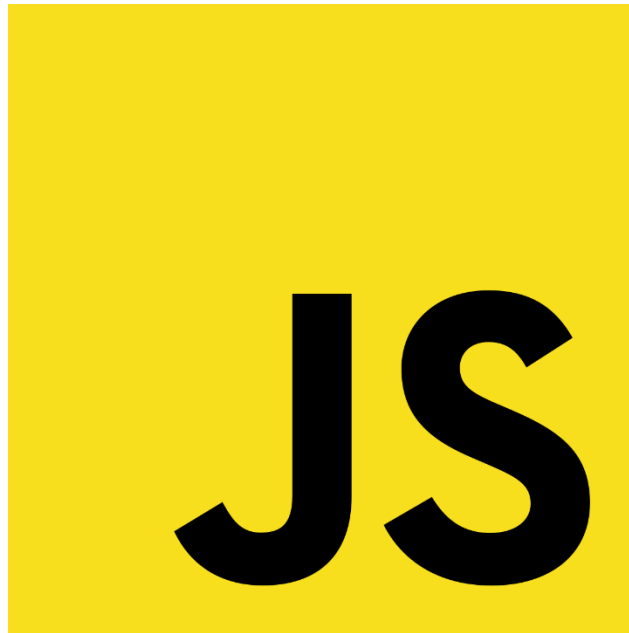


TAREA 01

DESARROLLO WEB EN ENTORNO CLIENTE



JUAN CALVENTE MONTORO

Contenido

Primera Parte. (RA1)	3
1. Elige e instala en tu sistema operativo al menos 2 navegadores web.....	3
2. ¿Por qué es conveniente tener instalados al menos dos navegadores diferentes en tu ordenador?.....	7
3. ¿Ves conveniente que Firefox abandone su motor y use blink?.....	8
4. Elige un editor para JavaScript e instálalo.....	9
Segunda Parte. (RA2)	16
1. Cadenas ordenadas.	16
2. Realice un programa en JavaScript que te pregunte por tres números que se introducen desde teclado usando prompt.....	17
3. División por restas sucesivas.	19
Ejemplo de salida.....	20

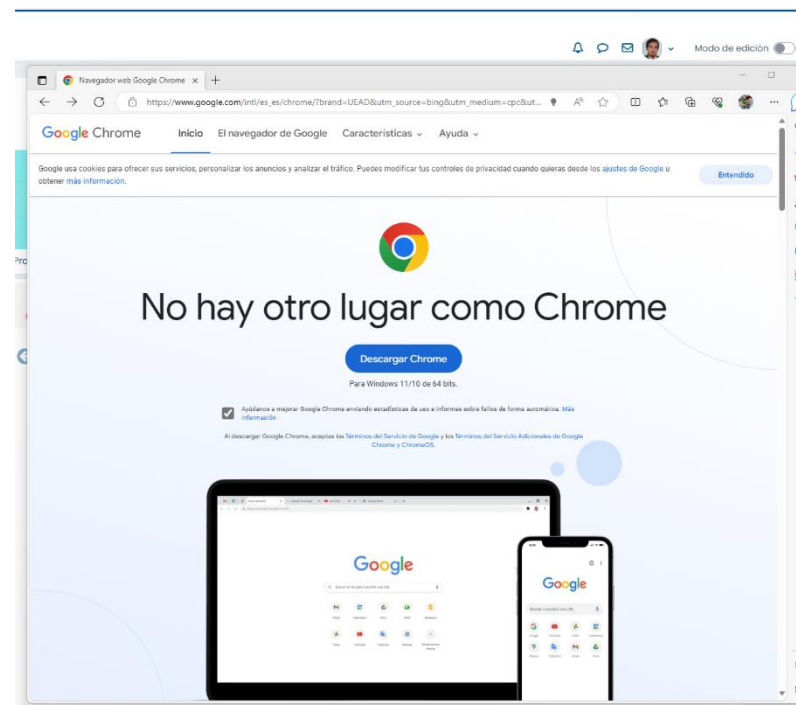
Primera Parte. (RA1).

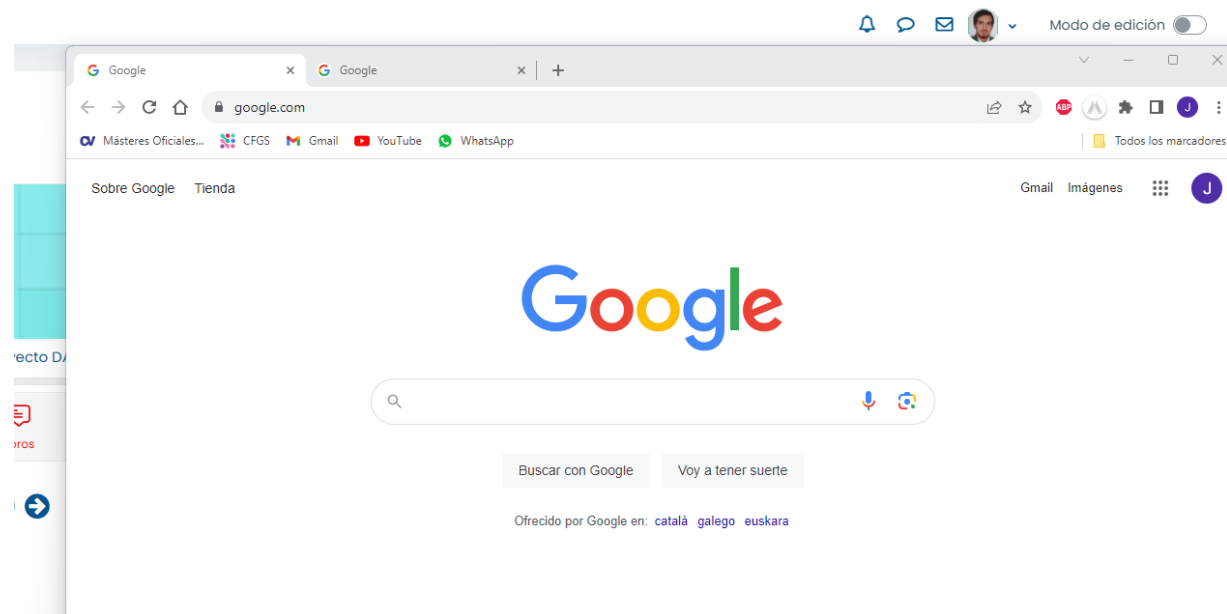
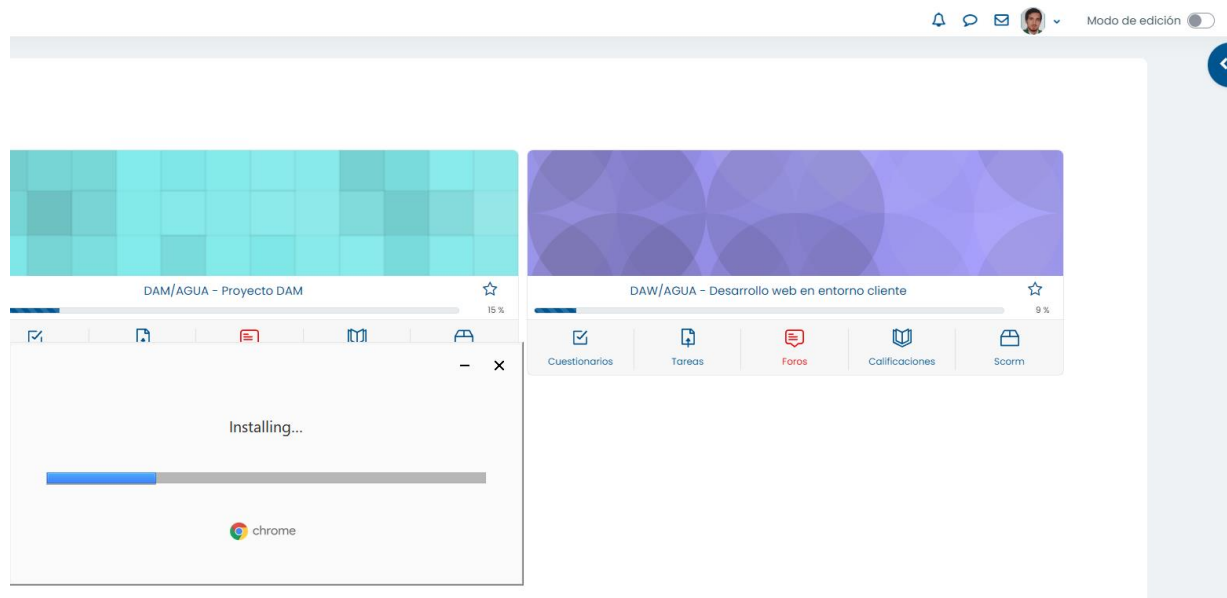
1. **Elige e instala en tu sistema operativo al menos 2 navegadores web (adicionales al que ya tienes instalado por defecto). Por ejemplo, si ya tienes instalado Firefox instala otros dos más.**
 - a. **Deben realizarse capturas autenticables del proceso de instalación. Autenticable es que aparezca tu usuario de distancia y la web en algún lado en otro navegador.**

Nuestro sistema operativo, Windows 11, trae preinstalado el navegador Edge, que es el que usaremos para descargar los otros dos exploradores.

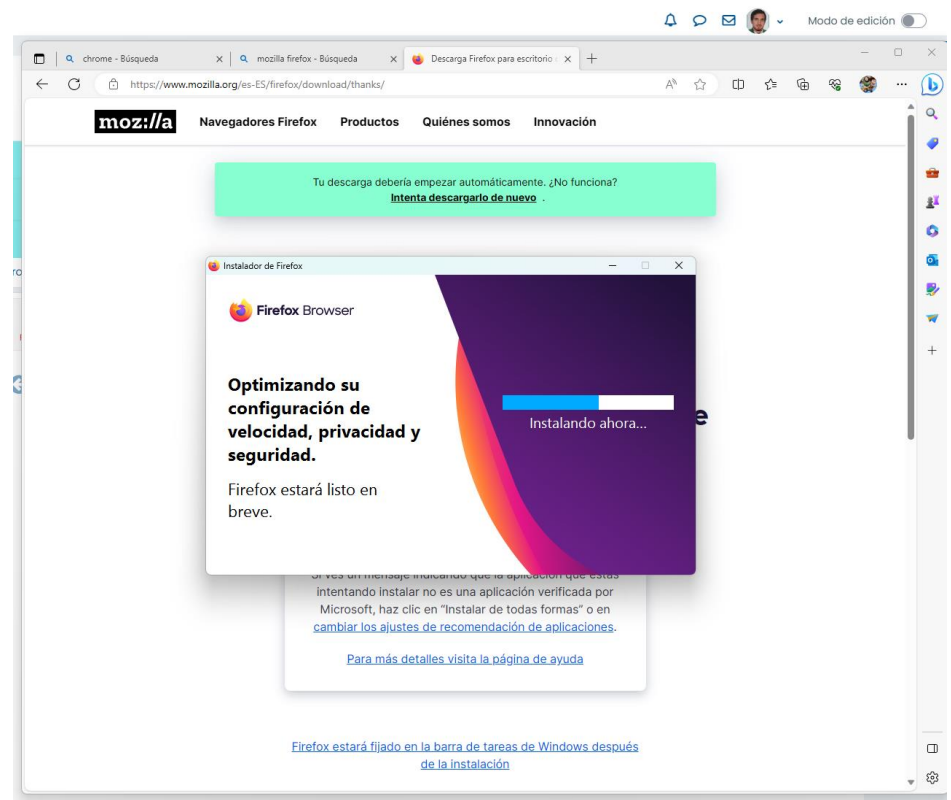
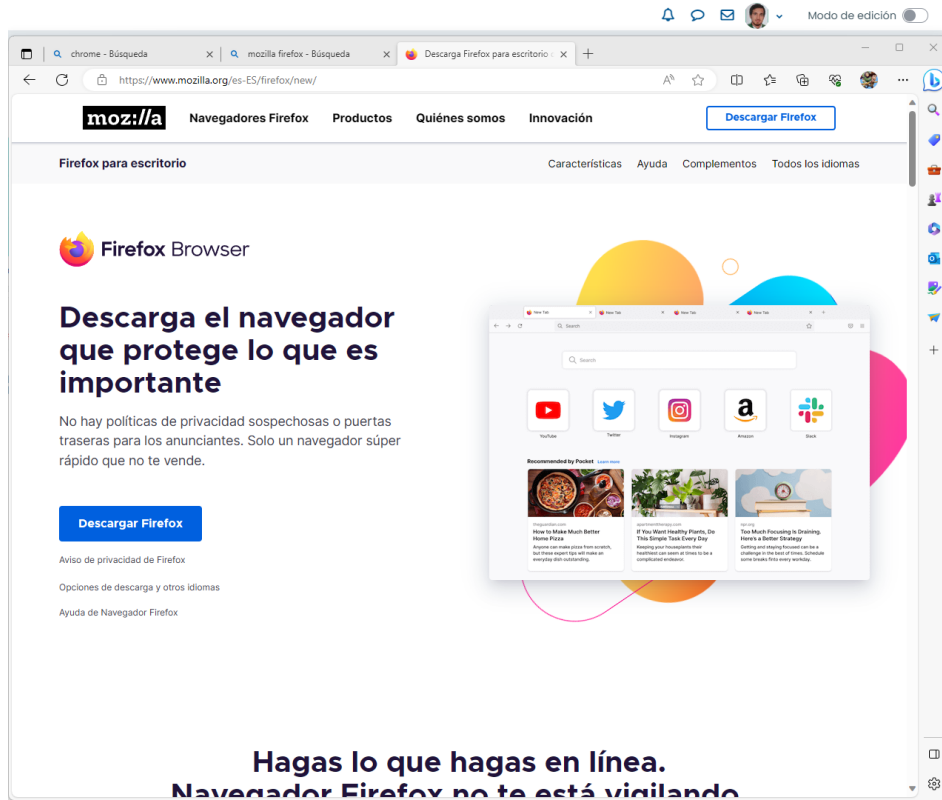
Descarga de Google Chrome

Descargamos el instalador desde la web oficial de Google haciendo click en “Descargar Chrome”. Tras esto, ejecutaremos el programa descargado y comenzará automáticamente la instalación. Cuando esta finalice, se lanzará automáticamente, como se aprecia en las siguientes imágenes.

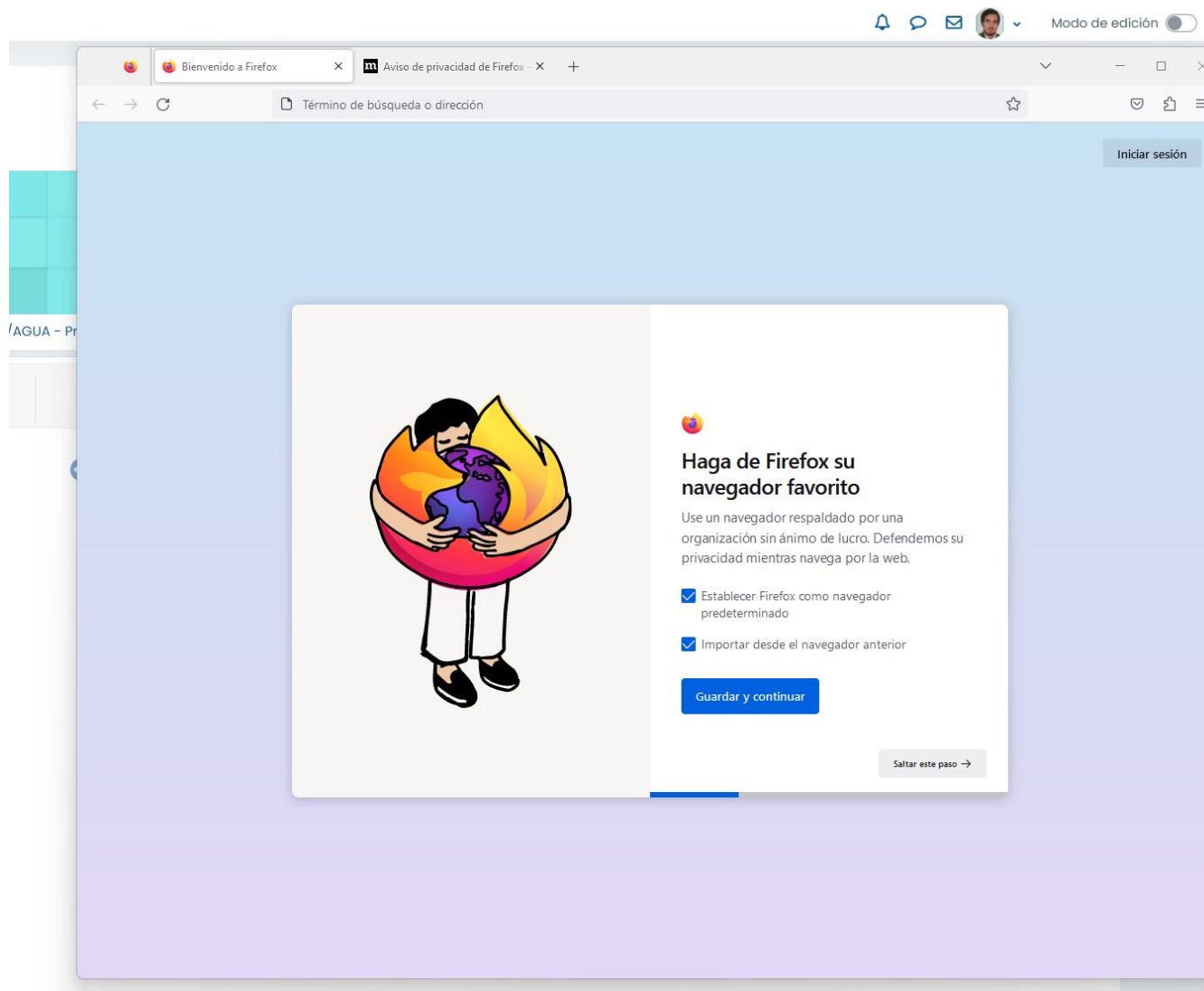




Descarga de Mozilla Firefox



Para descargar Firefox replicamos los mismos pasos que en el caso anterior. Desde la web oficial de Mozilla bajamos el instalador, lo ejecutamos y comenzará a instalarse el programa. Una vez se haya completado el proceso de instalación se abrirá el navegador con la pestaña de configuración.




2. ¿Por qué es conveniente tener instalados al menos dos navegadores diferentes en tu ordenador? Se abrirá un hilo en el foro. Comente ahí. Indica el nombre del motor HTML y JavaScript de los navegadores que has instalado.
- a. Adjunta la captura de la participación.

alendarío

ificaciones

Mozilla Firefox v. 117.0.1. Motor HTML: Gecko. Interprete js: SpiderMonkey

[Enlace permanente](#) [Marcar como leído](#) [Mostrar mensaje anterior](#) [Responder](#)

 **Re: Hilo 7. Hilo de navegadores alternativos.**
de Calvente Montoro, Juan - martes, 26 de septiembre de 2023, 12:42

Bajo mi punto de vista, existen varias razones para tener, al menos, dos navegadores instalados.

En cuanto a compatibilidad, los sitios y aplicaciones web pueden comportarse de manera diferente en diferentes navegadores, por lo que podemos encontrarnos el caso de visualizar o no parte del contenido en función del que estemos utilizando.

Respecto al desarrollo web, tener múltiples navegadores nos permite probar y verificar la compatibilidad de nuestros proyectos en diferentes entornos, dado que los motores que usan suelen ser distintos.

También es importante señalar que los navegadores se encuentran en un proceso constante de actualización. Cada uno añadirá nuevas características o funcionalidades que podremos probar y aprovechar.


Programas instalados para la tarea:

Google Chrome
Motor HTML - Blink
Motor Javascript - V8

Mozilla Firefox
Motor HTML - Gecko
Motor JavaScript - SpiderMonkey

[Enlace permanente](#) [Marcar como no leído](#) [Mostrar mensaje anterior](#) [Editar](#) [Borrar](#) [Responder](#) [Exportar al portafolios](#)

◀ Hilo 8. ¿Ves conveniente que Firefox abandone su motor y use blink?



3. ¿Ves conveniente que Firefox abandone su motor y use blink? Existirá un hilo en el foro. No abra el suyo.

- a. Realiza una captura de tu intervención y añádela al documento. Entre en el hilo específico que se ha abierto en el foro y haz un comentario sobre navegadores independientes. Realiza una captura de tu intervención y añádela al documento.


ilendario

ificaciones

de decisiones en el WHATWG (Web Hypertext Application Technology Working Group. Entidad que se encarga de los estándares en la web como W3C y que está formado por Apple, Mozilla, Opera.. y el editor que en este caso es Google)

Tal vez, la decisión de Mozilla, aunque no parece que vaya de la mano de sus principios fundadores y política (código abierto y software libre) tal vez venga dado por una posible estandarización de los motores de navegación o por ahorrarse el desarrollo continuo y mejora de su motor, puesto que aunque sea de código abierto y software libre su desarrollo recae en la empresa que lo diseñó. Sea como fuere, tal vez, esta decisión facilite la vida el desarrollo web y la estandarización de un navegador web... aunque se debería de ver si entra en juego lo de la libre competencia, el monopolio... temas que son harina de otro costal y de las cuales no estoy muy versado.

[Enlace permanente](#) [Marcar como leído](#) [Mostrar mensaje anterior](#) [Responder](#)

 **Re: Hilo 8. ¿Ves conveniente que Firefox abandone su motor y use blink?**
de [Calvente Montoro, Juan](#) - martes, 26 de septiembre de 2023, 13:24

Buenos días,

Blink se encuentra en navegadores como Chrome o Edge, con una importante cuota de mercado, lo que supone una más que probada compatibilidad con la mayoría de sitios web. La adopción de este motor por parte de Firefox facilitaría el desarrollo web, dado que habría menos variaciones entre un navegador y otro y nos llevaría, a largo plazo, a una mejora de eficiencia y rendimiento de Blink, pues la mayoría de los esfuerzos y apoyos recaerían sobre este motor.

No obstante, este cambio trae una serie de desventajas potenciales tales como la pérdida de independencia tecnológica, el elevado coste de transición de un motor a otro y una menor diversidad en el mercado.

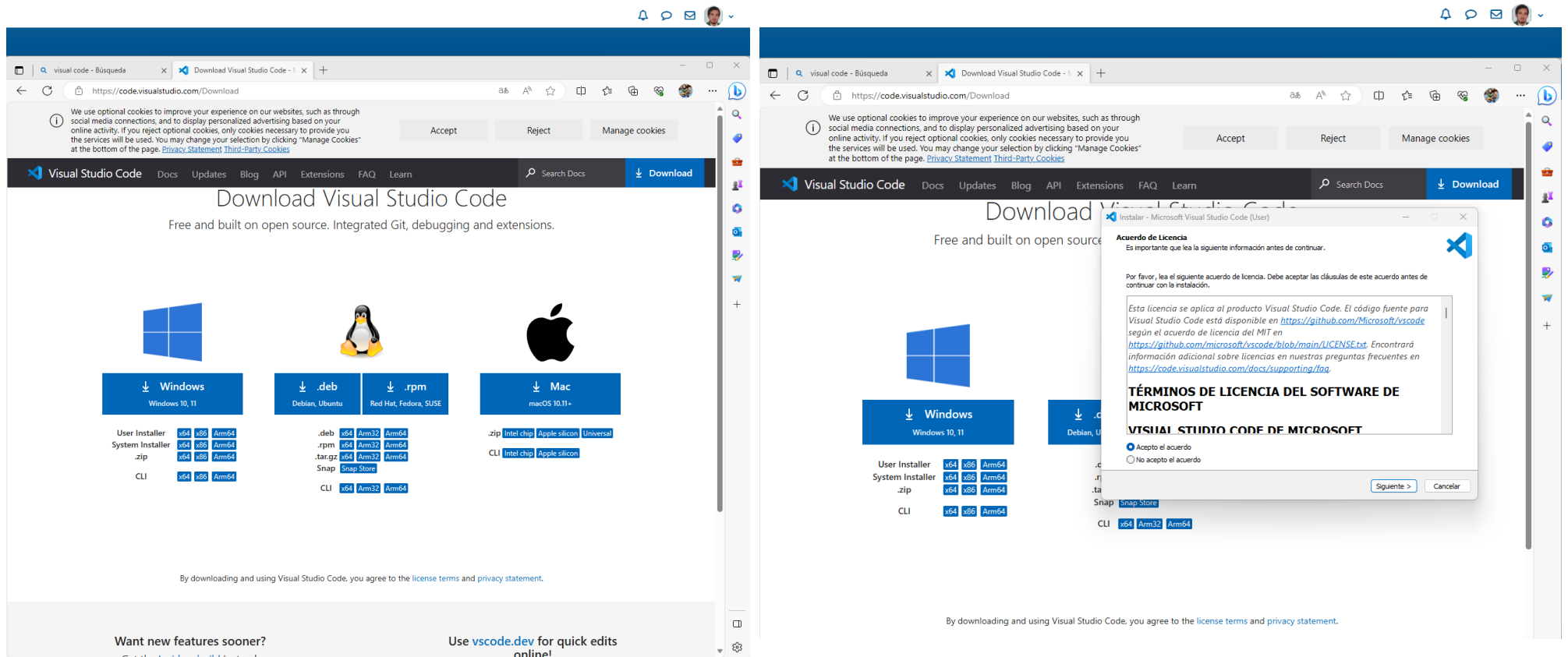
En base a ello, creo que, como usuarios, Firefox debería seguir utilizando Gecko. Este explorador no solo es una de las pocas opciones popularizadas y estandarizadas de código abierto, sino que aporta diversificación al ecosistema actual en cuanto a navegadores, fomentando la competencia entre ellos e impulsando el desarrollo de nuevas características e innovaciones.

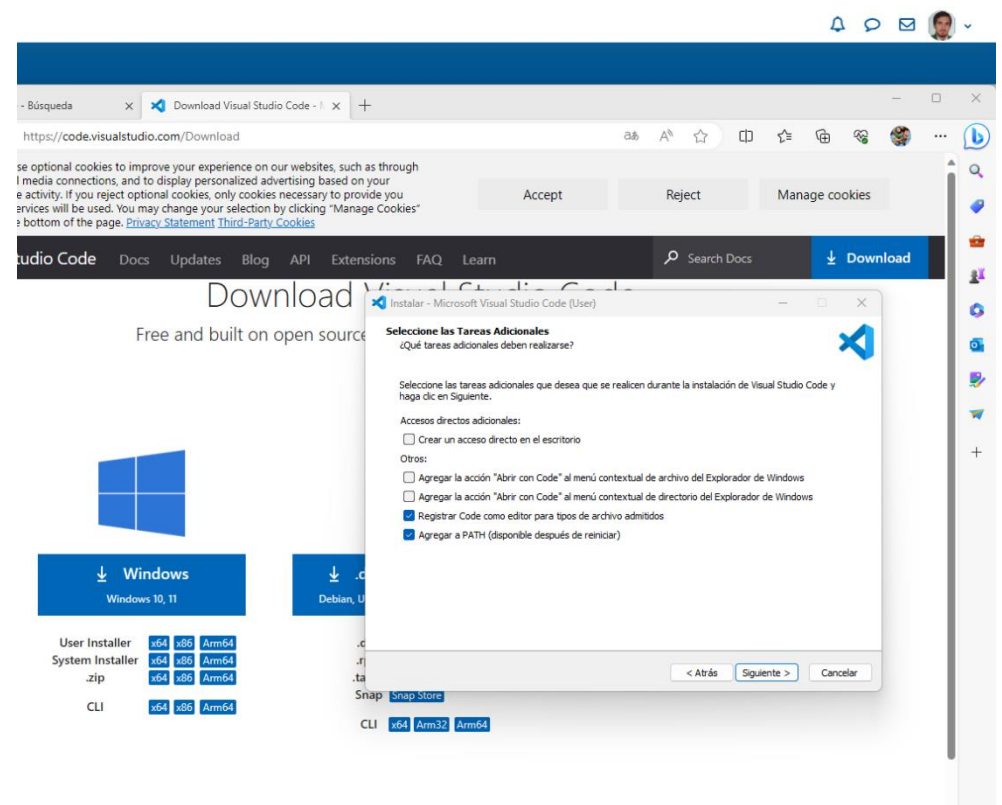
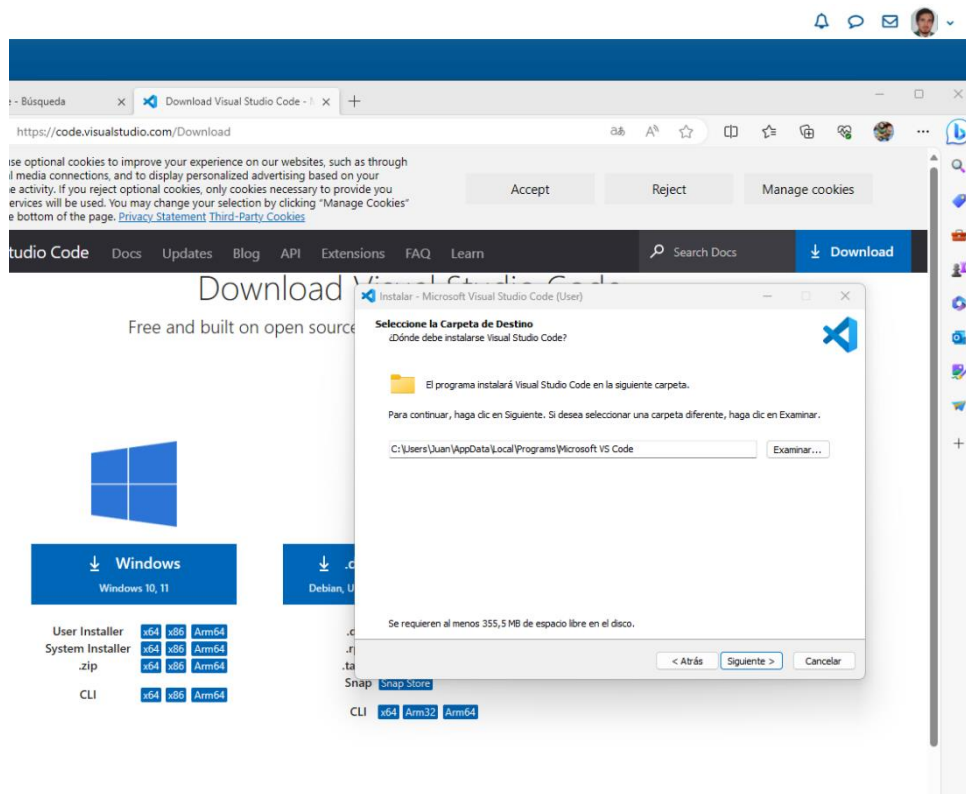
[Enlace permanente](#) [Marcar como no leído](#) [Mostrar mensaje anterior](#) [Editar](#) [Borrar](#) [Responder](#) [Exportar al portafolios](#)

◀ Hilo 7. Hilo de navegadores alternativos.

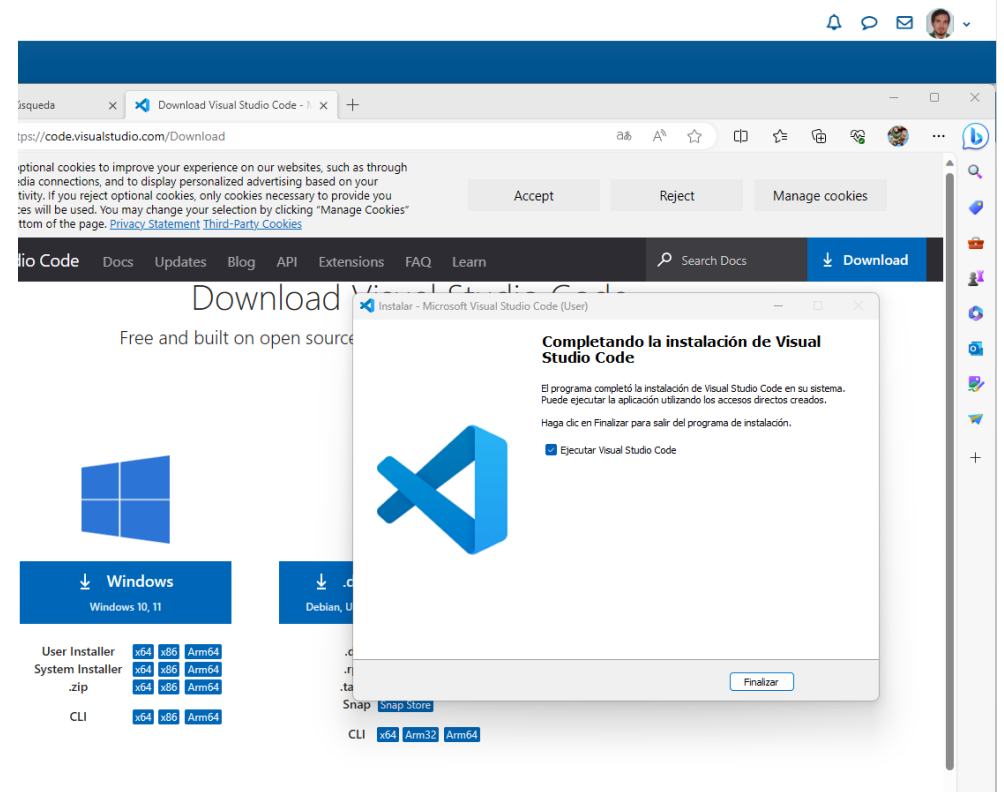
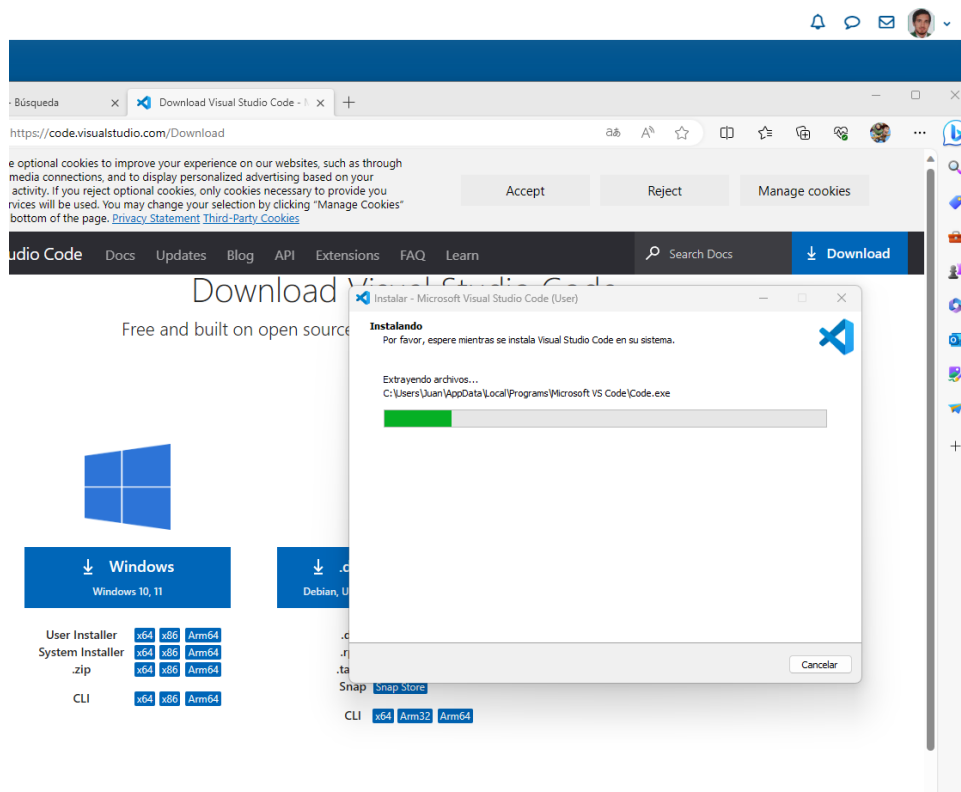
4. Elige un editor para JavaScript e instálalo. En los materiales y en los foros hay ejemplos de editores JavaScript.

Nos hemos decantado por instalar Visual Studio Code. Nos hemos descargado el instalador para la versión de nuestro sistema operativo desde la web oficial. Al ejecutarla, tendremos que aceptar los términos y condiciones.

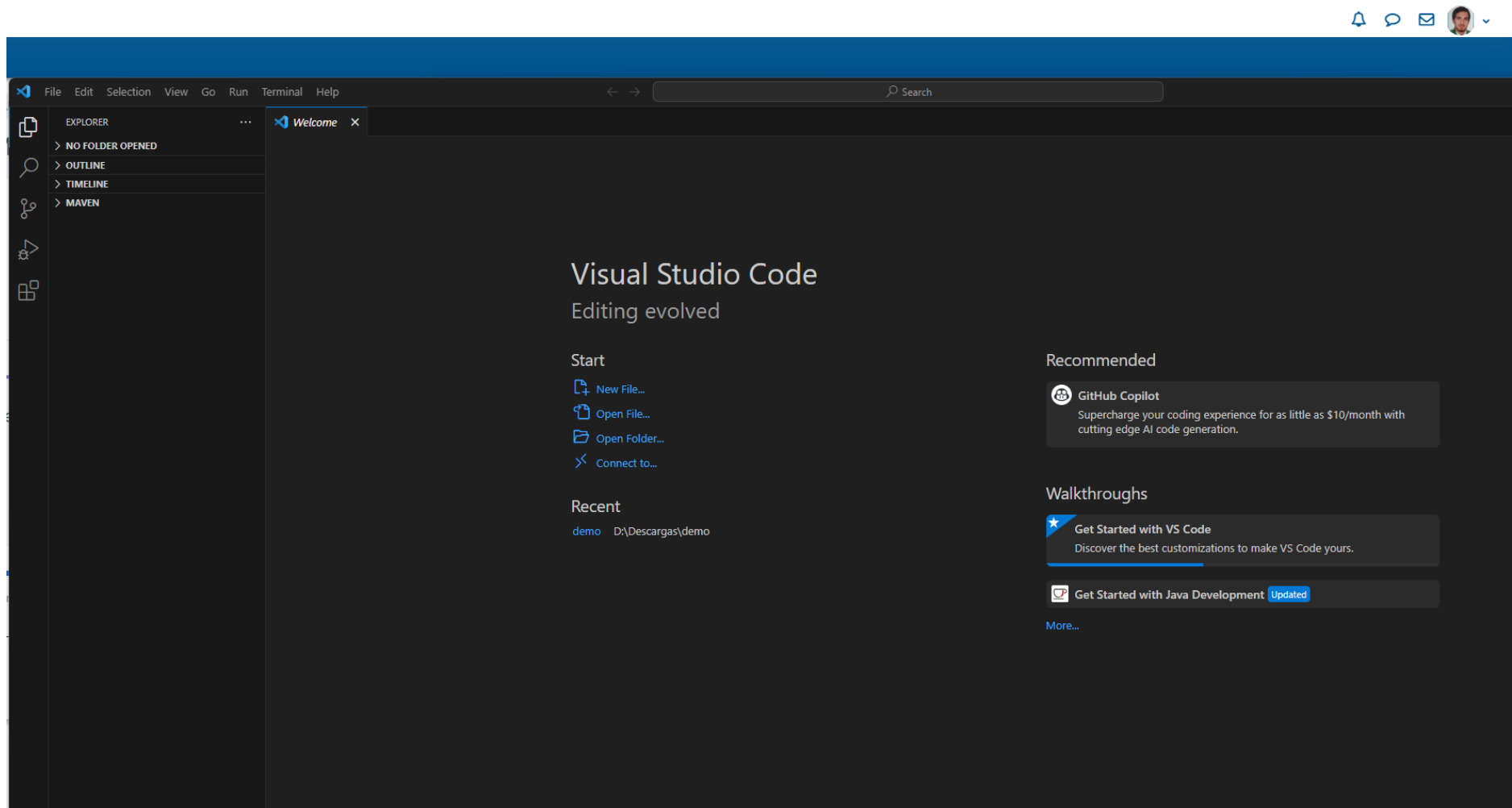




A través de las siguientes ventanas configuraremos las características y parámetros conforme a nuestras necesidades.



Finalmente, el programa comenzará a instalarse. Al finalizar, podremos lanzar directamente la aplicación desde el instalador.



a. Debes realizar una tabla comparativa de características para sustentar tu opinión. Si no la haces tendrás un 0 en este apartado.

EDITOR	DESCRIPCIÓN	VENTAJAS	DESVENTAJAS	LICENCIA
Visual Studio Code	Editor de código de Microsoft popular y muy personalizable.	<ul style="list-style-type: none"> - Amplia comunidad de usuarios y extensiones. - Integración sólida con Git. - Rendimiento rápido y consumo de recursos moderado. 	<ul style="list-style-type: none"> - Puede requerir configuración inicial para algunas características avanzadas. 	Código fuente abierto (MIT License)
Sublime Text	Editor de texto ligero y rápido con muchas características útiles.	<ul style="list-style-type: none"> - Rápido y altamente personalizable. - Gran cantidad de complementos disponibles. 	<ul style="list-style-type: none"> - No es de código abierto. - Las actualizaciones a veces son lentas. 	Comercial (Propietario)
Atom	Editor de código de GitHub con un enfoque en la comunidad y la extensibilidad.	<ul style="list-style-type: none"> - Totalmente personalizable con extensiones. - Desarrollo activo y actualizaciones frecuentes. 	<ul style="list-style-type: none"> - Puede consumir más recursos en comparación con otros editores ligeros. - Su rendimiento puede considerarse lento. 	Código fuente abierto (MIT License)
WebStorm	IDE (Entorno de Desarrollo Integrado) de JetBrains diseñado para JavaScript y web.	<ul style="list-style-type: none"> - Ofrece potentes características de depuración y refactorización. - Integración completa con herramientas de desarrollo web. 	Es un IDE completo y puede ser más pesado en términos de recursos. - Es de pago, aunque ofrece una versión de prueba.	Comercial (Propietario)
Notepad++	Editor de código de código abierto y ligero para Windows.	<ul style="list-style-type: none"> - Rápido y fácil de usar. - Soporte para una variedad de lenguajes de programación. 	<ul style="list-style-type: none"> - Disponible solo en Windows. - No es tan poderoso como algunos de los IDE más completos. 	Código fuente abierto (GPL License)

5. Para finalizar, usando la dirección de validación del W3C indicada en los apuntes, realiza la validación de la página de contenido informático SlashDot. Indica la solución solamente a 3 problemas diferentes.

Error nº 21:

21. **Error** Bad value for attribute `id` on element `input`: An ID must not be the empty string.

From line 684, column 4; to line 684, column 83

```
<input type="text" id="" class="" name="fhfilter" value="" placeholder="Search">
```

El atributo id no puede estar vacío, dado que se trata de un valor que debe de identificar de manera única el elemento en el documento.

```
<input type="text" id="id_1" class="" name="fhfilter" value="" placeholder="Search">
```

Error nº 30:

30. **Error** CSS: `font-weight`: Property `font-weight` doesn't exist.

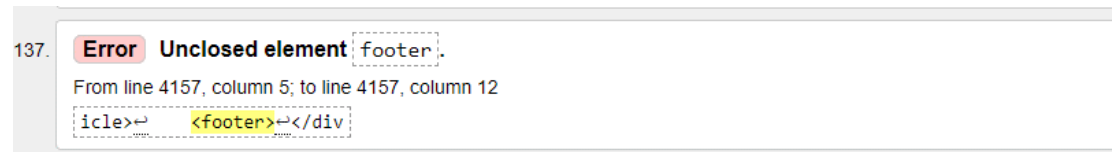
From line 918, column 13; to line 918, column 18

```
nt-weight: normal;
```

El error se encuentra en la propiedad "font-weight," que se usa para especificar el grosor o peso de una fuente. La propiedad correcta es "font-weight," y no "nt-weight."

```
font-weight: normal;
```

Error en la línea nº 115:



Se ha abierto la etiqueta <footer>, pero no se ha cerrado.

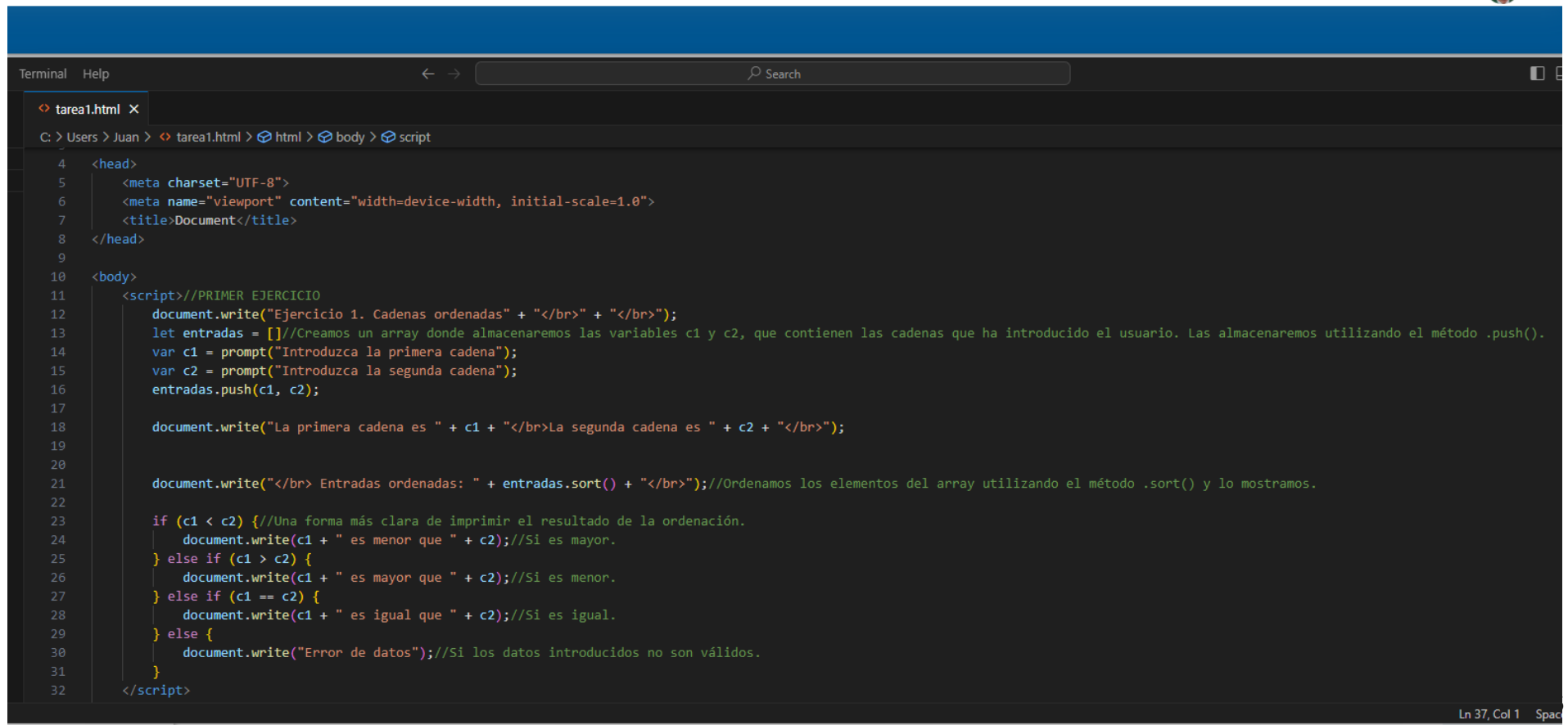
Código en el documento	Propuesta de solución
<pre>... <div class="modal-box"> Close <article class="modal-content"> </article> <footer> </div> ...</pre>	<pre>... <footer id="fhft" class="grid_24 nf"> <div class="modal-box"> Close <article class="modal-content"> </article> </div> </footer> ...</pre>

Segunda Parte. (RA2).

Vas a necesitar `prompt` y `document.write` para realizar la comunicación con el usuario.

1. Cadenas ordenadas.

- a. Pida dos cadenas.
- b. Muéstrelas en ordenadas de mayor a menor.
- c. Por ejemplo "zz" es menor que "aa".

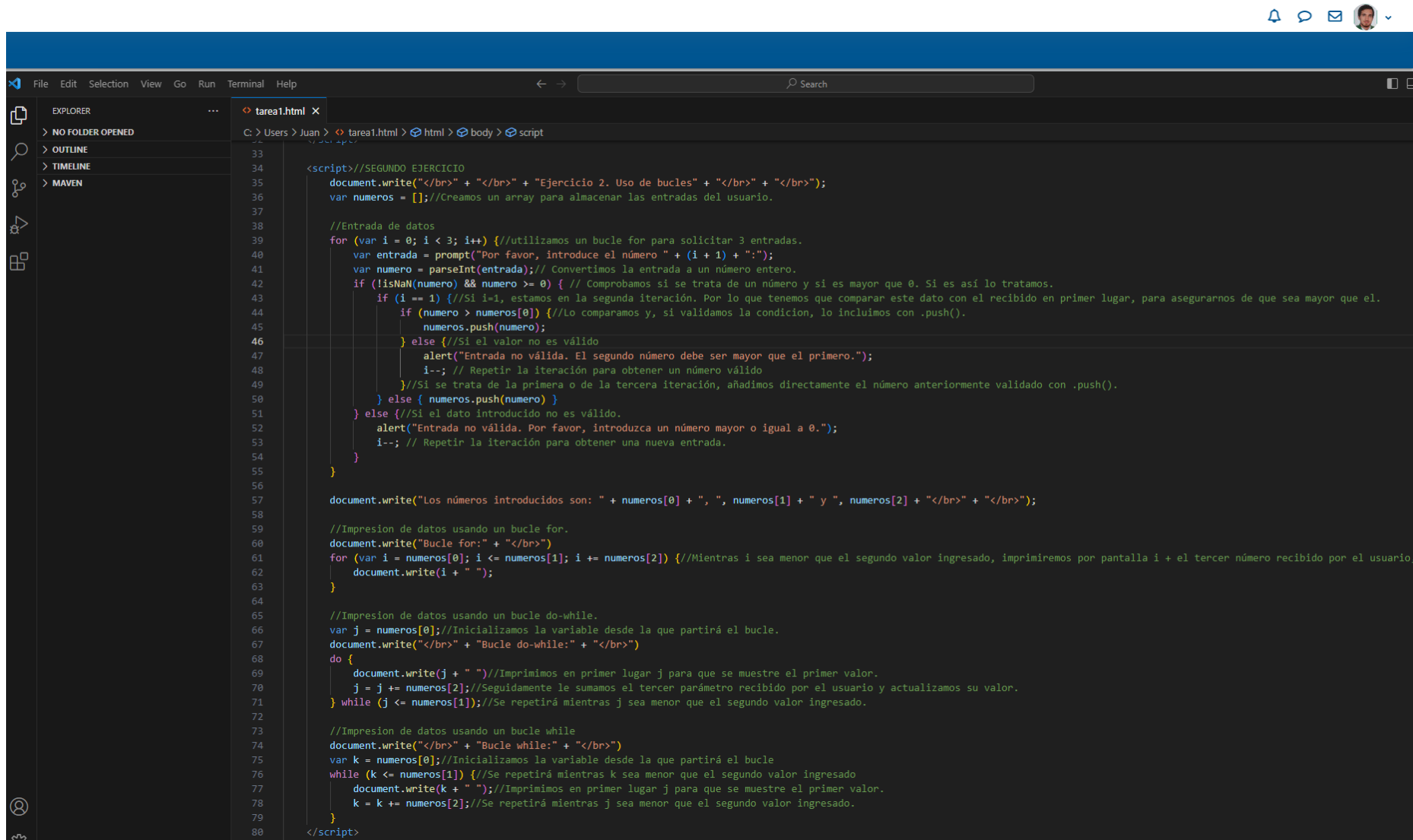


The screenshot shows a web browser window with a dark theme. The address bar shows the file path: `C:\Users\Juan> tarea1.html`. The browser's developer tools are open, showing the `script` tab. The code is as follows:

```
4 <head>
5   <meta charset="UTF-8">
6   <meta name="viewport" content="width=device-width, initial-scale=1.0">
7   <title>Document</title>
8 </head>
9
10 <body>
11   <script>//PRIMER EJERCICIO
12     document.write("Ejercicio 1. Cadenas ordenadas" + "<br>" + "<br>");
13     let entradas = []; //Creamos un array donde almacenaremos las variables c1 y c2, que contienen las cadenas que ha introducido el usuario. Las almacenaremos utilizando el método .push().
14     var c1 = prompt("Introduzca la primera cadena");
15     var c2 = prompt("Introduzca la segunda cadena");
16     entradas.push(c1, c2);
17
18     document.write("La primera cadena es " + c1 + "<br>La segunda cadena es " + c2 + "<br>");
19
20
21     document.write("<br>Entradas ordenadas: " + entradas.sort() + "<br>");//Ordenamos los elementos del array utilizando el método .sort() y lo mostramos.
22
23     if (c1 < c2) { //Una forma más clara de imprimir el resultado de la ordenación.
24       document.write(c1 + " es menor que " + c2); //Si es mayor.
25     } else if (c1 > c2) {
26       document.write(c1 + " es mayor que " + c2); //Si es menor.
27     } else if (c1 == c2) {
28       document.write(c1 + " es igual que " + c2); //Si es igual.
29     } else {
30       document.write("Error de datos"); //Si los datos introducidos no son válidos.
31     }
32   </script>
```

The status bar at the bottom right indicates "Ln 37, Col 1".

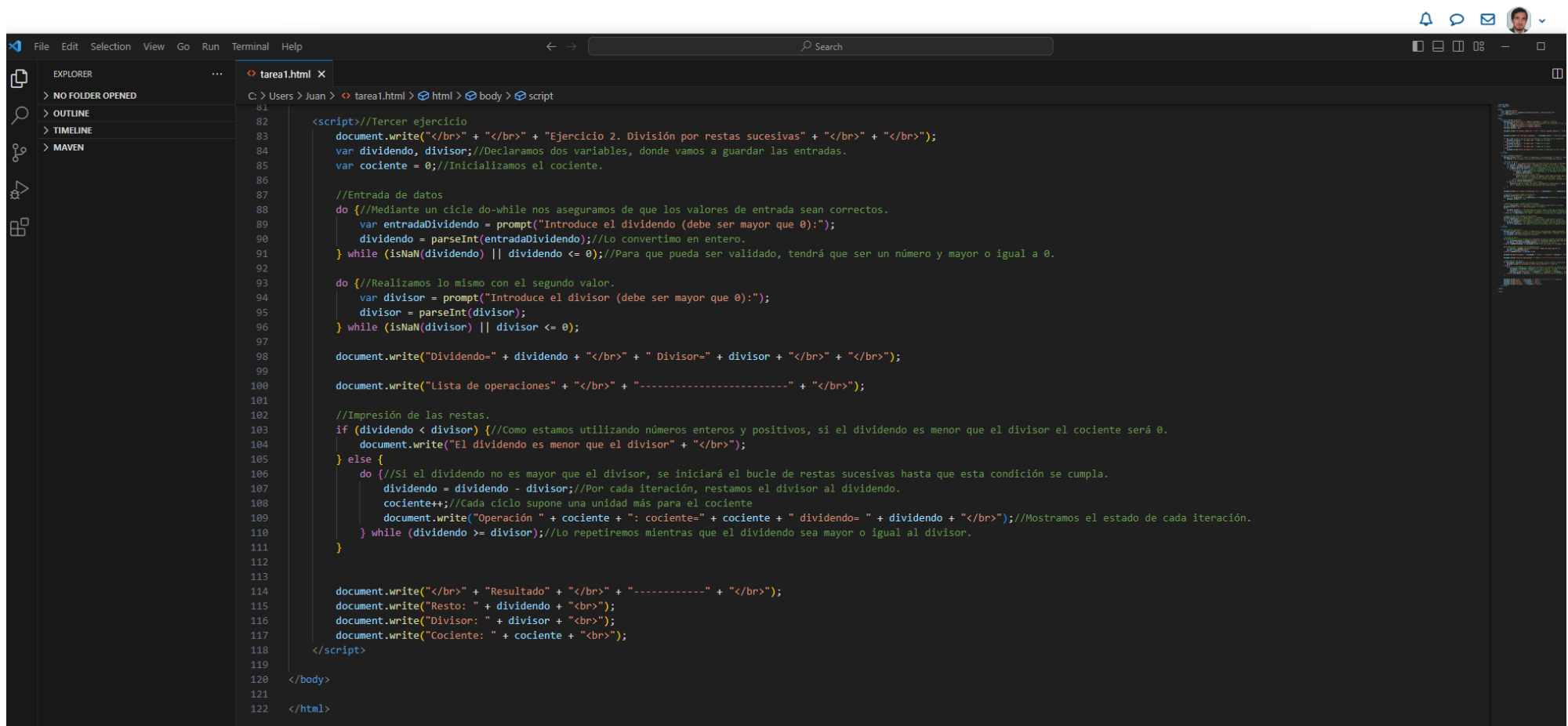
2. Realice un programa en JavaScript que te pregunte por tres números que se introducen desde teclado usando prompt.
- a. El primer número será el número inicial.
 - b. El segundo número ser el número final.
 - c. El tercer número será el incremento.
 - d. Así si introduces 1, 10 y 2 el resultado sería:
 - I. 1 3 5 7 9.
 - e. Si introduces 1, 10 y 1 el resultado sería:
 - I. 1 2 3 4 5 6 7 8 9 10.
 - f. Se debe realizar la misma tarea con tres bucles diferentes.
 - g. Debes comprobar que sea número. No debes aceptar números negativos.
 - h. Debes comprobar los límites mayores y menores de los mismos. Por ejemplo, que no sea inicio 10, fin 1 e incremento 1 ya que el bucle no termina



```
33
34
35 <script>//SEGUNDO EJERCICIO
36 document.write("<br>" + "<br>" + "Ejercicio 2. Uso de bucles" + "<br>" + "<br>");
37 var numeros = []; //Creamos un array para almacenar las entradas del usuario.
38
39 //Entrada de datos
40 for (var i = 0; i < 3; i++) { //utilizamos un bucle for para solicitar 3 entradas.
41     var entrada = prompt("Por favor, introduce el número " + (i + 1) + " :");
42     var numero = parseInt(entrada); // Convertimos la entrada a un número entero.
43     if (!isNaN(numero) && numero >= 0) { // Comprobamos si se trata de un número y si es mayor que 0. Si es así lo tratamos.
44         if (i == 1) { //Si i=1, estamos en la segunda iteración. Por lo que tenemos que comparar este dato con el recibido en primer lugar, para asegurarnos de que sea mayor que el.
45             if (numero > numeros[0]) { //Lo comparamos y, si validamos la condición, lo incluimos con .push().
46                 numeros.push(numero);
47             } else { //Si el valor no es válido
48                 alert("Entrada no válida. El segundo número debe ser mayor que el primero.");
49                 i--; // Repetir la iteración para obtener un número válido
50             } //Si se trata de la primera o de la tercera iteración, añadimos directamente el número anteriormente validado con .push().
51         } else { numeros.push(numero) }
52     } else { //Si el dato introducido no es válido.
53         alert("Entrada no válida. Por favor, introduzca un número mayor o igual a 0.");
54         i--; // Repetir la iteración para obtener una nueva entrada.
55     }
56 }
57
58 document.write("Los números introducidos son: " + numeros[0] + ", ", numeros[1] + " y ", numeros[2] + "<br>" + "<br>");
59
60 //Impresión de datos usando un bucle for.
61 document.write("Bucle for:" + "<br>");
62 for (var i = numeros[0]; i <= numeros[1]; i += numeros[2]) { //Mientras i sea menor que el segundo valor ingresado, imprimiremos por pantalla i + el tercer número recibido por el usuario
63     document.write(i + " ");
64 }
65
66 //Impresión de datos usando un bucle do-while.
67 var j = numeros[0]; //Iniciamos la variable desde la que partirá el bucle.
68 document.write("<br>" + "Bucle do-while:" + "<br>");
69 do {
70     document.write(j + " "); //Imprimimos en primer lugar j para que se muestre el primer valor.
71     j = j + numeros[2]; //Seguidamente le sumamos el tercer parámetro recibido por el usuario y actualizamos su valor.
72 } while (j <= numeros[1]); //Se repetirá mientras j sea menor que el segundo valor ingresado.
73
74 //Impresión de datos usando un bucle while
75 document.write("<br>" + "Bucle while:" + "<br>");
76 var k = numeros[0]; //Iniciamos la variable desde la que partirá el bucle
77 while (k <= numeros[1]) { //Se repetirá mientras k sea menor que el segundo valor ingresado
78     document.write(k + " "); //Imprimimos en primer lugar j para que se muestre el primer valor.
79     k = k + numeros[2]; //Se repetirá mientras j sea menor que el segundo valor ingresado.
80 }
81 </script>
```

3. División por restas sucesivas. Sin decimales.

- i. Usar un bucle do while. Aunque los do while los debemos usar para entradas y salidas, aquí lo haremos como un desafío.
- j. El algoritmo de restas sucesivas para realizar la división entera de dos números positivos. Este algoritmo consiste en ir restando al dividendo el divisor hasta obtener un resultado menor que el divisor, que será el resto de la división. El número de restas efectuado será el cociente.
- k. Mostrar el resultado el resto y el cociente.
- l. Se valorarán los comentarios y la calidad del código. Depende de haberlo hecho todo bien.



```
81
82 <script>//Tercer ejercicio
83 document.write("<br>" + "</br>" + "Ejercicio 2. División por restas sucesivas" + "</br>" + "</br>");
84 var dividendo, divisor;//Declaramos dos variables, donde vamos a guardar las entradas.
85 var cociente = 0;//Inicializamos el cociente.
86
87 //Entrada de datos
88 do { //Mediante un ciclo do-while nos aseguramos de que los valores de entrada sean correctos.
89     var entradaDividendo = prompt("Introduce el dividendo (debe ser mayor que 0):");
90     dividendo = parseInt(entradaDividendo);//Lo convertimo en entero.
91 } while (isNaN(dividendo) || dividendo <= 0); //Para que pueda ser validado, tendrá que ser un número y mayor o igual a 0.
92
93 do { //Realizamos lo mismo con el segundo valor.
94     var divisor = prompt("Introduce el divisor (debe ser mayor que 0):");
95     divisor = parseInt(divisor);
96 } while (isNaN(divisor) || divisor <= 0);
97
98 document.write("Dividendo=" + dividendo + "<br>" + " Divisor=" + divisor + "<br>" + "</br>");
99
100 document.write("Lista de operaciones" + "<br>" + "-----" + "</br>");
101
102 //Impresión de las restas.
103 if (dividendo < divisor) { //Como estamos utilizando números enteros y positivos, si el dividendo es menor que el divisor el cociente será 0.
104     document.write("El dividendo es menor que el divisor" + "<br>");
105 } else {
106     do { //Si el dividendo no es mayor que el divisor, se iniciará el bucle de restas sucesivas hasta que esta condición se cumpla.
107         dividendo = dividendo - divisor; //Por cada iteración, restamos el divisor al dividendo.
108         cociente++; //Cada ciclo supone una unidad más para el cociente
109         document.write("Operación " + cociente + ": cociente=" + cociente + " dividendo= " + dividendo + "<br>");//Mostramos el estado de cada iteración.
110     } while (dividendo >= divisor); //Lo repetiremos mientras que el dividendo sea mayor o igual al divisor.
111 }
112
113
114 document.write("<br>" + "Resultado" + "<br>" + "-----" + "</br>");
115 document.write("Resto: " + dividendo + "<br>");
116 document.write("Divisor: " + divisor + "<br>");
117 document.write("Cociente: " + cociente + "<br>");
118 </script>
119
120 </body>
121
122 </html>
```

Ejemplo de salida

