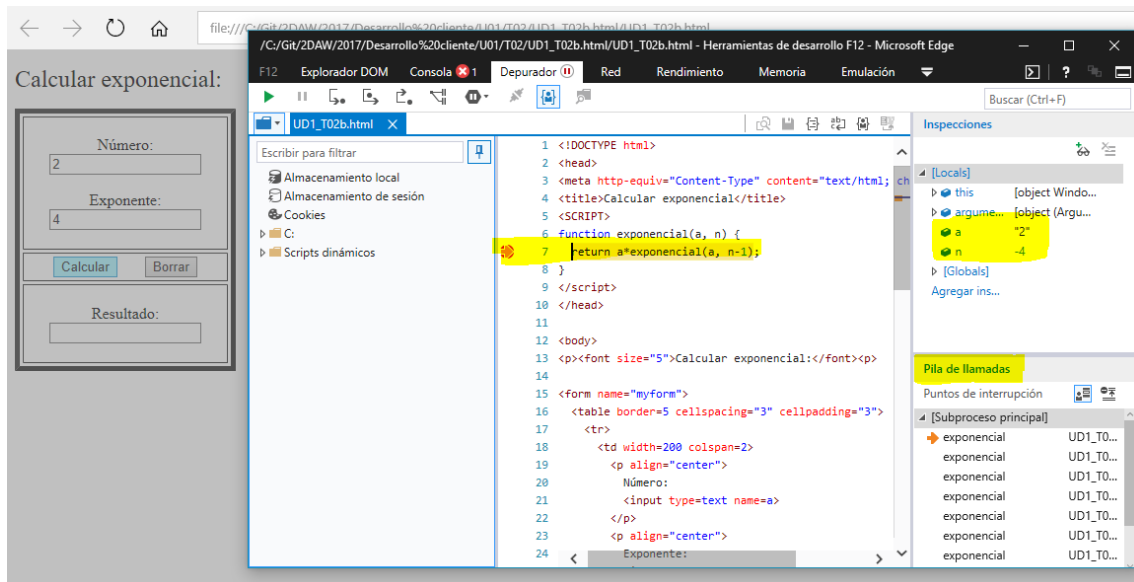
1. Se puede ver el punto de interrupción
2. El valor de las variables
3. Y el número de llamadas a la función exponencial.

En la pila de llamadas se puede ver la cantidad de llamadas realizadas a la función exponencial, se hacemos clic en cada una de ellas podemos ver el valor de las variables en cada llamada.

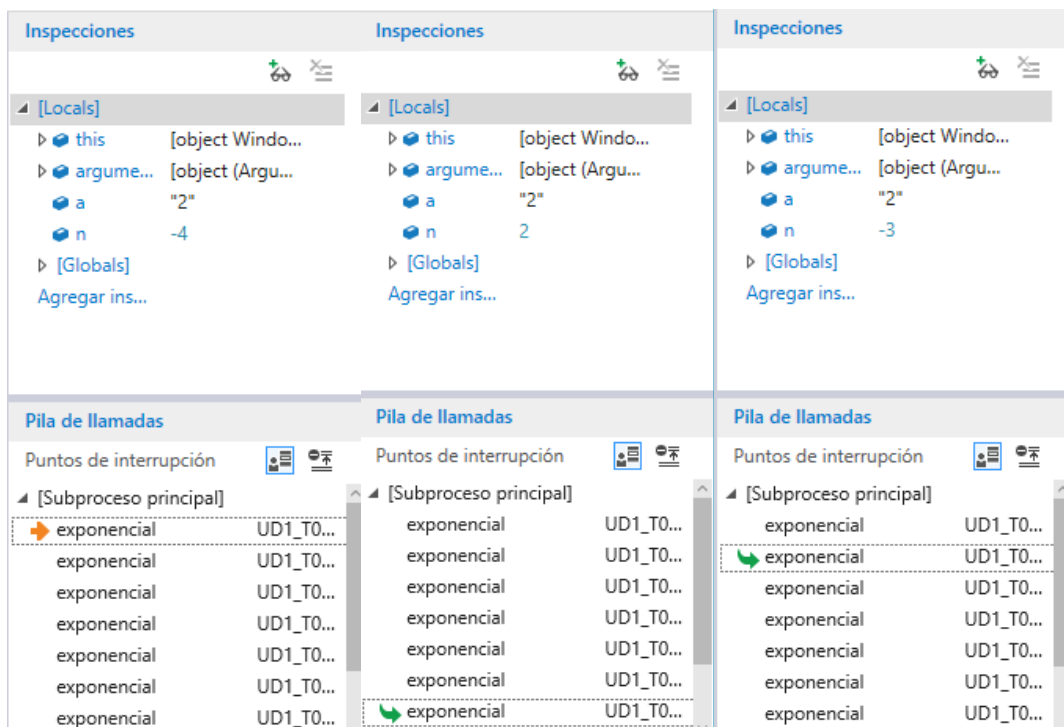


Edge

En Microsoft Edge es un poco diferente estéticamente, pero se muestra de la misma forma, se muestra el punto de depuración, el valor de las variables y la pila de llamadas en la misma pestaña, como vemos en la siguiente imagen:



Como en los otros navegadores, si pulsamos sobre la pila de llamadas en cada función exponencial, te muestra el valor de cada variable en ese momento de la ejecución de la función:



Ahí podemos ver la diferencia de valores entre las variables en la pila de llamadas.

Comparación entre navegadores

Tanto Chrome, como Firefox y Microsoft Edge tienen unas herramientas para desarrolladores que sinceramente se parecen bastante, ya que si se comparan las tres estéticamente y la forma en la que funcionan son casi iguales.

Cabe destacar que las opiniones son subjetivas y se realizar en un portátil de gama media, por lo que el rendimiento puede ser afectado.

En cuanto a rendimiento, Chrome y Firefox están a la par a la hora de realizar depuración, aunque Chrome suele tener algún que otro cuelgue a la hora de realizar la depuración. Sin embargo, Edge tiene serios problemas a la hora de cargar las páginas y hacer alguna que otra depuración.

En cuanto a la hora de acceder a las opciones que traen las herramientas de desarrolladores, ambas Chrome, Firefox y Edge están muy bien, ya que incluyen básicamente las mismas herramientas para utilizar.

Opinión personal

Cada desarrollador tiene una opinión diferente y un gusto diferente, personalmente yo utilizare Google Chrome por su rapidez en general y su estética simple. Coincido con mis compañeros de profesión que hay otros navegadores que pueden ofrecer mejor rendimiento, también es verdad que, en ciertos sistemas operativos o distribuciones, un navegador puede aumentar de rendimiento considerablemente, pero en mi opinión Google Chrome sigue reinando en el apartado de rendimiento y en el apartado de herramientas de desarrollo.