

Ejercicio HTML / CSS Nostromo

Introducción

Este ejercicio consta de lo siguiente:

- Una página principal llamada index.htm
- Una página sobre los aliens llamada alien.htm
- Una foto de la nave Nostromo
- Una serie de fotos de las tripulaciones (estaría bien unas fotos de más calidad y más pequeñas)
- Una serie de fotos de aliens (estaría bien cambiarles el tamaño a uno más pequeño)
- Dos tipos de letras descargadas de internet
- Un tipo de letra de Bootstrap para los iconos de las redes sociales



Para ello vamos a crear la siguiente estructura en nuestro proyecto:

```
css
  estilos.css
fuentes
  HomepageBaukasten Bold.ttf
  Open 24 Display St.ttf
imagenes
  tripulacion
    fotos de los tripulantes
  aliens
    fotos de los aliens
  nostromo.jpg
index.htm
alien.htm
```

Como el encabezado y pie es casi idéntico en ambas páginas, vamos a crear primero ambos en una de ellas y luego copiarlo en la otra, empezando por alien.htm por ser más fácil.

Podríamos también haber creado un fichero de estilos base y luego otros dos para lo propio de index.htm y de alien.htm y así tener una separación mejor.

Ejercicio HTML / CSS Nostromo

alien.htm (encabezado y pie)



Para el pie vamos a usar los iconos en formato de fuente de letra de <https://icons.getbootstrap.com/>. Para ello debemos añadir su CDN a nuestro fichero HTML con una etiqueta link o con @import en el fichero de estilos; vamos a hacer lo segundo. En la página de los iconos, bastante abajo después de los iconos, podemos encontrar la URL de la CDN:

`https://cdn.jsdelivr.net/npm/bootstrap-icons@1.13.1/font/bootstrap-icons.min.css"`

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

  <body>
    <header>
      <h1>USCSS Nostromo 'Aliens</h1>
      <nav>
        <a href="index.htm">inicio</a>
        <a href="">historia</a>
        <a href="alien.htm">alien</a>
        <a href="">contacto</a>
      </nav>
    </header>
    <main>

  </main>

    <footer>
      <div class="atribucion">
        Imágenes generadas por IA <!-- Podemos poner un texto directamente dentro de un elemento en
        bloque sin etiqueta p o similar-->
        <a href="https://perchance.org/ai-text-to-image-generator">
          https://perchance.org/ai-text-to-image-generator</a>
      </div>
      <div class="redes">
        <a href=""><i class="bi bi-twitter"></i></a>
        <a href=""><i class="bi bi-instagram"></i></a>
        <a href=""><i class="bi bi-envelope"></i></a>
      </div>
    </footer>
  </body>
</html>
```

Notas:

- Podemos ver cómo se añaden los iconos de Bootstrap en el pie usando etiquetas i y las clases adecuadas, las cuales podemos obtener de <https://icons.getbootstrap.com/>. Allí también podemos elegir usar el SVG en lugar de la fuente. Se usa una etiqueta i por convención ya que suena a icono
- Hasta que no añadamos el import en los estilos no veremos los iconos

Ejercicio HTML / CSS Nostromo

- Hemos dividido el pie en dos divs pues, como una parte tiene que ir a la izquierda y la otra a la derecha, lo mejor es usar Flex y luego usar justify-content: space-between. Si no metemos cada parte en un div, tendremos el texto, la imagen y cada uno de los enlaces en una columna diferente y no podríamos, o sería más difícil, ponerlos como queremos

index.htm (encabezado y pie)

Basta con copiar el contenido de la página alien.htm y cambiar solo el texto de atribuciones del pie.

estilos.css (encabezado, pie y lo básico)

```
/* Importamos los iconos de Bootstrap */
@import url("https://cdn.jsdelivr.net/npm/bootstrap-icons@1.13.1/font/bootstrap-icons.min.css");

/* Hacemos un CSS Reset de las propiedades que nos interesen. No siempre hay que asignar 0 a los
espacios, también podemos poner otros
valores que veamos más convenientes */
* {
    margin: 0;
    padding: 0;
    font-weight: 100%; /* Toda la letra del mismo tamaño */
    box-sizing: border-box; /* Para que el padding y borde no sume al ancho de un elemento */
    text-decoration: none; /* Quitamos todos los subrayados de los enlaces */
    font-weight: normal; /* Quitamos las negritas */
}

/* Importamos las fuentes a usar */
@font-face {
    font-family: homepage; /* Un nombre cualquiera que usaremos luego en otras reglas en esta misma
propiedad */
    src: url(../fuentes/Homepagebaukasten\ Bold.ttf);
    /*src:url(/fuentes/Homepagebaukasten\ Bold.ttf); Usando ruta absoluta*/
}
@font-face {
    font-family: open24;
    src: url(../fuentes/Open\ 24\ Display\ St.ttf);
    /*src: url(/fuentes/Open\ 24\ Display\ St.ttf);*/
}

/* Definimos los colores como variables con lo que sería más fácil en el futuro cambiar un color (pues
se cambia en un solo sitio) y
en el código queda también más claro el color usado ya que ponemos su nombre.
Recordemos que las variables deben comenzar por -- y no tene espacios o algunos símbolos especiales,
diferenciándose mayúsculas de minúsculas.
El valor asignado a una propiedad puede ser cualquier valor que podríamos asignar a una propiedad CSS
*/
:root {
    --colorTitulos: #66fcf1;
    --colorResaltado: rgb(124, 229, 124);
    /* Podemos crear una variable a partir de otra variable */
    --sombra: 2px 2px 8px var(--colorResaltado), -2px -2px 8px var(--colorResaltado);
    --colorTextoPie: #888;
    --colorEnlacesPie: white;
}

body {
    background-color: black;
    font-family: 'Segoe UI', Tahoma, Geneva, Verdana, sans-serif;
    /* Para que la página no sea muy ancha en monitores horizontales de mucha resolución (1200 u otro
valor que prefiramos).
También podemos aplicar este ancho máximo a elementos en lugar del body si queremos que el encabezado,
pié, ... ocupen todo el ancho */
    max-width: 1200px;
    margin: auto; /* La centramos horizontalmente. Recordemos que basta con marginr-left y margin-right
a auto */
}
```

Ejercicio HTML / CSS Nostromo

```
header {
  background-color: #1f2833;
  padding-bottom: .5rem; /* QUremos un poco de espacio entre el borde de lso enlaces y en final del
                             encabezado */
}

header h1 {
  color: var(--colorTitulos);
  /* Para hacer la letra responsiva:
   - vw: es un porcentaje del ancho del viewport (zona disponible)
   - le sumamos los rem para no tener problemas de accesibilidad Ya que 4vw en un móvil puede ser muy
      pequeño.
   En otros casos, puede ser conveniente indicar un mínimo y/o máximo para que no crezca o empequeñezca
      demasiado */
  font-size: calc(4vw);
  text-align: center;
  text-shadow: var(--sombra);
  font-family: homepage, sans-serif;
  /* Mientras no cargue la fuente que muestre una del sistema */
  font-display: swap;
}

header nav {
  /* Para colocar los menús separados y del mismo tamaño, una opción es usar flex o grid, entre otras.
  Aquí optamos por Grid */
  display: grid;
  grid-template-columns: repeat(4, 1fr); /* 4 columnas que ocupen todas el mismo espacio sobrante
                                           (1fr) */
  text-align: center; /* En este ejercicio, esta es la manera más sencilla de centrar los enlaces
                        dentro de su columna */
  gap: 2rem; /* Separación entre filas y entre columnas */
}

header nav a {
  color: var(--colorTitulos);
  border-bottom: 4px solid var(--colorTitulos);
  padding-bottom: .5rem; /* Para que no quede tan pegado el borde */
}

header nav a:hover { /* CUando pase el ratón por encima de un enlace */
  color: var(--colorResaltado);
  border-color: var(--colorResaltado);
  text-shadow: var(--sombra);
}

footer {
  border-top: 1px solid #333;
  margin-top: 2rem;
  /* La manera más sencilla de poner una parte a la izuquierda y otra a la derecha es flex con space-
  between */
  display: flex;
  justify-content: space-between;
  padding: 1rem;
  /* El color y tamaño, entre otros, también afecta a los iconos pues son también letras, aunque
  algunos iconos de otros
  podrían no dejar cambiar el color o solo uno de lso colores. No es el caso. */
  font-size: .8rem;
  color: var(--colorTextoPie);
}

footer a {
  color: var(--colorEnlacesPie);
}

.atribucion a {
  /* Como quitamos el subrayado en CSS Reset, ahora lo añadimos */
  text-decoration: underline;
}
```

Ejercicio HTML / CSS Nostromo

```
.redes a {  
  margin-right: 2rem;  
}
```

alien.htm (resto)

Ahora nos queda añadir el código que va dentro del main. Tendremos una serie de fotos con un texto a su derecha. Para ello podemos usar flex o grid, además del antiguo float; en este caso optamos por flex, debiendo poner también el título y párrafos de cada foto dentro de un contenedor para que el flex solo afecte a este, tal como explicamos anteriormente. Optamos por un div pero podría ser otra etiqueta contendora.

Ponemos solo un div en este documento, ya que todos son iguales en sus etiquetas, pero harían falta otros tres en el documento HTML.

```
<div class="foto">  
    
  <div>  
    <h2>En las entrañas del conducto</h2>  
    <p>  
      La criatura se arrastra silenciosamente por el angosto túnel  
      metálico, sus movimientos apenas perceptibles entre las sombras. Las  
      paredes húmedas reflejan el brillo intermitente de luces de  
      emergencia, mientras su cuerpo aceitado se desliza sin esfuerzo,  
      como si la nave misma lo acogiera como huésped natural.  
    </p>  
    <p>  
      Cada zancada suena lejana, amortiguada por el eco de acero. Su  
      respiración es casi imperceptible, pero constante, como una amenaza  
      latente. Nadie sabe por dónde vendrá, pero el conducto ya lo ha  
      reclamado como suyo.  
    </p>  
  </div>  
</div>
```

estilos.css (fotos de alien.htm)

```
.foto {  
  display: flex;  
  gap: 40px; /* Espacio entre las filas y columnas */  
  margin-bottom: 20px;  
  margin-top: 20px;  
  padding: 1rem;  
}  
.foto img {  
  /* Le damos un ancho máximo para que no sea demasiado grande la foto */  
  max-width: 30%;  
  /* Queremos que cada foto esté dentro de un octógono. Para ello podemos usar la propiedad clip-path  
  que nos permite crear áreas de recorte  
  de muchos tipos. Para crearlas lo mejor es acudir a páginas como https://bennettfeely.com/clippy/  
  Anteriormente se llamaba a esta propiedad clip */  
  clip-path: polygon(30% 0%, 70% 0%, 100% 30%, 100% 70%, 70% 100%, 30% 100%, 0% 70%, 0% 30%);  
  /* Si empequeñecemos la pantalla mucho y el texto crece de alto más que la foto, está quedará  
  deformada, pues flex por defecto usa stretch para que todas las columnas tengan el mismo, lo que  
  hace que la foto se alargue. Una solución sería  
  colocar la imagen dentro de un div, con lo que lo que se alargaría sería el div, no la foto. Otra  
  opción es usar align-self con center, flex-start, ...  
  Si usamos flex-start, en este caso, podríamos hacerlo en la propiedad align-items de .foto, ya que  
  no afectaría al texto */  
  align-self: flex-start;  
}
```

Ejercicio HTML / CSS Nostromo

```
/* Las fotos impares (odd) en otro color de fondo. even para las pares */
.foto:nth-of-type(odd) {
  /* Para usar la variable --colorFondo pero algo cambiada, podemos usar la función color-mix. En
  este caso le damos algo de transparencia al color */
  background-color: color-mix(in srgb, var(--colorFondo) 70%, transparent);
}

.foto h2 {
  color: var(--colorTitulos);
  font-family: homepage, sans-serif;
  font-size: calc(1rem + 1vw);
  border-bottom: 2px solid var(--colorTitulos);
  margin-top: 1rem;
  margin-bottom: 1rem;
  padding: .4rem 1rem;
}

.foto p {
  color: white;
  margin-top: 1rem;
  margin-bottom: 1rem;
}
```

index.htm

Para la sección sobre la nave vamos a crear un contenedor flex y así poder tener dos columnas. Como en casos anteriores, colocaremos los textos dentro de un div contenedor para que vayan en la segunda columna.

```
<section id="nave">
  <h2>Nave</h2>
  <div>
    
    <div>
      <p>
        La <strong>USCSS Nostromo</strong> es una nave remolcadora
        espacial registrada en la compañía <em>Weyland-Yutani</em>. Su
        misión principal es el transporte de mineral desde colonias
        espaciales a la Tierra. Tiene una capacidad de carga enorme, una
        tripulación reducida y sistemas de inteligencia artificial
        avanzados como "MU-TH-UR 6000".
      </p>
      <h3>Características técnicas:</h3>
      <ul>
        <li>Registro: 180924609</li>
        <li>Largo: 334 metros</li>
        <li>Peso: 63,000 toneladas</li>
        <li>Velocidad: FTL (con nave auxiliar)</li>
        <li>IA: MU-TH-UR 6000</li>
        <li>Propietario: Weyland-Yutani Corporation</li>
      </ul>
    </div>
  </div>
</section>
```

Para la sección de la tripulación vamos a optar por un grid para así conseguir de manera fácil que todas las filas tengan el mismo alto, pues algunas fotos tienen menos alto. Dentro de cada tripulante vamos a usar un flex para poner la foto a la izquierda y los textos a la derecha.

```
<section id="tripulacion">
  <h2>Tripulación</h2>
  <div class="tripulantes">
    <div class="tripulante">
      
      <div class="tarjeta-datos">
```

Ejercicio HTML / CSS Nostromo

```
        <h3>Ellen Ripley</h3>
        <h4>Suboficial de vuelo</h4>
    </div>
    <a href="" class="masInfo">+ info</a>
</div>

<div class="tripulante">
    
    <div class="tarjeta-datos">
        <h3>Dallas</h3>
        <h4>Capitán</h4>
    </div>
    <a href="" class="masInfo">+ info</a>
</div>

<!-- Aquí irán el resto de tripulantes -->

</section>
```

Estilos.css (nave)

Primero aprovechamos los estilos que creamos para los h2 de las fotos para usarlos también con el h2 de nave:

```
.foto h2, #nave h2 {
    /* Lo que ya estaba */
}

/* En este caso anidamos los estilos como ejemplo */
#nave {
    color: white;
    > div { /* Al div hijo directo de #nave (para no afectar a otros div */
        display: flex; /* Para poner la foto a la izquierda y el texto a la derecha */
        gap: 30px;
        margin: 1rem;
    }
    img {
        max-width: 250px; /* Para que no crezca demasiado */
        /* Al igual que explicamos con las fotos de los aliens, usamos esta propiedad para que la
        foto no se alargue demasiado de alto y se deforme */
        align-self: flex-start;
    }
    h3 {
        font-weight: bold;
        margin-top: 1rem;
        margin-bottom: 1rem;
    }
    ul {
        margin-left: 2rem;
    }
    li {
        margin-bottom: 1rem;
    }
    li:hover {
        margin-left: 1rem; /* Ponemos el texto un poco hacia adentro cuando pasa el ratón por encima */
    }
    li:nth-of-type(odd) { /* Los li impares */
        color: #45a29e; /* No usamos variables aquí ya que solo aparecen una vez */
    }
    li:nth-of-type(even) { /* Los lie pares */
        color: #99e6e2;
    }
}
```

Ejercicio HTML / CSS Nostromo

Estilos.css (tripulación)

Primero aprovechamos los estilos que creamos para los h2 de las fotos y nave para usarlos también con el h2 de tripulación:

```
.foto h2, #nave h2, #tripulacion h2 {
    /* Lo que ya estaba */
}

.tripulantes {
    display: grid;
    /* con auto-fit creamos todas las columnas que se puedan, cada una de un tamaño entre 200px y 20vw */
    grid-template-columns: repeat(auto-fit, minmax(200px, 20vw));
    gap: 40px;
    /*Si no usamos la propiedad grid-template-rows para indicar las filas deseadas, grid va creando las
    que hagan falta, pero puede que de un alto
    diferente. Con esta propiedad indicamos que alto tendrán las filas que se vayan creando */
    grid-auto-rows: 1fr;
}

.tripulante {
    font-family: open24;
    background-color: #1f2833;
    border: 1px solid #45a29e;
    border-radius: 5px;
    overflow: hidden; /* Para que la imagen o un color de fondo de un hijo no salga por fuera de la caja,
    con lo que no se vería el borde redondeado */
    box-shadow: 0 0 10px #45a29e55;
    display: flex;
    /* Para poder poner el texto 'más info' encima de la foto y en la zona inferior, vamos a usar
    Posicionamiento absoluto en el enlace de m'ás info'. Por ello debemos cambiar el posicionamiento
    de tripulante, que por defecto es static, para que sea su padre, y no el body u otro padre con
    distinto posicionamiento de static. Y lo más fácil y que no 'estropea' nada es darle posicionamiento
    relativo */
    position: relative;
}

.tripulante:hover {
    /* Agrandamos un poco la caja al pasar el ratón por encima */
    transform: scale(1.02);
    /* Para que no lo haga de golpe; que lo haga en 0.3 segundos */
    transition: transform 0.3s ease;
    border-color: greenyellow;
}

.tripulante img {
    /* Para que la imagen ocupe algo menos */
    width: 55%;
}

.tripulante h3 {
    margin: 0.5rem 0 0.25rem;
    color: #66fcf1;
}

.tripulante h4 {
    margin: 0;
    color: #45a29e;
    font-weight: normal;
    font-size: 0.95rem;
}

.tripulante p {
    font-size: 0.9rem;
    margin-top: 0.5rem;
    line-height: 1.4;
}
```


Ejercicio HTML / CSS Nostromo

```
.tarjeta-datos {  
  padding: 0 1rem;  
}  
  
.masInfo {  
  /* Para colocar el enlace sobre la foto */  
  position: absolute;  
  bottom: 10px; /* 10px por encima de la parte inferior del contenedor */  
  left: 10px; /* 10px dentro de la parte izquierda del contenedor */  
  padding: 0.3rem 1rem;  
  background-color: rgba(0, 0, 0, 0.554);  
  color: white;  
  font-family: "Segoe UI", Tahoma, Geneva, Verdana, sans-serif;  
  /* Con el & indicamos que es el mismo elemento, no un hijo:  
    - &hover equivale a .masInfo:hover  
    - hover equivale a .masInfo :hover */  
  &:hover {  
    background-color: #45a29e;  
  }  
}
```