

# Diseño para dispositivos móviles



*Diseño de Sitios Web*

# Diseño Responsive

- El diseño Web Responsive hace que la página web se vea “bien” en todos los dispositivos (computadoras, tablets y teléfonos)
- Utiliza CSS and HTML para redimensionar, ocultar, ajustar, aumentar o mover el contenido para que se vea bien en cualquier pantalla.

# Diseño Responsive



# Usando clases css

- Es posible lograr el ordenamiento de los elementos si se colocan en clases de manera tal que cuando el tamaño de pantalla cambie, los elementos fluyan
- [Ejemplo](#)

# Usando Viewport

- `<meta name="viewport" content="width=device-width, initial-scale=1.0">`
- `<meta>` el elemento viewport le da instrucciones al navegador de como debe controlar las dimensiones de la página.
- `width=device-width` establece el ancho de la página para seguir el tamaño del ancho de pantalla del dispositivo.
- `initial-scale=1.0` establece el nivel de zoom inicial cuando se carga la página por primera

sin usar viewport

usando viewport



# Recomendaciones

- No utilizar elementos de gran tamaño y fijo. Ej. Imágenes
- No confiarse en usar un único viewport. Por los diferentes tamaños de pantalla.
- Usar css con valores relativos considerando el 100% de tamaño

# Usando Grilla

- Usar una grilla para el diseño de la página facilita la ubicación de los elementos en la página.
- Una grilla responsive tiene por lo general 12 columnas que ocupan un total del 100% las cuales se expanden o contraen cuando se redimensiona la página.
- Ej de grilla  
[http://www.w3schools.com/css/tryresponsive\\_grid.htm](http://www.w3schools.com/css/tryresponsive_grid.htm)

# Construir una grilla

- Asegurarme que todos los elementos unifiquen el borde y el padding

```
* {  
  box-sizing: border-box;  
}
```

Ej

[http://www.w3schools.com/css/tryit.asp?filename=tryresponsive\\_webpag](http://www.w3schools.com/css/tryit.asp?filename=tryresponsive_webpag)

- Calcular el tamaño de la grilla  $100\% / 12 \text{ columns} = 8.33\%$ .
- Crear la clase col-

```
[class*="col-"] {  
  float: left;  
  padding: 15px;  
  border: 1px solid red;  
}
```

```
.col-1 {width: 8.33%;}  
.col-2 {width: 16.66%;}  
.col-3 {width: 25%;}  
.col-4 {width: 33.33%;}  
.col-5 {width: 41.66%;}  
.col-6 {width: 50%;}  
.col-7 {width: 58.33%;}  
.col-8 {width: 66.66%;}  
.col-9 {width: 75%;}  
.col-10 {width: 83.33%;}  
.col-11 {width: 91.66%;}  
.col-12 {width: 100%;}
```

# Creo filas

```
<html>
```

```
....
```

```
<div class="row">
```

```
  <div class="col-3">...</div>
```

```
  <div class="col-9">...</div>
```

```
</div>
```

```
...
```

```
</html>
```

[Ejemplo](#)

# Media Queries

- Las **media queries** son útiles cuando deseas modificar tu página web o aplicación en función del tipo de dispositivo (como una impresora o una pantalla) o de características y parámetros específicos (como la resolución de la pantalla o el ancho del viewport del navegador).
- Se utilizan para:
  - Aplicar estilos condicionales con las [reglas-at @media](#) e [@import](#) de [CSS](#).
  - Indicar medios específicos en los elementos [<link>](#), [<source>](#) y otros elementos [HTML](#).

# @Media queries

- Permite personalizar los estilos CSS basándonos en las características del dispositivo como, por ejemplo: tv, pc, tablet, movil o si esta horizontal (paisaje) o en vertical (retrato), etc; con ello y conociendo los tamaños más comunes de dispositivos de salida:
  - 320px
  - 480px
  - 600px
  - 768px
  - 900px
  - 1200px

# Como se definen

- Varias consultas se pueden combinar utilizando operadores lógicos (and, or ,not). No distinguen entre mayúsculas y minúsculas.
- El resultado de la consulta es "verdadero" cuando el tipo de medio (si se especifica) coincide con el dispositivo en el que se está mostrando el documento y todas las expresiones en el *media query* son "verdaderas". En este caso, se aplica los estilos correspondientes, siguiendo las reglas usuales de cascada.

# Diferencias entre las propiedades y características de los medios de comunicación

- Las propiedades son usadas en declaraciones para dar información sobre la forma de presentar un documento. Para las características de los medios se utilizan expresiones para describir los requisitos del dispositivo de salida.
- La mayoría de las características de los medios aceptan el opcional ' min- 'o' max- '. Prefijos para expresar «mayor o igual que» y «menor o igual que» en las limitaciones.
- Las propiedades requieren siempre un valor para formar una declaración. Las características de los medios, por otro lado, también se puede utilizar sin un valor. Las funciones de medios que llevan el prefijo min / max no se puede utilizar sin un valor. Sería una consulta (Media Queries) incorrecta.
- Las propiedades pueden aceptar valores más complejos, por ejemplo, los cálculos que involucran a otros valores. Los medios de comunicación (Media features) solo aceptan valores individuales: una palabra clave, un número o un número con un identificador de la unidad. Las únicas excepciones de funciones de medios (Media features) son el 'aspect-ratio' y 'device-aspect-ratio'.

# Ejemplo

```
<link rel="stylesheet" media="(max-width: 800px)" href="example.css" />
```

```
/* Para 960px */
```

```
@media only screen and (max-width: 980px) and (min-width: 821px) {
```

```
}
```

```
/* Para 800px */
```

```
@media only screen and (max-width: 820px) and (min-width: 621px) {
```

```
}
```

```
/* Para 480px */
```

```
@media only screen and (max-width: 500px) and (min-width: 341px) {
```

```
}
```

```
/* Para 320px */
```

# Media en CSS2

CSS2: define la misma lista de html 4 de types (**'print', 'screen', 'print', entre otros**) y CSS acepta algunos y añade **'embossed'** y **'speech'**. También, **'all'** es usado para aplicar la hoja de estilo a todas las media types.

## Documento.html

```
<link rel="stylesheet" type="text/css" media="screen" href="pantalla.css">
```

## Pantalla.css

```
@media only screen {  
  * { font-family: times }  
}
```

# Media en CSS3

- Media Queries consiste en una media type y cero o más expresiones que compruebe las condiciones de determinadas características de los Media Features.

documento.html

```
<link rel="stylesheet" media="screen and (color)"  
href="ejemplo.css" />
```

# Media Features

- **Width:** El valor es de longitud (length) y se aplica a tipos de medios visuales y táctiles. Acepta los prefijos min / max. Con 'width' media feature especificamos el ancho de la zona de visualización dirigido del dispositivo de salida.
- **Height:** El valor es de longitud (length) y se aplica a tipos de medios visuales y táctiles. Acepta los prefijos min / max. Con 'height' media feature especificamos el alto de la zona de visualización dirigido del dispositivo de salida.

# Media Features

- Orientation: portrait | landscape No acepta los prefijos min / max.
- El valor de «orientation» es 'portrait' (retrato) cuando el valor de la **altura** del medio de salida es mayor o igual al valor de la **anchura** de ese medio.  
De lo contrario el valor es 'landscape' (paisaje).

# Ejemplos

- Esta Media Queries expresa que hoja de estilo se va a aplicar en dispositivos con ventana (la parte de la pantalla / papel donde se procesa el documento) con anchuras entre 400 y 700 píxeles y con las reglas CSS siguientes:

```
@media screen and (min-width: 400px) and (max-width: 700px) {  
body{  
background-color:#555;  
font-family: Times New Roman;  
}  
header h3{  
    text-align:center;  
}  
}
```

# Ejemplos

- `@media all and (orientation:portrait) { /*Reglas CSS*/ }`
- `@media all and (orientation:landscape) { /*Reglas CSS*/ }`

# Ejemplos

- `@media only screen and (max-width: 500px) {  
 body {  
 background-color: lightblue;  
 }  
}`

# estilo.css

```
body {  
  margin: 0px;  
  padding: 0px;  
  background-color: #990000;  
}  
@media (max-width: 480px) {  
  body {  
    background-color: #3333FF;  
  }  
}
```

# Otros

- Usando w3.css: Es un framework definido por w3c para facilitar el diseño responsive

```
<link rel="stylesheet"  
href="http://www.w3schools.com/lib/w3.css">
```