



HOJAS DE ESTILO

Cascade Style Sheets (CSS)

Ejercicios Resueltos

Botones.....	2
B1.- Crea botones de tipo alerta.....	2
B2.- Crea botones de tipo outline.....	3
B3.- Crea un paginado.....	4
B4.- Botón para mostrar/ocultar contraseña.....	5
Tablas.....	7
T1.- Tabla con estilo cebra.....	7
T2.- Tabla responsive con scroll horizontal.....	9
T3.- Tabla responsive convertida a contenedores en dispositivos pequeños.....	11
T4.- Tabla con cabecera fija.....	15
Barras de progreso.....	17
BP1.- Barra de progreso con porcentaje (sin javascript).....	17
Sombras.....	18
S1.- Aplicación de sombra exterior.....	18
S2.- Aplicación de sombra interior.....	19
Migas de pan (breadcrumbs).....	19
Transiciones.....	20
TSN1.- Probando las transiciones.....	20
TSN2.- Menú con persiana de color.....	24
TSN3.- Menú con efecto zoom-in.....	27
Transformaciones.....	28
TFN1.- Rota, deforma y estira.....	28
TFN2.- Rotar botón.....	29
TFN3.- Rotar imagen con información en anverso.....	30
TFN4.- Caída de imagen hasta final de pantalla.....	32
TFN5.- Rebote de pelota.....	33
TFN6.- En una galaxia lejana.....	36
TFN7.- Botón que se descuelga y cae.....	41
Flex.....	44
F1.- Contenedores en línea.....	44
F2.- Web con barra de navegación, área de contenido y pie.....	46
F3.- Encabezado con texto o logo y menú.....	48
F4.- Encabezado con texto arriba y menú debajo.....	51
F5.- Menú de navegación de ancho completo y que crezca.....	53



Botones

B1.- Crea botones de tipo alerta

Crea los siguientes botones de tipo alerta:



SOLUCIÓN:

Código HTML

```
<button class="btn primary">Primary</button>
<button class="btn secondary">Secondary</button>
<button class="btn success">Success</button>
<button class="btn warning">Warning</button>
<button class="btn danger">Danger</button>
```

Código CSS

```
.btn {
  border: none;
  color: white;
  padding: 14px 28px;
  cursor: pointer;
  border-radius: 5px;
}

.primary {background-color: #007bff;}
.primary:hover {background: #0b7dda;}

.secondary {background-color: #e7e7e7; color: black;}
.secondary:hover {background: #ddd;}

.success {background-color: #04AA6D;}
.success:hover {background-color: #46a049;}

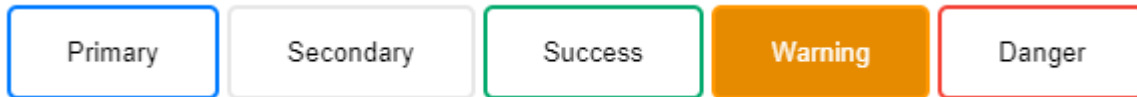
.warning {background-color: #ff9800;}
.warning:hover {background: #e68a00;}

.danger {background-color: #f44336;}
.danger:hover {background: #da190b;}
```



B2.- Crea botones de tipo outline

Crear botones de tipo outline, del mismo estilo que los botones de Bootstrap. Es decir que cuando pases el ratón por encima se rellena del color del borde, como ejemplo véase el Warning:



SOLUCIÓN:

Código HTML

```
<button class="btn primary">Primary</button>
<button class="btn secondary">Secondary</button>
<button class="btn success">Success</button>
<button class="btn warning">Warning</button>
<button class="btn danger">Danger</button>
```

Código CSS

```
.btn {
  border: none;
  color: black;
  padding: 14px 28px;
  cursor: pointer;
  border-radius: 5px;
}

.primary {background-color: white; border: 2px solid #007bff;}
.primary:hover {background: #0b7dda; color: white;}

.secondary {background-color: white; border: 2px solid #e7e7e7;}
.secondary:hover {background: #ddd; color: white;}

.success {background-color: white; border: 2px solid #04AA6D;}
.success:hover {background-color: #46a049; color: white;}

.warning {background-color: white; border: 2px solid #ff9800;}
.warning:hover {background: #e68a00; color: white;}

.danger {background-color: white; border: 2px solid #f44336;}
.danger:hover {background: #da190b; color: white;}
```



B3.- Crea un paginado

Ejemplo básico de cómo crear un paginado simple utilizando CSS



SOLUCIÓN:

Código HTML

```
<ul class="pagination">
  <li><a href="#">«</a></li>
  <li><a href="#">1</a></li>
  <li><a href="#">2</a></li>
  <li><a class="active" href="#">3</a></li>
  <li><a href="#">4</a></li>
  <li><a href="#">5</a></li>
  <li><a href="#">6</a></li>
  <li><a href="#">7</a></li>
  <li><a href="#">»</a></li>
</ul>
```

Código CSS

```
ul.pagination {
  display: inline-block;
  padding: 0;
  margin: 0;
}

ul.pagination li {display: inline;}

ul.pagination li a {
  color: black;
  float: left;
  padding: 9px 18px;
  text-decoration: none;
}

ul.pagination li a.active {
  background-color: #4CAF50;
  color: white;
}
```



```
ul.pagination li a.active:hover {background-color: #397f3b;}  
ul.pagination li a:hover:not(.active) {background-color: #ddd;}
```

B4.- Botón para mostrar/ocultar contraseña

Agregar un botón para mostrar u ocultar la contraseña en un campo de texto de forma sencilla utilizando CSS y JavaScript.

Se trata de una funcionalidad muy buena que no puede faltar en tu página web ya que en dispositivos pequeños, sobretodo, va a ayudar mucho al usuario cuando se está equivocando al teclear su contraseña.

Escribo contraseña:

Contraseña

MOSTRAR

Pulso "MOSTRAR":

Contraseña

OCULTAR

¡Hazlo por ti mismo! Si tienes duda puedes consultar la solución.

SOLUCIÓN:

Código HTML

```
<div class="campo">  
  <label for="password">Contraseña</label>  
  <input type="password" name="password" id="password">  
  <span>Mostrar</span>  
</div>
```

Código CSS

```
.campo {  
  position: relative;  
}  
  
label {  
  width: 100%;  
}
```



```
input {  
  width: 100%;  
  padding: 7px;  
}  
  
.campo span {  
  position: absolute;  
  right: 5px;  
  top: 39px;  
  text-transform: uppercase;  
  cursor: pointer;  
  padding: 2px 10px;  
  background-color: #dadada;  
}
```

Código JavaScript

```
document.querySelector('.campo span').addEventListener('click', e => {  
  const passwordInput = document.querySelector('#password');  
  if (e.target.classList.contains('show')) {  
    e.target.classList.remove('show');  
    e.target.textContent = 'Ocultar';  
    passwordInput.type = 'text';  
  } else {  
    e.target.classList.add('show');  
    e.target.textContent = 'Mostrar';  
    passwordInput.type = 'password';  
  }  
});
```



Tablas

T1.- Tabla con estilo cebra

Se desea dar el siguiente formato cebra a la tabla:

Nombre	Apellidos	Edad	Nómina
Martin	Gonzalez	56	2.560€
Martin	Gonzalez	56	2.560€
Martin	Gonzalez	56	2.560€
Total			7.680 €

Donde al elemento `<table>` se le quita el espaciado de las celdas, establecer un borde fino, le doy un ancho y lo centro en el contenedor.

A las celdas en sí, le doy un borde sólido de color gris suave y le coloco un espaciado interior de 5px.

Con `tr:nth-child(even)` puedes iluminar las filas que sean pares.

Con `td:nth-child(n + 3)` estoy seleccionando aquellas columnas a partir de la tercera para, en este caso, centrar sus valores.

Con `tbody tr:hover` estoy dándole un color de fondo diferente cuando pase el ratón por encima.

Y, por último, le doy estilo a la cabecera para ponerle un fondo negro y el texto de color blanco.

SOLUCIÓN:

Código HTML.

```
<table>
  <thead>
    <tr>
      <th>Nombre</th>
      <th>Apellidos</th>
      <th>Edad</th>
      <th>Nómina</th>
    </tr>
```



```
</thead>
<tbody>
  <tr>
    <td>Martin</td>
    <td>Gonzalez</td>
    <td>56</td>
    <td>2.560€</td>
  </tr>
  <tr>
    <td>Martin</td>
    <td>Gonzalez</td>
    <td>56</td>
    <td>2.560€</td>
  </tr>
  <tr>
    <td>Martin</td>
    <td>Gonzalez</td>
    <td>56</td>
    <td>2.560€</td>
  </tr>
  <tr>
    <td>Martin</td>
    <td>Gonzalez</td>
    <td>56</td>
    <td>2.560€</td>
  </tr>
</tbody>
<tfoot>
  <tr>
    <th colspan="3">Total</th>
    <th class="text-center">7.680 €</th>
  </tr>
</tfoot>
</table>
```

Código CSS.

```
table {
  border-spacing: 0;
  border-collapse: collapse;
  width: 800px;
  margin: 0px auto;
}
```




```
td, th {  
    border: 1px solid #ccc;  
    padding: 5px;  
}  
  
tr:nth-child(even) {  
    background-color: #eee;  
}  
  
td:nth-child(n + 3),  
th:nth-child(n + 3) {  
    text-align: center;  
}  
  
tbody tr:hover {  
    background-color: aquamarine;  
}  
  
thead {  
    background-color: black;  
    color: white;  
}
```

T2.- Tabla responsive con scroll horizontal

La tabla se muestra al mismo tamaño original pero el usuario tiene que desplazar el dedo para visualizar el contenido oculto que hay a la derecha ya que la tabla no se muestra de forma completa.

Visualmente, en dispositivos pequeños el usuario ve una barra de scroll horizontal como la siguiente:

Nombre	Apellidos	Email	Edad	N
Lorem	Ipsum	demo@demo.com	56	2
Lorem	Ipsum	demo@demo.com	56	2
Lorem	Ipsum	demo@demo.com	56	2

¡Hazlo por ti mismo! Si tienes duda puedes consultar la solución.



SOLUCIÓN:

Código HTML.

```
<div class="table-responsive">
  <table>
    <thead>
      <tr>
        <th>Nombre</th>
        <th>Apellidos</th>
        <th>Email</th>
        <th>Edad</th>
        <th>Nómina</th>
        <th>Ingresos</th>
        <th>Gastos</th>
        <th>Beneficios</th>
        <th>Impuestos</th>
      </tr>
    </thead>
    <tbody>
      <tr>
        <td>Lorem</td>
        <td>Ipsum</td>
        <td>demo@demo.com</td>
        <td>56</td>
        <td>2.560€</td>
        <td>2.560€</td>
        <td>1.000€</td>
        <td>1.560€</td>
        <td>390</td>
      </tr>
      <tr>
        <td>Lorem</td>
        <td>Ipsum</td>
        <td>demo@demo.com</td>
        <td>56</td>
        <td>2.560€</td>
        <td>2.560€</td>
        <td>1.000€</td>
        <td>1.560€</td>
        <td>390</td>
      </tr>
      <tr>
        <td>Lorem</td>
```



```
<td>Ipsum</td>
<td>demo@demo.com</td>
<td>56</td>
<td>2.560€</td>
<td>2.560€</td>
<td>1.000€</td>
<td>1.560€</td>
<td>390</td>
</tr>
</tbody>
</table>
</div>
```

Código CSS.

```
.table-responsive {
  display: block;
  width: 100%;
  overflow-x: auto;
}
```

T3.- Tabla responsive convertida a contenedores en dispositivos pequeños

Consiste en disponer de la tabla original en pantallas de escritorio y para dispositivos más pequeños disponer de un contenedor para mostrar los datos de la tabla en 2 columnas: cabecera y datos

Quedaría algo como:

Nombre	Lorem
Apellidos	Ipsum
Email	demo@demo.com
Edad	56
Nómina	2.560€
Ingresos	2.560€
Gastos	1.000€
Beneficios	1.560€
Impuestos	390

Nombre	Lorem
Apellidos	Ipsum
Email	demo@demo.com



¡Hazlo por ti mismo! Si tienes duda puedes consultar la solución.

SOLUCIÓN:

Código HTML.

```
<div class="table">
  <div class="item">
    <div class="row">
      <div class="thead">Nombre</div>
      <div class="tbody">Lorem</div>
    </div>
    <div class="row">
      <div class="thead">Apellidos</div>
      <div class="tbody">Ipsum</div>
    </div>
    <div class="row">
      <div class="thead">Email</div>
      <div class="tbody">demo@demo.com</div>
    </div>
    <div class="row">
      <div class="thead">Edad</div>
      <div class="tbody">56</div>
    </div>
    <div class="row">
      <div class="thead">Nómina</div>
      <div class="tbody">2.560€</div>
    </div>
    <div class="row">
      <div class="thead">Ingresos</div>
      <div class="tbody">2.560€</div>
    </div>
    <div class="row">
      <div class="thead">Gastos</div>
      <div class="tbody">1.000€</div>
    </div>
    <div class="row">
      <div class="thead">Beneficios</div>
      <div class="tbody">1.560€</div>
    </div>
    <div class="row">
      <div class="thead">Impuestos</div>
      <div class="tbody">390</div>
    </div>
  </div>
</div>
```



```
<div class="row">
  <div class="thead">Nombre</div>
  <div class="tbody">Lorem</div>
</div>
<div class="row">
  <div class="thead">Apellidos</div>
  <div class="tbody">Ipsum</div>
</div>
<div class="row">
  <div class="thead">Email</div>
  <div class="tbody">demo@demo.com</div>
</div>
<div class="row">
  <div class="thead">Edad</div>
  <div class="tbody">56</div>
</div>
<div class="row">
  <div class="thead">Nómina</div>
  <div class="tbody">2.560€</div>
</div>
<div class="row">
  <div class="thead">Ingresos</div>
  <div class="tbody">2.560€</div>
</div>
<div class="row">
  <div class="thead">Gastos</div>
  <div class="tbody">1.000€</div>
</div>
<div class="row">
  <div class="thead">Beneficios</div>
  <div class="tbody">1.560€</div>
</div>
<div class="row">
  <div class="thead">Impuestos</div>
  <div class="tbody">390</div>
</div>
</div>
<div class="item">
  <div class="row">
    <div class="thead">Nombre</div>
    <div class="tbody">Lorem</div>
  </div>
  <div class="row">
    <div class="thead">Apellidos</div>
    <div class="tbody">Ipsum</div>
  </div>
</div>
<div class="row">
```



```
<div class="thead">Email</div>
<div class="tbody">demo@demo.com</div>
</div>
<div class="row">
  <div class="thead">Edad</div>
  <div class="tbody">56</div>
</div>
<div class="row">
  <div class="thead">Nómina</div>
  <div class="tbody">2.560€</div>
</div>
<div class="row">
  <div class="thead">Ingresos</div>
  <div class="tbody">2.560€</div>
</div>
<div class="row">
  <div class="thead">Gastos</div>
  <div class="tbody">1.000€</div>
</div>
<div class="row">
  <div class="thead">Beneficios</div>
  <div class="tbody">1.560€</div>
</div>
<div class="row">
  <div class="thead">Impuestos</div>
  <div class="tbody">390</div>
</div>
</div>
</div>
```

Código CSS.

```
div.table {
  display: none;
}

@media screen and (max-width: 920px) {
  table {
    display: none;
  }

  div.table {
    display: block;
    width: 100%;
  }
}
```



```
div.item {  
  margin-bottom: 10px;  
}  
  
div.row {  
  display: flex;  
  justify-content: start;  
  align-items: center;  
  flex-direction: row;  
}  
  
div.tbody, div.thead {  
  border: 1px solid #ccc;  
  padding: 5px;  
  text-align: left;  
  width: 50%;  
}  
  
div.thead {  
  background-color: #eee;  
}  
}
```

T4.- Tabla con cabecera fija

Tabla en la que su cabecera te sigue con el scroll para mantenerla siempre visible.

Para que tu tabla tenga una cabecera fija tan solo tienes que agregar los siguientes estilos a la etiqueta <thead> de tu tabla.

SOLUCIÓN:

Código HTML.

```
<table>  
  <thead>  
    <tr>  
      <th>Nombre</th>  
      <th>Apellidos</th>  
      <th>Edad</th>  
      <th>Nómina</th>  
    </tr>  
  </thead>
```



```
<tbody>
  <tr>
    <td>Lorem</td>
    <td>Ipsum</td>
    <td>56</td>
    <td>2.560€</td>
  </tr>
  <tr>
    <td>Lorem</td>
    <td>Ipsum</td>
    <td>56</td>
    <td>2.560€</td>
  </tr>
  <tr>
    <td>Lorem</td>
    <td>Ipsum</td>
    <td>56</td>
    <td>2.560€</td>
  </tr>
</tbody>
<tfoot>
  <tr>
    <th colspan="3">Total</th>
    <th class="text-center">7.680 €</th>
  </tr>
</tfoot>
</table>
```

Código CSS.

```
thead {
  position: sticky;
  top: 0;
}
```




Barras de progreso

BP1.- Barra de progreso con porcentaje (sin javascript)

Cuando un proceso tarda en dar un resultado es muy importante informar al usuario de ello y no dejar una página en blanco cargando que no dice nada y puede acabar desesperando al usuario en la espera.

Para evitar esto es mostrar algo que indique que la página se está cargando, las barras de progreso pueden ser ideales para estas situaciones.

Crea la siguiente barra de progreso:



¡Hazlo por ti mismo! Si tienes duda puedes consultar la solución.

SOLUCIÓN:

Código HTML.

```
<div class="progress">  
  <div class="progress-bar" style="width:75%;">  
    <span class="progress-bar-text">75%</span>  
  </div>  
</div>
```

Código CSS.

```
.progress {  
  height: 35px;  
  width: 100%;  
  border: 1px solid #428bca;  
  border-radius: 5px;  
  background-color: #e6f3fa;  
  margin-bottom: 15px;  
}  
  
.progress-bar {
```



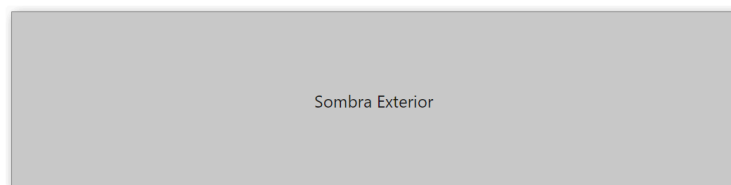
```
height: 100%;  
background: #428bca;  
display: flex;  
align-items: center;  
transition: width 0.25s;  
border-radius: 5px;  
}
```

```
.progress-bar-text {  
margin-left: 10px;  
font-weight: bold;  
color: #cce7f5;  
}
```

Sombras

S1.- Aplicación de sombra exterior

Cómo aplicar sombras a un contenedor, caja o div utilizando la nueva propiedad box-shadow de CSS3.



SOLUCIÓN:

Código HTML.

```
<div class="sombra-exterior">Sombra Exterior</div>
```

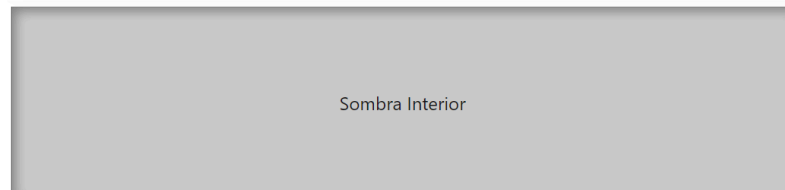
Código CSS.

```
.sombra-exterior {  
box-shadow: 5px 5px 15px gray;  
}
```



S2.- Aplicación de sombra interior

Cómo aplicar sombras a un contenedor, caja o div utilizando la nueva propiedad box-shadow de CSS3.



SOLUCIÓN:

Código HTML.

```
<div class="sombra-interior">Sombra Interior</div>
```

Código CSS.

```
.sombra-interior {  
    box-shadow: 5px 5px 15px gray inset;  
}
```

Migas de pan (breadcrumbs)

Crear la típica ruta navegación de dónde se encuentra el usuario:

Home / Contact / Blog / **Services**

SOLUCIÓN:

Código HTML.

```
<ul class="breadcrumb">  
    <li><a href="#">Home</a></li>  
    <li><a href="#">Contact</a></li>  
    <li><a href="#">Blog</a></li>  
    <li><a class="active" href="#">Services</a></li>  
</ul>
```

Código CSS.



```
ul.breadcrumb {  
    padding: 8px 16px;  
    list-style: none;  
    background-color: #eee;  
}  
ul.breadcrumb li {display: inline;}  
ul.breadcrumb li+li:before {  
    padding: 8px;  
    color: black;  
    content: "\00a0";  
}  
ul.breadcrumb a:link, ul.breadcrumb a:visited {  
    text-decoration:none;  
    color: black;  
}  
  
ul.breadcrumb li a.active {  
    color: #4CAF50;  
    font-weight: bold;  
}  
ul.breadcrumb li a.active:hover {  
    color: #397f3b;  
    font-weight: bold;  
}
```

Transiciones

TSN1.- Probando las transiciones

Crea un contenedor de 180 píxeles de ancho y de alto, incluye un texto en su interior y aplícale un color de fondo. Realiza las siguientes transiciones cuando el usuario pase por encima del contenedor el puntero del ratón:

- Cambia el color de fondo del contenedor.
- Cambia el tamaño de las letras del contenedor.
- Cambia el color de las letras del contenedor.
- Cambia el valor del interlineado.
- Cambia el valor de la propiedad border-radius del contenedor.
- Cambia el valor de la sombra del contenedor.
- Cambia la opacidad del contenedor.



- Cambia tres propiedades más a tu elección.

SOLUCIÓN:

Código HTML.

```
<div class="container">
  <div class="caja caja1">Background</div>
  <div class="caja caja2">Font-size</div>
  <div class="caja caja3">Color</div>
  <div class="caja caja4">Line-height</div>
  <div class="caja caja5">Border-radius</div>
  <div class="caja caja6">Box-shadow</div>
  <div class="caja caja7">Opacity</div>
  <div class="caja caja8">Border</div>
  <div class="caja caja9">Margin</div>
  <div class="caja caja10">Padding</div>
</div>
```

Código CSS.

```
.container{
  position: relative;
}
.caja{
  width: 180px;
  height: 180px;
  background-color: #30B037;
  color: #FFF;
  text-align: center;
  line-height: 180px;
  margin: 15px;
  float: left;
  font-size: 18px;
  font-family: Arial;
}
.caja1{
  background-color: #C0392B;
  -webkit-transition: background-color 1s linear;
  transition: background-color 1s linear;
}
.caja1:hover{
  background-color: #3F51B5;
}
```



```
.caja2{
  background-color: #9B59B6;
  -webkit-transition: font-size 1s ease;
  transition: font-size 1s ease;
}
.caja2:hover{
  font-size: 28px;
}
.caja3{
  background-color: #2980B9;
  -webkit-transition: color 1s linear;
  transition: color 1s linear;
}
.caja3:hover{
  color: yellow;
}
.caja4{
  background-color: #3498DB;
  -webkit-transition: 1s linear;
  transition: 1s linear;
}
.caja4:hover{
  line-height: 300px;
}
.caja5{
  border-radius: 0px;
  background-color: #17A589;
  -webkit-transition: 1s linear;
  transition: 1s linear;
}
.caja5:hover{
  border-radius: 20px;
}
.caja6{
  -webkit-transition: 1s linear;
  transition: 1s linear;
  background-color: #F1C40F;
}
.caja6:hover{
  -webkit-box-shadow: 10px 10px 14px 2px rgba(0,0,0,0.47);
  box-shadow: 10px 10px 14px 2px rgba(0,0,0,0.47);
}
.caja7{
  -webkit-transition: 1s linear;
```



```
    transition: 1s linear;
    background-color: #E67E22;
}
.caja7:hover{
    opacity: 0.3;
}
.caja8{
    -webkit-transition: 1s linear;
    transition: 1s linear;
    background-color: #95A5A6;
}
.caja8:hover{
    border-style: solid;
    border-color: #95A5A6;
    border-color: #000;
}
.caja9{
    -webkit-transition: 1s linear;
    transition: 1s linear;
    background-color: #2C3E50;
}
.caja9:hover{
    margin: 80px;
}
.caja10{
    -webkit-transition: 1s linear;
    transition: 1s linear;
    background-color: black;
}
.caja10:hover{
    padding: 30px;
}
```



TSN2.- Menú con persiana de color

Tenemos un menú sencillo al que se le aplica una transición de desplegar un color como persiana y que termina al pie de la barra de menú. En la imagen podemos ver como el amarillo de “Servicios” empezaba a bajar:



SOLUCIÓN:

Código HTML.

```
<ul>
<li><a class="menu servicios" href="#" data-title="Servicios"><span>
  <span>Servicios</span>
</span></a></li>
<li><a class="menu proyectos" href="#" data-title="Proyectos">
  <span>
    <span>Proyectos</span>
  </span>
</a></li>
<li><a class="menu recursos" href="#" data-title="Recursos">
  <span>
    <span>Recursos</span>
  </span>
</a></li>
<li><a class="menu contacto" href="#" data-title="Contacto">
  <span>
    <span>Contacto</span>
  </span>
</a></li>
</ul>
```

Código CSS.

```
.menu {
  display: block;
  position: relative;
```




```
height: 76px;
width: 120px;
margin: 0;
overflow: hidden;
}

/* Bordes superiores */

.menu.servicios {
border-top: #ffc000 0.3em solid;
}

.menu.recursos {
border-top: #b81800 0.3em solid;
}

.menu.proyectos {
border-top: #00a53c 0.3em solid;
}

.menu.contacto {
border-top: #9600b4 0.3em solid;
}

/* Atributos y transición */

.menu > span {
display: block;
position: absolute;
overflow: hidden;
left: 0;
top: 0;
width: 100%;
height: 0%;
transition: 0.5s ease-in-out;
-webkit-transition: 0.5s ease-in-out;
}

.menu:after, .menu > span > span {
display: block;
text-align: center;
border-radius: 0em;
padding: 2em 0 1.5em;
}
```



```
.menu:after {
  content: attr(data-title);
  width: 100%;
  background: #000;
  color: #fff;
}

.menu > span > span {
  width: 120px;
  color: #fff;
}

.menu.servicios > span > span {
  background: #ffc000;
}

.menu.proyectos > span > span {
  background: #00a53c;
}

.menu.recursos > span > span {
  background: #b81800;
}

.menu.contacto > span > span {
  background: #9600b4;
}

/* Lo que pasa con hover */

.menu:hover > span {
  height: 100%;
}

/* Dando formato a lista */

ul {
  list-style-type: none;
  margin: 0;
  padding: 0;
}

li a {
```



```
display: inline;  
float: left;  
text-decoration: none;  
font-family: sans;  
}
```

TSN3.- Menú con efecto zoom-in

Crea un menú de navegación utilizando transiciones y transformaciones en CSS. A continuación, se presenta el código HTML y las especificaciones CSS que debes implementar:

Código HTML:

```
<nav>  
  <ul>  
    <li><a href="#">Inicio</a></li>  
    <li><a href="#">Servicios</a></li>  
    <li><a href="#">Acerca de</a></li>  
    <li><a href="#">Contacto</a></li>  
  </ul>  
</nav>
```

Instrucciones:

1. Utiliza el código HTML proporcionado como base para tu página.
2. Quitar viñetas de la lista: Elimina las viñetas del listado.
3. Aplicar márgenes horizontales: Añade márgenes horizontales de 5px a los enlaces del menú para separarlos ligeramente.
4. Eliminar subrayado: Elimina el subrayado de los enlaces en el menú.
5. Definir color de texto: Establece el color de texto en blanco (#fff) para que los enlaces se muestren en color blanco.
6. Establecer relleno: Define un relleno de 10px en la parte superior y inferior y 15px en los lados para dar espacio a los enlaces del menú.
7. Agregar transiciones suaves: Implementa una transición suave de 0.3 segundos en todas las propiedades para que los cambios de estilo sean fluidos.
8. Color de fondo predeterminado: Define un color de fondo predeterminado para los enlaces en el menú. Puedes usar el color #3498db.
9. Bordes redondeados: Aplica bordes redondeados (border-radius) con un valor de 5px a los enlaces.
10. Alinear en línea horizontalmente: Utiliza float: left para alinear los enlaces horizontalmente en el menú.



11. Estilos al pasar el mouse: Personaliza los estilos para los enlaces cuando el mouse pasa sobre ellos. Cambia el color de fondo a #1f587d y escala ligeramente los enlaces utilizando transformaciones.

SOLUCIÓN:

Código CSS.

```
ul {  
  list-style-type: none; /*También list-style*/  
}  
a {  
  margin: 0 5px;  
  text-decoration: none;  
  color: #fff;  
  padding: 10px 15px;  
  transition: all 0.3s ease;  
  background-color: #3498db;  
  border-radius: 5px;  
  float: left;  
}  
a:hover {  
  background-color: #1f587d;  
  -webkit-transform: scale(1.1);  
  transform: scale(1.1);  
}
```

Transformaciones

TFN1.- Rota, deforma y estira

Haciendo uso de propiedad transform y aplicando rotate, skew y scale hacer que el cuadro siguiente gire 150°, se deforme y se agrande:





SOLUCIÓN:

Código HTML:

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Cuadro deformable</title>
  <link rel="stylesheet" href="stylesheet.css">
</head>
  <body>
    <div class="caja">Mírame</div>
  </body>
</html>
```

Código CSS:

```
.caja {
  width: 200px;
  height: 200px;
  background-color: #ff6347;
  transition: transform 2s;
}

.caja:hover {
  transform: rotate(180deg) skew(20deg) scale(1.5);
}
```

TFN2.- Rotar botón



Pasa el ratón

Crea el siguiente botón aplicando HTML y CSS haciendo uso de propiedades *transform*, *rotate*, *skew* y *scale*. De manera que rote sobre su eje vertical y se deforme un poco.

SOLUCIÓN:

Código HTML:

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
```



```
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<title>Botón giratorio</title>
<link rel="stylesheet" href="stylesheet.css">
</head>
</head>
<body>
  <button class="boton">Pasa el ratón</button>
</body>
</html>
```

Código CSS:

```
.boton {
  padding: 10px 20px;
  font-size: 20px;
  background-color: #008080;
  color: white;
  border: none;
  border-radius: 12px; /* Bordes redondeados */
  cursor: pointer;
  transition: transform 2s;
}

.boton:hover {
  transform: rotateY(180deg) skew(10deg) scale(1.2);
}
```

TFN3.- Rotar imagen con información en anverso

A partir de una foto aplicando HTML y CSS y haciendo uso de propiedad transform, rotate, skew y scale así como de transform-origin y transform-style. Se pide: al pasar el ratón por encima de la imagen (frotal) gire sobre su eje vertical 180° y muestre los datos de la imagen (trasera): título, autor, fecha:

Frontal



Trasera

Título: loren ipsu
Autor: Cesar Augusto
Fecha: 01-01-01



SOLUCIÓN:

Código HTML:

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Tarjeta giratoria...</title>
  <link rel="stylesheet" href="stylesheet.css">
</head>
  <body>
    <div class="tarjeta">
      <div class="cara frente">Frente</div>
      <div class="cara atrás">Atrás</div>
    </div>
  </body>
</html>
```

Código CSS:

```
.tarjeta {
  width: 200px;
  height: 200px;
  position: relative;
  perspective: 1000px;
  transform-style: preserve-3d;
  transition: transform 2s;
}

.cara {
  width: 100%;
  height: 100%;
  position: absolute;
  backface-visibility: hidden;
  line-height: 200px;
  text-align: center;
  font-size: 24px;
  color: white;
}
```



```
.frente {  
  background-color: #ff6347;  
}  
  
.atras {  
  background-color: #32cd32;  
  transform: rotateY(180deg);  
}  
  
.tarjeta:hover {  
  transform: rotateY(180deg);  
}
```

TFN4.- Caída de imagen hasta final de pantalla

Que la imagen del apartado anterior al pasar el ratón por encima caiga hasta el final de la pantalla.

SOLUCIÓN:

Código HTML:

```
<!DOCTYPE html>  
<html lang="en">  
<head>  
  <meta charset="UTF-8">  
  <meta name="viewport" content="width=device-width, initial-scale=1.0">  
  <title>Caída de imagen</title>  
  <link rel="stylesheet" href="stylesheet.css">  
</head>  
<body>  
  <div class="contenedor">  
      
  </div>  
</body>  
</html>
```

Código CSS:

```
.contenedor {  
  width: 200px;
```




```
height: 200px;
position: relative;
perspective: 1000px;
}

.imagen {
width: 100%;
height: 100%;
object-fit: cover;
transition: transform 2s;
transform-origin: center center;
transform-style: preserve-3d;
}

.contenedor:hover .imagen {
transform: translateY(100vh);
}
```

TFN5.- Rebote de pelota

Dado el código siguiente implementar el CÓDIGO PERDIDO para que la pelota rebote.

NOTA: puede hacer uso de top y animation-timing-function

```
<!DOCTYPE html>
<html>

<head>
  <style>
    @keyframes rebote {
      /* CÓDIGO PERDIDO */
    }

    .suelo {
      position: absolute;
      top: 260px;
      bottom: 0;
      left: 0;
      width: 100%;
```



```
        height: 10px;
        background-color: #804000;
    }

    .pelota {
        /* La animación comenzará 1 segundo después de cargar la
página */
        position: relative;
        animation: rebote 2s linear infinite;
        left:0;top:0;
    }
</style>
</head>

<body>
    <div class="suelo"></div>
    <div class="pelota" id="pelota">
        <svg width="120" height="120" viewBox="0 0 120 120"
xmlns="http://www.w3.org/2000/svg">
            <circle cx="50" cy="50" r="20" fill="red"/>
        </svg>
    </div>
</body>
```

SOLUCIÓN:

Código HTML:

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Rebote de pelota</title>
    <link rel="stylesheet" href="stylesheet.css">
</head>
</head>
```



```
<body>
  <div class="suelo"></div>
  <div class="pelota" id="pelota">
    <svg width="120" height="120" viewBox="0 0 120 120"
    xmlns="http://www.w3.org/2000/svg">
      <circle cx="50" cy="50" r="20" fill="red"/>
    </svg>
  </div>
</body>
</html>
```

Código CSS:

```
@keyframes rebote {
  16% {
    top: 90px;
    animation-timing-function: ease-out;}

  32% {top: 50px;
    animation-timing-function: ease-in;}

  48% {
    top: 190px;
    animation-timing-function: ease-out;}

  62% {
    top: 100px;
    animation-timing-function: ease-in;}

  78% {
    top: 190px;
    animation-timing-function: ease-out;}

  90% {top: 150px;
    animation-timing-function: ease-in; }

  100% {top: 190px;
    animation-timing-function: ease-out;
    left: 100%;}
}

.suelo {
  position: absolute;
```



```
top: 260px;
bottom: 0;
left: 0;
width: 100%;
height: 10px;
background-color: #804000;
}

.pelota {
  /* La animación comenzará 1 segundo después de cargar la página */
  position: relative;
  animation: rebote 2s linear infinite;
  left: 0; top: 0;
}
```

TFN6.- En una galaxia lejana....

Vamos a recrear la Intro de STAR WARS. Dado el html reponer las secciones perdidas de código. Por cierto, no te olvides de añadir un fondo de estrellas.



Código:

```
<!DOCTYPE html>
<html>
<head>
<style>
body {
  width: 100%;
  height: 100%;
  background: #000;
  overflow: hidden;
}
```



```
.fade {
  position: relative;
  width: 100%;
  min-height: 60vh;
  top: -25px;
  background-image: linear-gradient(0deg, transparent, black 75%);
  z-index: 1;
}

.star-wars {
  display: flex;
  justify-content: center;
  position: relative;
  height: 800px;
  color: #feda4a;
  font-family: /* CÓDIGO PERDIDO */;
  font-size: 500%;
  font-weight: 600;
  letter-spacing: 6px;
  line-height: 150%;
  perspective: /* CÓDIGO PERDIDO */;
  text-align: justify;
}

.crawl {
  position: /* CÓDIGO PERDIDO */;
  top: 99999px;
  transform-origin: 50% 100%;
  animation: /* CÓDIGO PERDIDO */;
}

.crawl > .title {
  font-size: 90%;
  text-align: center;
}

.crawl > .title h1 {
  margin: 0 0 100px;
```



```
text-transform: uppercase;
}

@keyframes crawl {
  0% {
    /* CÓDIGO PERDIDO */
  }
  100% {
    /* CÓDIGO PERDIDO */
  }
}

</style>
</head>
<body>

  <section class="star-wars">
    <div class="crawl">
      <div class="title">
        <p>Episode IV</p>
        <h1>A New Hope</h1>
      </div>

      <p>It is a period of civil war. Rebel spaceships, striking
from a hidden base, have won their first victory against the evil
Galactic Empire.</p>

      <p>During the battle, Rebel spies managed to steal secret
plans to the Empire's ultimate weapon, the DEATH STAR, an armored
space station with enough power to destroy an entire planet.</p>

      <p>Pursued by the Empire's sinister agents, Princess Leia
races home aboard her starship, custodian of the stolen plans that
can save her people and restore freedom to the galaxy...</p>
    </div>
  </section>
</body>
</html>
```



SOLUCIÓN:

Código HTML:

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>En una galaxia lejana...</title>
  <link rel="stylesheet" href="stylesheet.css">
</head>
<body>
  <section class="star-wars">
    <div class="crawl">
      <div class="title">
        <p>Episode IV</p>
        <h1>A New Hope</h1>
      </div>
      <p>It is a period of civil war. Rebel spaceships, striking from a hidden
base, have won their first victory
      against the evil Galactic Empire.</p>
      <p>During the battle, Rebel spies managed to steal secret plans to
the Empire's ultimate weapon, the DEATH
      STAR, an armored space station with enough power to destroy an
entire planet.</p>
      <p>Pursued by the Empire's sinister agents, Princess Leia races
home aboard her starship, custodian of the
      stolen plans that can save her people and restore freedom to the
galaxy...</p>
    </div>
  </section>
</body>
</html>
```

Código CSS:

```
body {
  width: 100%;
  height: 100%;
  background: #000;
  overflow: hidden;
}
```



```
.fade {
  position: relative;
  width: 100%;
  min-height: 60vh;
  top: -25px;
  background-image: linear-gradient(0deg, transparent, black 75%);
  z-index: 1;
}

.star-wars {
  display: flex;
  justify-content: center;
  position: relative;
  height: 800px;
  color: #feda4a;
  font-family: 'Pathway Gothic One', sans-serif;
  font-size: 500%;
  font-weight: 600;
  letter-spacing: 6px;
  line-height: 150%;
  perspective: 400px;
  text-align: justify;
}

.crawl {
  position: relative;
  top: 99999px;
  transform-origin: 50% 100%;
  animation: crawl 60s linear;
}

.crawl>.title {
  font-size: 90%;
  text-align: center;
}

.crawl>.title h1 {
  margin: 0 0 100px;
  text-transform: uppercase;
}

@keyframes crawl {
  0% {
```




```
        top: -100px;  
        transform: rotateX(20deg) translateZ(0);  
    }  
  
    100% {  
        top: -6000px;  
        transform: rotateX(25deg) translateZ(-2500px);  
    }  
}
```

TFN7.- Botón que se descuelga y cae.

Crear un botón de broma de manera que cuando se pase el ratón por encima este se descuelgue de la esquina superior izquierda y luego se descuelgue y caiga a la parte de abajo de la pantalla.

NOTA: puedes anidar animaciones con animation-name separándolas por comas.

Ejemplo: animation-name: spin, caída;

SOLUCIÓN:

Código HTML:

```
<!DOCTYPE html>  
<html lang="en">  
<head>  
    <meta charset="UTF-8">  
    <meta name="viewport" content="width=device-width, initial-scale=1.0">  
    <title>Botón de Cristal Roto con Efecto de Caída</title>  
    <link rel="stylesheet" href="stylesheet.css">  
</head>  
<body>  
    <button class="boton-cristal">Botón de broma</button>  
</body>  
</html>
```

Código CSS:

```
@keyframes spin {  
    0% {  
        transform-origin: 0 0;  
        transform: rotate(0deg);  
    }  
}
```



```
25% {
  transform-origin: 0 0;
  transform: rotate(30deg);
}

50% {
  transform-origin: 0 0;
  transform: rotate(60deg);
}

100% {
  transform-origin: 0 0;
  transform: rotate(95deg);
}
}

@keyframes caida {
  0% {
    transform-origin: 0 0;
    transform: rotate(96deg);
  }

  100% {
    transform-origin: 0 0;
    transform: translateY(calc(100vh - 100%));
  }
}

body {
  display: flex;
  align-items: center;
  justify-content: center;
  height: 100vh;
  margin: 0;
  background-color: #001f3f;
  overflow: hidden;
}
```



```
.boton-cristal {
  position: relative;
  padding: 15px 30px;
  background-color: transparent;
  border: 2px solid #7FDBFF;
  color: #7FDBFF;
  font-size: 16px;
  cursor: pointer;
  overflow: hidden;
  z-index: 1;
}

.boton-cristal:hover {
  animation-name: spin, caida;
  animation-duration: 1.5s, 3s;
  animation-timing-function: linear, cubic-bezier(0.390, 0.575, 0.565, 1.000);
  animation-direction: forwards, forwards;
  animation-delay: 0s, 1.4s;
}

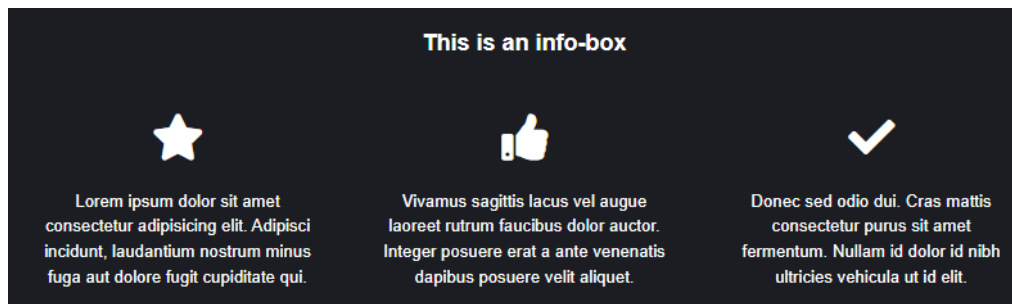
.boton-cristal::before {
  content: "";
  position: absolute;
  top: 50%;
  left: 50%;
  width: 0;
  height: 0;
  background: radial-gradient(circle, rgba(255, 255, 255, 0.5) 0%, rgba(255, 255, 255, 0) 80%);
  transform: translate(-50%, -50%);
  opacity: 0;
  transition: width 0.5s, height 0.5s, opacity 0.5s;
  z-index: -1;
}
```



Flex

F1.- Contenedores en línea

Crea tres contenedores que se encuentren inicialmente en línea y que se vayan situando unos bajo de los otros según el ancho del dispositivo.



SOLUCIÓN:

Código HTML:

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Botón de Cristal Roto con Efecto de Caída</title>
  <link rel="stylesheet" href="stylesheet.css">
  <script src="https://kit.fontawesome.com/509deda241.js"
crossorigin="anonymous"></script>
</head>
<body>
  <div class="info-box">
    <h2>This is an info-box</h2>
    <div id="flex-p">
      <p class="big"><i class="fas fa-star"></i><br>Lorem ipsum dolor sit amet
consectetur adipisicing elit. Adipisci incidunt, laudantium nostrum minus fuga aut
dolore fugit cupiditate qui.</p>
      <p class="big"><i class="fas fa-thumbs-up"></i><br>Vivamus sagittis lacus vel
augue laoreet rutrum faucibus dolor auctor. Integer posuere erat a ante venenatis
dapibus posuere velit aliquet.</p>
      <p class="big"><i class="fas fa-check"></i><br>Donec sed odio dui. Cras mattis
consectetur purus sit amet fermentum. Nullam id dolor id nibh ultricies vehicula ut id
elit.</p>
    </div>
  </div>
</body>
</html>
```



```
</div>  
</div>  
</body>
```

Código CSS:

```
*{  
    margin: 0px;  
    padding: 0 px;  
    font-family: "Roboto", Arial, Tahoma, sans-serif;  
}  
.info-box{  
    background-color: #1C1D22;  
    margin: 0px auto;  
    padding: 110px 0;  
    color: white;  
    text-align: center;  
}  
.info-box h2{  
    margin-bottom: 30px;  
}  
#flex-p{  
    justify-content: center;  
    display: flex;  
    flex-wrap: wrap;  
    align-items: center;  
}  
#flex-p p{  
    padding: 30px;  
    max-width: 300px;  
}  
p.big{  
    font-size: 1.1em;  
    line-height: 1.5em;  
}  
.fa-star, .fa-thumbs-up, .fa-check{  
    font-size: 2.7em;  
    display: block;  
    color: white;  
    margin-bottom: 30px;  
}
```



F2.- Web con barra de navegación, área de contenido y pie

Crea una página web que contenga una barra de navegación horizontal en la parte superior, un pie de página en la parte inferior, y un área de contenido principal que tenga una sección principal y una barra lateral. Utiliza Flexbox para hacer que la barra lateral esté siempre a la derecha de la sección principal y para que la barra de navegación y el pie de página se extiendan a lo largo del ancho de la ventana del navegador.



SOLUCIÓN:

Código HTML:

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Botón de Cristal Roto con Efecto de Caída</title>
  <link rel="stylesheet" href="stylesheet.css">
  <script src="https://kit.fontawesome.com/509deda241.js"
crossorigin="anonymous"></script>
</head>

<body>
<header class="navbar">
  <nav>
    <ul>
      <li><a href="#">Inicio</a></li>
      <li><a href="#">Acerca</a></li>
      <li><a href="#">Servicios</a></li>
      <li><a href="#">Contacto</a></li>
    </ul>
  </nav>
</header>
<main class="content-area">
  <section class="main-content">
```



```
<h1>Título del Contenido Principal</h1>
<p>Este es el contenido principal de la página.</p>
</section>
<aside class="sidebar">
  <h2>Barra Lateral</h2>
  <p>Contenido de la barra lateral.</p>
</aside>
</main>
</body>
<footer class="footer">
  <p>Derechos reservados © 2023</p>
</footer>
</html>
```

Código CSS:

```
* {
  box-sizing: border-box;
  margin: 0;
  padding: 0;
}

body {
  display: flex;
  flex-direction: column;
  /* min-height: 100vh;*/
}

.navbar {
  background-color: #333;
  color: white;
  padding: 1em;
}

nav ul {
  list-style: none;
  display: flex;
}

.navbar nav a {
  color: white;
  text-decoration: none;
  padding: 0.5em;
}
```



```
.content-area {
  display: flex;
  flex-grow: 1;
  padding: 1em;
}

.main-content {
  flex-grow: 3;
  padding: 1em;
  background-color: #f8f8f8;
}

.sidebar {
  flex-grow: 1;
  padding: 1em;
  background-color: #e7e7e7;
}

.footer {
  background-color: #333;
  color: white;
  padding: 1em;
  text-align: center;
}

@media (max-width: 768px) {
  .content-area {
    flex-direction: column;
  }
}
```

F3.- Encabezado con texto o logo y menú

Crea un encabezado que tenga un elemento a la izquierda (texto o imagen) y otro a la derecha (menú de navegación).

Para este ejercicio se utiliza un contenedor principal `<header>` y se distribuye el espacio de los elementos contenidos con **justify-content: space-between**. Los elementos del menú de navegación se establecen de tipo flex para alineación horizontal uno detrás del otro.



Nombre del Sitio

Inicio

Acerca

Servicios

Contacto

Dos contenedores alineados uno a la derecha y otro a la izquierda con justify-content: space-between

Menú de navegación con los elementos de tipo flex para alineación horizontal uno detrás del otro.

SOLUCIÓN:

Código HTML:

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Botón de Cristal Roto con Efecto de Caída</title>
  <link rel="stylesheet" href="stylesheet.css">
  <script src="https://kit.fontawesome.com/509deda241.js"
crossorigin="anonymous"></script>
</head>
<body>
  <header>
    <h1>Nombre del Sitio</h1>
    <nav>
      <ul>
        <li><a href="#">Inicio</a></li>
        <li><a href="#">Acerca</a></li>
        <li><a href="#">Servicios</a></li>
        <li><a href="#">Contacto</a></li>
      </ul>
    </nav>
  </header>
  <br>
  <p>Dos contenedores alineados uno a la derecha y otro a la izquierda con
justify-content: space-between </p>
  <p>Menú de navegación con los elementos de tipo flex para alineación
horizontal uno detrás del otro. </p>
</body>
</html>
```



Código CSS:

```
* {
  margin: 0;
  padding: 0;
  font-family: Arial, sans-serif;
  font-size: 16px;
}

header {
  background-color: #3498db;
  color: #fff;
  padding: 15px;
  display: flex;
  justify-content: space-between;
  align-items: center;
}

h1 {
  margin: 0;
}

nav ul {
  list-style: none;
  display: flex;
}

nav li {
  margin-right: 20px;
}

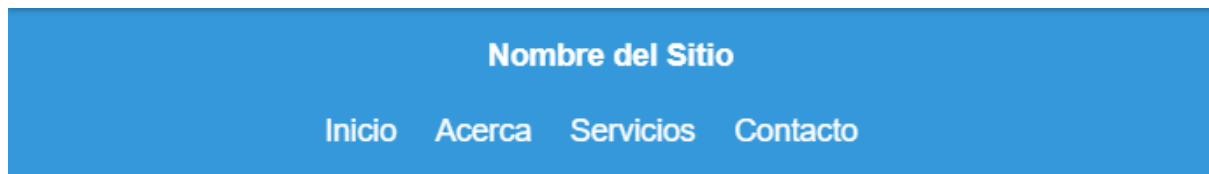
a {
  text-decoration: none;
  color: #fff;
}
```



F4.- Encabezado con texto arriba y menú debajo

Crea un encabezado que tenga un elemento arriba (texto o imagen) y otro abajo (menú de navegación). Ambos elementos deben encontrarse centrados.

Para este ejercicio se utiliza un contenedor principal `<header>` y se alinean verticalmente sus elementos hijos con **flex-direction: column**. Los elementos del menú de navegación se establecen de tipo flex para alineación horizontal uno detrás del otro.



Dos contenedores alineados verticalmente
Menú de navegación con los elementos de tipo flex para alineación horizontal uno detrás del otro.

SOLUCIÓN:

Código HTML:

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Botón de Cristal Roto con Efecto de Caída</title>
  <link rel="stylesheet" href="stylesheet.css">
  <script src="https://kit.fontawesome.com/509deda241.js"
crossorigin="anonymous"></script>
</head>
<body>
  <header>
    <h1>Nombre del Sitio</h1>
    <nav>
      <ul>
        <li><a href="#">Inicio</a></li>
        <li><a href="#">Acerca</a></li>
        <li><a href="#">Servicios</a></li>
        <li><a href="#">Contacto</a></li>
      </ul>
    </nav>
  </header>
</body>
</html>
```



```
</header>
<br>
<p>Dos contenedores alineados verticalmente</p>
<p>Menú de navegación con los elementos de tipo flex para alineación
horizontal uno detrás del otro.
</body>
</html>
```

Código CSS:

```
* {
  margin: 0;
  padding: 0;
  font-family: Arial, sans-serif;
  font-size: 16px;
}
header {
  background-color: #3498db;
  color: #fff;
  padding: 15px;
  display: flex;
  flex-direction: column;
  align-items: center;
}
h1 {
  margin: 0;
  padding-bottom: 20px;
}
nav ul {
  list-style: none;
  display: flex;
}
nav li {
  margin-right: 20px;
}
a {
  text-decoration: none;
  color: #fff;
}
```



F5.- Menú de navegación de ancho completo y que crezca

Crea un menú de navegación que ocupe el ancho total de la pantalla y que sus elementos crezcan para ocupar de forma uniforme el ancho total.

Para este ejercicio utilizamos la propiedad **flex-grow** para que los elementos del menú crezcan hasta ocupar el ancho total del menú de navegación.

Nombre del Sitio

Inicio	Acerca	Servicios	Contacto
--------	--------	-----------	----------

Dos contenedores alineados verticalmente

Menú de navegación con los elementos de tipo flex para alineación horizontal uno detrás del otro. Como tiene elementos li y dentro elementos a. Primero ocupamos el espacio disponible con flex-grow:1 en los li. Después hacemos que los enlaces ocupen todo su contenedor padre haciendo display:block

SOLUCIÓN:

Código HTML:

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Botón de Cristal Roto con Efecto de Caída</title>
  <link rel="stylesheet" href="stylesheet.css">
  <script src="https://kit.fontawesome.com/509deda241.js"
crossorigin="anonymous"></script>
</head>
<body>
  <header>
    <h1>Nombre del Sitio</h1>
    <nav>
      <ul>
        <li><a href="#">Inicio</a></li>
        <li><a href="#">Acerca</a></li>
        <li><a href="#">Servicios</a></li>
        <li><a href="#">Contacto</a></li>
      </ul>
    </nav>
  </header>
  <br>
```



```
<p>Dos contenedores alineados verticalmente</p>
<p>Menú de navegación con los elementos de tipo flex para alineación
horizontal uno detrás del otro. Como tiene elementos li y dentro elementos a.
Primero ocupamos el espacio disponible con flex-grow:1 en los li. Después
hacemos que los enlaces ocupen todo su contenedor padre haciendo
display:block </p>
</body>
</html>
```

Código CSS:

```
* {
  margin: 0;
  padding: 0;
  font-family: Arial, sans-serif;
  font-size: 16px;
}
header {
  display: flex;
  flex-direction: column;
  align-items: center;
}
h1 {
  margin: 20px;
}
nav {
  width: 100%; /* Ancho completo del contenedor */
  background-color: #3498db;
}
nav ul {
  display: flex; /* Activar Flexbox */
  list-style-type: none; /* Quitar estilos de lista */
  padding: 0; /* Quitar padding predeterminado */
  margin: 0; /* Quitar margen predeterminado */
}
nav li {
  flex-grow: 1; /* Crecer para ocupar el espacio disponible */
  text-align: center; /* Centrar el texto */
}
nav a {
  display: block; /* Hacer que el enlace ocupe todo el espacio del li */
  margin: 0;
```



```
color: white;
padding: 10px; /* Espaciado para una mejor visualización */
text-decoration: none; /* Quitar el subrayado del enlace */
border: 1px solid white;
}
nav a:hover {
    background-color: #217dbb;
}
```