

2º DAW

Tarea 1.1 Hola mundo

Desarrollo Web en Entorno Cliente

Rafael López Cruz



19

Contenido

Apartado 1.....	3
Ejercicio 1	3
Visual Studio Code.....	3
Sublime Text.....	5
Atom.....	6
Ejercicio 2	7
Visual Studio Code.....	7
Sublime Text.....	10
Atom.....	11
Apartado 2.....	14
Ejercicio 1	14
Google Chrome	14
Mozilla Firefox.....	14
Microsoft Edge	15
Ejercicio 2	15
Google Chrome	15
Mozilla Firefox.....	15
Microsoft Edge	16
Ejercicio 3	16
Google Chrome	16
Mozilla Firefox.....	17
Microsoft Edge	17
Ejercicio 4	18
Google Chrome	18
Mozilla Firefox.....	18
Microsoft Edge	19
Ejercicio 5	19
Google Chrome	19
Mozilla Firefox.....	20
Microsoft Edge	20

Implementa los ejemplos propuestos en el apartado 1.4 del tema y pruébalo con los distintos navegadores web (Chrome, Edge, Firefox....)

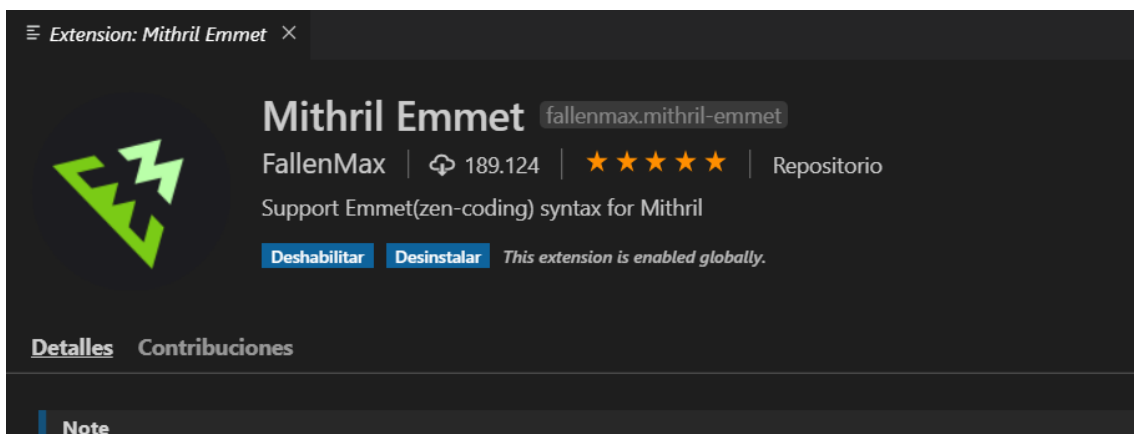
Utiliza al menos tres entornos de desarrollo: IntelliJ, Sublime, Atom, Visual Studio Code,...

Apartado 1. Realiza un tutorial por cada uno de los editores donde aparezca:

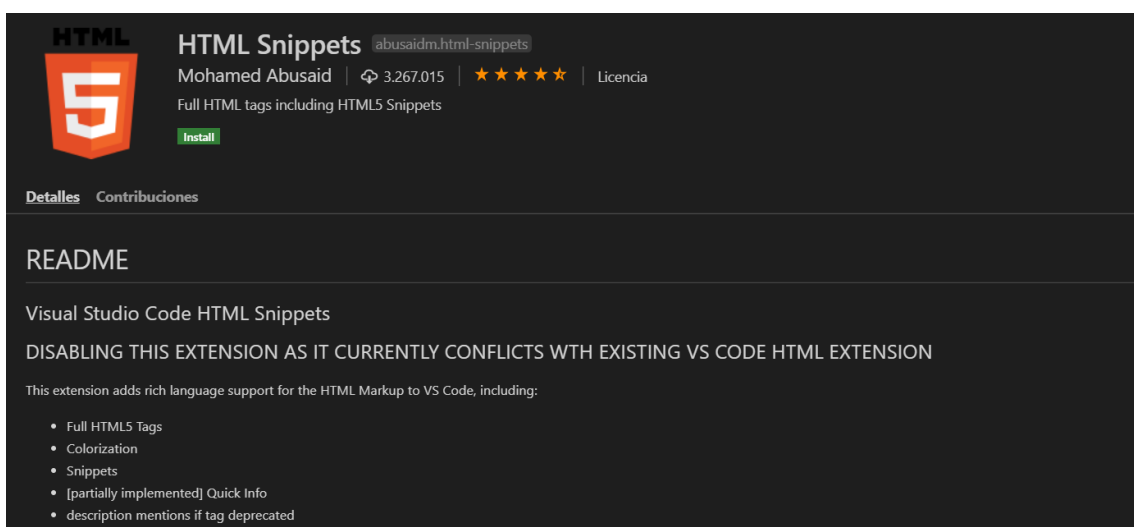
Ejercicio 1. Instalaciones necesarias para reconocimiento de código html y javascript

Visual Studio Code

Visual Studio Code ya trae por defecto soporte para HTML pero si queremos tener más funcionalidades vamos a instalar la siguiente extensión llamada emmet que trae soporte para diferentes lenguajes de marcas.



También vamos a instalar la extensión de HTML Snippets para un mayor soporte para todas las etiquetas de HTML5-

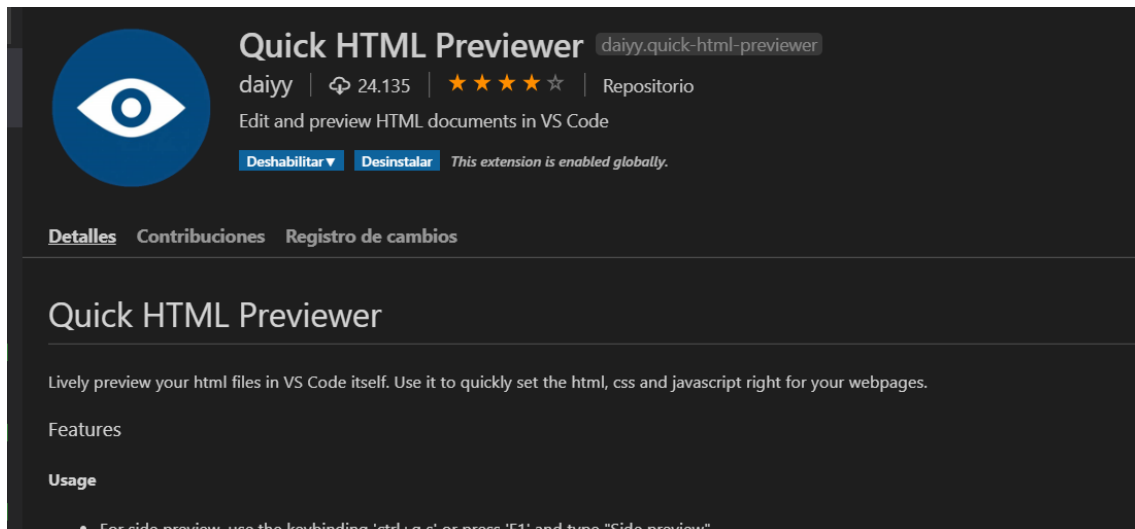


Tarea 1.1 Hola mundo

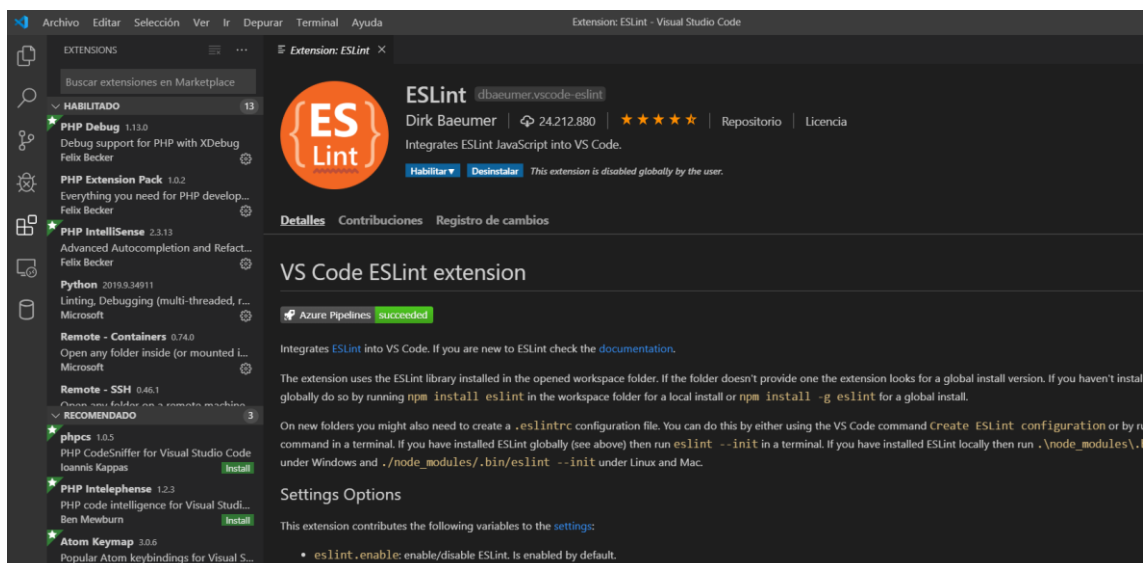
Desarrollo Web en Entorno Cliente

Rafael López Cruz

Vamos a aprovechar y vamos a instalar la siguiente extensión llamada Quick HTML Previewer para tener una previsualización del código HTML que vamos escribiendo.



Para instalar el soporte de JavaScript tendremos que instalar la extensión ESLint.



Sublime Text

Vamos a instalar la extensión de Emmet Css Snippets y Emmet Style Reflector para Sublime Text para tener soporte para HTML y CSS.

Emmet

Emmet Css Snippets

Emmet CSS completions for Sublime Text

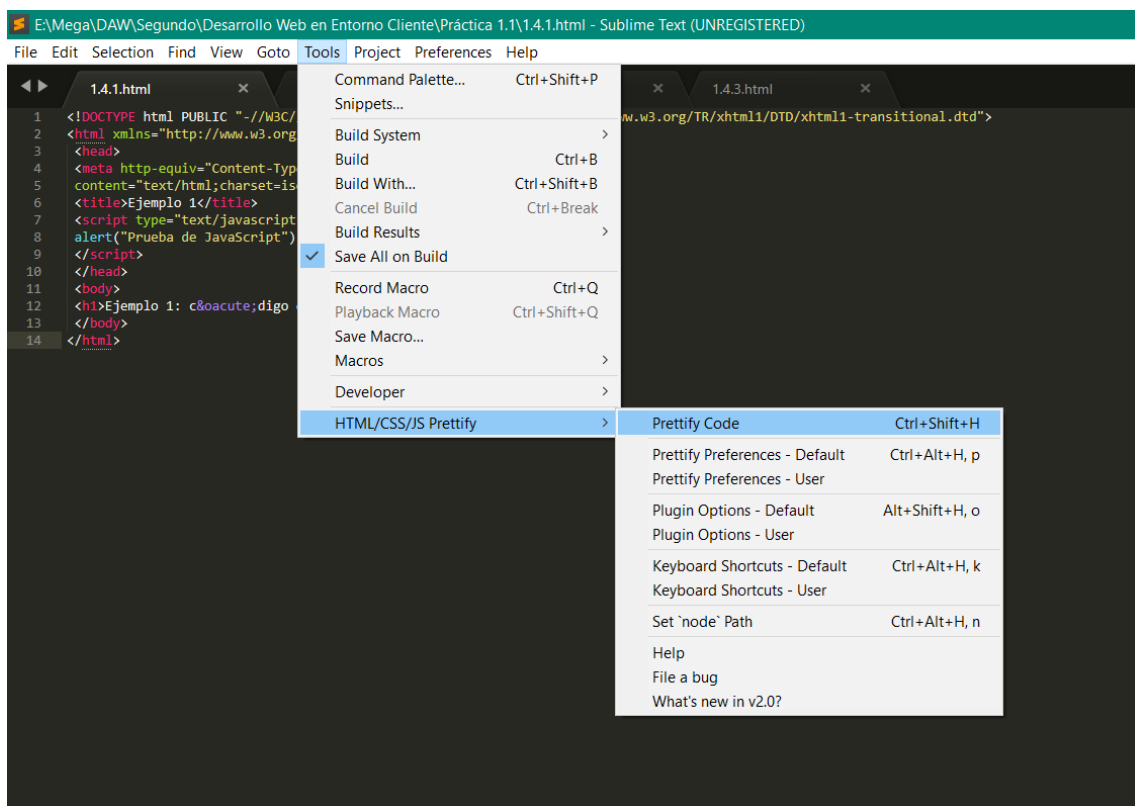
install v2014.05.11.15.10.35; p233.github.com/Emmet-Css-Snippets-for-Sublime-Text-2/

Emmet Style Reflector

Reflect Emmet HTML expansion in Sass/LESS

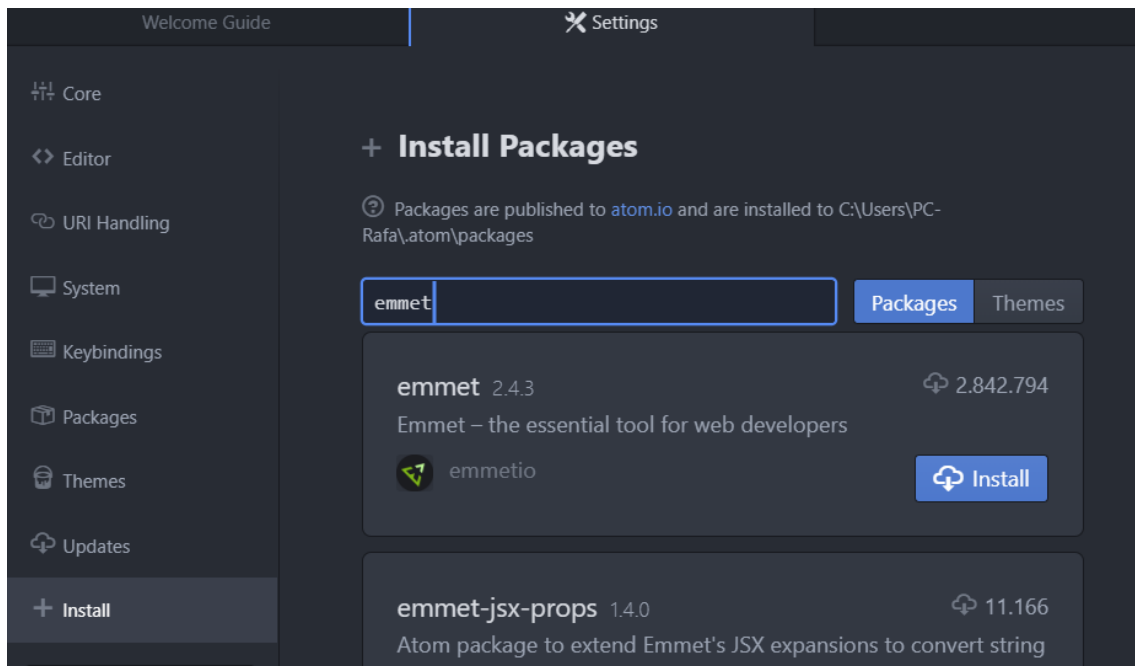
install v1.0.2; github.com/eecolella/emmet-style-reflector

Para tener soporte para JS Vanilla en Sublime Text vamos a instalar la extensión *HTML/CSS/JS Prettify*.

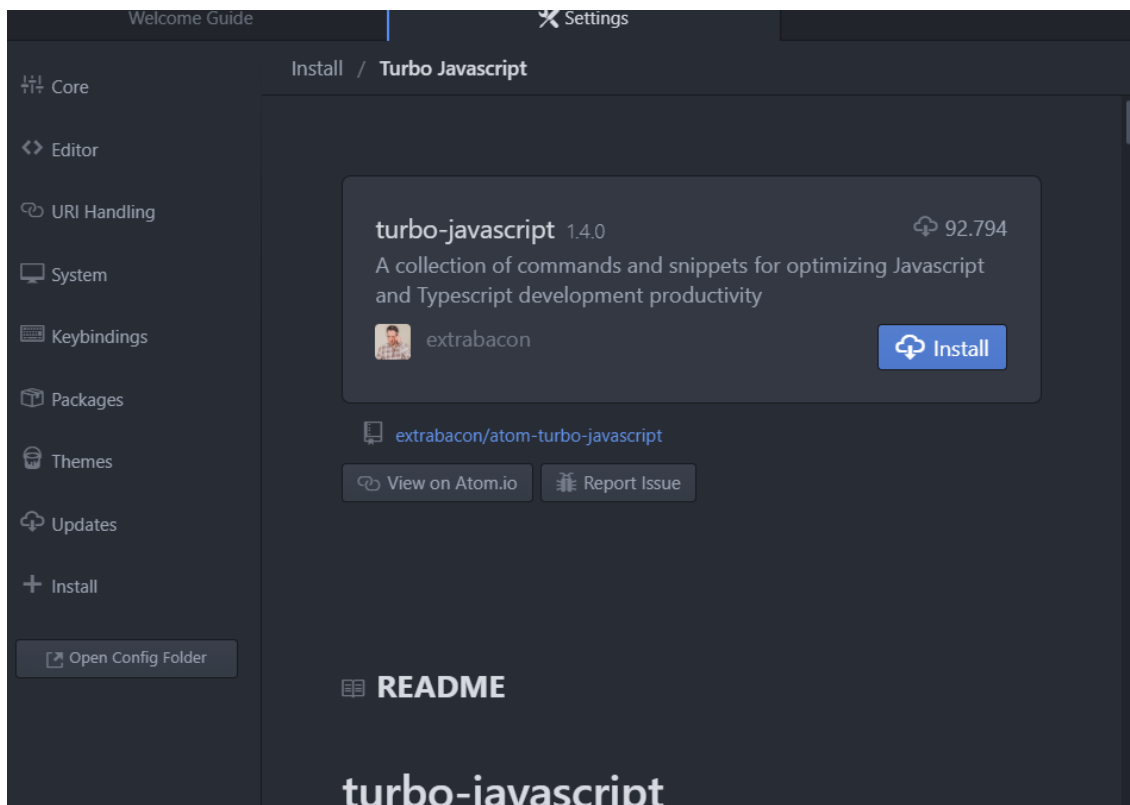


Atom

Vamos a instalar la siguiente extensión llamada emmet que trae soporte para diferentes lenguajes de marcas.



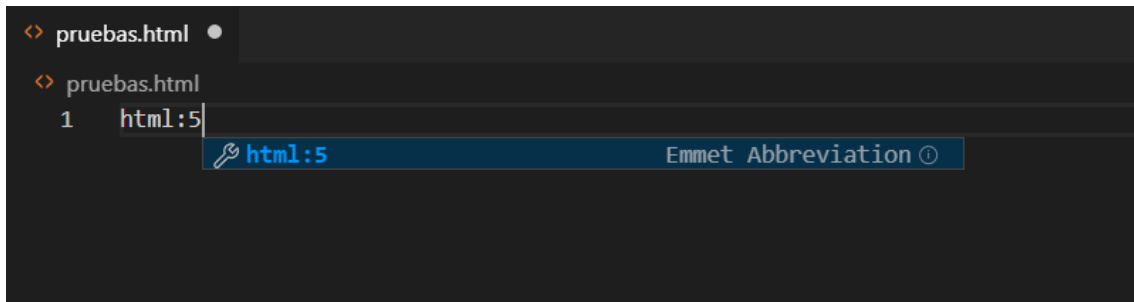
Para dar soporte a JavaScript vamos a instalar la extensión turbo-javascript.



Ejercicio 2. Atajos en ambos lenguajes. (Mínimo de 5)

Visual Studio Code

- Estructura de HTML5

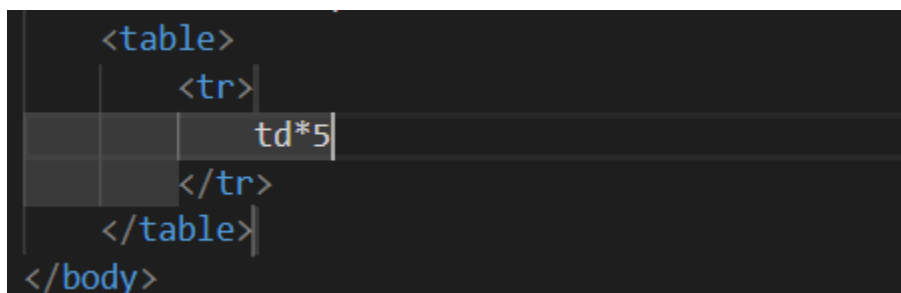


```
<> pruebas.html •
<> pruebas.html
1  html:5
    html:5 Emmet Abbreviation ⓘ
```



```
<> pruebas.html x
<> pruebas.html > html
1  <!DOCTYPE html>
2  <html lang="en">
3  <head>
4      <meta charset="UTF-8">
5      <meta name="viewport" content="width=device-width, initial-scale=1.0">
6      <meta http-equiv="X-UA-Compatible" content="ie=edge">
7      <title>Document</title>
8  </head>
9  <body>
10
11 </body>
12 </html>
```

- Soporte para cualquier etiqueta HTML y multiplicación de etiquetas.



```
<table>
  <tr>
    td*5
  </tr>
</table>
</body>
```

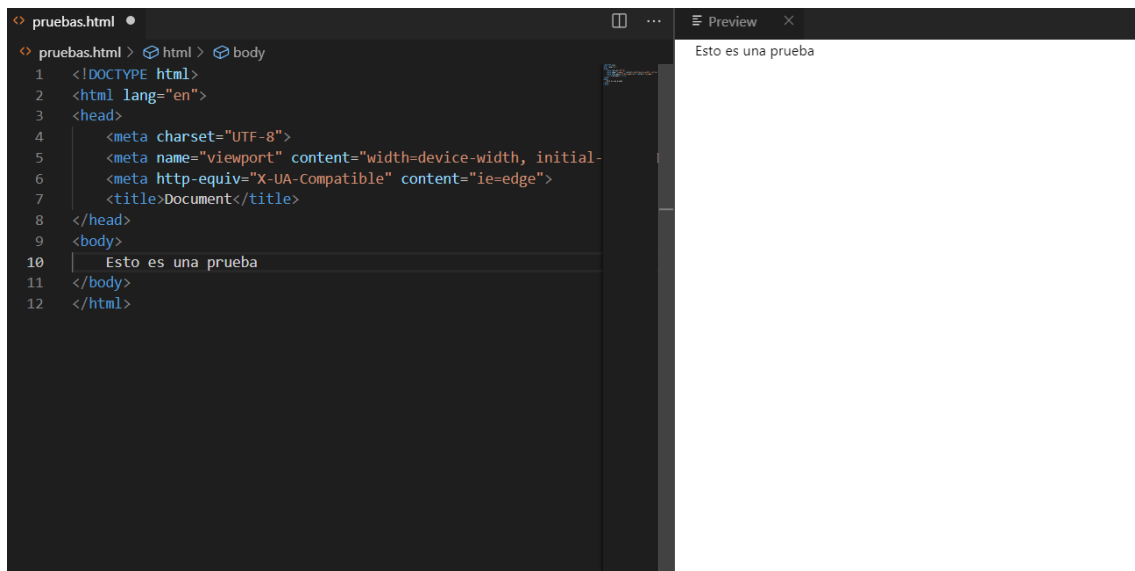
Tarea 1.1 Hola mundo

Desarrollo Web en Entorno Cliente

Rafael López Cruz

```
<table>
<tr>
  <td></td>
  <td></td>
  <td></td>
  <td></td>
  <td></td>
</tr>
</table>
</body>
```

- Ctrl + q + s: Preview del HTML y CSS que estamos editando a tiempo real.



- Script. Soporte para la etiqueta script y añadiendo un src.

```
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <meta http-equiv="X-UA-Compatible" content="ie=edge">
  <title>Document</title>
  <script>
</head>
<body>
  Esto es una prueba
</body>
</html>
```

- Soporte para cualquier función o método de JavaScript.

```
<title>Document</title>
<script>alert</script>
</head>
<body>
  Esto es una
</body>
</html>
```

Sublime Text

- html o html:5. Nos completa cualquier estructura de una etiqueta HTML.

```
15
16 html
```

```
<!DOCTYPE html>
<html>
<head>
  <title>*/title>
</head>
<body>

</body>
</html>
```

- Creación de estructuras. Por ejemplo vamos a crear una lista donde tenga 5 elementos dentro, para ello escribimos `ul>li*5` y pulsamos TAB.

```
ul>li*5|
```

```
<ul>
  <li>*/li>
  <li></li>
  <li></li>
  <li></li>
  <li></li>
</ul>
```

- `script/script:src`. Crea automáticamente la etiqueta `script` y si le indicamos el `src` nos lleva directamente al atributo para escribirlo.

```
script
script:src|
```

```
<script></script>
<script src="|"></script>
```

- table+. Crea la estructura completa de una tabla.

```
table+
```

```
<table>
  <tr>
    <td*/td>
  </tr>
</table>
```

- btn. Si escribimos btn y pulsamos TAB se nos escribe la etiqueta button.

```
btn
```

```
<button*/button>
```

Atom

- html:5. Pulsando Ctrl+E nos completa la estructura.

```
html:5
```

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <meta http-equiv="X-UA-Compatible" content="ie=edge">
  <title>Document</title>
</head>
<body>
</body>
</html>
```

- Creación de estructuras. Por ejemplo vamos a crear una tabla con 5 columnas y 5 filas y al final se le añade un footer.

```
(table>tr*5>(td*5))+footer
```

```
<table>
  <tr>
    <td></td>
    <td></td>
    <td></td>
    <td></td>
    <td></td>
  </tr>
  <tr>
    <td></td>
    <td></td>
    <td></td>
    <td></td>
    <td></td>
  </tr>
  <tr>
    <td></td>
    <td></td>
    <td></td>
    <td></td>
    <td></td>
  </tr>
  <tr>
    <td></td>
    <td></td>
    <td></td>
    <td></td>
    <td></td>
  </tr>
  <tr>
    <td></td>
    <td></td>
    <td></td>
    <td></td>
    <td></td>
  </tr>
</table>
<footer></footer>
```

- ret. Escribe return y punto y coma.

```
<script>
  ret
  → | ret      return
  <
  charset
  <---
```

```
8      <script>  
9          return ;  
10     </script>
```

- cl. Escribe un console.log.

```
<script>  
  cl  
  →| cl      console.log  
<  →| class  Class  
<  <> col  
<  <> colgroup  
</script>
```

```
<script>  
  console.log()  
</script>
```

- v y v=. Nos define una variable.

```
8      <script>  
9          v  
10     →| v      var statement  
11     <  →| v=    var assignment  
12     <  <> var  
13     <  <> video  
14     <  <>  
15     </script>
```

```
<script>  
  var name = value  
</script>
```

Tarea 1.1 Hola mundo

Desarrollo Web en Entorno Cliente

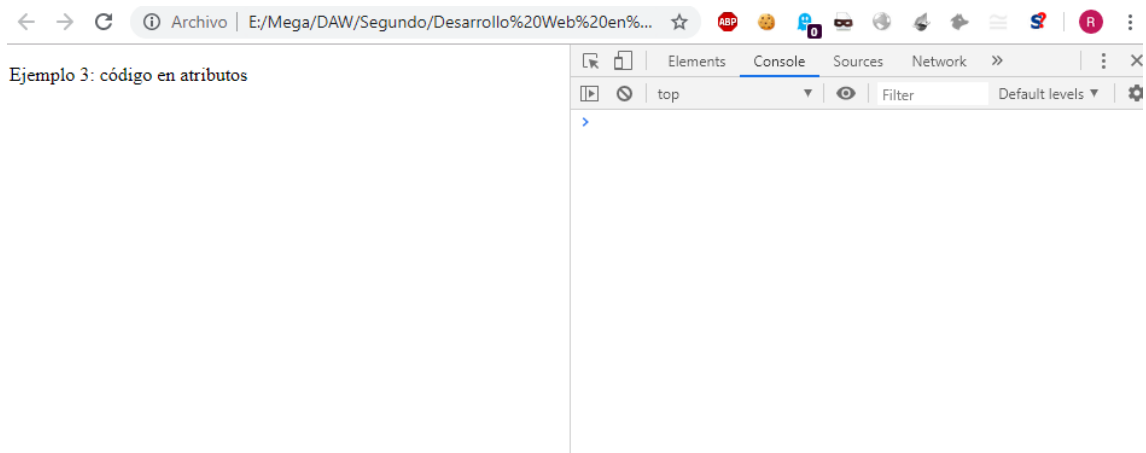
Rafael López Cruz

Apartado 2. En los tres navegadores indica:

Ejercicio 1. Teclas de acceso rápido que muestren las herramientas de desarrollo.

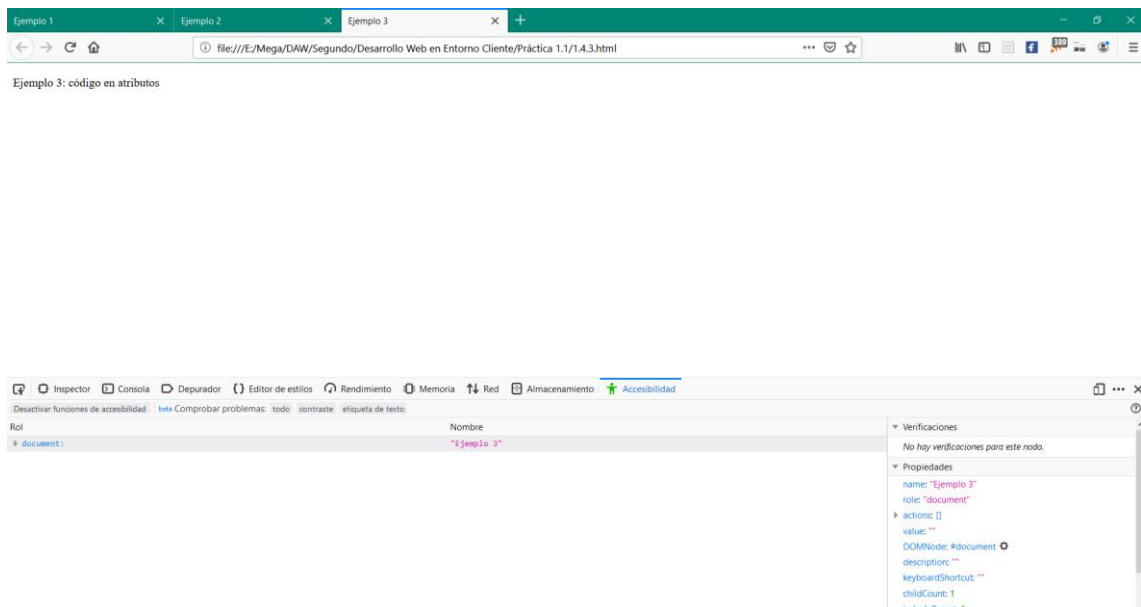
Google Chrome

Si pulsamos la tecla F12 podremos visualizar las herramientas de desarrollador.



Mozilla Firefox

Si pulsamos la tecla F12 podremos visualizar las herramientas de desarrollador.



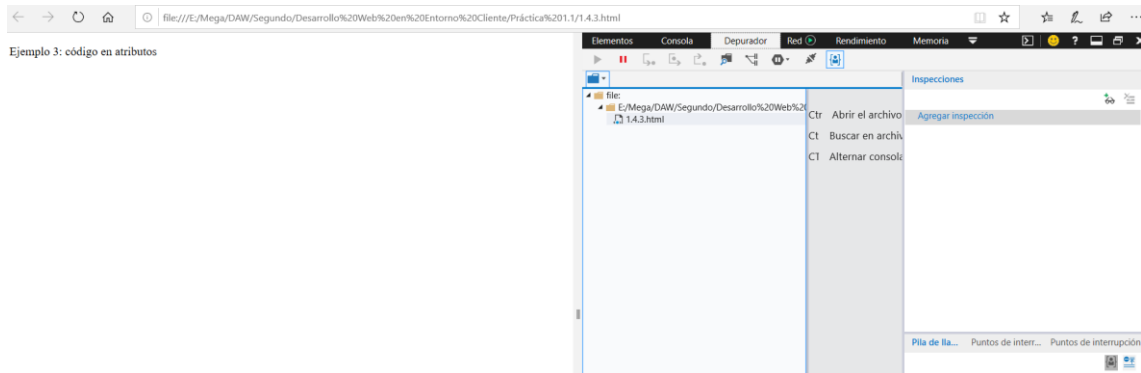
Tarea 1.1 Hola mundo

Desarrollo Web en Entorno Cliente

Rafael López Cruz

Microsoft Edge

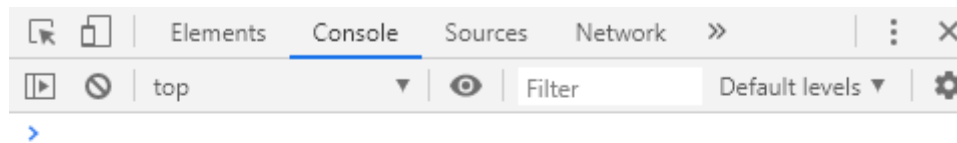
Si pulsamos la tecla F12 podremos visualizar las herramientas de desarrollador.



Ejercicio 2. Acceso a la consola

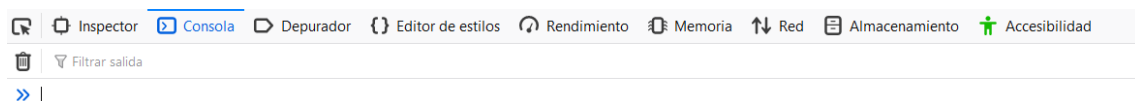
Google Chrome

Pulsamos en la pestaña “Consola”.



Mozilla Firefox

Pulsamos en la pestaña “Consola”.



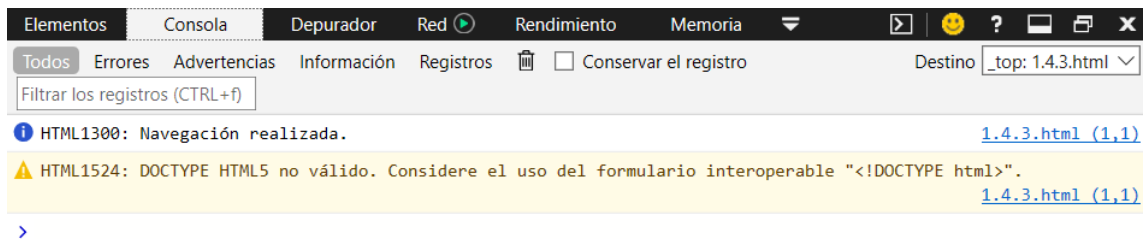
Tarea 1.1 Hola mundo

Desarrollo Web en Entorno Cliente

Rafael López Cruz

Microsoft Edge

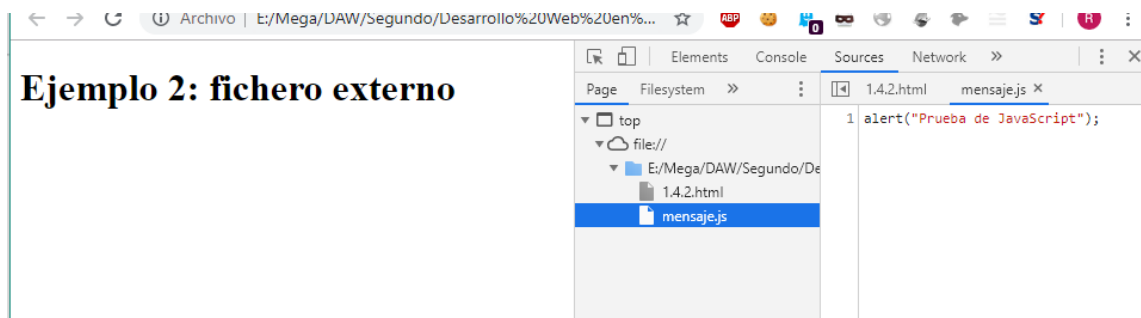
Pulsamos en la pestaña “Consola”.



Ejercicio 3. Acceso a los ficheros (mensaje.js en nuestro caso)

Google Chrome

En el apartado “Sources” podemos visualizar los ficheros cargados.



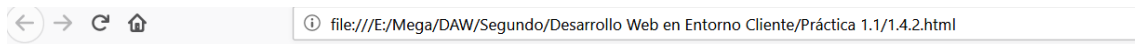
Tarea 1.1 Hola mundo

Desarrollo Web en Entorno Cliente

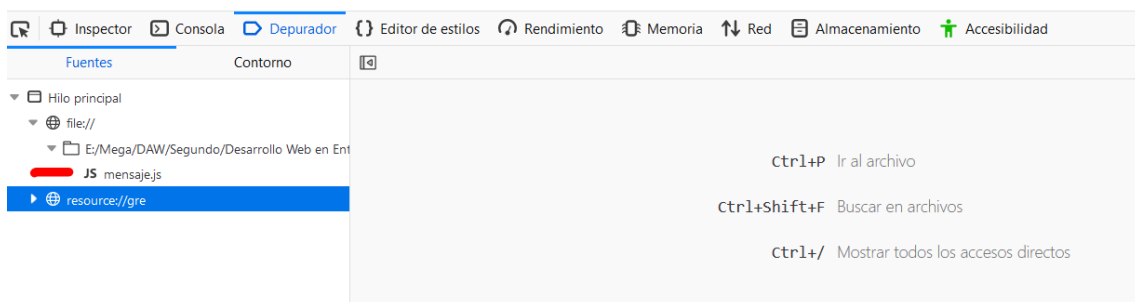
Rafael López Cruz

Mozilla Firefox

En el apartado “Depurador” podemos visualizar los ficheros cargados.

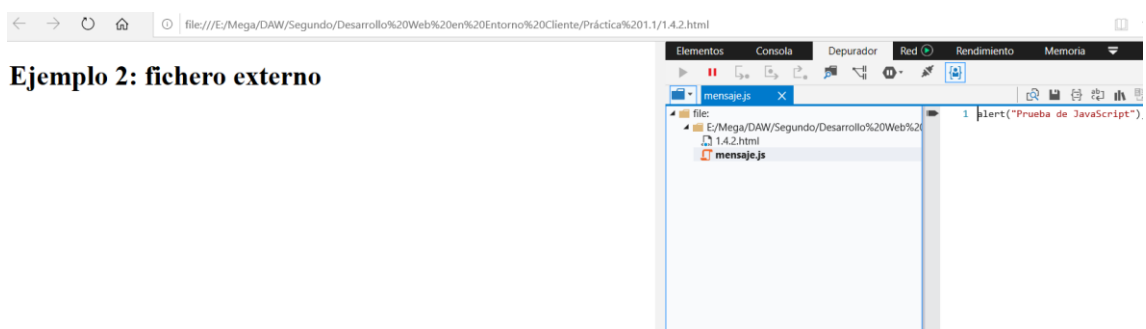


Ejemplo 2: fichero externo



Microsoft Edge

En el apartado “Depurador” podemos visualizar los ficheros cargados.



Tarea 1.1 Hola mundo

Desarrollo Web en Entorno Cliente

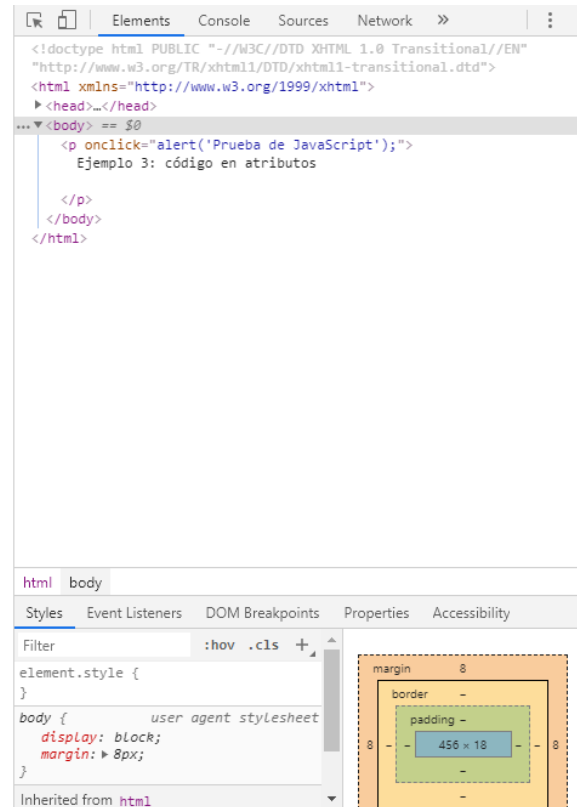
Rafael López Cruz

Ejercicio 4. Reglas CSS

Google Chrome

En el apartado “Elements” en “Styles” encontramos las reglas CSS.

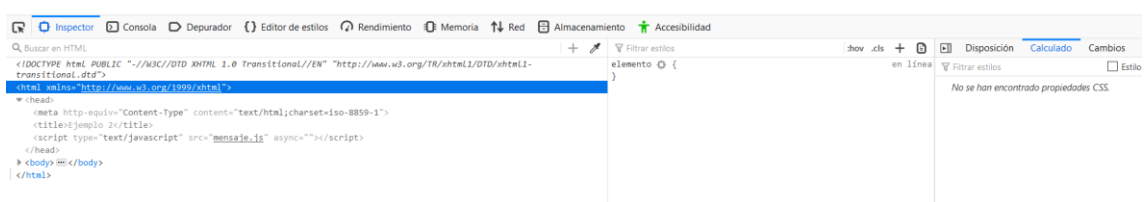
Ejemplo 3: código en atributos



Mozilla Firefox

En el apartado “Inspector” en “Calculado” encontramos las reglas CSS.

Ejemplo 2: fichero externo



Tarea 1.1 Hola mundo

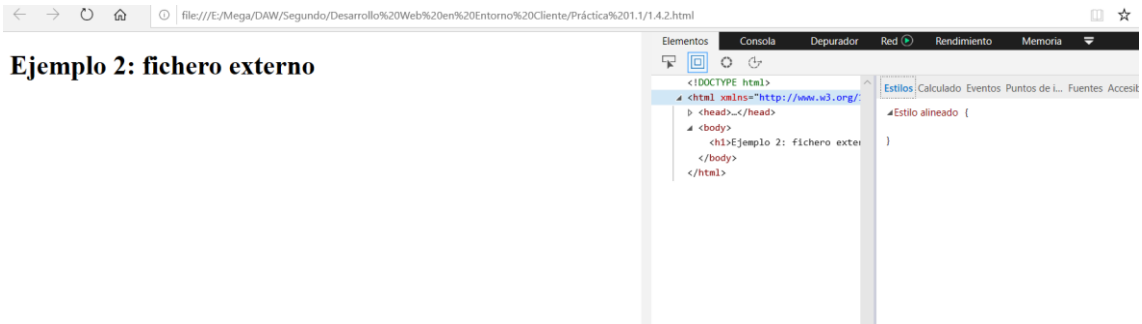
Desarrollo Web en Entorno Cliente

Rafael López Cruz

Microsoft Edge

En el apartado “Elementos” en “Estilos” encontramos las reglas CSS.

Ejemplo 2: fichero externo

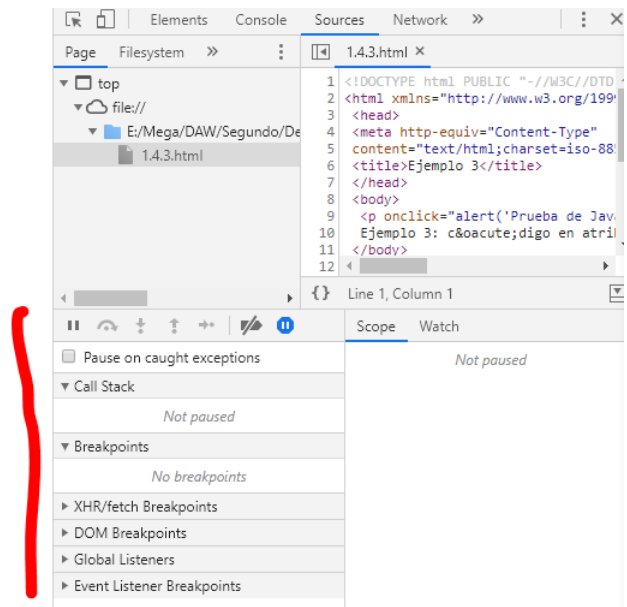


Ejercicio 5. Depurador

Google Chrome

En el apartado “Sources” podemos visualizar el depurador.

Ejemplo 3: código en atributos



Tarea 1.1 Hola mundo

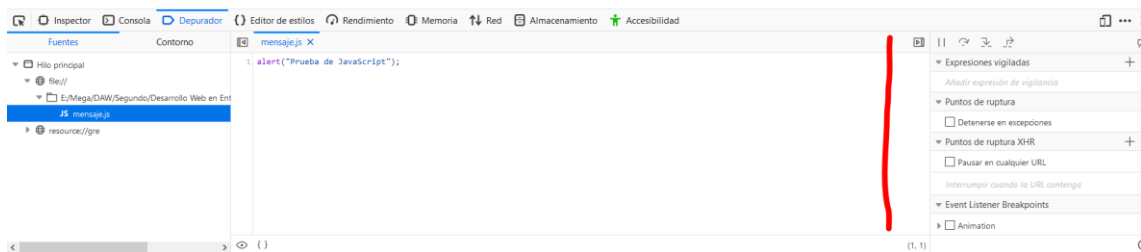
Desarrollo Web en Entorno Cliente

Rafael López Cruz

Mozilla Firefox

En el apartado “Depurador” podemos visualizar el depurador.

Ejemplo 2: fichero externo



Microsoft Edge

En el apartado “Depurador” podemos visualizar el depurador.

Ejemplo 2: fichero externo

