Diseño Web Responsivo

Juan Carlos Trejo

Módulo 1

Introducción al diseño web responsivo

Introducción a RWD

El Responsive Web Design (Diseño Web Responsivo) permite que los sitios web se adapten a diferentes tamaños de pantalla, desde computadoras de escritorio hasta móviles. Utiliza técnicas como media queries, unidades relativas, cajas y otras unidades para cambiar el diseño según el dispositivo del usuario.

Introducción a RWD

Los elementos clave del diseño responsivo incluyen:

- Media Queries: permiten aplicar estilos según el tamaño de pantalla.
- Grid y Flexbox: sistemas de diseño CSS modernos que se ajustan dinámicamente.
- Imágenes flexibles: se redimensionan dentro de su contenedor.
- Tipografías escalables: uso de unidades relativas como em, rem, %.

Introducción a RWD

Importancia del RWD

Hoy en día, más del 50% del tráfico web proviene de dispositivos móviles. RWD mejora:

- Accesibilidad
- Experiencia de usuario
- Prioriza sitios mobile-friendly
- Reducción de mantenimiento (una sola versión del sitio)

Introducción a RWD

RWD vs Diseño adaptativo

Característica	RWD	Diseño Adaptativo
Diseño fluido	Sí	No, usa diseños fijos predefinidos
Media queries	Sí	Opcional
Flexibilidad total	Alta	Menor
Mantenimiento	Menor	Mayor (diseños separados por dispositivo)

Introducción a RWD

Algunas buenas practicas

- Diseñar con contenido en mente.
- Evitar tamaños fijos.
- Usar unidades relativas.
- Pruebas continuas en diferentes dispositivos.
- Optimizar recursos.
- Minimizar el uso de frameworks si no son necesarios.

Introducción a RWD

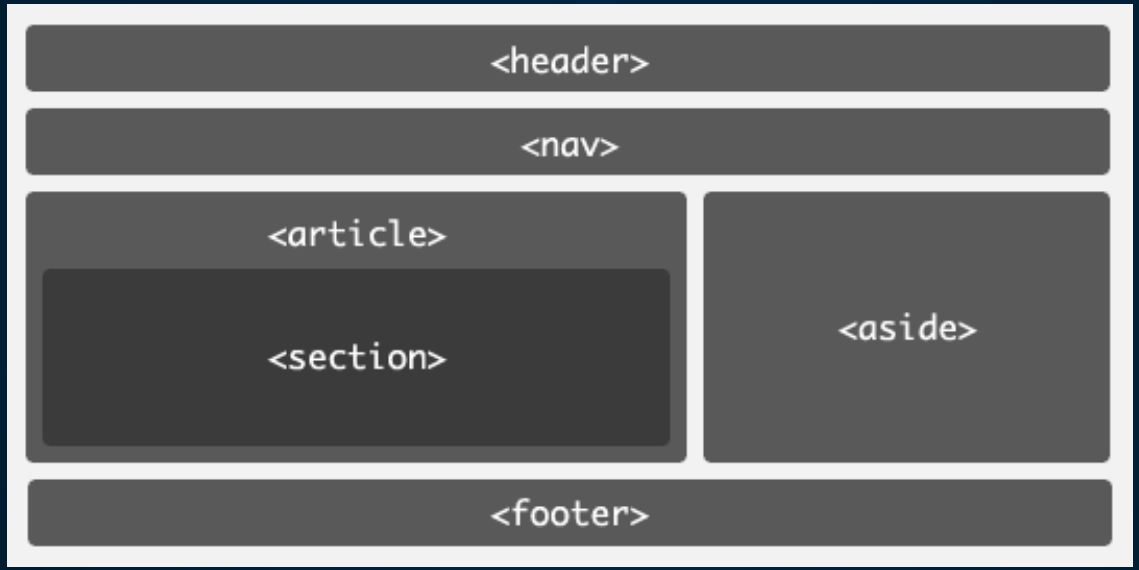
Elementos semánticos

```html
<!DOCTYPE html>
<html lang="es">

<head>
    <meta charset="UTF-8" />
    <meta name="viewport" content="width=device-width, initial-scale=1.0" />
    <title>Mi sitio responsivo</title>
    <link rel="stylesheet" href="estilos.css" />
</head>

<body>
    <!-- contenido -->
</body>

</html>
```

En conjunto, estos elementos establecen una base sólida para que un sitio web pueda adaptarse y responder eficazmente a diferentes tamaños de pantalla y dispositivos, lo cual es la esencia del diseño web responsivo.

Introducción a RWD

Elementos semánticos



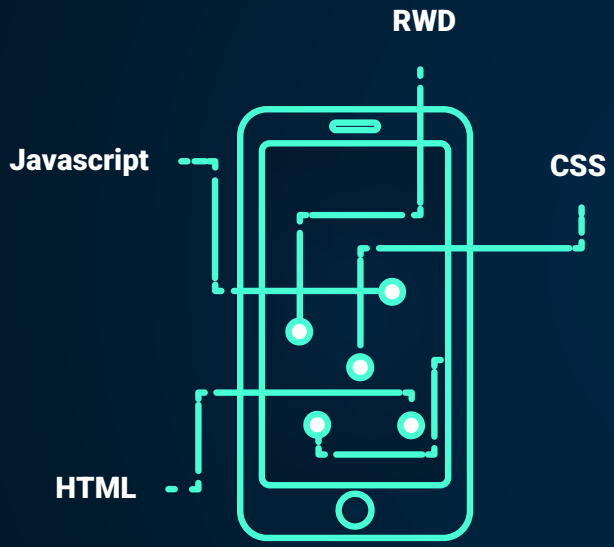
Semantics - Glossary | MDN

Introducción a RWD

Elementos semánticos

Elemento Semántico	
<header>	Contenido introductorio o navegación principal.
<nav>	Enlaces de navegación del sitio.
<main>	Contenido principal y único de la página.
<section>	Agrupación temática de contenido.
<article>	Contenido independiente y autónomo.
<aside>	Contenido lateral relacionado.
<footer>	Pie de página con info extra (derechos de autor).

Semantics - Glossary | MDN

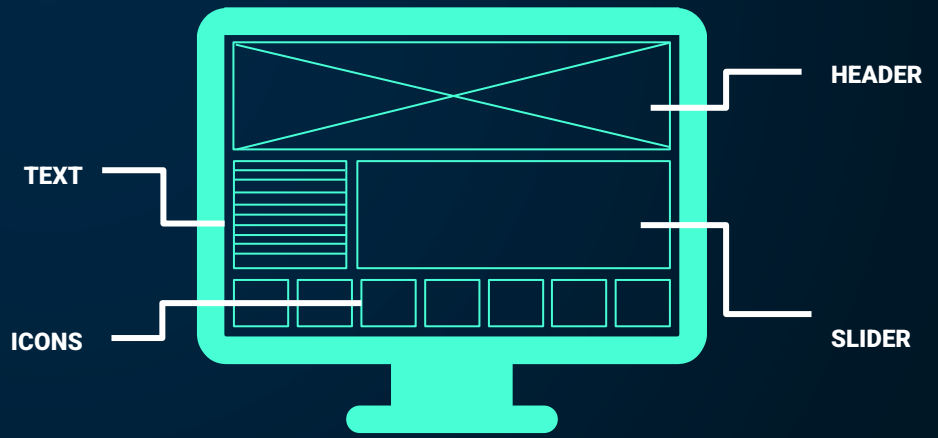


Módulo 2

Elementos del RWD

Elementos del RWD

CSS (Cascading Style Sheets) se utiliza para definir la apariencia de los elementos HTML.



Elementos del RWD

Posiciones de los elementos

`position` es una propiedad que controla cómo un elemento se posiciona dentro del flujo normal del documento de una página web.

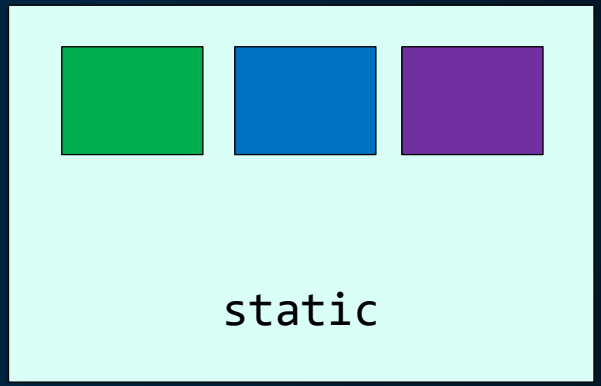
Es decir, determina cómo se ubica y se comporta un elemento en relación con otros elementos o con su contenedor.

Valor	
static	Valor por defecto (flujo normal del documento)
relative	Se mueve en relación a su posición original
absolute	Se posiciona respecto al ancestro más cercano que no tenga static
fixed	Fijo respecto a la ventana (scroll no lo afecta)
sticky	Se comporta como relative hasta que alcanza cierto punto y se vuelve fixed

Elementos del RWD

Posición de los elementos

`static` es el valor por defecto de todos los elementos HTML. El elemento se posiciona según el flujo normal del documento.

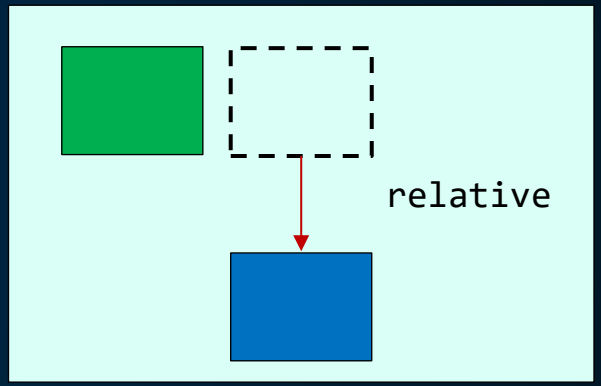


position - CSS | MDN

Elementos del RWD

Posición de los elementos

Usando la posición `relative` el elemento se posiciona relativo a su posición original en el flujo normal del documento.

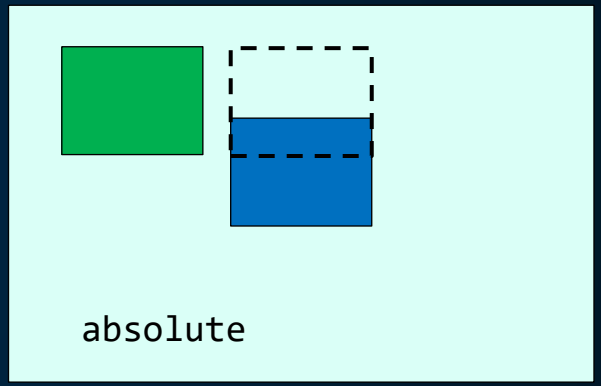


position - CSS | MDN

Elementos del RWD

Posición de los elementos

Cuando se usa **`absolute`**, el elemento se posiciona relativo al ancestro posicionado más cercano, es decir, el elemento padre que tenga position relative, absolute, fixed, o sticky.

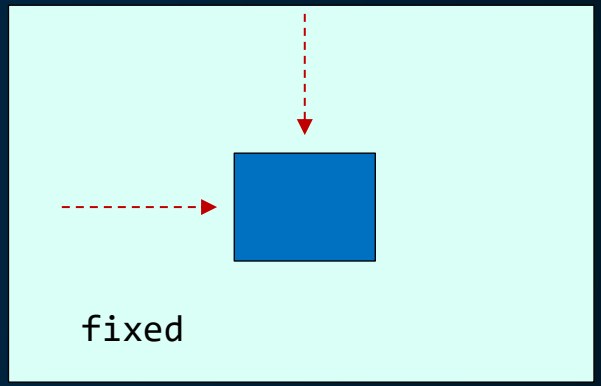


position - CSS | MDN

Elementos del RWD

Posición de los elementos

`fixed` se posiciona fuera del flujo del documento, pero se fija con respecto al viewport y no depende de ningún contenedor padre.

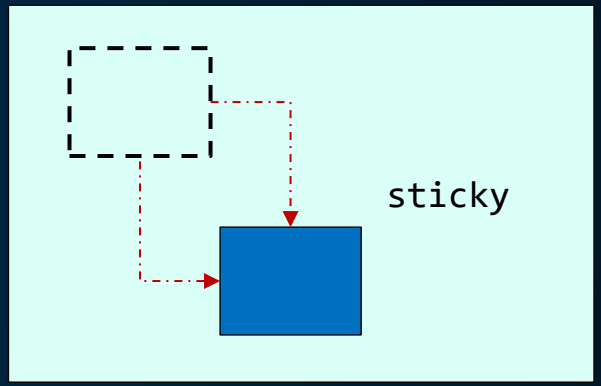


position - CSS | MDN

Elementos del RWD

Posición de los elementos

`sticky` es un híbrido entre relative y fixed. Es decir, se comporta como relative hasta que su posición de desplazamiento (scroll position) alcanza un umbral determinado, momento en el que se vuelve fixed.



position - CSS | MDN

Elementos del RWD

Valor	Descripción	¿Afecta al flujo normal?	¿Se puede mover?	¿Se mueve con scroll?	¿A quién toma como referencia?
static	Valor por defecto. El elemento sigue el flujo normal del documento.	No	No	Sí	No se posiciona, es parte del flujo normal.
relative	Se posiciona respecto a su posición original en el flujo normal.	No	Sí	Sí	Su posición original en el documento.
absolute	Se elimina del flujo normal y se posiciona respecto a un contenedor padre.	Sí	Sí	Sí	El ancestro más cercano con posición relative, absolute o fixed. Si no hay, usa viewport
fixed	Se elimina del flujo y queda fijo en pantalla, incluso al hacer scroll.	Sí	Sí	No	Siempre con referencia al viewport (pantalla).
sticky	Se comporta como relative hasta que se activa el scroll, y luego se “pega”.	No	Sí	Depende	Su contenedor inmediato. Cambia de relative a fixed al activarse.

Elementos del RWD

Sombras

Las sombras en CSS permiten añadir profundidad, relieve y atractivo visual a los elementos de texto y caja

- Sombra de caja
- Sombra de texto

box-shadow - CSS | MDN

Elementos del RWD

Sombras

La propiedad `box-shadow` se aplica a elementos HTML que son cajas como div, p, img, etc. y que permiten crear una o más sombras alrededor del marco de un elemento.

```
box-shadow: [offset-x] [offset-y] [blur-radius] [spread-radius] [color] [inset];
```

Elementos del RWD

Sombras

La propiedad `text-shadow` se aplica a elementos que contienen texto. Permite crear una o más sombras detrás del texto.

```
text-shadow: [offset-x] [offset-y] [blur-radius] [color];
```

Elementos del RWD

Sombras

Parámetro	Descripción	¿En box-shadow?	¿En text-shadow?
`offset-x`	Desplazamiento horizontal de la sombra (positivo: derecha, negativo: izquierda).	Sí	Sí
`offset-y`	Desplazamiento vertical de la sombra (positivo: abajo, negativo: arriba).	Sí	Sí
`blur-radius`	Grado de difuminado de la sombra (mayor valor más difuso).	Sí	Sí
`spread-radius`	Expansión o contracción de la sombra (positivo: crece, negativo: encoge).	Sí	No
`color`	Color de la sombra.	Sí	Sí
`inset`	Convierte la sombra de externa a interna.	Sí	No

Elementos del RWD

Bordes

La propiedad `border-radius` permite suavizar las esquinas de cualquier elemento HTML, transformándolas de ángulos rectos a curvas.

```css
/* Todas las esquinas tendrán un radio de 10 píxeles */
border-radius: 10px;

 /*
   Superior Izquierda: 10px
   Superior Derecha: 20px
   Inferior Derecha: 30px
   Inferior Izquierda: 40px
 */
border-radius: 10px 20px 30px 40px;
```

border-radius - CSS | MDN

Elementos del RWD

Bordes

Si se necesita un control aún más preciso, es posible aplicar `border-radius` a cada esquina de forma individual:

- `border-top-left-radius`
- `border-top-right-radius`
- `border-bottom-right-radius`
- `border-bottom-left-radius`

Elementos del RWD

Gradientes

Los gradientes permiten mezclar dos o más colores de forma suave y continua en un elemento, en lugar de usar un solo color plano. Esto añade profundidad sin la necesidad de usar imágenes.

Using CSS gradients - CSS | MDN

Elementos del RWD

Gradientes

Un gradiente lineal crea una transición de color a lo largo de una línea recta.

```
/*
Palabras clave de dirección: to top, to right, to bottom, to left.
Se puede combinar, por ejemplo, to top right (hacia la esquina superior derecha).

Ángulo: Se especifica en grados (deg). 0deg es hacia arriba, 90deg es hacia la derecha.
Por defecto, el gradiente va de arriba a abajo (to bottom).

*/
background-image: linear-gradient([direccion o angulo], [color1], [color2], ...);
```

Elementos del RWD

Gradientes

Un gradiente radial crea una transición de color que se irradia desde un punto central.

```
/*
Forma (opcional): Puede ser circle o ellipse (valor por default).

Tamanio (opcional): Define el tamaño del gradiente.

    * closest-corner, farthest-corner (por default): Basado en la distancia a la esquina
    más cercana o más lejana.

    * closest-side, farthest-side: Basado en la distancia al lado más cercano o más lejano.

Posición (opcional): Indica el centro del gradiente. Se especifica con palabras clave como
center (por defecto), top, bottom, left, right, o con coordenadas (20% 30%).

*/
background-image: radial-gradient([forma] [tamanio] at [posicion], [color1], [color2])
```

Elementos del RWD

Animaciones

Las animaciones permiten hacer que los elementos de una página web cambien de estilo suavemente a lo largo del tiempo, creando efectos visuales. Para crear una animación, se necesitan dos componentes principales:

- @keyframes rules: Definen las diferentes etapas de la animación.
- Propiedades de animación: Se aplican a los elementos HTML para controlar cómo se reproduce la animación definida por @keyframes.

Using CSS animations - CSS | MDN

Elementos del RWD

Animaciones

Las reglas `@keyframes` es donde se especifica cómo el estilo de un elemento cambia en puntos específicos durante la duración de la animación. Se definen usando un nombre de animación para referenciarse y porcentajes que representan el progreso de la animación.

```css
@keyframes nombre-de-mi-animacion {
    0% {
      /* Estilos al inicio de la animación */
      opacity: 0;
    }
    50% {
      /* Estilos a la mitad de la animación */
      opacity: 0.5;
    }
    100% {
      /* Estilos al final de la animación */
      opacity: 1;
    }
  }
```

Elementos del RWD

Animaciones

Una vez que se ha definido una regla `@keyframes`, se aplican las propiedades de animación a un elemento HTML para controlar cómo se reproduce esa animación.

`animation-name`	Especifica el nombre de la regla @keyframes que se quiere aplicar al elemento. Es obligatoria.
`animation-duration`	Define cuánto tiempo tarda una animación en completarse un ciclo.
`animation-timing-function`	Controla la velocidad de la animación a lo largo de su duración (cómo se acelera o desacelera).
`animation-delay`	Especifica un retraso antes de que la animación comience a reproducirse.
`animation-iteration-count`	Define cuántas veces debe repetirse la animación.
`animation-direction`	Especifica los estilos que se aplican al elemento antes o después de que la animación se ejecute.

Elementos del RWD

Animaciones

Una vez que se ha definido una regla `@keyframes`, se aplican las propiedades de animación a un elemento HTML para controlar cómo se reproduce esa animación.

```
/*
    nombre: slideIn
    duración: 2s
    función de tiempo: ease-out
    retraso: 0.5s
    conteo de iteraciones: infinite
    dirección: alternate
    modo de relleno: forwards
*/
.elemento-animado {
    animation: slideIn 2s ease-out 0.5s infinite alternate forwards;
  }
```

Elementos del RWD

Animaciones

Las funciones de transformación en CSS permiten aplicar cambios geométricos a los elementos HTML, modificando su posición, tamaño, orientación o forma dentro del espacio del documento.

Función	Descripción	Sintaxis	
`translate()`	Mueve en ambos ejes (X, Y).	`translate(50px, 20px)`	Mueve 50px a la derecha y 20px hacia abajo.
`scale()`	Escala ancho y alto a la vez.	`scale(2, 0.5)`	Duplica el ancho y reduce a la mitad la altura.
`rotate()`	Rota el elemento en 2D.	`rotate(45deg)`	Rota 45 grados en sentido horario.
`skew()`	Inclina ambos ejes.	`skew(20deg, 10deg)`	Deforma en X y Y.

transform - CSS | MDN

Elementos del RWD

Animaciones

Las transiciones permiten animar de forma gradual los cambios en los estilos de un elemento en lugar de que un cambio de estilo ocurra de inmediato.

```
transition: [property] [duration] [timing-function] [delay];

transition: background-color 0.3s ease-in-out 0.1s;
```

Elementos del RWD

Animaciones

Las timing functions (funciones de temporización) determinan cómo cambia la velocidad de una animación o transición a lo largo del tiempo.

Función	Descripción
linear	Velocidad constante (sin aceleración)
ease	Comienza lento, acelera, y termina lento
ease-in	Comienza lento, luego acelera
ease-out	Comienza rápido y desacelera
ease-in-out	Lento al inicio y al final, rápido en medio
steps(n)	Divide el tiempo en n pasos bruscos

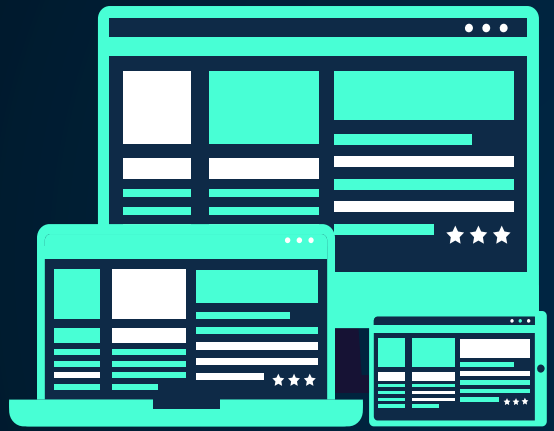
animation-timing-function - CSS | MDN

Elementos del RWD

Animaciones

Un “evento” es una condición que se activa por la interacción del usuario, que cambia el estado del elemento y aplica estilos diferentes.

	¿Cuándo se activa?
:hover	Cuando el puntero está sobre el elemento
:active	Mientras se hace clic (presionado, antes de soltar)
:focus	Cuando el elemento recibe foco (como un input seleccionado)
:visited	Enlaces que ya fueron visitados
:link	Enlaces que aún no han sido visitados
:checked	Cuando un checkbox o radio está marcado
:disabled	Cuando un input o botón está deshabilitado
:enabled	Cuando un input o botón está habilitado
:target	Cuando el elemento es el destino de un enlace con #id



Módulo 3

Box model

Box model

Unidades de medida

Las medidas se emplean para definir la altura, anchura y márgenes de los elementos y para establecer el tamaño de letra del texto. Todas las medidas se indican como un valor numérico entero o decimal seguido de una unidad de medida

Box model

Unidades de medida absolutas

Las unidades de medida absolutas tienen un valor fijo y no cambian según el entorno (pantalla, zoom, etc.). Se usan cuando quieres un tamaño constante.

CSS values and units - Learn web development | MDN

Box model

Unidades de medida absolutas

Unidad	Descripción	Equivalencia
px	pixeles	1 px = 1 punto en pantalla
cm	centímetros	1 cm = 37.8 px
mm	milímetros	1 mm = 3.78 px
in	pulgadas	1 in = 96 px = 2.54 cm
pt	puntos	1 pt = 1/72 in = 1.33 px
pc	picas	1 pc = 12 pt = 16 px

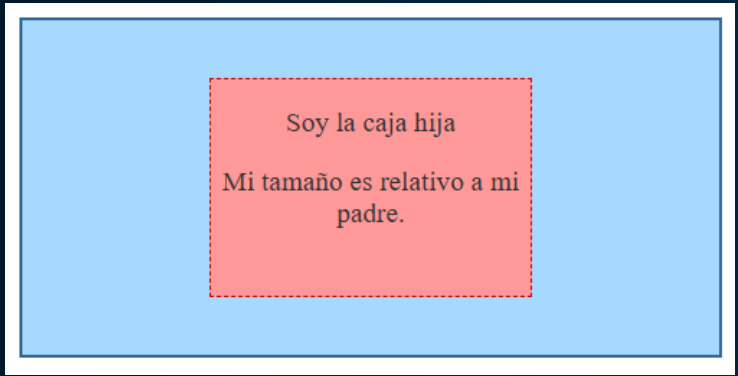
Box model

Unidades de medida relativas

Las unidades de medida relativas cambian dependiendo del contexto, como el tamaño de la fuente de un contenedor padre o del mismo tamaño del contenedor.

Box model

Unidades de medida relativas



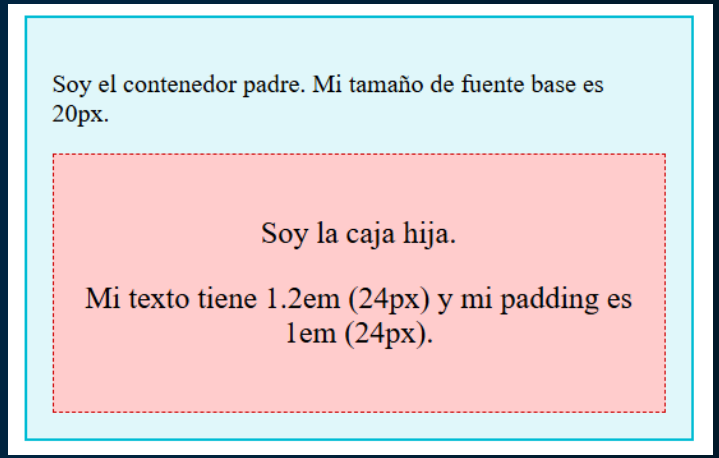
La unidad de porcentaje (%) es una de las unidades relativas más comunes y versátiles. Su valor se calcula en relación con el tamaño de su elemento padre o, en algunos casos, con el tamaño de la ventana gráfica (viewport).

Box model

Unidades de medida relativas

El valor de `em` se calcula en relación con el `font-size` del elemento padre.

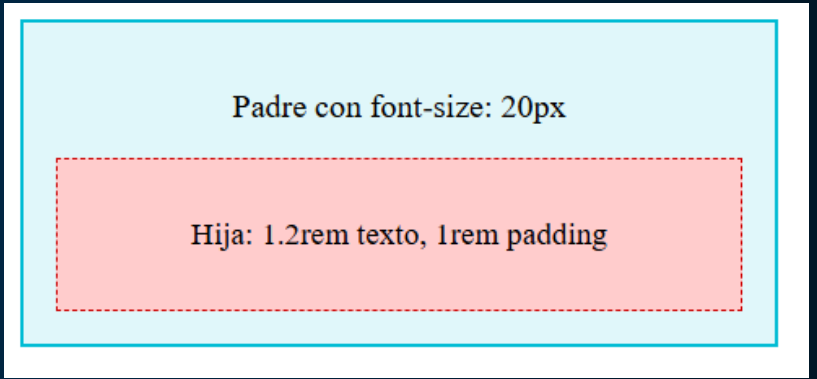
Por ejemplo, si el padre tiene un `font-size` de 16px y un elemento hijo tiene `font-size` de 1.5em, su tamaño de fuente será de 24px (1.5 * 16px).



Box model

Unidades de medida relativas

La unidad `rem` es similar a `em`, pero con una diferencia de que `rem` siempre se basa en el tamaño de fuente del elemento raíz del documento (el elemento <html>). Esto evita la acumulación de `em` en elementos anidados.



Box model

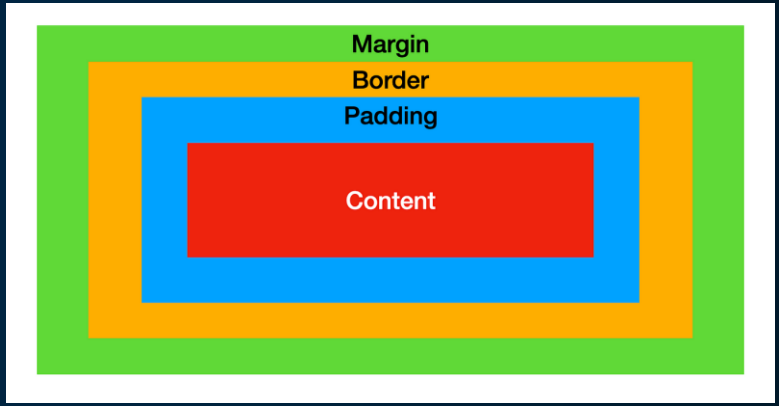
Unidades de medida relativas

Unidad	Relativa a...
%	el tamaño del contenedor padre
em	el tamaño de fuente del elemento padre
rem	el tamaño de fuente del html

Box model

El Box Model (Modelo de Caja) es la forma en que el navegador representa y calcula el tamaño de cada elemento HTML en una página web.

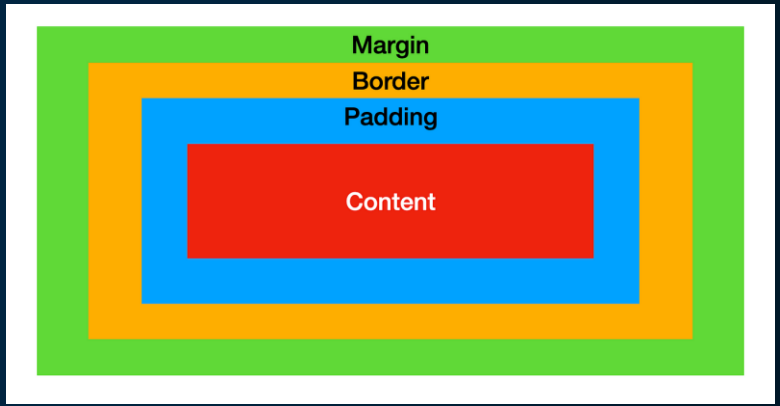
Cada elemento se comporta como si fuera una caja rectangular compuesta por varias capas



The box model - Learn web development | MDN

Box model

- margin: espacio externo entre elementos.
- padding: espacio interno entre el borde y el contenido.
- border: línea que rodea al padding.
- width y height: anchura y altura tamaño del contenido



Box model

La propiedad display determina cómo se comporta un elemento HTML en el flujo del documento. Afecta:

- Si el elemento empieza en una nueva línea.
- Si puede contener otros elementos.
- Cómo se alinean o distribuyen sus elementos hijos.

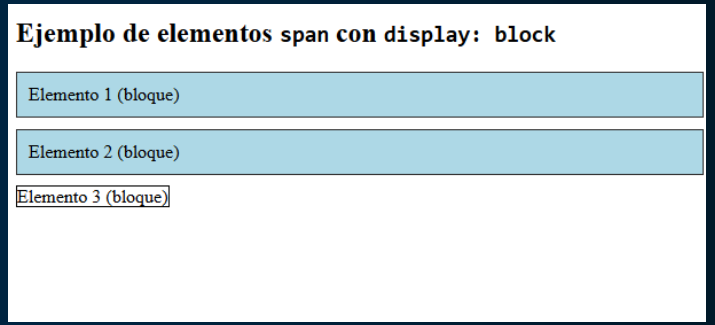
display - CSS | MDN

Box model

display: block

Características:

- Ocupa todo el ancho disponible.
- Empieza en una nueva línea.
- Se puede modificar con propiedades `width, height, margin, padding.`

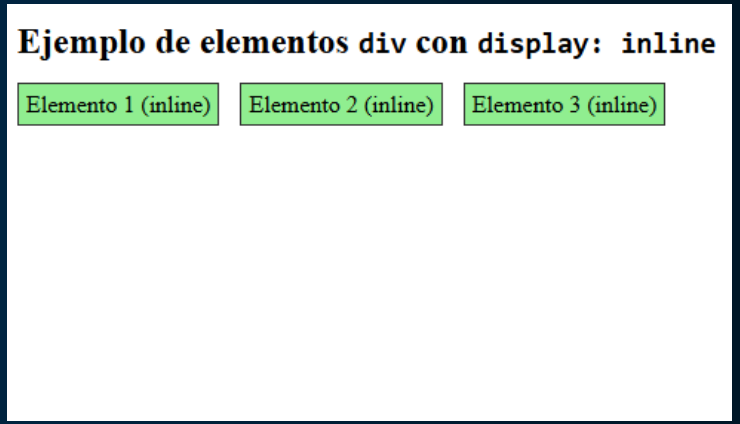


Box model

display: inline

Características:

- No inicia una nueva línea.
- Solo ocupa el espacio necesario para su contenido.
- No es posible modificar **`width, height`**. Sólo funcionan los márgenes horizontales.

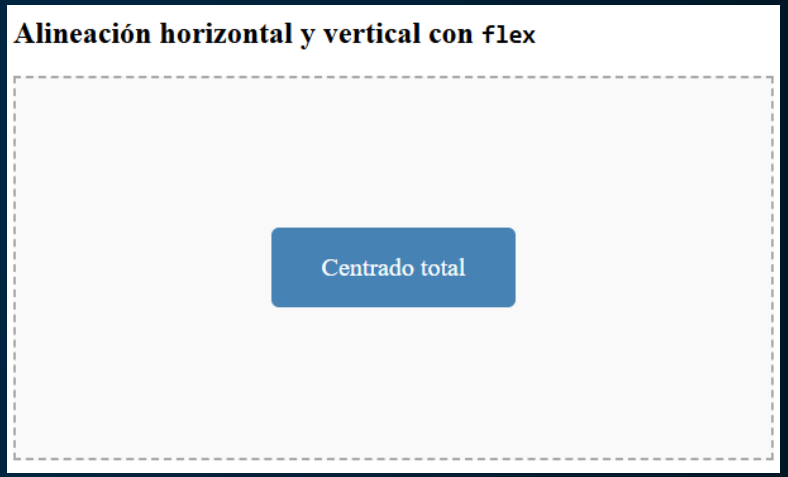


Box model

display: flex

Características:

- Convierte al elemento en un contenedor flexible.
- Sus hijos se alinean horizontal o verticalmente de forma dinámica y adaptable.
- Es posible distribuir, alinear y espaciar elementos con propiedades como `justify-content`, `align-items`.

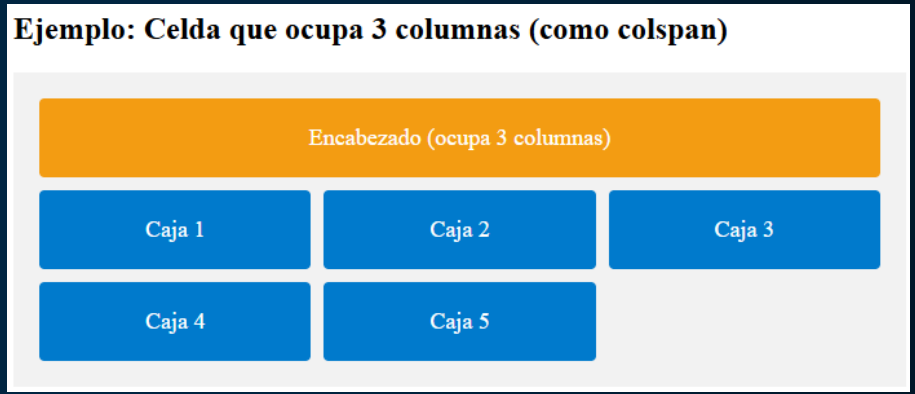


Box model

display: grid

Características:

- Convierte al elemento en un contenedor de tipo grid o filas y columas.
- Es posible definir filas y columnas para alinear los hijos en un diseño tipo tabla.



Basic concepts of grid layout - CSS | MDN

Box model

Fracciones

¿Qué es fr en CSS Grid?

fr significa "fraction" y es una unidad exclusiva de CSS Grid que representa una porción del espacio disponible en el contenedor.

Box model

Fracciones

¿Cómo funciona fr?

El valor **`1fr`** indica que el elemento debe ocupar una parte proporcional del espacio restante después de calcular márgenes, paddings y elementos de tamaño fijo.

Se puede usar múltiples fracciones para repartir el espacio.

```
grid-template-columns: 1fr 2fr 1fr;
```

Esto significa:

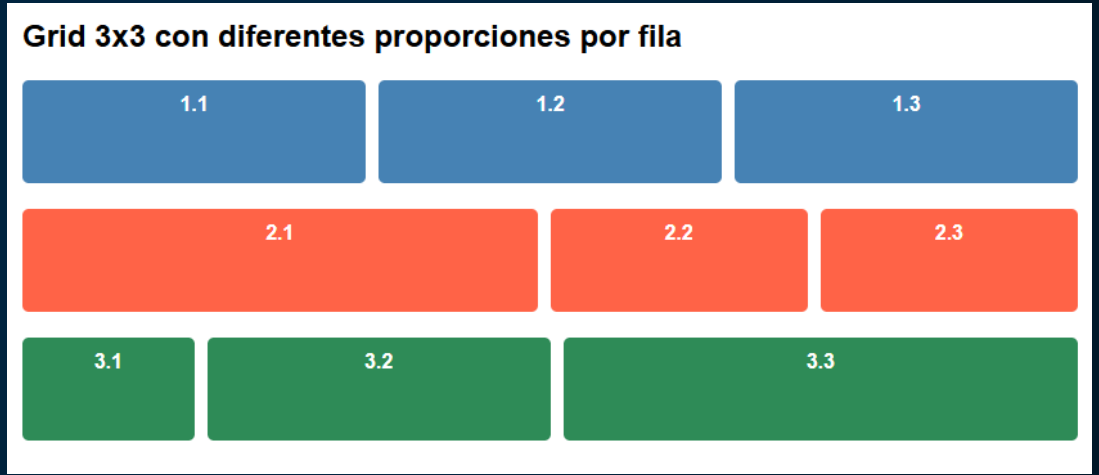
- La primera columna ocupará 1 parte.
- La segunda columna 2 partes
- La tercera columna 1 parte

`Total: 4 partes: proporciones de 25%, 50%, 25%`

Box model

Fracciones

¿Cómo podemos crear este grid usando diferentes valor de fr?



Box model

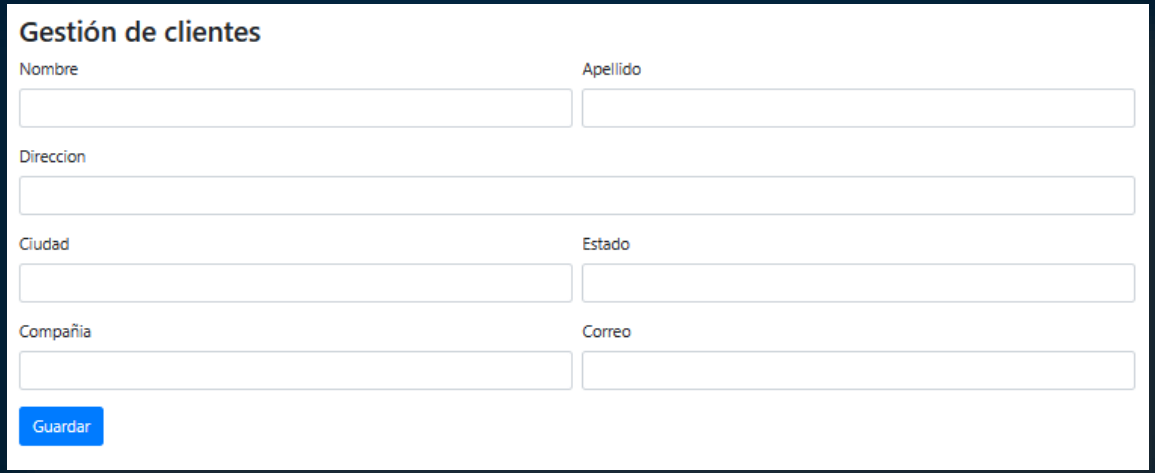
Fracciones

¿Es posible realizar este layout con display?

Box model

Fracciones

¿Es posible realizar este form con display?



Box model

Media queries

Las media queries permiten aplicar estilos condicionales según el tamaño u orientación del dispositivo del usuario.

Se usan para adaptar el diseño a distintos tamaños de pantalla y permiten cambiar propiedades como el tamaño del texto, el layout, los márgenes, etc.

```
@media (condición) {
    /* Reglas CSS que se aplican si
    se cumple la condición */
}


/* Cuando el ancho de la pantalla sea menor
o igual a 768px, cambia el color de fondo */
@media (max-width: 768px) {
    body {
      background-color: lightgray;
    }
}
```

Using media queries - CSS | MDN

Box model

Media queries

Propiedad en Media Query	¿Para qué sirve?
`max-width`	Aplica estilos cuando el ancho del viewport es igual o menor al valor.
`min-width`	Aplica estilos cuando el ancho del viewport es igual o mayor al valor.
`orientation`	Aplica estilos basándose en la orientación del viewport: portrait (vertical) o landscape (horizontal).
`hover`	Detecta si el dispositivo de entrada principal permite la funcionalidad de "hover" (pasar el cursor por encima).

```css
@media (min-width: 768px) and (orientation: landscape) {
    .sidebar {
        display: block;
    }
}
```

Box model

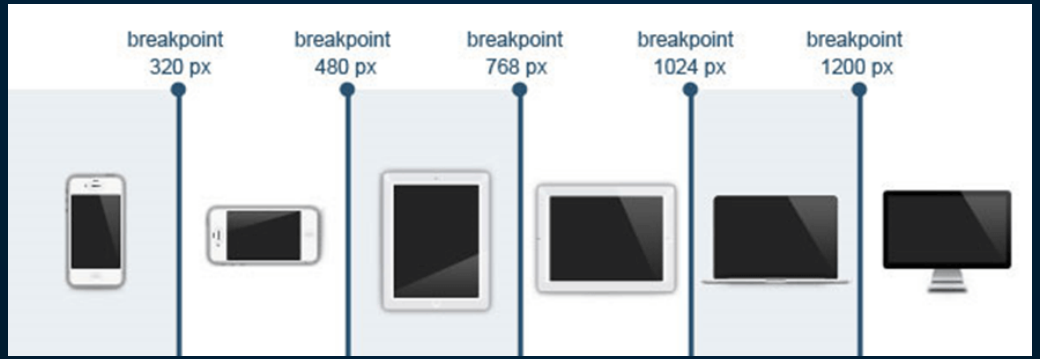
Puntos de quiebre

Los puntos de ruptura (breakpoints) son anchos específicos donde el diseño debe adaptarse. No son reglas fijas, pero sí hay tamaños populares basados en los dispositivos más usados.

Box model

Puntos de quiebre

Dispositivo	Ancho común (px)
Móviles pequeños	320–480
Teléfonos en general	480–767
Tablets/Laptops	768–1024
Laptops/PCs medianos	1025–1279
Pantallas grandes	1280+



Box model

Medios

Los medios (imágenes, videos, iframes, etc.) deben escalar automáticamente según el tamaño del contenedor o del dispositivo. Esto garantiza que no se desborden ni pierdan proporciones.

Box model

Medios - Imágenes

La forma más común y sencilla de hacer que una imagen sea responsiva es usar la propiedades CSS max-width, height y display.

```css
img {
    /* la imagen no excede el ancho de su contenedor */
    max-width: 100%;
    /* mantiene la proporcion */
    height: auto;
    /* evita espacios no deseados debajo de la imagen */
    display: block;
  }
```

Box model

Medios - Videos

Los videos embebidos, como los de YouTube, no son responsivos por defecto. Para hacerlos adaptables, se necesita un contenedor con estilos especiales.

```html
<div class="video-container">
    <iframe
        src="https://www.youtube.com/embed/VIDEO_ID"
        frameborder="0"
        allowfullscreen>
    </iframe>
</div>
```

```css
.video-container {
    position: relative;
    padding-bottom: 56.25%; /* Proporción 16:9 */
    height: 0;
    overflow: hidden;
    max-width: 100%;
  }

  .video-container iframe {
    position: absolute;
    top: 0;
    left: 0;
    width: 100%;
    height: 100%;
  }
```

Módulo 4

Bootstrap

Bootstrap

Bootstrap es un framework de código abierto desarrollado que facilita el desarrollo de sitios y aplicaciones web responsivos y consistentes en navegadores y dispositivos.

Bootstrap

Características

Bootstrap ofrece una colección de herramientas ya listas, como:

- Sistema de rejilla (grid) para crear layouts responsivos.
- Componentes preconstruidos como botones, formularios, alertas, cards, sliders, barras de navegación, etc.
- Utilidades de CSS para espaciado, colores, tipografía y visibilidad.
- Soporte para Jquery y JavaScript para agregar comportamiento dinámico.

Bootstrap

Bootstrap 4 vs 5

1. Eliminación de JQuery

- Bootstrap 4: Depende de jQuery para funcionalidades JavaScript
- Bootstrap 5: Elimina jQuery, usando solo JavaScript puro (Vanilla JS)

2. Sistema de rejilla (Grid)

- Bootstrap 4: 5 breakpoints (xs, sm, md, lg, xl).
- Bootstrap 5: Agrega un sexto breakpoint xxl (≥1400px).

Bootstrap

Plantilla

Una plantilla en Bootstrap es un conjunto de archivos HTML, CSS y a veces JavaScript que ya están maquetados y listos para personalizar y reutilizar. Y crear de manera rápida una página web o parte de ella.

Bootstrap

Sistema de rejilla

Es un sistema de diseño basado en filas y columnas que permite crear layouts que se adaptan a diferentes tamaños de pantalla.

- Contenedor (.container o .container-fluid)
- Fila (.row)
- Columnas (.col)

```
<div class="container">
    <div class="row">
      <div class="col-4">Columna 1 (4 columnas)</div>
      <div class="col-8">Columna 2 (8 columnas)</div>
    </div>
</div>
```

Grid system · Bootstrap v5.3

Bootstrap

Sistema de rejilla

Cuántas columnas caben en el grid de Bootstrap?

Bootstrap divide cada fila (.row) en 12 columnas virtuales. Eso significa que dentro de una fila, la suma de las columnas debe ser 12 o menos para que estén en la misma línea.

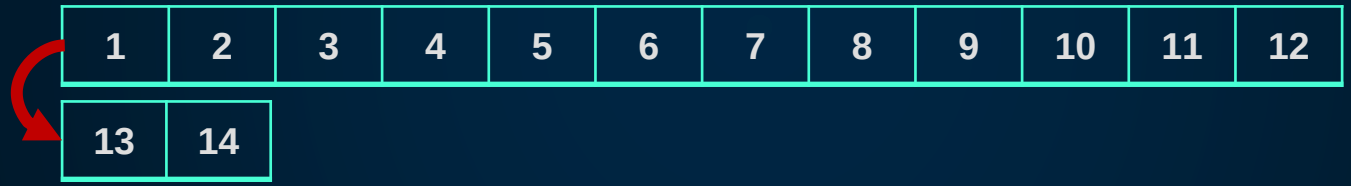
1	2	3	4	5	6	7	8	9	10	11	12

Columns · Bootstrap v5.3

Sistema de rejilla

¿Qué pasa si hay más de 12 columnas?

Si la suma de las columnas en una fila supera las 12 Bootstrap envuelve automáticamente las columnas excedentes en una nueva línea. Es decir, las columnas sobrantes bajan a una nueva fila visual, aunque estén en la misma fila en el HTML.




```html
<div class="row">
    <div class="col-6">Col 1</div>
    <div class="col-6">Col 2</div>
    <!-- Esta se va a la siguiente línea -->
    <div class="col-4">Col 3</div>
</div>
```

Bootstrap

Sistema de rejilla

¿Qué clases se usan para definir columnas?

1. Clase automática (.col): Divide el espacio restante en partes iguales.

```
<div class="row">
    <div class="col">Col A</div>
    <div class="col">Col B</div>
</div>
```

Bootstrap

Sistema de rejilla

¿Qué clases se usan para definir columnas?

2. Clases con número (.col-{n}): Especifican cuántas columnas (de 12) debe ocupar el elemento html.

```html
<div class="row">
    <div class="col-4">Col 1 (4 de 12 columnas)</div>
    <div class="col-8">Col 2 (8 de 12 columnas)</div>
</div>
```

¿Qué clases se usan para definir columnas?

3. Clases responsivas (.col-{breakpoint}-{n}): Permiten cambiar la distribución dependiendo del ancho de la pantalla:

```
<div class="row">
    <div class="col-12 col-md-6">Columna</div>
</div>
```

Bootstrap

Contenedores

Un contenedor es un elemento que se utiliza para centrar y alinear el contenido de forma responsiva. Sirve como envoltorio y como punto de partida para el sistema de grid (.row y .col) de Bootstrap.

```
<div class="container">
    <div class="row">
        <div class="col">Contenido</div>
    </div>
</div>
```

Bootstrap

Contenedores

Clase	Uso
`.container`	Fijo y responsivo ya que cambia su ancho máximo según el tamaño de pantalla. Diseño centrado y con márgenes, ideal para la mayoría de sitios.
`.container-fluid`	Siempre ocupa el 100% del ancho disponible, sin importar el tamaño del dispositivo. Útil cuando se quiere aprovechar todo el ancho de la pantalla.
`.container-{breakpoint}`	Funciona como .container-fluid hasta cierto tamaño, y después se comporta como .container (fijo). Se usa para tener más control por tamaño de pantalla.

Bootstrap

Contenedores

Los breakpoints (o puntos de interrupción) son los tamaños de pantalla en los que el diseño web cambia para adaptarse a diferentes dispositivos. Bootstrap usa estos puntos de interrupción para aplicar estilos responsivos, es decir, que se ajusten automáticamente al tamaño del dispositivo del usuario.

Tamaño	Clase base	Ancho mínimo	Dispositivos comunes
Pequeño	sm	576px	Teléfonos grandes
Mediano	md	768px	Tablets
Grande	lg	992px	Laptops
Extra grande	xl	1200px	Desktops grandes
Extra extra grande	xxl	1400px	Pantallas muy grandes

Breakpoints · Bootstrap v5.3

Módulo 5

JQuery

JQuery

jQuery es una biblioteca de JavaScript que simplifica la manipulación del DOM, el manejo de eventos, las animaciones y las peticiones AJAX aunque JavaScript moderno ha incorporado muchas de las capacidades que antes sólo jQuery proporcionaba.

jQuery API Documentation

JQuery

jQuery aún se sigue usando en:

- Sistemas heredados o mantenidos por años.
- Proyectos donde se integran plugins jQuery (como datepickers, datatables, etc.)
- Proyectos donde se busca compatibilidad con navegadores antiguos.
- Equipos de trabajo que prefieren una sintaxis más sencilla y concisa.

JQuery

Filosofia

Cuando jQuery nació en 2006, el DOM de los navegadores no era fácil de manipular. Había que escribir mucho código y además variaba de un navegador a otro. Por eso, jQuery introdujo una forma basada en los selectores de CSS para hacer “consultas” al DOM.

```
// Seleccionar todos los <p> dentro de un div
// con clase .contenedor
$(".contenedor p").addClass("resaltado");
```

JQuery

Filosofia

- Tratar el DOM como una base de datos de nodos, es decir, hacer queries para filtrar, mapear y modificar resultados.
- Está pensado como una colección de elementos sobre la que puedes encadenar.
- No importa cómo lo hace el navegador por debajo, solo se “piden” datos y “aplican” cambios.

JQuery

Filosofia

Selección del DOM

Permite seleccionar elementos HTML usando sintaxis similar a CSS. Es el núcleo de jQuery y lo que le dio su popularidad.

$() es un alias de la función jQuery() y es la función principal o global en jQuery que se utiliza para seleccionar elementos del DOM

```
// Seleccionar todos los <p> dentro de un div
// con clase .contenedor
$(".contenedor p").addClass("resaltado");
```

JQuery

Filosofia

Manipulacion del DOM	Permite modificar el contenido, atributos y estructura de los elementos HTML seleccionados.	• html() • text()
Eventos	Maneja eventos como clics, envíos de formularios, teclas presionadas, etc.	• click() • submit()
Efectos y animaciones	Permite mostrar, ocultar o animar elementos sin necesidad de usar CSS o JS	• hide() • show()
AJAX	Facilita llamadas HTTP asincrónicas para cargar datos sin recargar la página.	• ajax() • get()
Utilidades	Funciones auxiliares para operaciones comunes con arrays, objetos, tipos de datos, etc	• each • map
Manipulacion del CSS	Permite leer y escribir propiedades CSS directamente desde JavaScript.	• css() • offset()

¡Gracias!