



## Bootstrap (3)

# Base del Grid (no olvidar)

	<b>Extra small devices Phones (&lt;768px)</b>	<b>Small devices Tablets (≥768px)</b>	<b>Medium devices Desktops (≥992px)</b>	<b>Large devices Desktops (≥1200px)</b>
Grid behavior	Horizontal at all times	Collapsed to start, horizontal above breakpoints	Collapsed to start, horizontal above breakpoints	Collapsed to start, horizontal above breakpoints
Max container width	None (auto)	750px	970px	1170px
Class prefix	<b>.col-xs-</b>	<b>.col-sm-</b>	<b>.col-md-</b>	<b>.col-lg-</b>
# of columns	12	12	12	12
Max column width	Auto	60px	78px	95px

## <head>Meta</head> (1)

- Para asegurarnos de que se va a 'renderizar' la página tal y como queremos, con un escalado 1:1 (se reinicia el zoom que tengamos) aplicamos el initial-scale=1.0
- Y para asegurarnos de que el ancho en el que nos basamos es el correcto aplicamos el width=device-width.

```
<meta name="viewport" content="width=device-width, initial-scale=1.0">
```

## <head>Meta</head> (2)

- Si quisieramos deshabilitar que el usuario haga zoom, más parecido a una app nativa. Se suele utilizar `maximum-scale=1.0` con `user-scalable=no`, con uno te aseguras la escala máxima y con otro que el usuario no pueda escalarlo.
- Bootstrap no recomienda el `user-scalable` en todas las

```
<meta name="viewport" content="width=device-width,  
                                initial-scale=1.0,  
                                maximum-scale=1.0,  
                                user-scalable=no">
```

# Nesting

- Si queremos anidar (nesting) dentro de un elemento de tipo **col-\*-\*** simplemente tenemos que añadir otra row dentro y también las nuevas **col-\*.\***. Estas tendrán el elemento '**padre**' como si fuera un elemento de tipo row.

```
<div class="col-lg-6 col-sm-3" style="background-color: #9fa">Nivel 1:  
  <div class="row">  
    <div class="col-lg-4">  
      Nivel 2: Yo estoy anidado!!
```

# Reordenando (push & pull)

- Si queremos modificar que elemento está situado en cada lugar dependiendo del dispositivo utilizaremos las clases `col-*-pull-*` y `col-*-push-*`
- En este ejemplo en dispositivos lg ( $\geq 1200\text{px}$ ) intercambiamos la posición de B y A

```
<div class="col-sm-4 col-lg-push-4">  
  <div>B</div>  
</div>  
<div class="col-sm-4 col-lg-pull-4">  
  <div>A</div>  
</div>
```

# Cracking IE8 and lower (IE)

- Dado que utiliza muchos elementos de CSS3 y HTML5 vamos a añadir unos scripts para que los navegadores de IE < a IE9 puedan funcionar (HTML5 y CSS3 no están soportados por dichos navegadores)

```
<!-- [if lt IE 9]>  
    <script src = "https://oss.maxcdn.com/libs/html5shiv/3.7.0/html5shiv.js"></script>  
    <script src="https://oss.maxcdn.com/libs/respond.js/1.4.2/respond.min.js"></script>  
<![endif]-->  
</head>
```

# IE-edge (IE)

- Para que los navegadores de IE utilicen la ultima versión de su 'engine' se utiliza el siguiente código, más info en <https://www.modern.ie/en-us/performance/how-to-use-x-ua-compatible>

```
<meta http-equiv="X-UA-Compatible" content="IE=edge">
```