

Detectar el Navegador con la regla CSS @supports

Dado que los navegadores tienen su propia decisión de qué características incluir, nosotros los desarrolladores web a menudo tenemos que identificar si el navegador es compatible con funciones específicas y, a continuación llenamos el vacío utilizando polyfills (si es necesario). La forma más popular de manejar este tipo de situaciones es mediante el uso de Modernizr pero sabes que también podemos hacerlo usando sólo CSS?

Hay una nueva regla que se formula en el proyecto de W3C, llamada **@supports**. En este post, que le guiará a través de cómo funciona esta regla para que pueda detectar las características del navegador a través de CSS.

El uso de la regla @supports

A diferencia de Modernizr, la regla **@supports** requiere la propiedad y el valor especificado. Digamos que queremos aplicar CSS3 red sólo para el navegador que lo soporte. Podemos escribir de esta manera.

```
[code lang=»css»]
```

```
@support (display: grid;) {  
  .selector {  
    display: grid;  
    grid-column: 3;  
    grid-row: 1;  
    grid-row-span: 2;  
  }  
}
```

```
[/code]
```

Las reglas CSS bajo el **@support** sólo se aplicará si la condición devuelve true. Teniendo en cuenta el ejemplo anterior, se aplicará si el navegador soporta la pantalla: grid. O bien, los navegadores se aplicarán otras normas equivalentes fuera **@support**.

Condicionales CSS

Además, nos permite combinar varias condiciones utilizando y, o, y no el operador. Por ejemplo, digamos que queremos aplicar las reglas sólo si el navegador cumple con las siguientes condiciones:

- Es compatible con cualquier cuadrícula CSS3 o CSS3 Flexbox.
- Es compatible con Columnas. CSS3

En ese caso, podemos escribir las reglas condicionales, de la siguiente manera.

```
[code lang=»css»]
@support
((display: grid;) or (display:flexbox;))
and (column-count:3;) {
/*
Las declaraciones CSS van aqui
*/
}
[/code]
```

Recuerde que si se incluyen múltiples operadores en un único conjunto de reglas, cada operador debe ser agrupado en paréntesis como hemos mostrado anteriormente.

Escribiendo de la siguiente manera no es válido, y va a confundir el navegador.

```
[code lang=»css»]
@support
```

```
(display: grid;) or (display:flexbox;) and (column-  
count:3;)    {  
/*  
Las declaraciones CSS van aqui  
*/  
}  
[/code]
```

Esta característica es ideal y sería muy útil en varias ocasiones.

Para las pruebas, sin embargo, los navegadores deben soportar esta norma, de lo contrario este método fallará. Y en el momento de escribir este artículo, Opera parece ser el único navegador que ha implementado esta característica. Así, hasta que la especificación es estable y ha sido cubierto por todos los otros navegadores, retrasar el uso de este método para poner a prueba durante las etapas de producción.