Diseño web receptivo heredado

En esta Certificación de diseño web receptivo, aprenderá los lenguajes que usan los desarrolladores para crear páginas web: HTML (lenguaje de marcado de hipertexto) para el contenido y CSS (hojas de estilo en cascada) para el diseño.

Primero, creará una aplicación de fotos de gatos para aprender los conceptos básicos de HTML y CSS. Más adelante, aprenderá técnicas modernas como variables CSS mediante la creación de un pingüino y las prácticas recomendadas de accesibilidad mediante la creación de un formulario web.

Finalmente, aprenderá cómo crear páginas web que respondan a diferentes tamaños de pantalla mediante la creación de una tarjeta de Twitter con Flexbox y un diseño de blog complejo con CSS Grid.

Nota: Algunas extensiones del navegador, como los bloqueadores de anuncios y las extensiones del modo oscuro, pueden interferir con las pruebas. Si tiene problemas, le recomendamos que deshabilite las extensiones que modifican el contenido o el diseño de las páginas mientras realiza el curso.

CURSO:

HTML básico y HTML5

HTML es un **lenguaje de marcado** que utiliza una sintaxis o notación especial para describir la estructura de una página web al navegador.

Los elementos HTML suelen tener **etiquetas de apertura y cierre** que rodean y dan sentido al contenido. Por ejemplo, diferentes elementos pueden describir texto como encabezado, párrafo o elemento de lista.

En este curso, **creará una aplicación de fotos de gatos** para aprender algunos de los elementos HTML más comunes: los componentes básicos de cualquier página web.

Por favor, disminuya la velocidad y lea esto.

freeCodeCamp es un camino comprobado hacia su primer trabajo como desarrollador de

software.

Más de 40 000 personas obtuvieron trabajos de desarrollador después de completar esto, incluso en grandes empresas como Google y Microsoft.

Si es nuevo en la programación, le recomendamos que comience desde el principio y obtenga estas certificaciones en orden.

Para obtener cada certificación, construya sus 5 proyectos requeridos y apruebe todas sus pruebas.

Puede agregar estas certificaciones a su currículum o LinkedIn. Pero más importante que las certificaciones es la práctica que obtienes en el camino.

Si te sientes abrumado, es normal. La programación es difícil.

La práctica es la clave. Práctica práctica práctica.

Y este plan de estudios le dará miles de horas de práctica de programación práctica.

Y si quieres aprender más teoría matemática e informática, también tenemos miles de horas de cursos en video en el canal de YouTube de freeCodeCamp .

Si desea obtener un trabajo de desarrollador o clientes independientes, las habilidades de programación serán solo una parte del rompecabezas. También necesita construir su red personal y su reputación como desarrollador.

Puedes hacerlo en Twitter y GitHub, y también en el foro de freeCodeCamp .

¡Feliz codificación!

- Quincy Larson, el maestro que fundó freeCodeCamp.org

Por cierto, cada viernes envío un email con 5 enlaces sobre programación e informática. Los envío a unos 4 millones de personas. ¿Quieres que te envíe esto también?

LE HE DICHO QUE NO.

Saluda a los elementos HTML

Bienvenido a los desafíos de codificación HTML de freeCodeCamp. Estos lo guiarán a través del desarrollo web paso a paso.

Primero, comenzará creando una página web simple usando HTML. Puede editar el código en su editor de código, que está incrustado en esta página web.

¿Ves el código en tu editor de código que dice <h1>Hello</h1>? Eso es un elemento HTML.

La mayoría de los elementos HTML tienen una etiqueta de apertura y una etiqueta de cierre.

Las etiquetas de apertura se ven así:

<h1>

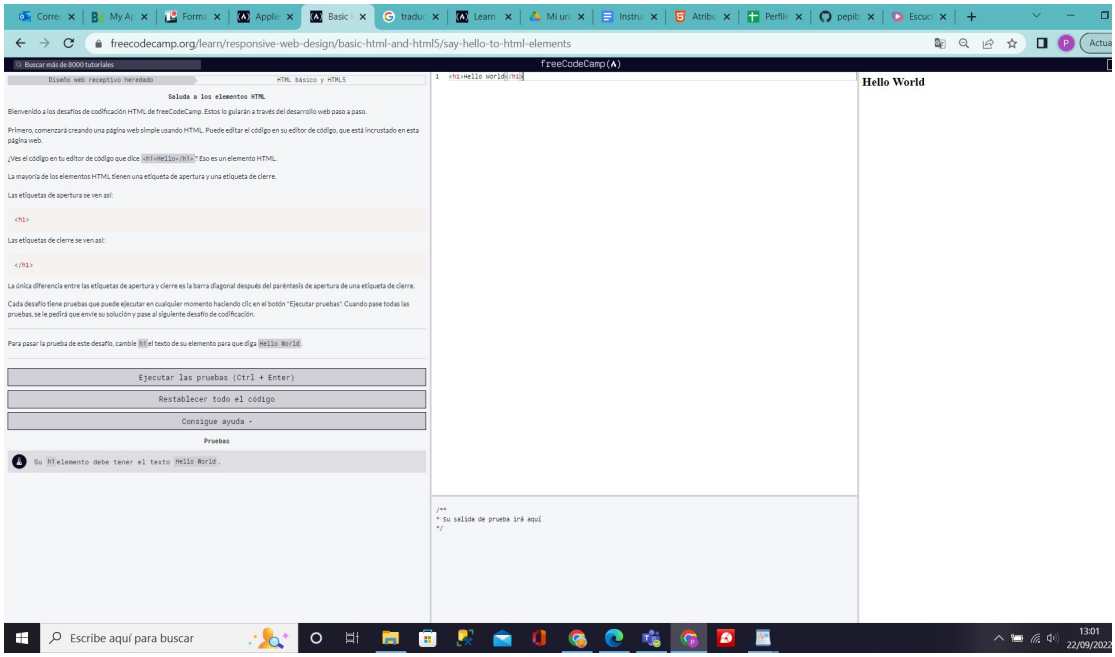
Las etiquetas de cierre se ven así:

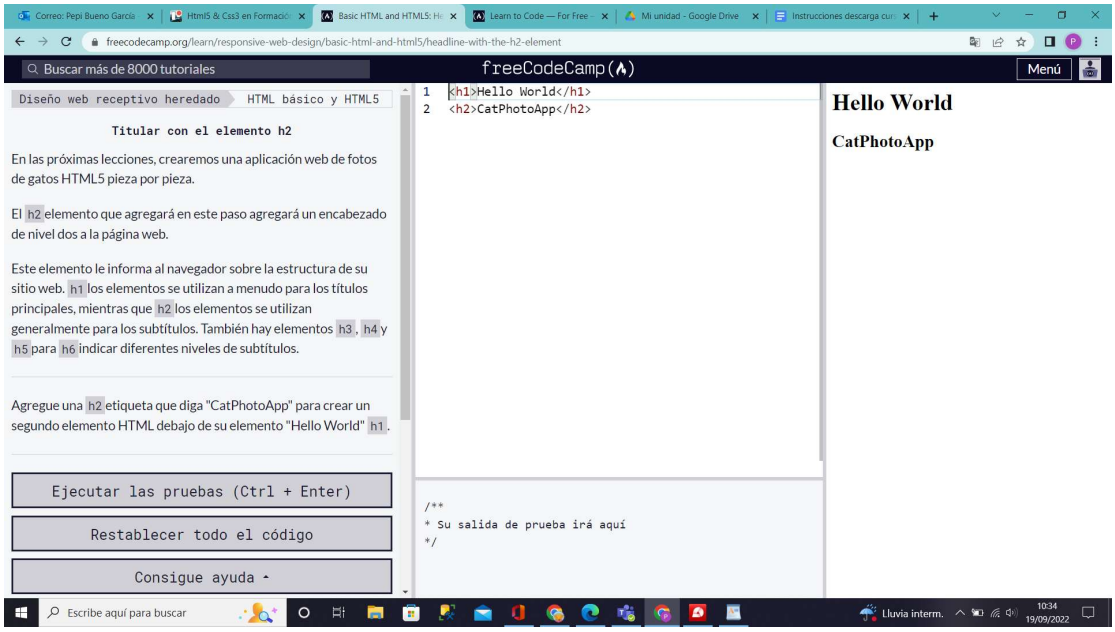
</h1>

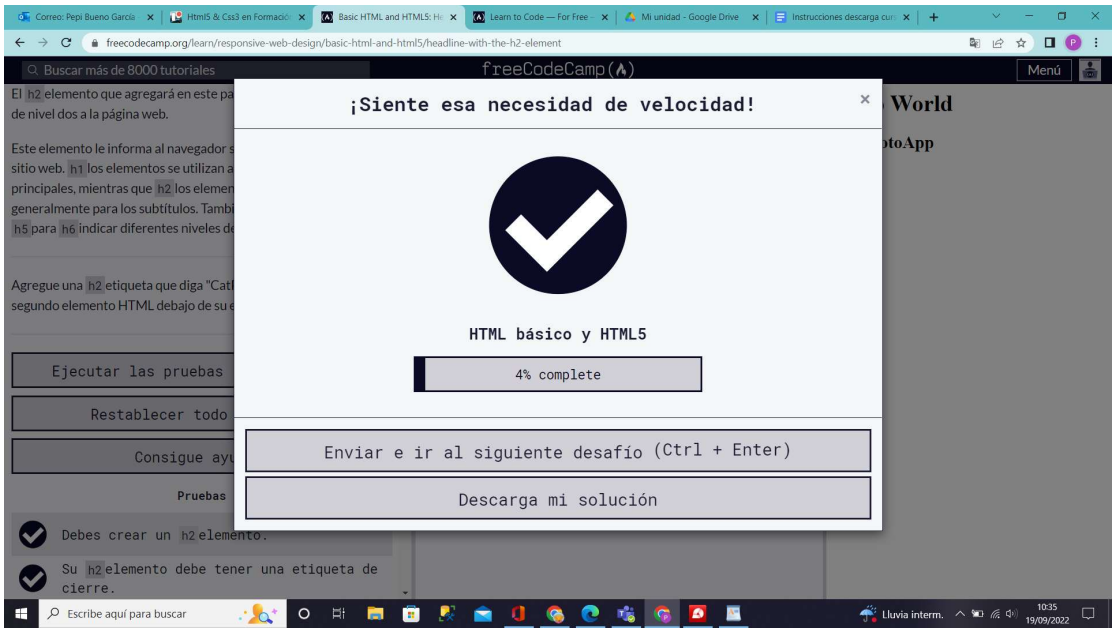
La única diferencia entre las etiquetas de apertura y cierre es la barra diagonal después del paréntesis de apertura de una etiqueta de cierre.

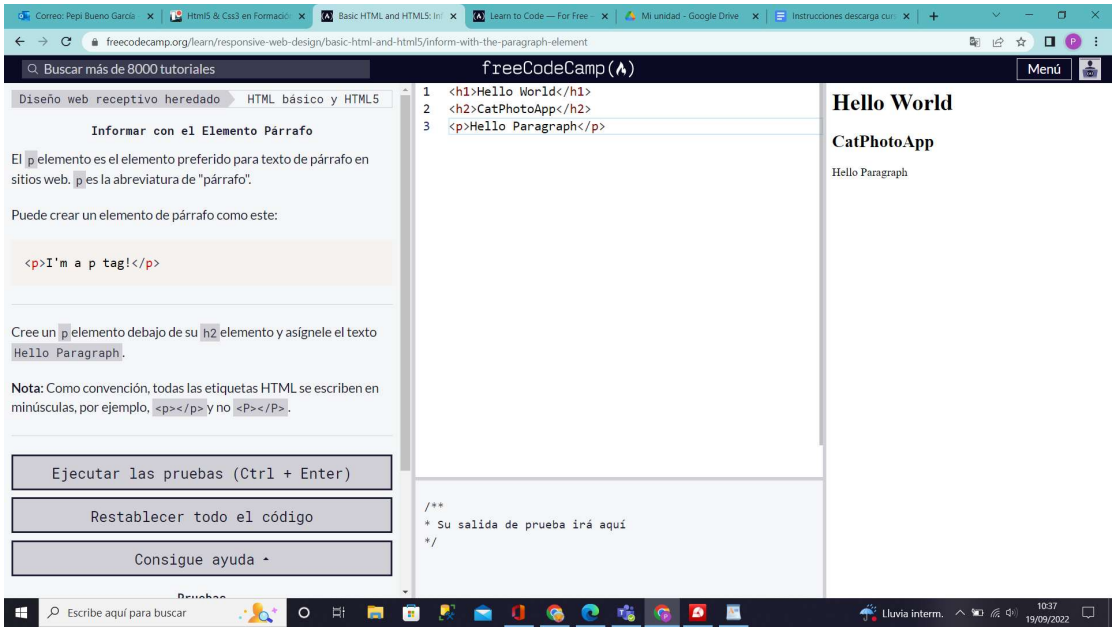
Cada desafío tiene pruebas que puede ejecutar en cualquier momento haciendo clic en el botón "Ejecutar pruebas". Cuando pase todas las pruebas, se le pedirá que envíe su solución y pase al siguiente desafío de codificación.

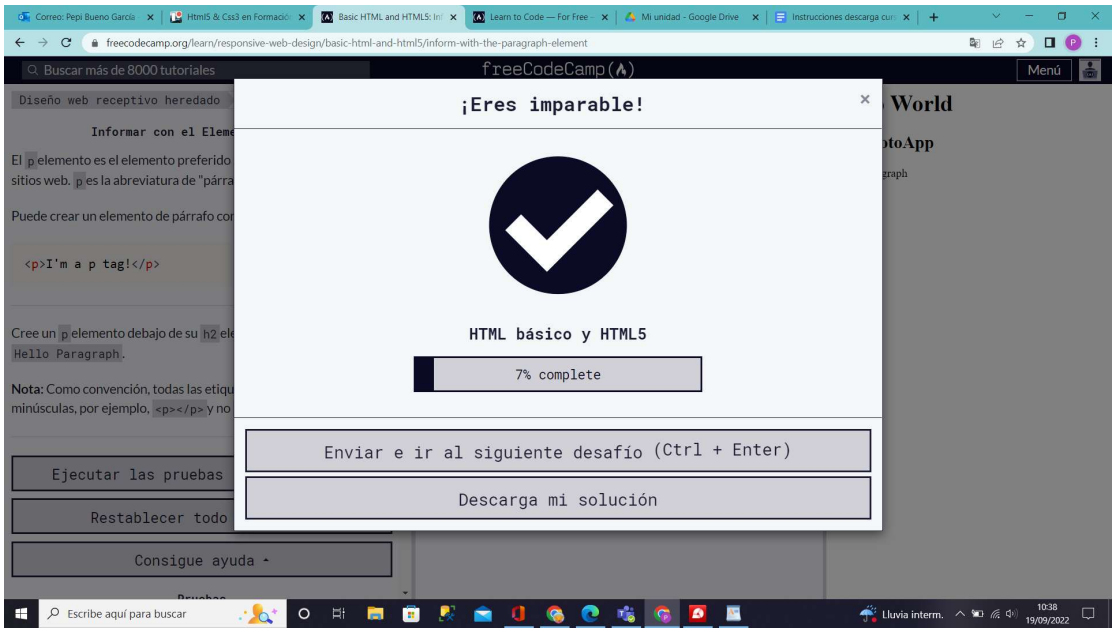
Para pasar la prueba de este desafío, cambie h1el texto de su elemento para que diga Hello World.

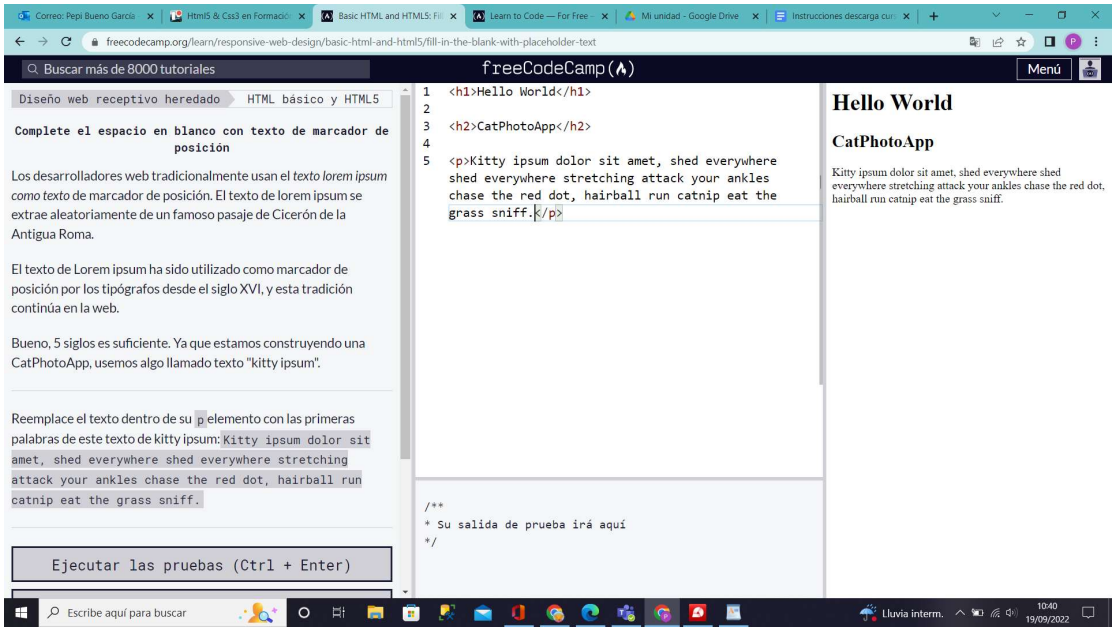


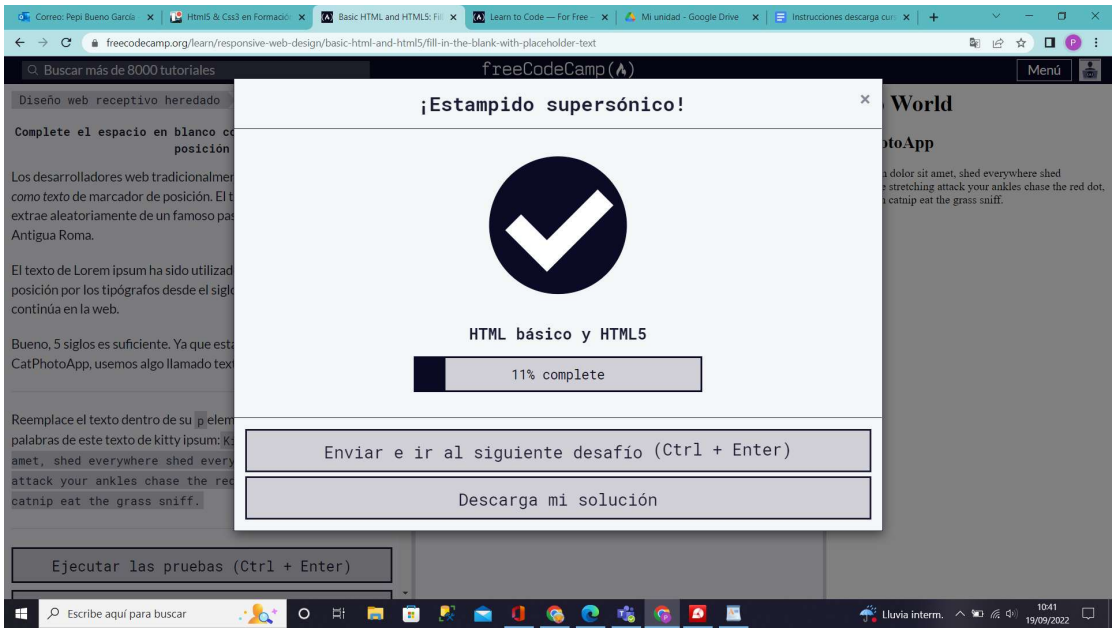

freecodecamp.org/learn/responsive-web-design/basic-html-and-html5/headline-with-the-h2-element
Buscar más de 8000 tutoriales
freeCodeCamp
Menú
Diseño web receptivo heredado
HTML básico y HTML5
Titular con el elemento h2
En las próximas lecciones, crearemos una aplicación web de fotos de gatos HTML5 pieza por pieza.
El h2 elemento que agregará en este paso agregará un encabezado de nivel dos a la página web.
Este elemento le informa al navegador sobre la estructura de su sitio web. h1 los elementos se utilizan a menudo para los títulos principales, mientras que h2 los elementos se utilizan generalmente para los subtítulos. También hay elementos h3, h4 y h5 para h6 indicar diferentes niveles de subtítulos.
Agregue una h2 etiqueta que diga "CatPhotoApp" para crear un segundo elemento HTML debajo de su elemento "Hello World" h1.
Ejecutar las pruebas (Ctrl + Enter)
Restablecer todo el código
Consigue ayuda
1 <h1>Hello World</h1>
2 <h2>CatPhotoApp</h2>
/**
* Su salida de prueba irá aquí
*/
Hello World
CatPhotoApp

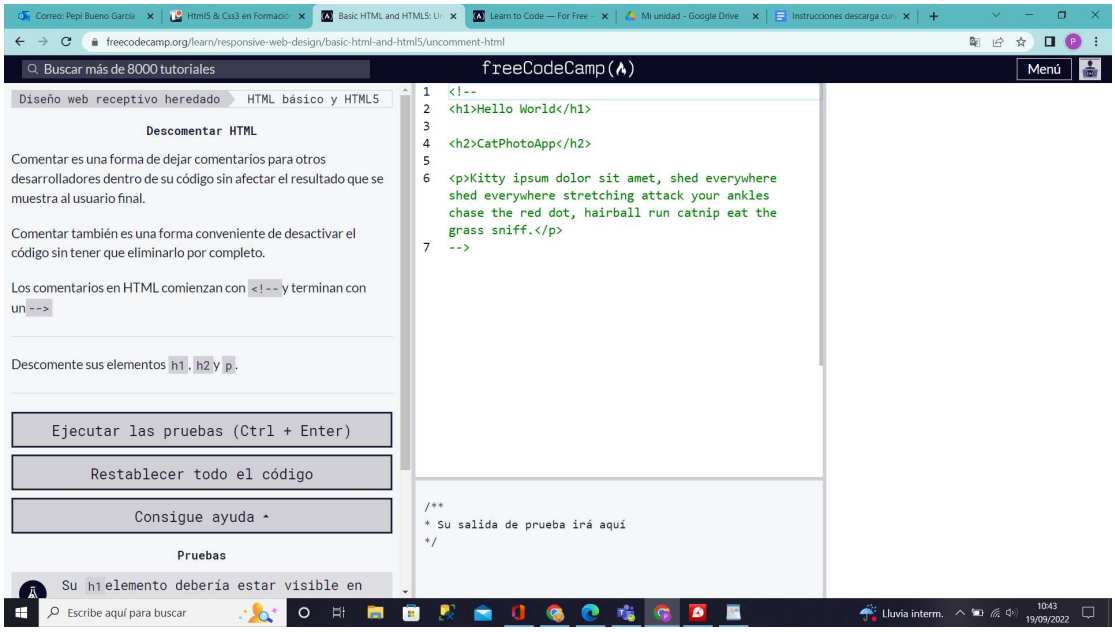

freecodecamp.org/learn/responsive-web-design/basic-html-and-html5/headline-with-the-h2-element
¡Siente esa necesidad de velocidad!
HTML básico y HTML5
4% complete
Enviar e ir al siguiente desafío (Ctrl + Enter)
Descarga mi solución
Pruebas
Debes crear un h2 elemento.
Su h2 elemento debe tener una etiqueta de cierre.

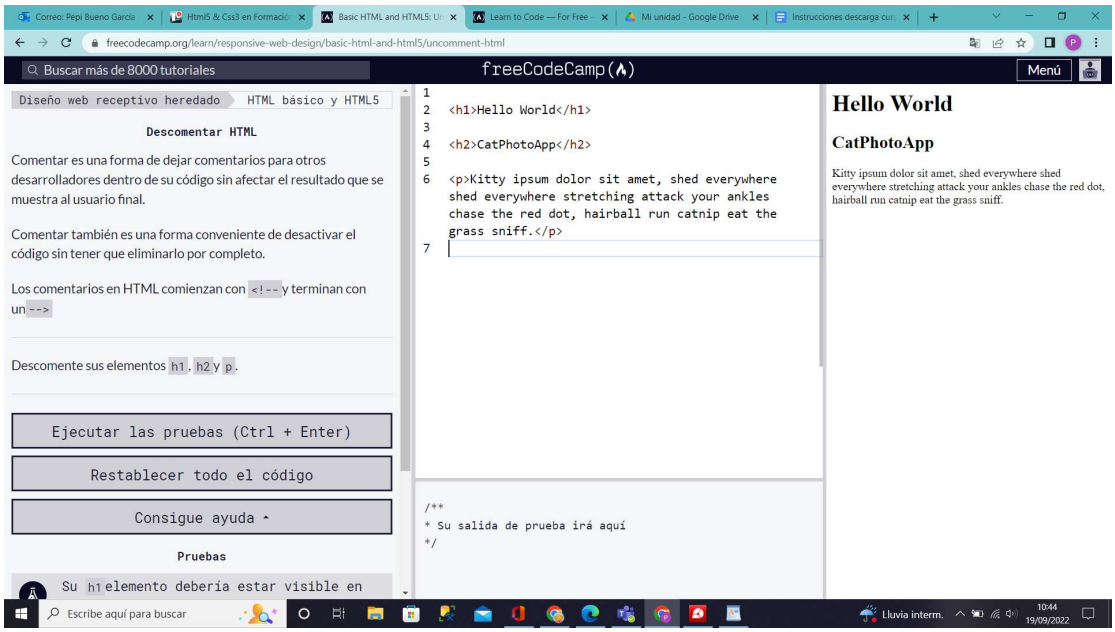

freecodecamp.org/learn/responsive-web-design/basic-html-and-html5/inform-with-the-paragraph-element
Buscar más de 8000 tutoriales
freeCodeCamp
Menú
Diseño web receptivo heredado
HTML básico y HTML5
Informar con el Elemento Párrafo
El p elemento es el elemento preferido para texto de párrafo en sitios web. p es la abreviatura de "párrafo".
Puede crear un elemento de párrafo como este:
I'm a p tag!
Cree un p elemento debajo de su h2 elemento y asígnele el texto Hello Paragraph.
Nota: Como convención, todas las etiquetas HTML se escriben en minúsculas, por ejemplo, y no <P></P>.
Ejecutar las pruebas (Ctrl + Enter)
Restablecer todo el código
Consigue ayuda
1 <h1>Hello World</h1>
2 <h2>CatPhotoApp</h2>
3 Hello Paragraph
/**
* Su salida de prueba irá aquí
*/
Hello World
CatPhotoApp
Hello Paragraph
Escribe aquí para buscar
Lluvia interm.
10:37
19/09/2022

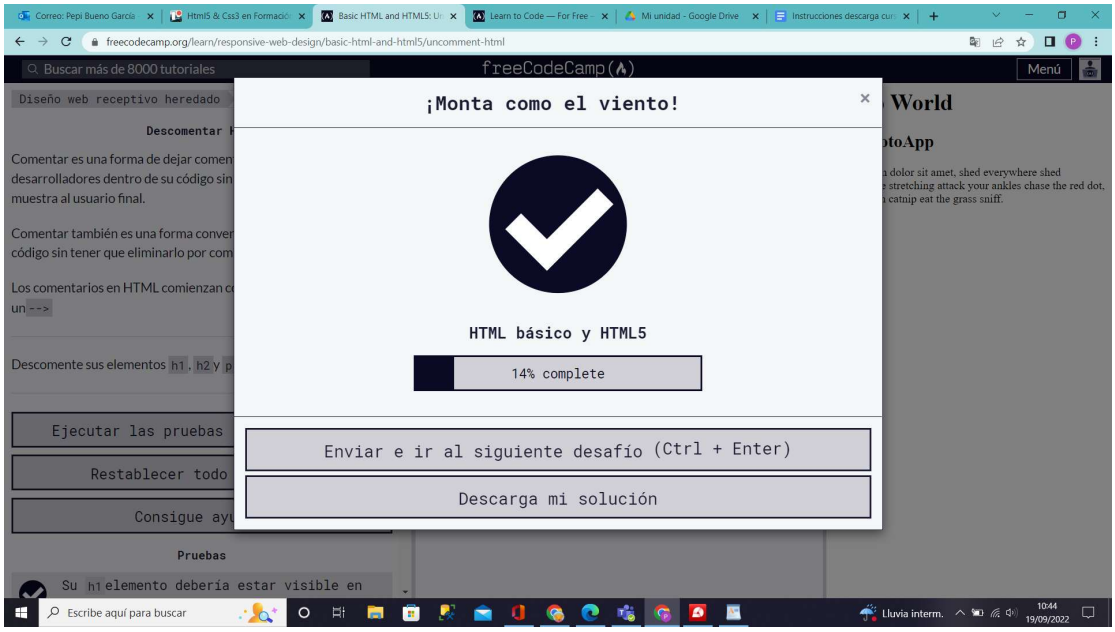

freecodecamp.org/learn/responsive-web-design/basic-html-and-html5/inform-with-the-paragraph-element
¡Eres imparable!
HTML básico y HTML5
7% complete
Enviar e ir al siguiente desafío (Ctrl + Enter)
Descarga mi solución
Consigue ayuda
Escribe aquí para buscar
Lluvia interm.
10:38
19/09/2022

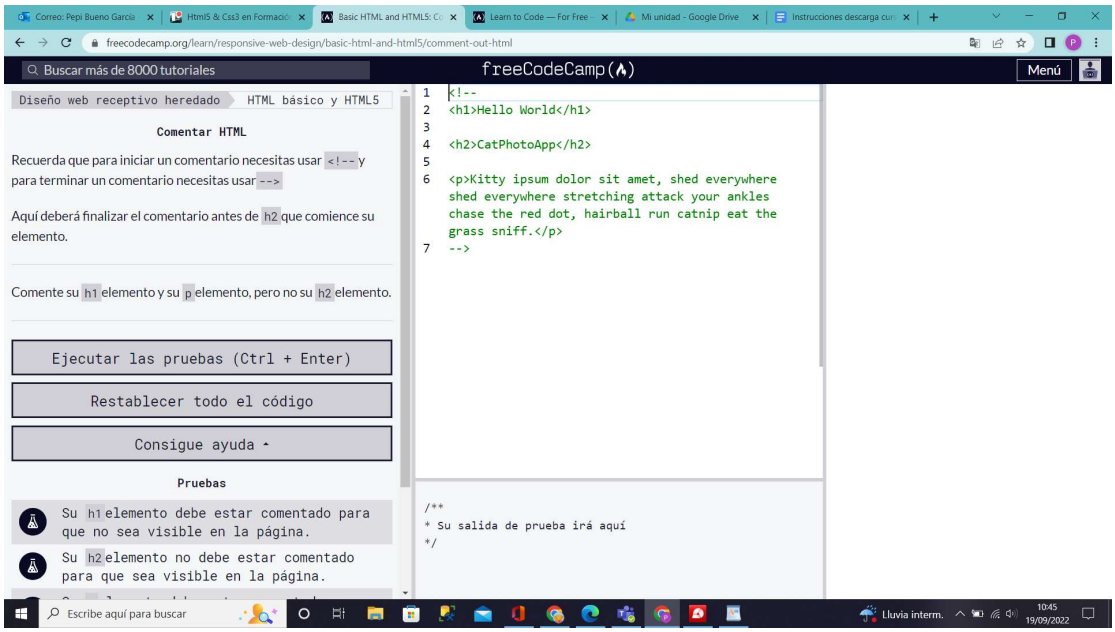

freecodecamp.org/learn/responsive-web-design/basic-html-and-html5/fill-in-the-blank-with-placeholder-text
Buscar más de 8000 tutoriales
freeCodeCamp
Menú
Diseño web receptivo heredado
HTML básico y HTML5
Complete el espacio en blanco con texto de marcador de posición
Los desarrolladores web tradicionalmente usan el texto lorem ipsum como texto de marcador de posición. El texto de lorem ipsum se extrae aleatoriamente de un famoso pasaje de Cicerón de la Antigua Roma.
El texto de Lorem ipsum ha sido utilizado como marcador de posición por los tipógrafos desde el siglo XVI, y esta tradición continúa en la web.
Bueno, 5 siglos es suficiente. Ya que estamos construyendo una CatPhotoApp, usemos algo llamado texto "kitty ipsum".
Reemplace el texto dentro de su p elemento con las primeras palabras de este texto de kitty ipsum: Kitty ipsum dolor sit amet, shed everywhere shed everywhere stretching attack your ankles chase the red dot, hairball run catnip eat the grass sniff.
Ejecutar las pruebas (Ctrl + Enter)
1 <h1>Hello World</h1>
2
3 <h2>CatPhotoApp</h2>
4
5 Kitty ipsum dolor sit amet, shed everywhere shed everywhere stretching attack your ankles chase the red dot, hairball run catnip eat the grass sniff.
/**
* Su salida de prueba irá aquí
*/
Hello World
CatPhotoApp
Kitty ipsum dolor sit amet, shed everywhere shed everywhere stretching attack your ankles chase the red dot, hairball run catnip eat the grass sniff.

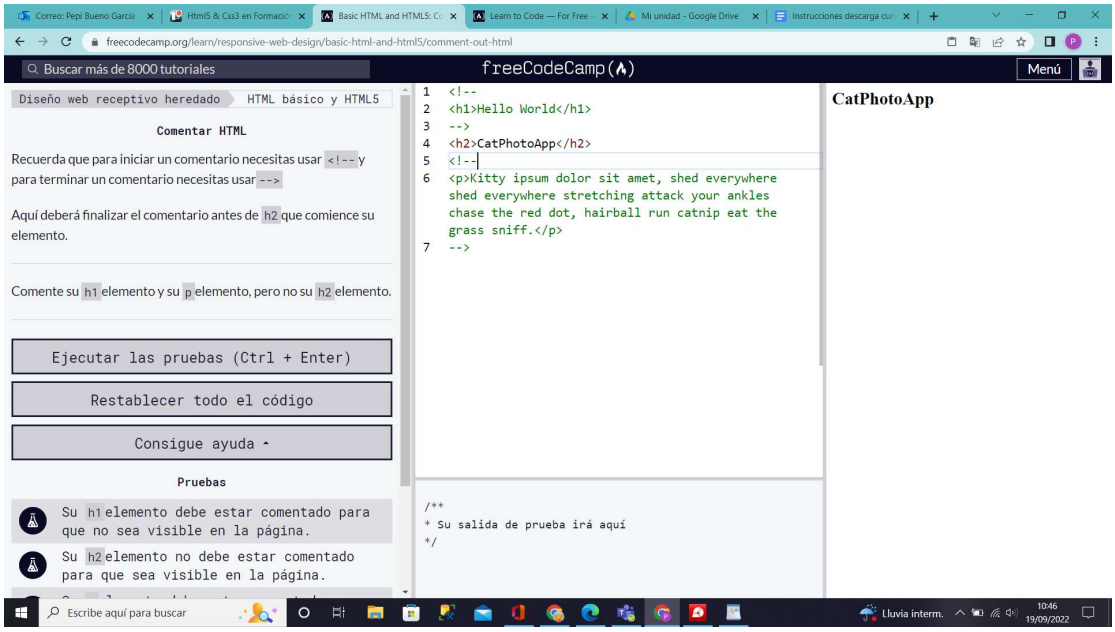

¡Estampido supersónico!
HTML básico y HTML5
11% complete
Enviar e ir al siguiente desafío (Ctrl + Enter)
Descarga mi solución

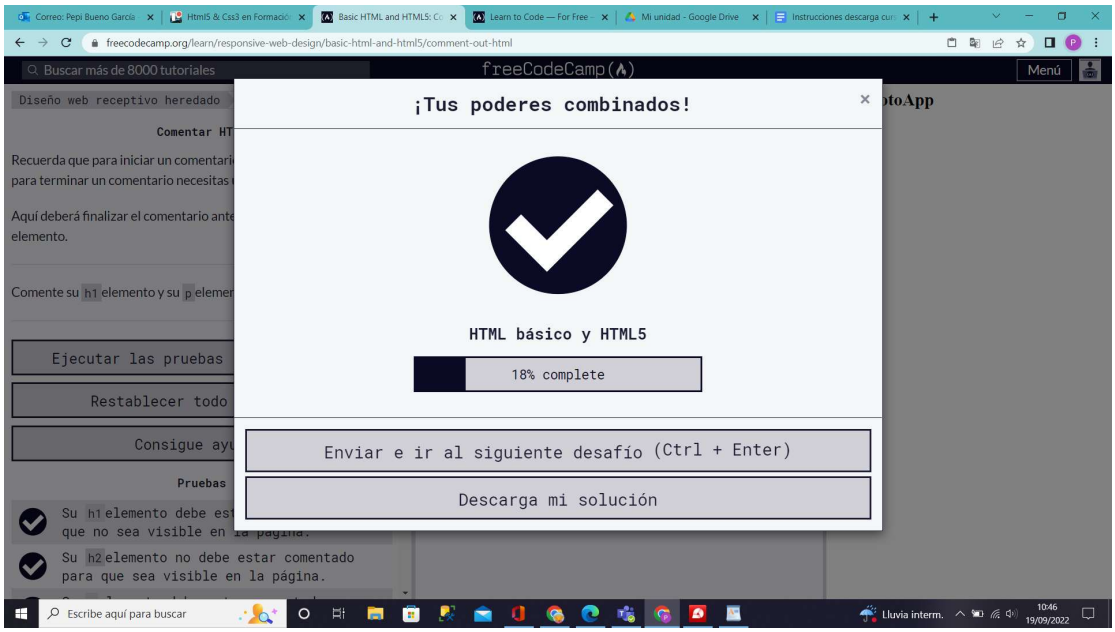

freecodecamp.org/learn/responsive-web-design/basic-html-and-html5/uncomment-html
Buscar más de 8000 tutoriales
freeCodeCamp
Menú
Diseño web receptivo heredado
HTML básico y HTML5
Descomentar HTML
Comentar es una forma de dejar comentarios para otros desarrolladores dentro de su código sin afectar el resultado que se muestra al usuario final.
Comentar también es una forma conveniente de desactivar el código sin tener que eliminarlo por completo.
Los comentarios en HTML comienzan con <!-- y terminan con un -->
Descomente sus elementos h1, h2 y p.
Ejecutar las pruebas (Ctrl + Enter)
Restablecer todo el código
Consigue ayuda
Pruebas
Su h1 elemento debería estar visible en
<!--
<h1>Hello World</h1>
<h2>CatPhotoApp</h2>
Kitty ipsum dolor sit amet, shed everywhere shed everywhere stretching attack your ankles chase the red dot, hairball run catnip eat the grass sniff.
-->
/**
* Su salida de prueba irá aquí
*/

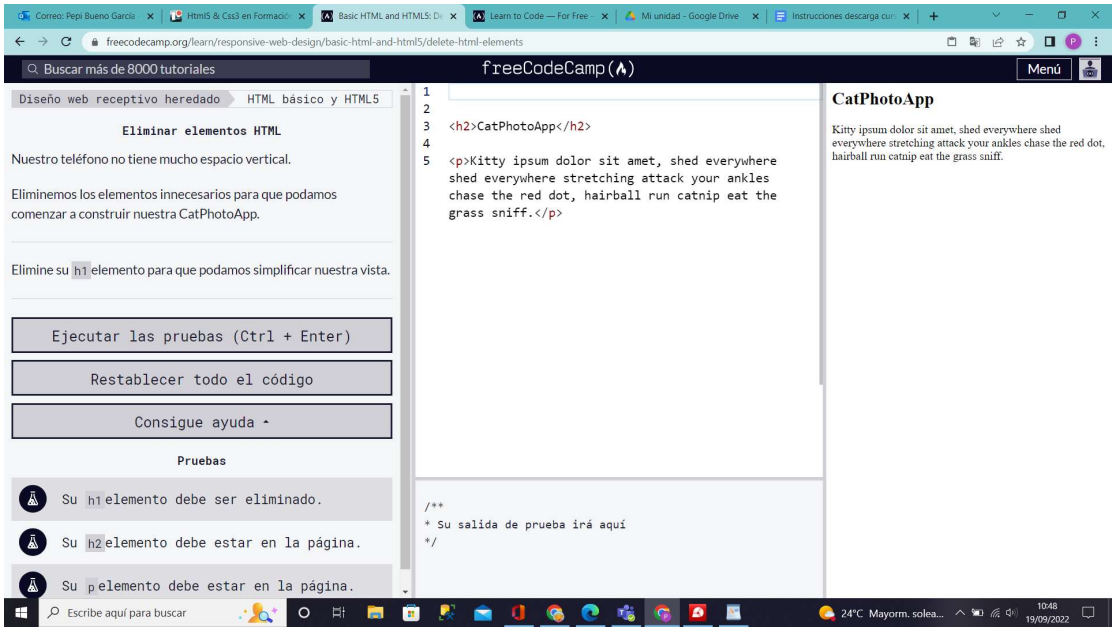

freecodecamp.org/learn/responsive-web-design/basic-html-and-html5/uncomment-html
Buscar más de 8000 tutoriales
freeCodeCamp
Menú
Diseño web receptivo heredado
HTML básico y HTML5
Descomentar HTML
Comentar es una forma de dejar comentarios para otros desarrolladores dentro de su código sin afectar el resultado que se muestra al usuario final.
Comentar también es una forma conveniente de desactivar el código sin tener que eliminarlo por completo.
Los comentarios en HTML comienzan con <!-- y terminan con un -->
Descomente sus elementos h1, h2 y p.
Ejecutar las pruebas (Ctrl + Enter)
Restablecer todo el código
Consigue ayuda
Pruebas
Su h1 elemento debería estar visible en
<h1>Hello World</h1>
<h2>CatPhotoApp</h2>
Kitty ipsum dolor sit amet, shed everywhere shed everywhere stretching attack your ankles chase the red dot, hairball run catnip eat the grass sniff.
/**
* Su salida de prueba irá aquí
*/
Hello World
CatPhotoApp
Kitty ipsum dolor sit amet, shed everywhere shed everywhere stretching attack your ankles chase the red dot, hairball run catnip eat the grass sniff.

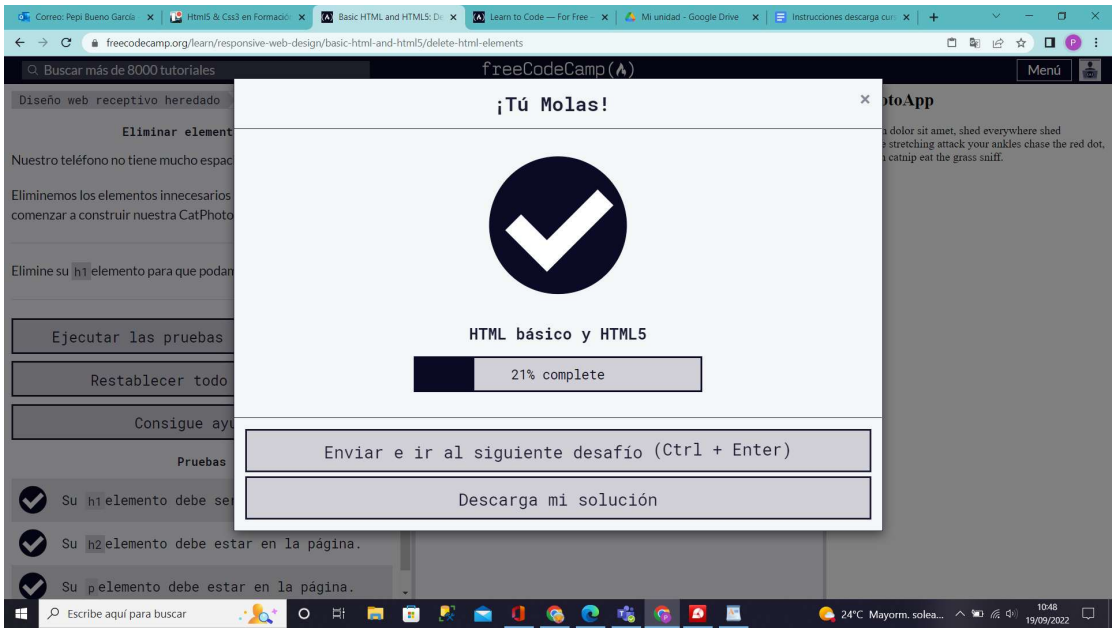

freecodecamp.org/learn/responsive-web-design/basic-html-and-html5/uncomment-html
Buscar más de 8000 tutoriales
freeCodeCamp
Menú
Diseño web receptivo heredado
¡Monta como el viento!
HTML básico y HTML5
14% complete
Enviar e ir al siguiente desafío (Ctrl + Enter)
Descarga mi solución
Ejecutar las pruebas
Restablecer todo
Consigue ayu
Pruebas
Su h1 elemento debería estar visible en
Escribe aquí para buscar
Lluvia interm.
10:44
19/09/2022

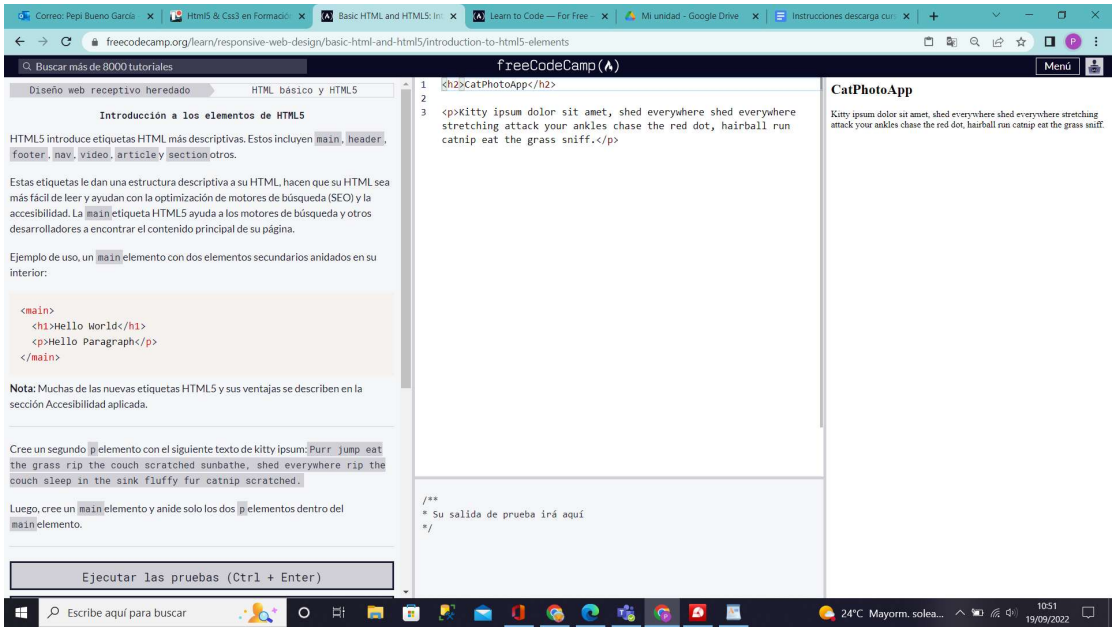

freecodecamp.org/learn/responsive-web-design/basic-html-and-html5/comment-out-html
Buscar más de 8000 tutoriales
freeCodeCamp
Menú
Diseño web receptivo heredado
HTML básico y HTML5
Comentar HTML
Recuerda que para iniciar un comentario necesitas usar <!-- y para terminar un comentario necesitas usar -->
Aquí deberá finalizar el comentario antes de h2 que comience su elemento.
Comente su h1 elemento y su p elemento, pero no su h2 elemento.
Ejecutar las pruebas (Ctrl + Enter)
Restablecer todo el código
Consigue ayuda
Pruebas
Su h1 elemento debe estar comentado para que no sea visible en la página.
Su h2 elemento no debe estar comentado para que sea visible en la página.
<!--
<h1>Hello World</h1>
<h2>CatPhotoApp</h2>
Kitty ipsum dolor sit amet, shed everywhere shed everywhere stretching attack your ankles chase the red dot, hairball run catnip eat the grass sniff.
-->
/**
* Su salida de prueba irá aquí
*/
Escribe aquí para buscar
Lluvia interm.
10:45
19/09/2022

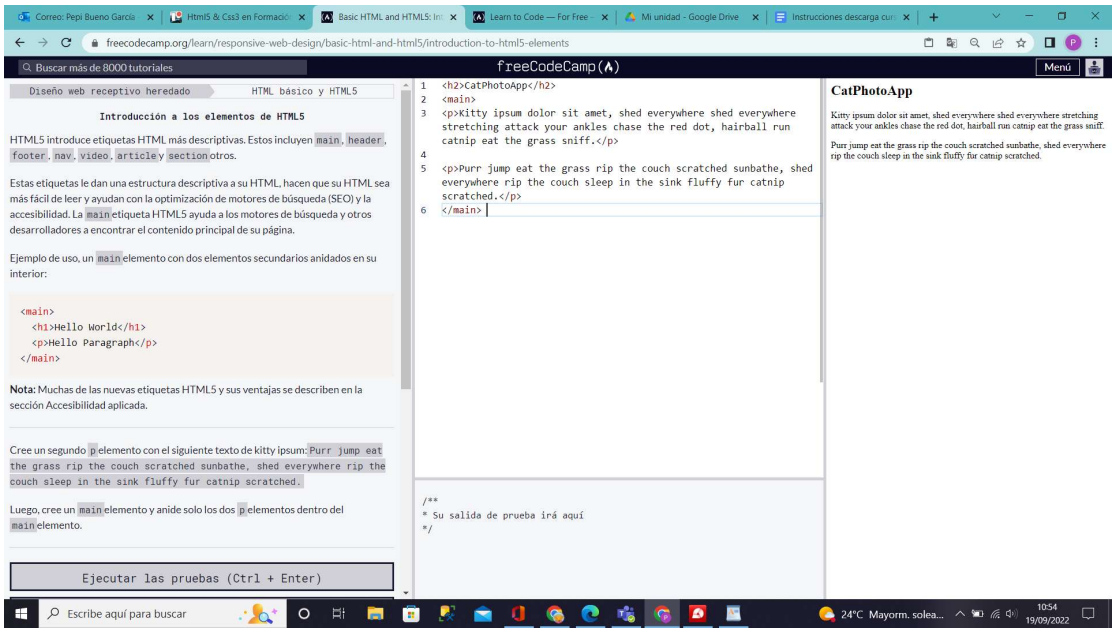

freecodecamp.org/learn/responsive-web-design/basic-html-and-html5/comment-out-html
Buscar más de 8000 tutoriales
freeCodeCamp
Menú
Diseño web receptivo heredado
HTML básico y HTML5
Comentar HTML
Recuerda que para iniciar un comentario necesitas usar <!-- y para terminar un comentario necesitas usar -->
Aquí deberá finalizar el comentario antes de h2 que comience su elemento.
Comente su h1 elemento y su p elemento, pero no su h2 elemento.
Ejecutar las pruebas (Ctrl + Enter)
Restablecer todo el código
Consigue ayuda
Pruebas
Su h1 elemento debe estar comentado para que no sea visible en la página.
Su h2 elemento no debe estar comentado para que sea visible en la página.
<!--
<h1>Hello World</h1>
-->
<h2>CatPhotoApp</h2>
<!--
Kitty ipsum dolor sit amet, shed everywhere shed everywhere stretching attack your ankles chase the red dot, hairball run catnip eat the grass sniff.
-->
/**
* Su salida de prueba irá aquí
*/
CatPhotoApp

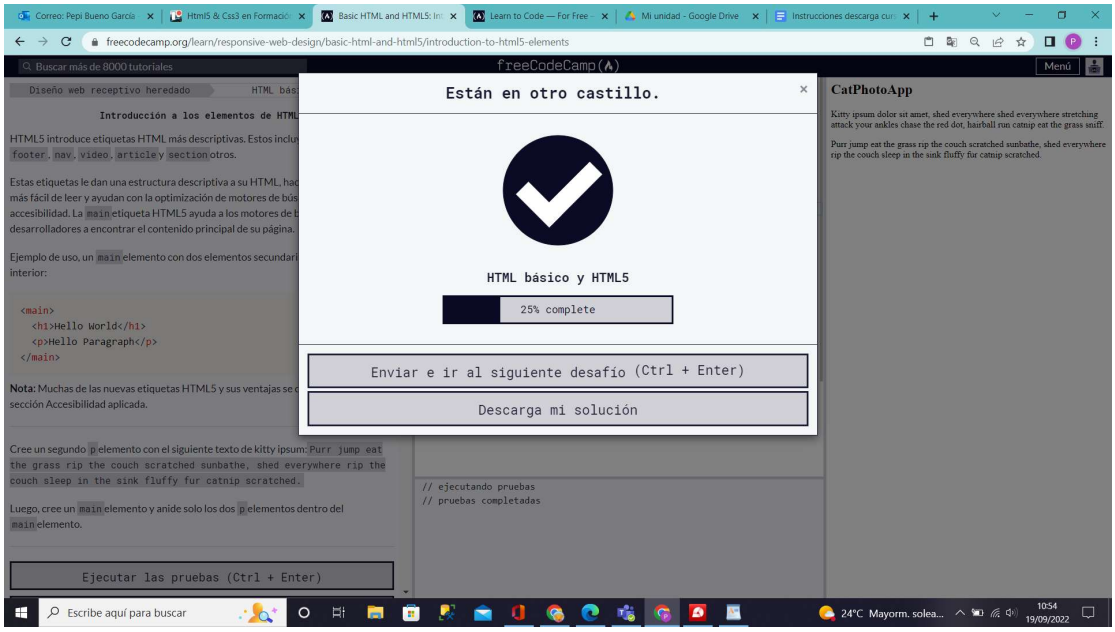

freecodecamp.org/learn/responsive-web-design/basic-html-and-html5/comment-out-html
¡Tus poderes combinados!
HTML básico y HTML5
18% complete
Enviar e ir al siguiente desafío (Ctrl + Enter)
Descarga mi solución

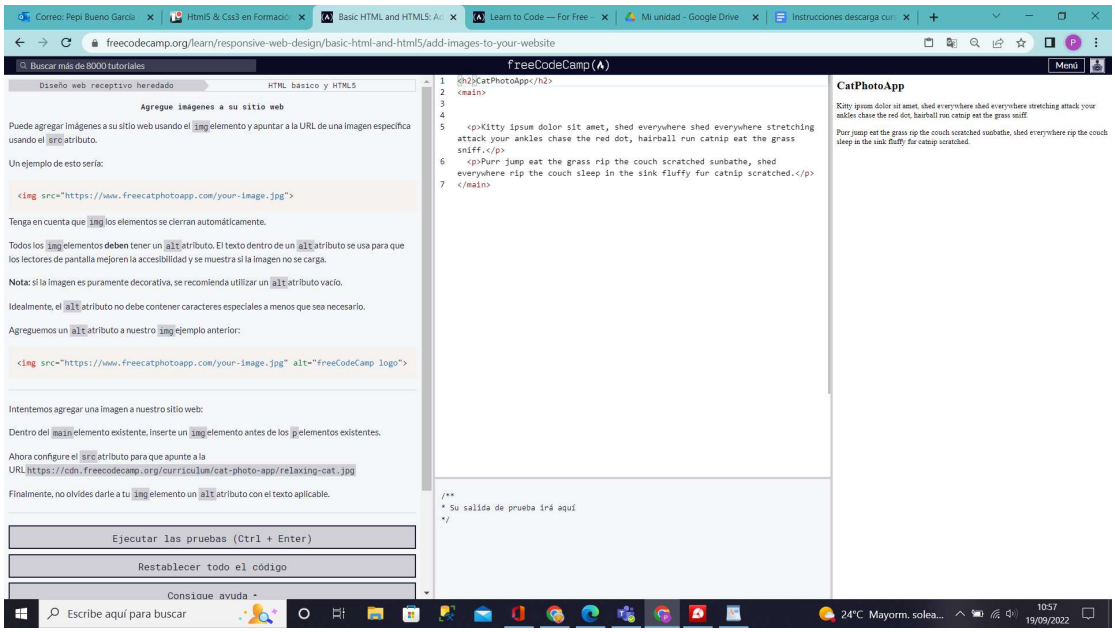

freecodecamp.org/learn/responsive-web-design/basic-html-and-html5/delete-html-elements
Buscar más de 8000 tutoriales
freeCodeCamp
Menú
Diseño web receptivo heredado
HTML básico y HTML5
Eliminar elementos HTML
Nuestro teléfono no tiene mucho espacio vertical.
Eliminemos los elementos innecesarios para que podamos comenzar a construir nuestra CatPhotoApp.
Elimine su h1 elemento para que podamos simplificar nuestra vista.
Ejecutar las pruebas (Ctrl + Enter)
Restablecer todo el código
Consigue ayuda
Pruebas
Su h1 elemento debe ser eliminado.
Su h2 elemento debe estar en la página.
Su p elemento debe estar en la página.
1
2
3 <h2>CatPhotoApp</h2>
4
5 Kitty ipsum dolor sit amet, shed everywhere shed everywhere stretching attack your ankles chase the red dot, hairball run catnip eat the grass sniff.
/**
* Su salida de prueba irá aquí
*/
CatPhotoApp
Kitty ipsum dolor sit amet, shed everywhere shed everywhere stretching attack your ankles chase the red dot, hairball run catnip eat the grass sniff.

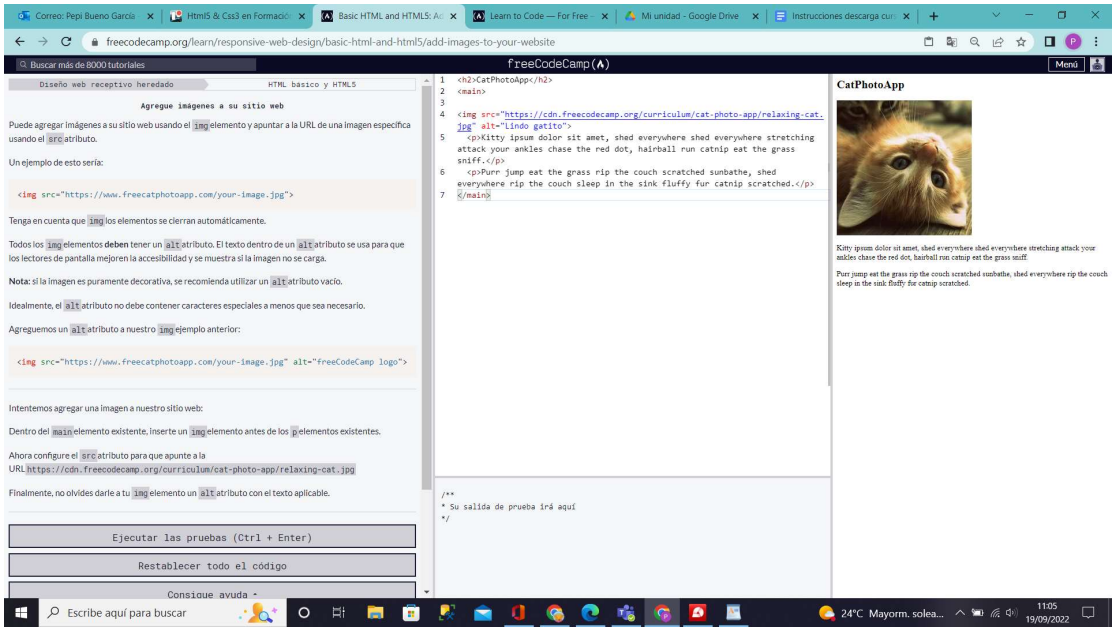

freecodecamp.org/learn/responsive-web-design/basic-html-and-html5/delete-html-elements
¡Tú Molas!
HTML básico y HTML5
21% complete
Enviar e ir al siguiente desafío (Ctrl + Enter)
Descarga mi solución

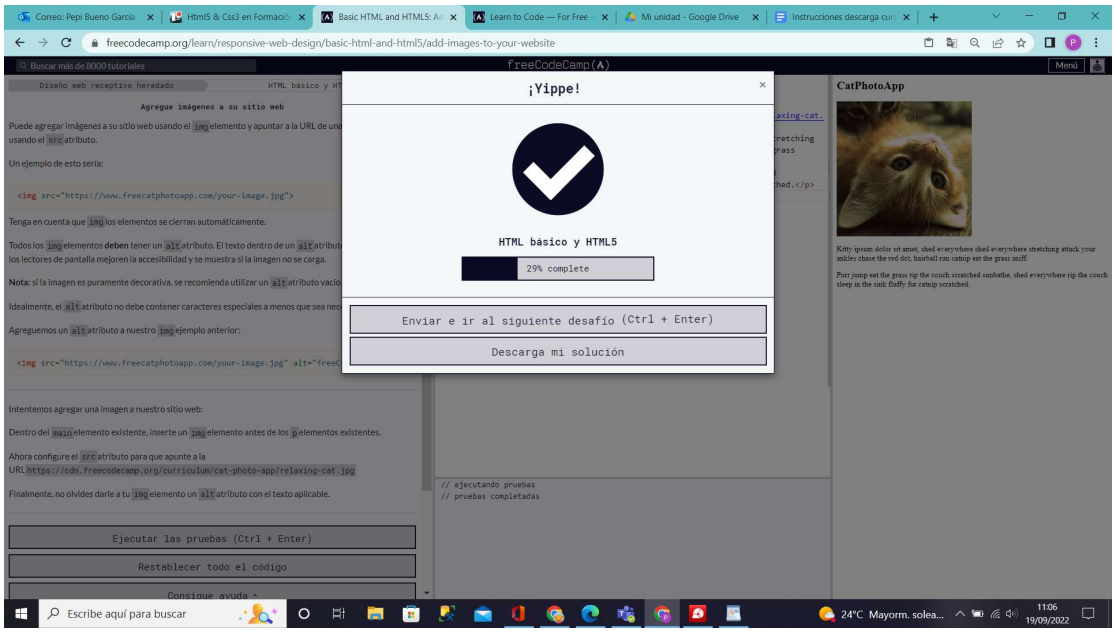

freecodecamp.org/learn/responsive-web-design/basic-html-and-html5/introduction-to-html5-elements
Buscar más de 8000 tutoriales
freeCodeCamp
Menú
Diseño web receptivo heredado
HTML básico y HTML5
Introducción a los elementos de HTML5
HTML5 introduce etiquetas HTML más descriptivas. Estos incluyen main, header, footer, nav, video, article y section otros.
Estas etiquetas le dan una estructura descriptiva a su HTML, hacen que su HTML sea más fácil de leer y ayudan con la optimización de motores de búsqueda (SEO) y la accesibilidad. La main etiqueta HTML5 ayuda a los motores de búsqueda y otros desarrolladores a encontrar el contenido principal de su página.
Ejemplo de uso, un main elemento con dos elementos secundarios anidados en su interior:
<main>
<h1>Hello World</h1>
Hello Paragraph
</main>
Nota: Muchas de las nuevas etiquetas HTML5 y sus ventajas se describen en la sección Accesibilidad aplicada.
Cree un segundo p elemento con el siguiente texto de kitty ipsum: Purr jump eat the grass rip the couch scratched sunbathe, shed everywhere rip the couch sleep in the sink fluffy fur catnip scratched.
Luego, cree un main elemento y anide solo los dos p elementos dentro del main elemento.
Ejecutar las pruebas (Ctrl + Enter)
<h2>CatPhotoApp</h2>
Kitty ipsum dolor sit amet, shed everywhere shed everywhere stretching attack your ankles chase the red dot, hairball run catnip eat the grass sniff.
/**
* Su salida de prueba irá aquí
*/
CatPhotoApp
Kitty ipsum dolor sit amet, shed everywhere shed everywhere stretching attack your ankles chase the red dot, hairball run catnip eat the grass sniff.
Escribe aquí para buscar
24°C Mayorm. solea...
10:51
19/09/2022

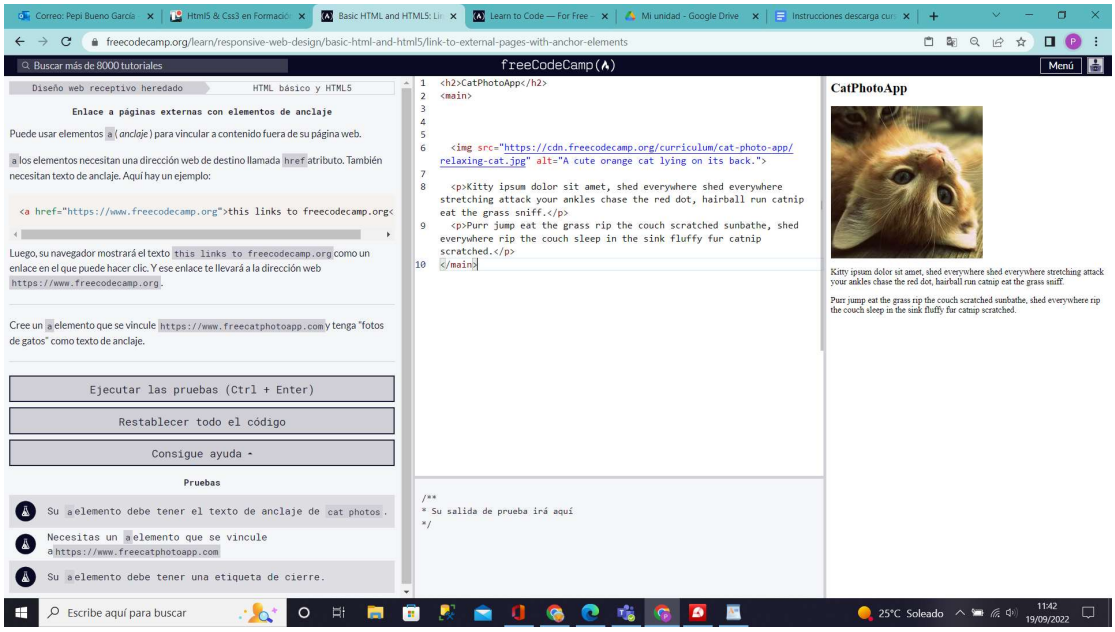

freecodecamp.org/learn/responsive-web-design/basic-html-and-html5/introduction-to-html5-elements
Buscar más de 8000 tutoriales
freeCodeCamp
Menú
Diseño web receptivo heredado
HTML básico y HTML5
Introducción a los elementos de HTML5
HTML5 introduce etiquetas HTML más descriptivas. Estos incluyen main, header, footer, nav, video, article y section otros.
Estas etiquetas le dan una estructura descriptiva a su HTML, hacen que su HTML sea más fácil de leer y ayudan con la optimización de motores de búsqueda (SEO) y la accesibilidad. La main etiqueta HTML5 ayuda a los motores de búsqueda y otros desarrolladores a encontrar el contenido principal de su página.
Ejemplo de uso, un main elemento con dos elementos secundarios anidados en su interior:
<main>
<h1>Hello World</h1>
Hello Paragraph
</main>
Nota: Muchas de las nuevas etiquetas HTML5 y sus ventajas se describen en la sección Accesibilidad aplicada.
Cree un segundo p elemento con el siguiente texto de kitty ipsum: Purr jump eat the grass rip the couch scratched sunbathe, shed everywhere rip the couch sleep in the sink fluffy fur catnip scratched.
Luego, cree un main elemento y anide solo los dos p elementos dentro del main elemento.
Ejecutar las pruebas (Ctrl + Enter)
<h2>CatPhotoApp</h2>
<main>
Kitty ipsum dolor sit amet, shed everywhere shed everywhere stretching attack your ankles chase the red dot, hairball run catnip eat the grass sniff.
Purr jump eat the grass rip the couch scratched sunbathe, shed everywhere rip the couch sleep in the sink fluffy fur catnip scratched.
</main>
/**
* Su salida de prueba irá aquí
*/
CatPhotoApp
Kitty ipsum dolor sit amet, shed everywhere shed everywhere stretching attack your ankles chase the red dot, hairball run catnip eat the grass sniff.
Purr jump eat the grass rip the couch scratched sunbathe, shed everywhere rip the couch sleep in the sink fluffy fur catnip scratched.
Escribe aquí para buscar
24°C Mayorm. solea...
10:54
19/09/2022

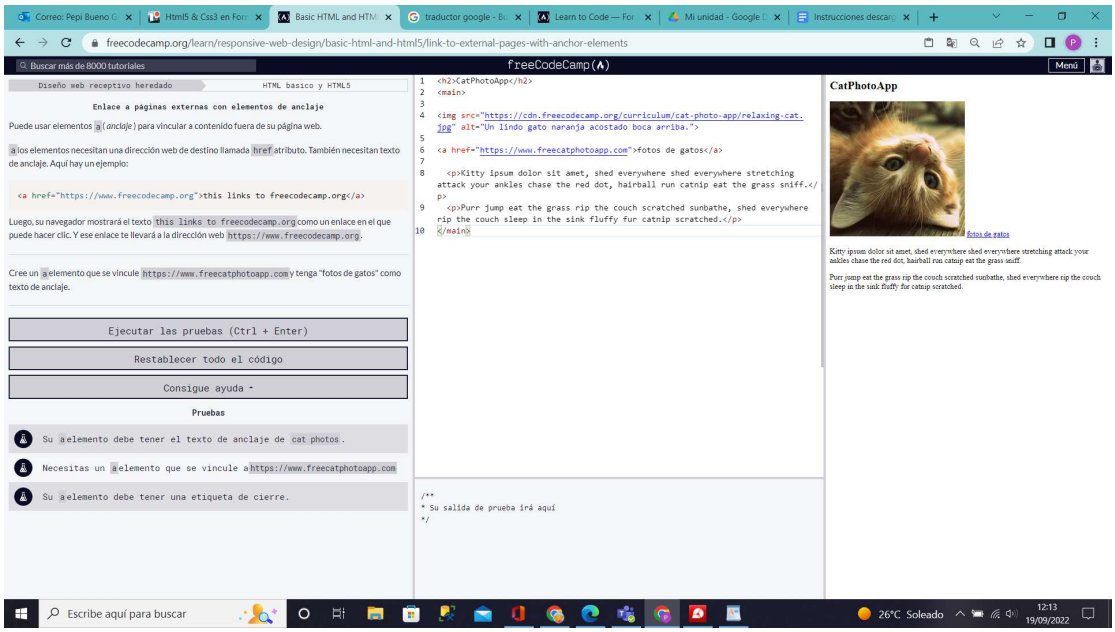

freecodecamp.org/learn/responsive-web-design/basic-html-and-html5/introduction-to-html5-elements
freeCodeCamp
Están en otro castillo.
HTML básico y HTML5
25% complete
Enviar e ir al siguiente desafío (Ctrl + Enter)
Descarga mi solución
CatPhotoApp
Ejecutar las pruebas (Ctrl + Enter)


freecodecamp.org/learn/responsive-web-design/basic-html-and-html5/add-images-to-your-website
freeCodeCamp
Agregue imágenes a su sitio web
CatPhotoApp
Ejecutar las pruebas (Ctrl + Enter)
Restablecer todo el código


freecodecamp.org/learn/responsive-web-design/basic-html-and-html5/add-images-to-your-website
freeCodeCamp
Diseño web receptivo heredado
HTML basico y HTML5
Agregue imágenes a su sitio web
Puede agregar Imágenes a su sitio web usando el img elemento y apuntar a la URL de una imagen específica usando el src atributo.
Un ejemplo de esto sería:
<img src="https://www.freecatphotoapp.com/your-image.jpg">
Tenga en cuenta que img los elementos se cierran automáticamente.
Todos los img elementos deben tener un alt atributo. El texto dentro de un alt atributo se usa para que los lectores de pantalla mejoren la accesibilidad y se muestra si la imagen no se carga.
Nota: si la imagen es puramente decorativa, se recomienda utilizar un alt atributo vacío.
Idealmente, el alt atributo no debe contener caracteres especiales a menos que sea necesario.
Agreguemos un alt atributo a nuestro img ejemplo anterior:
<img src="https://www.freecatphotoapp.com/your-image.jpg" alt="freeCodeCamp logo">
Intentemos agregar una imagen a nuestro sitio web:
Dentro del main elemento existente, inserte un img elemento antes de los p elementos existentes.
Ahora configure el src atributo para que apunte a la URL https://cdn.freecodecamp.org/curriculum/cat-photo-app/relaxing-cat.jpg
Finalmente, no olvides darle a tu img elemento un alt atributo con el texto aplicable.
Ejecutar las pruebas (Ctrl + Enter)
Restablecer todo el código
Consigue ayuda
<h2>CatPhotoApp</h2>
<main>
<img src="https://cdn.freecodecamp.org/curriculum/cat-photo-app/relaxing-cat.jpg" alt="Lindo gatito">
Kitty ipsum dolor sit amet, shed everywhere shed everywhere stretching attack your ankles chase the red dot, hairball run catnip eat the grass sniff.
Purr jump eat the grass rip the couch scratched sunbathe, shed everywhere rip the couch sleep in the sink fluffy fur catnip scratched.
</main>
/**
* Su salida de prueba irá aquí
*/
CatPhotoApp
Kitty ipsum dolor sit amet, shed everywhere shed everywhere stretching attack your ankles chase the red dot, hairball run catnip eat the grass sniff.
Purr jump eat the grass rip the couch scratched sunbathe, shed everywhere rip the couch sleep in the sink fluffy fur catnip scratched.
Escribe aquí para buscar
24°C Mayorm. solea...
11:05
19/09/2022


freecodecamp.org/learn/responsive-web-design/basic-html-and-html5/add-images-to-your-website
freeCodeCamp
¡Yippe!
HTML básico y HTML5
29% complete
Enviar e ir al siguiente desafío (Ctrl + Enter)
Descarga mi solución
// ejecutando pruebas
// pruebas completadas
CatPhotoApp
Escribe aquí para buscar
24°C Mayorm. solea...
11:06
19/09/2022


freecodecamp.org/learn/responsive-web-design/basic-html-and-html5/link-to-external-pages-with-anchor-elements
freeCodeCamp
Diseño web receptivo heredado
HTML básico y HTML5
Enlace a páginas externas con elementos de anclaje
Puede usar elementos a (anclaje) para vincular a contenido fuera de su página web.
a los elementos necesitan una dirección web de destino llamada href atributo. También necesitan texto de anclaje. Aquí hay un ejemplo:
<a href="https://www.freecodecamp.org">this links to freecodecamp.org<
Luego, su navegador mostrará el texto this links to freecodecamp.org como un enlace en el que puede hacer clic. Y ese enlace te llevará a la dirección web https://www.freecodecamp.org.
Cree un a elemento que se vincule https://www.freecatphotoapp.com y tenga "fotos de gatos" como texto de anclaje.
Ejecutar las pruebas (Ctrl + Enter)
Restablecer todo el código
Consigue ayuda
Pruebas
Su a elemento debe tener el texto de anclaje de cat photos.
Necesitas un a elemento que se vincule a https://www.freecatphotoapp.com
Su a elemento debe tener una etiqueta de cierre.
<h2>CatPhotoApp</h2>
<main>
<img src="https://cdn.freecodecamp.org/curriculum/cat-photo-app/relaxing-cat.jpg" alt="A cute orange cat lying on its back.">
Kitty ipsum dolor sit amet, shed everywhere shed everywhere stretching attack your ankles chase the red dot, hairball run catnip eat the grass sniff.
Purr jump eat the grass rip the couch scratched sunbathe, shed everywhere rip the couch sleep in the sink fluffy fur catnip scratched.
</main>
/**
* Su salida de prueba irá aquí
*/
CatPhotoApp
Kitty ipsum dolor sit amet, shed everywhere shed everywhere stretching attack your ankles chase the red dot, hairball run catnip eat the grass sniff.
Purr jump eat the grass rip the couch scratched sunbathe, shed everywhere rip the couch sleep in the sink fluffy fur catnip scratched.


freecodecamp.org/learn/responsive-web-design/basic-html-and-html5/link-to-external-pages-with-anchor-elements
freeCodeCamp
Diseño web receptivo heredado
HTML básico y HTML5
Enlace a páginas externas con elementos de anclaje
Puede usar elementos a (anclaje) para vincular a contenido fuera de su página web.
a los elementos necesitan una dirección web de destino llamada href atributo. También necesitan texto de anclaje. Aquí hay un ejemplo:
<a href="https://www.freecodecamp.org">this links to freecodecamp.org</a>
Luego, su navegador mostrará el texto this links to freecodecamp.org como un enlace en el que puede hacer clic. Y ese enlace te llevará a la dirección web https://www.freecodecamp.org.
Cree un a elemento que se vincule https://www.freecatphotoapp.com y tenga "fotos de gatos" como texto de anclaje.
Ejecutar las pruebas (Ctrl + Enter)
Restablecer todo el código
Consigue ayuda
Pruebas
Su a elemento debe tener el texto de anclaje de cat photos.
Necesitas un a elemento que se vincule a https://www.freecatphotoapp.com
Su a elemento debe tener una etiqueta de cierre.
<h2>CatPhotoApp</h2>
<main>
<img src="https://cdn.freecodecamp.org/curriculum/cat-photo-app/relaxing-cat.jpg" alt="Un lindo gato naranja acostado boca arriba.">
<a href="https://www.freecatphotoapp.com">fotos de gatos</a>
Kitty ipsum dolor sit amet, shed everywhere shed everywhere stretching attack your ankles chase the red dot, hairball run catnip eat the grass sniff.
Purr jump eat the grass rip the couch scratched sunbathe, shed everywhere rip the couch sleep in the sink fluffy fur catnip scratched.
</main>
/**
* Su salida de prueba irá aquí
*/
CatPhotoApp
fotos de gatos
Kitty ipsum dolor sit amet, shed everywhere shed everywhere stretching attack your ankles chase the red dot, hairball run catnip eat the grass sniff.
Purr jump eat the grass rip the couch scratched sunbathe, shed everywhere rip the couch sleep in the sink fluffy fur catnip scratched.

freecodecamp.org/learn/responsive-web-design/basic-html-and-html5/link-to-external-pages-with-anchor-elements

Diseño web receptivo heredado — HTML básico y HTML5

Enlace a páginas externas con elementos de anclaje

Puede usar elementos `a` (*anclaje*) para vincular a contenido fuera de su página web.

`a` los elementos necesitan una dirección web de destino llamada `href` atributo. También necesitan texto de anclaje. Aquí hay un ejemplo:

```
<a href="https://www.freecodecamp.org">this links to freecodecamp.org</a>
```

Luego, su navegador mostrará el texto `this links to freecodecamp.org` como un enlace en el que puede hacer clic. Y ese enlace te llevará a la dirección web `https://www.freecodecamp.org`.

Cree un `a` elemento que se vincule `https://www.freecatphotoapp.com` y tenga "fotos de gatos" como texto de anclaje.

Ejecutar las pruebas (Ctrl + Enter)

Restablecer todo el código

Consigue ayuda

Pruebas

- Su `a` elemento debe tener el texto de anclaje de `cat photos`.
- Necesitas un `a` elemento que se vincule a `https://www.freecatphotoapp.com`
- Su `a` elemento debe tener una etiqueta de cierre.

```
<h2>CatPhotoApp</h2>
<main>

<img src="https://cdn.freecodecamp.org/curriculum/cat-photo-app/relaxing-cat.jpg" alt="Un lindo gato naranja acostado boca arriba.">

<a href="https://www.freecatphotoapp.com">fotos de gatos</a>

  <p>Kitty ipsum dolor sit amet, shed everywhere shed everywhere stretching attack your ankles chase the red dot, hairball run catnip eat the grass sniff.</p>
  <p>Purr jump eat the grass rip the couch scratched sunbathe, shed everywhere rip the couch sleep in the sink fluffy fur catnip scratched.</p>
</main
```

```
/**
* Su salida de prueba irá aquí
*/
```

CatPhotoApp

fotos de gatos

Kitty ipsum dolor sit amet, shed everywhere shed everywhere stretching attack your ankles chase the red dot, hairball run catnip eat the grass sniff.

Purr jump eat the grass rip the couch scratched sunbathe, shed everywhere rip the couch sleep in the sink fluffy fur catnip scratched.

https://www.freecatphotoapp.com

Escribe aquí para buscar — 26°C Soleado — 12:14 19/09/2022

freecodecamp.org/learn/responsive-web-design/basic-html-and-html5/link-to-external-pages-with-anchor-elements

Legacy Responsive Web Design — Basic HTML and HTML5

Link to External Pages with Anchor Elements

You can use `a` (*anchor*) elements to link to content outside of your web page.

`a` elements need a destination web address called an `href` attribute. They also need anchor text. Here's an example:

```
<a href="https://www.freecodecamp.org">this links to freecodecamp.org</a>
```

Then your browser will display the text `this links to freecodecamp.org` as a link you can click. And that link will take you to the web address `https://www.freecodecamp.org`.

Create an `a` element that links to `https://www.freecatphotoapp.com` and has "cat photos" as its anchor text.

Run the Tests (Ctrl + Enter)

Reset All Code

Get Help

Tests

- Your `a` element should have the anchor text of `cat photos`.
- You need an `a` element that links to `https://www.freecatphotoapp.com`
- Your `a` element should have a closing tag.

```
<h2>CatPhotoApp</h2>
<main>

<img src="https://cdn.freecodecamp.org/curriculum/cat-photo-app/relaxing-cat.jpg" alt="Un lindo gato naranja acostado boca arriba.">

<a href="https://www.freecatphotoapp.com">cat photos</a>

  <p>Kitty ipsum dolor sit amet, shed everywhere shed everywhere stretching attack your ankles chase the red dot, hairball run catnip eat the grass sniff.</p>
  <p>Purr jump eat the grass rip the couch scratched sunbathe, shed everywhere rip the couch sleep in the sink fluffy fur catnip scratched.</p>
</main>
```

```
/**
* Your test output will go here
*/
```

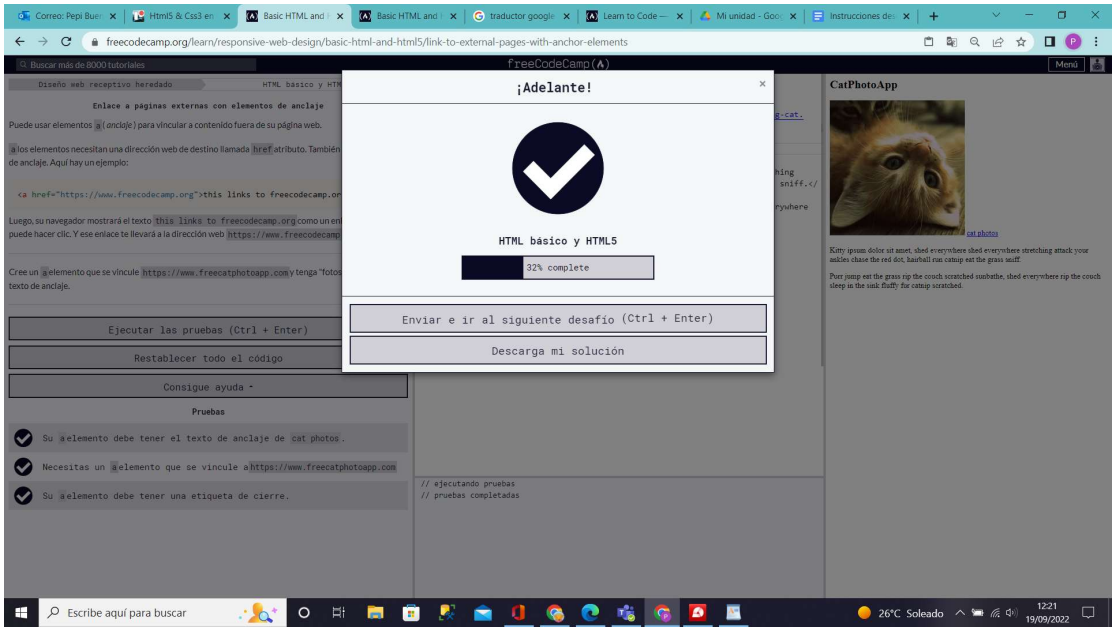
CatPhotoApp

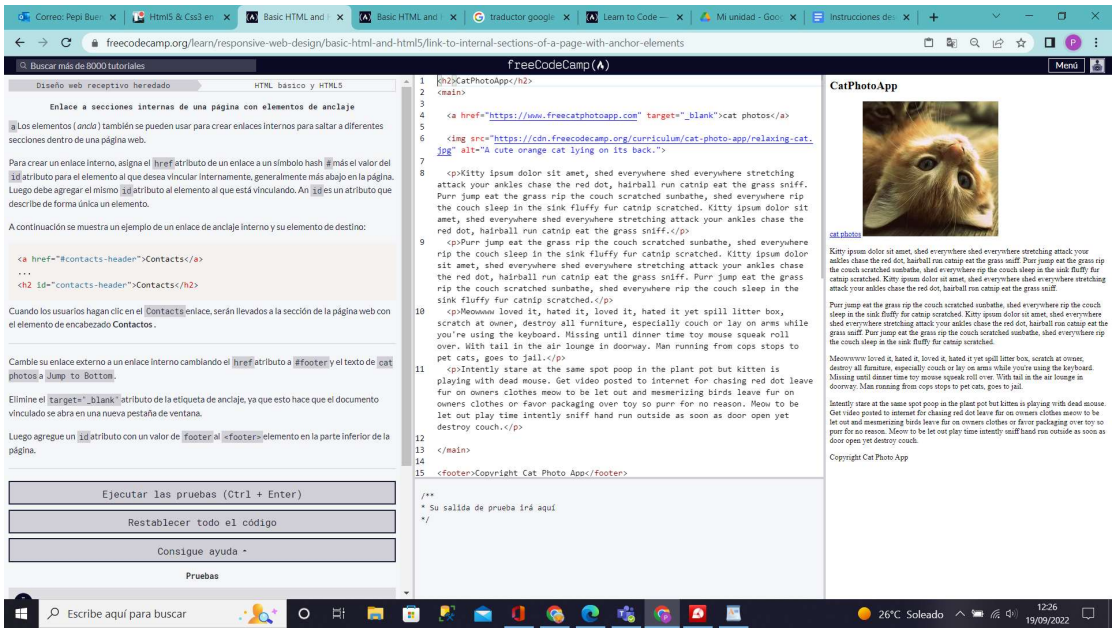
cat photos

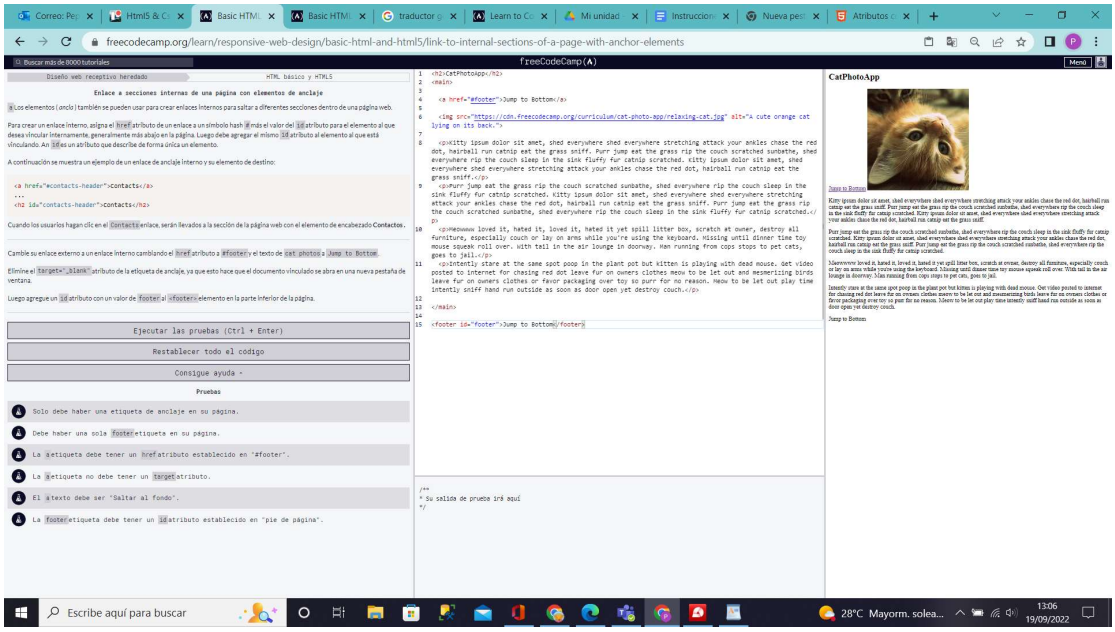
Kitty ipsum dolor sit amet, shed everywhere shed everywhere stretching attack your ankles chase the red dot, hairball run catnip eat the grass sniff.

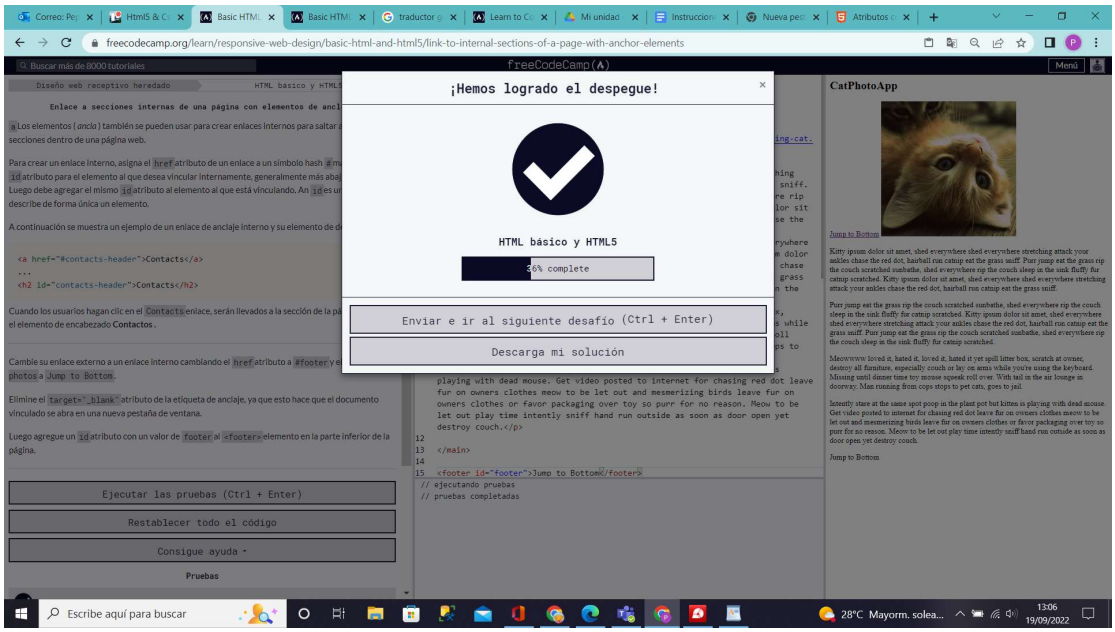
Purr jump eat the grass rip the couch scratched sunbathe, shed everywhere rip the couch sleep in the sink fluffy fur catnip scratched.

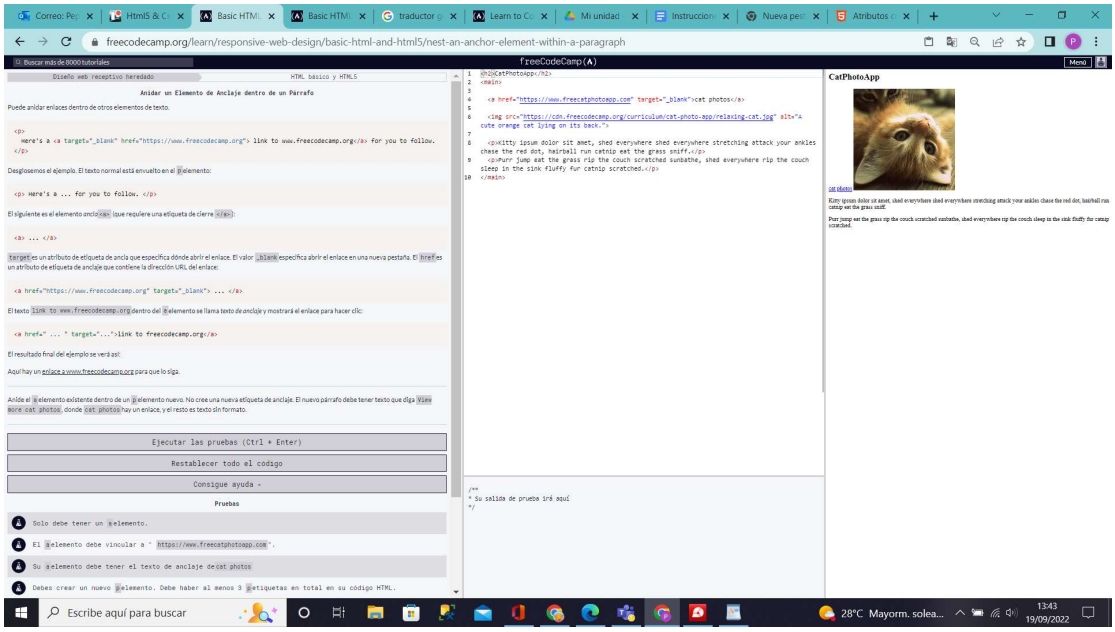
Escribe aquí para buscar — 26°C Soleado — 12:19 19/09/2022

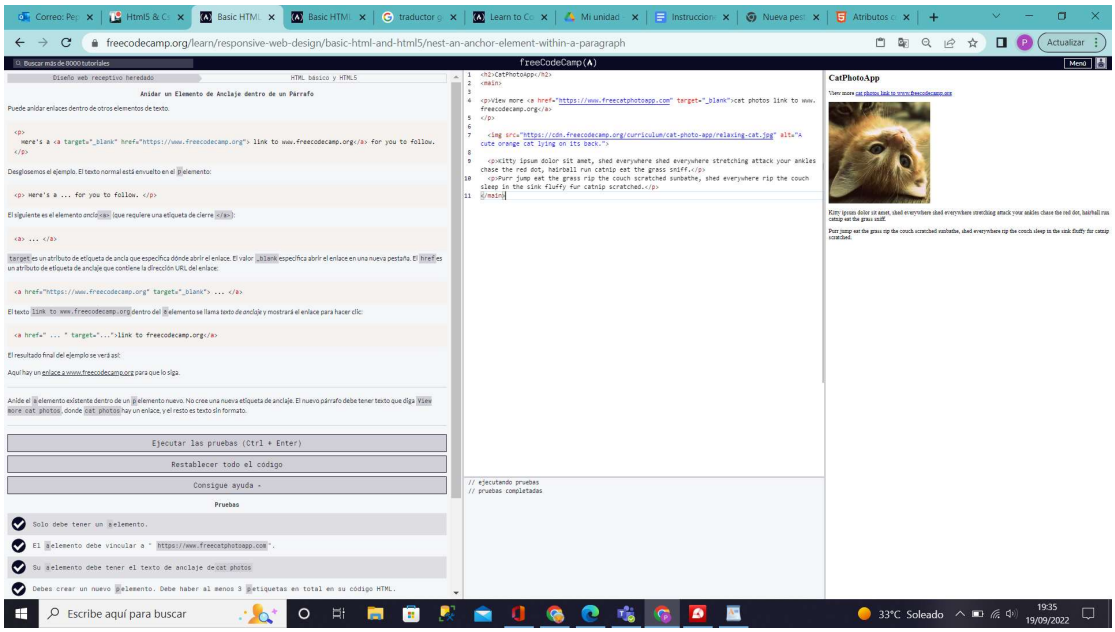

freecodecamp.org/learn/responsive-web-design/basic-html-and-html5/link-to-external-pages-with-anchor-elements
freeCodeCamp
¡Adelante!
HTML básico y HTML5
32% complete
Enviar e ir al siguiente desafío (Ctrl + Enter)
Descarga mi solución
Ejecutar las pruebas (Ctrl + Enter)
Restablecer todo el código
Consigue ayuda
Pruebas
CatPhotoApp

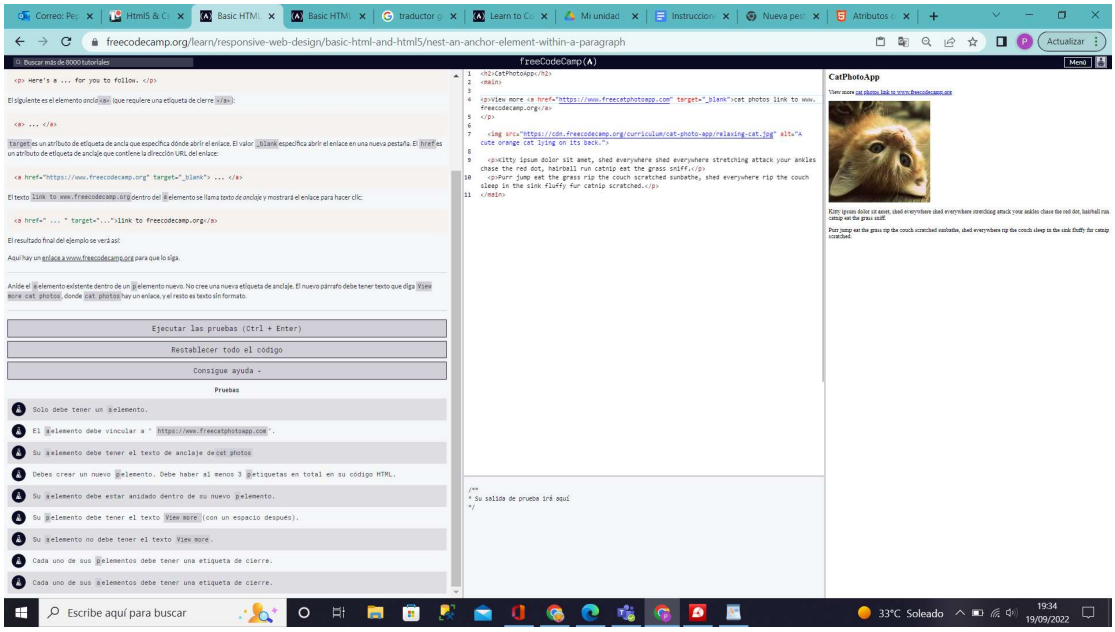

freecodecamp.org/learn/responsive-web-design/basic-html-and-html5/link-to-internal-sections-of-a-page-with-anchor-elements
freeCodeCamp
Ejecutar las pruebas (Ctrl + Enter)
Restablecer todo el código
Consigue ayuda
Pruebas
CatPhotoApp

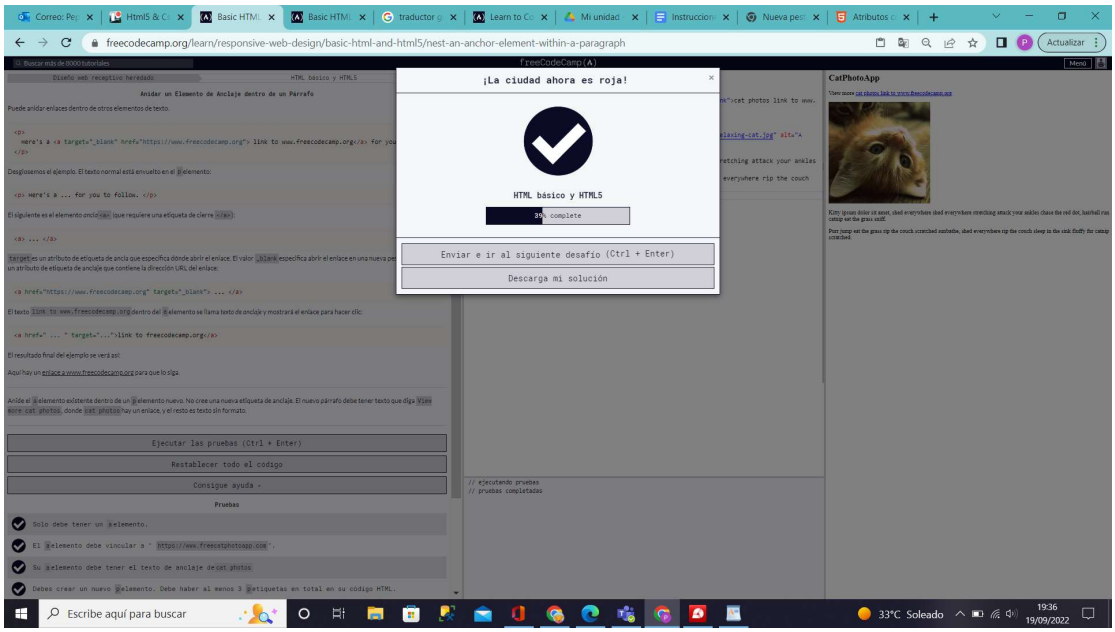

freecodecamp.org/learn/responsive-web-design/basic-html-and-html5/link-to-internal-sections-of-a-page-with-anchor-elements
freeCodeCamp(A)
Diseño web receptivo heredado
HTML básico y HTML5
Enlace a secciones internas de una página con elementos de anclaje
Ejecutar las pruebas (Ctrl + Enter)
Restablecer todo el codigo
Consigue ayuda
Pruebas
Solo debe haber una etiqueta de anclaje en su página.
Debe haber una sola footer etiqueta en su página.
La a etiqueta debe tener un href atributo establecido en "#footer".
La a etiqueta no debe tener un target atributo.
El a texto debe ser "Saltar al fondo".
La footer etiqueta debe tener un id atributo establecido en "pie de página".
CatPhotoApp
Jump to Bottom

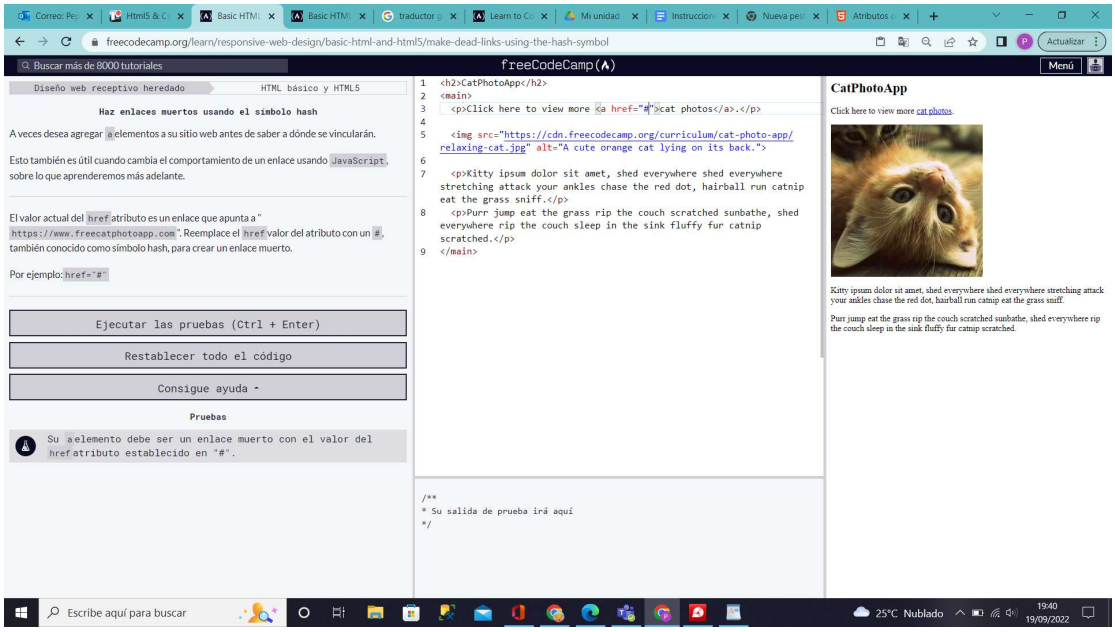

freecodecamp.org/learn/responsive-web-design/basic-html-and-html5/link-to-internal-sections-of-a-page-with-anchor-elements
freeCodeCamp(A)
¡Hemos logrado el despegue!
HTML básico y HTML5
66% complete
Enviar e ir al siguiente desafío (Ctrl + Enter)
Descarga mi solución
Ejecutar las pruebas (Ctrl + Enter)
Restablecer todo el código
Consigue ayuda
Pruebas
CatPhotoApp
Jump to Bottom

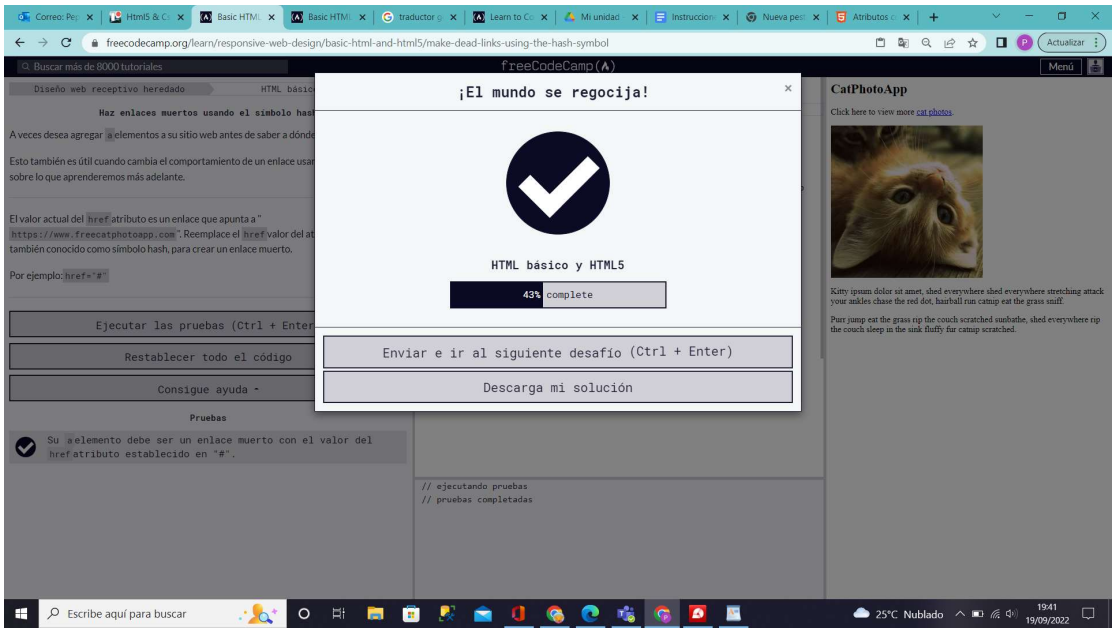

freecodecamp.org/learn/responsive-web-design/basic-html-and-html5/nest-an-anchor-element-within-a-paragraph
freeCodeCamp(A)
Diseño web receptivo heredado
HTML básico y HTML5
Anidar un Elemento de Anclaje dentro de un Párrafo
Puede anidar enlaces dentro de otros elementos de texto.
Desglosemos el ejemplo. El texto normal está envuelto en el p elemento:
El siguiente es el elemento ancla <a> que requiere una etiqueta de cierre </a>:
target es un atributo de etiqueta de ancla que especifica dónde abrir el enlace. El valor _blank especifica abrir el enlace en una nueva pestaña. El href es un atributo de etiqueta de anclaje que contiene la dirección URL del enlace:
El texto link to www.freecodecamp.org dentro del a elemento se llama texto de anclaje y mostrará el enlace para hacer clic:
El resultado final del ejemplo se verá así:
Aquí hay un enlace a www.freecodecamp.org para que lo siga.
Anide el a elemento existente dentro de un p elemento nuevo. No cree una nueva etiqueta de anclaje. El nuevo párrafo debe tener texto que diga View more cat photos, donde cat photos hay un enlace, y el resto es texto sin formato.
Ejecutar las pruebas (Ctrl + Enter)
Restablecer todo el código
Consigue ayuda
Pruebas
Solo debe tener un a elemento.
El a elemento debe vincular a " https://www.freecatphotoapp.com ".
Su a elemento debe tener el texto de anclaje de cat photos
Debes crear un nuevo p elemento. Debe haber al menos 3 p etiquetas en total en su código HTML.
CatPhotoApp
Escribe aquí para buscar
28°C Mayorm. solea...
13:43
19/09/2022

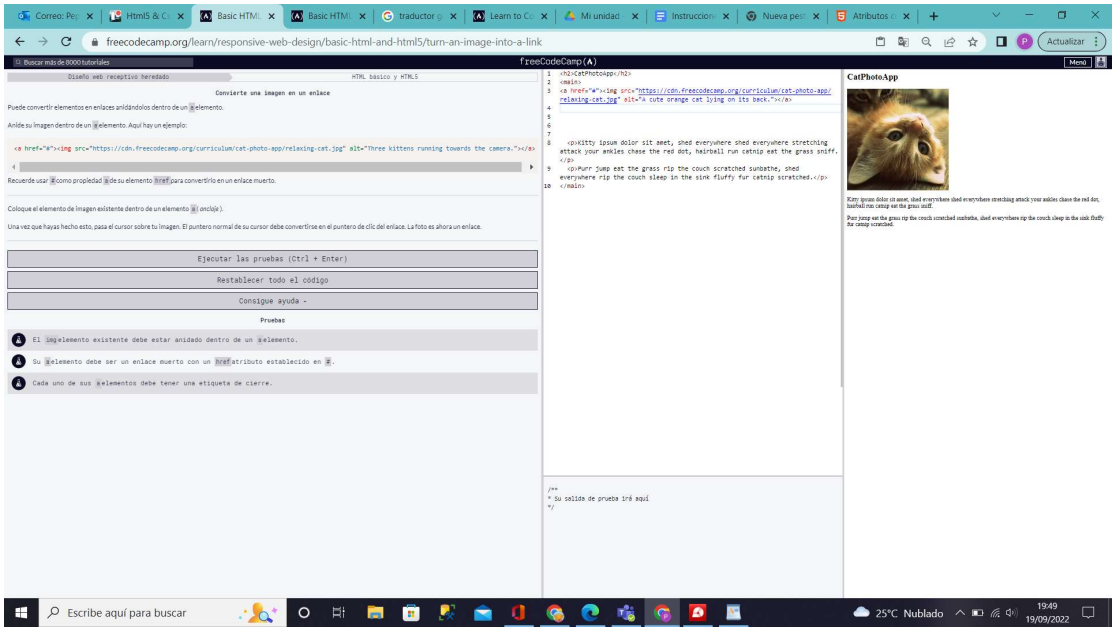

freecodecamp.org/learn/responsive-web-design/basic-html-and-html5/nest-an-anchor-element-within-a-paragraph
freeCodeCamp(A)
Diseño web receptivo heredado
HTML básico y HTML5
Anidar un Elemento de Anclaje dentro de un Párrafo
Puede anidar enlaces dentro de otros elementos de texto.
Desglosemos el ejemplo. El texto normal está envuelto en el p elemento:
El siguiente es el elemento ancla <a> que requiere una etiqueta de cierre </a>:
target es un atributo de etiqueta de ancla que especifica dónde abrir el enlace. El valor _blank especifica abrir el enlace en una nueva pestaña. El href es un atributo de etiqueta de anclaje que contiene la dirección URL del enlace:
El texto link to www.freecodecamp.org dentro del a elemento se llama texto de anclaje y mostrará el enlace para hacer clic:
El resultado final del ejemplo se verá así:
Aquí hay un enlace a www.freecodecamp.org para que lo siga.
Anide el a elemento existente dentro de un p elemento nuevo. No cree una nueva etiqueta de anclaje. El nuevo párrafo debe tener texto que diga View more cat photos, donde cat photos hay un enlace, y el resto es texto sin formato.
Ejecutar las pruebas (Ctrl + Enter)
Restablecer todo el código
Consigue ayuda
Pruebas
Solo debe tener un a elemento.
El a elemento debe vincular a " https://www.freecatphotoapp.com ".
Su a elemento debe tener el texto de anclaje de cat photos
Debes crear un nuevo p elemento. Debe haber al menos 3 p etiquetas en total en su código HTML.
// ejecutando pruebas
// pruebas completadas
CatPhotoApp
Escribe aquí para buscar
33°C Soleado
19:35
19/09/2022

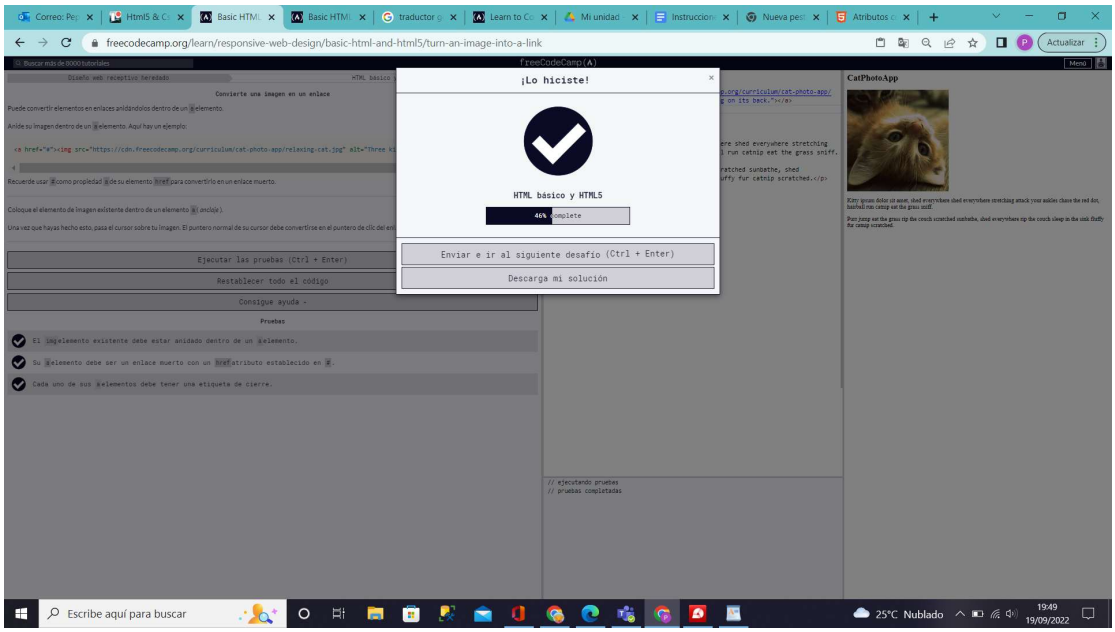

freecodecamp.org/learn/responsive-web-design/basic-html-and-html5/nest-an-anchor-element-within-a-paragraph
freeCodeCamp(A)
CatPhotoApp
Ejecutar las pruebas (Ctrl + Enter)
Restablecer todo el código
Consigue ayuda -
Pruebas
Solo debe tener un a elemento.
El a elemento debe vincular a " https://www.freecatphotoapp.com ".
Su a elemento debe tener el texto de anclaje de cat photos
Debes crear un nuevo p elemento. Debe haber al menos 3 p etiquetas en total en su código HTML.
Su a elemento debe estar anidado dentro de su nuevo p elemento.
Su p elemento debe tener el texto View more (con un espacio después).
Su a elemento no debe tener el texto View more.
Cada uno de sus p elementos debe tener una etiqueta de cierre.
Cada uno de sus a elementos debe tener una etiqueta de cierre.
/**
* Su salida de prueba irá aquí
*/
Escribe aquí para buscar
33°C Soleado
19:34
19/09/2022

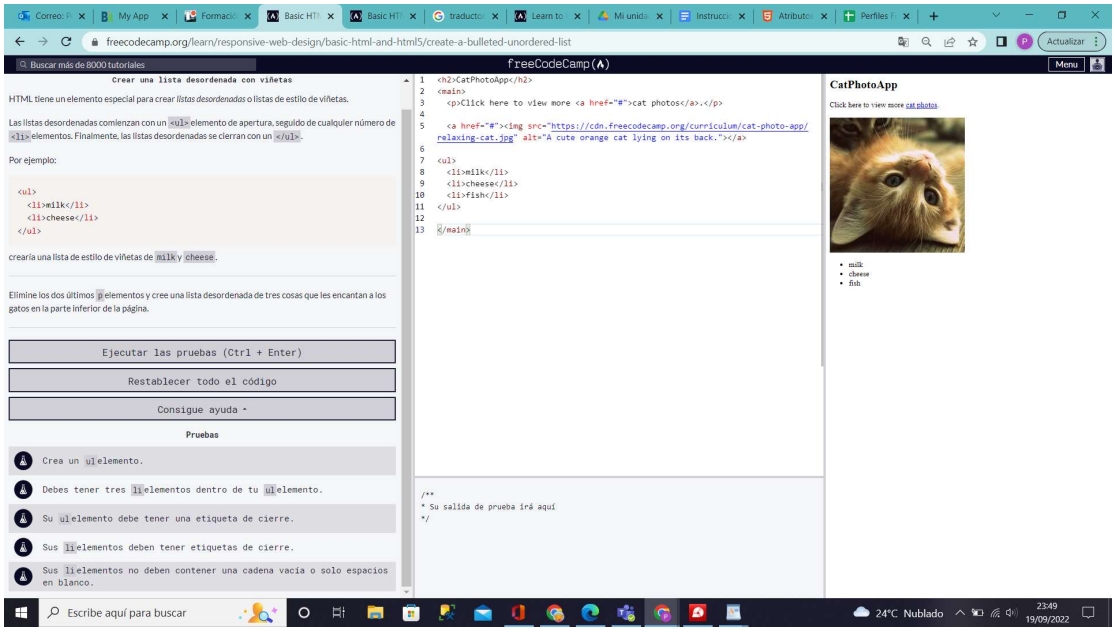

freecodecamp.org/learn/responsive-web-design/basic-html-and-html5/nest-an-anchor-element-within-a-paragraph
freeCodeCamp(A)
¡La ciudad ahora es roja!
HTML básico y HTML5
3% complete
Enviar e ir al siguiente desafío (Ctrl + Enter)
Descarga mi solución
CatPhotoApp
Ejecutar las pruebas (Ctrl + Enter)
Restablecer todo el código
Consigue ayuda -
Pruebas
// ejecutando pruebas
// pruebas completadas
Escribe aquí para buscar
33°C Soleado
19:36
19/09/2022

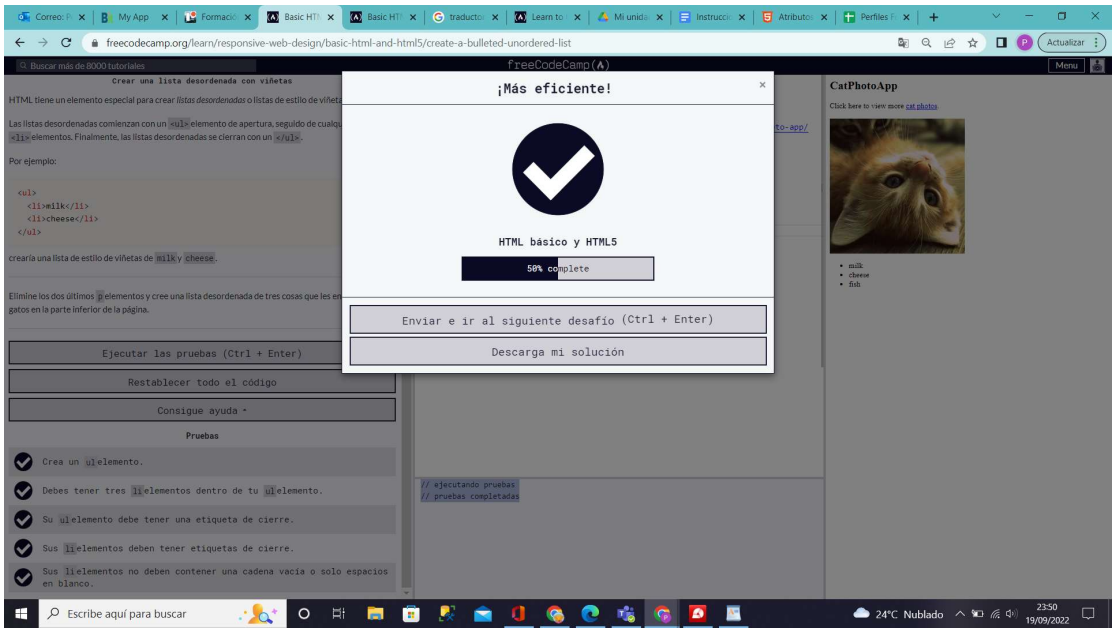

freecodecamp.org/learn/responsive-web-design/basic-html-and-html5/make-dead-links-using-the-hash-symbol
freeCodeCamp
Diseño web receptivo heredado
HTML básico y HTML5
Haz enlaces muertos usando el símbolo hash
A veces desea agregar a elementos a su sitio web antes de saber a dónde se vincularán.
Esto también es útil cuando cambia el comportamiento de un enlace usando JavaScript, sobre lo que aprenderemos más adelante.
El valor actual del href atributo es un enlace que apunta a " https://www.freecatphotoapp.com". Reemplace el href valor del atributo con un #, también conocido como símbolo hash, para crear un enlace muerto.
Por ejemplo: href="#"
Ejecutar las pruebas (Ctrl + Enter)
Restablecer todo el código
Consigue ayuda
Pruebas
Su a elemento debe ser un enlace muerto con el valor del href atributo establecido en "#".
<h2>CatPhotoApp</h2>
<main>
Click here to view more <a href="#">cat photos</a>.
<img src="https://cdn.freecodecamp.org/curriculum/cat-photo-app/relaxing-cat.jpg" alt="A cute orange cat lying on its back.">
Kitty ipsum dolor sit amet, shed everywhere shed everywhere stretching attack your ankles chase the red dot, hairball run catnip eat the grass sniff.
Purr jump eat the grass rip the couch scratched sunbathe, shed everywhere rip the couch sleep in the sink fluffy fur catnip scratched.
</main>
/**
* Su salida de prueba irá aquí
*/
CatPhotoApp
Click here to view more cat photos.
Kitty ipsum dolor sit amet, shed everywhere shed everywhere stretching attack your ankles chase the red dot, hairball run catnip eat the grass sniff.
Purr jump eat the grass rip the couch scratched sunbathe, shed everywhere rip the couch sleep in the sink fluffy fur catnip scratched.
Escribe aquí para buscar
25°C Nublado
19:40
19/09/2022

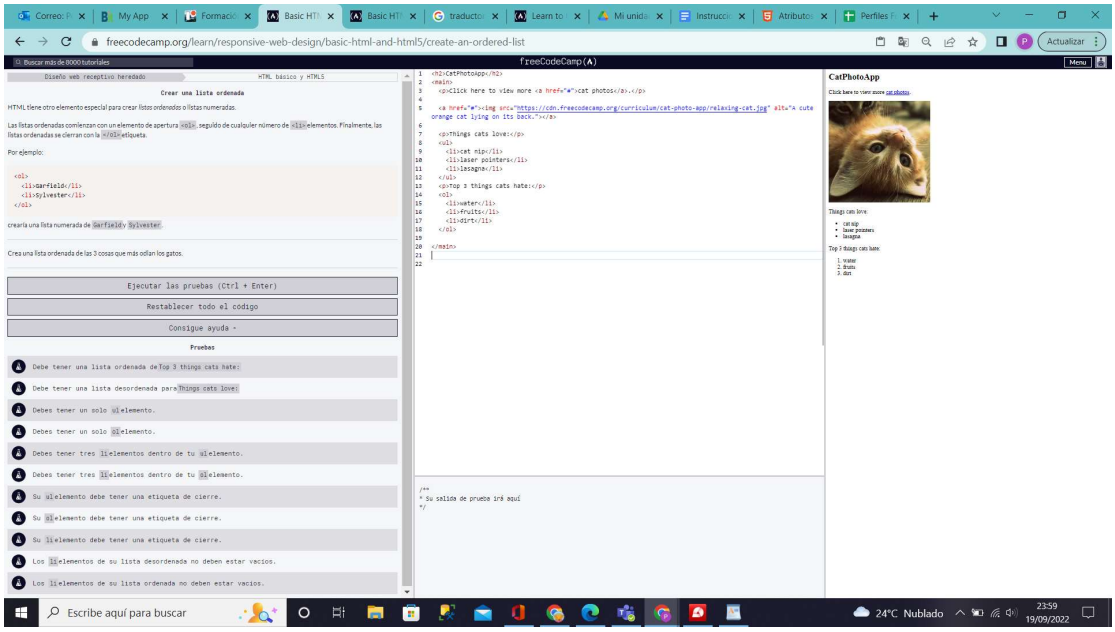

freecodecamp.org/learn/responsive-web-design/basic-html-and-html5/make-dead-links-using-the-hash-symbol
¡El mundo se regocija!
HTML básico y HTML5
43% complete
Enviar e ir al siguiente desafío (Ctrl + Enter)
Descarga mi solución
// ejecutando pruebas
// pruebas completadas
CatPhotoApp
Escribe aquí para buscar
25°C Nublado
19:41
19/09/2022

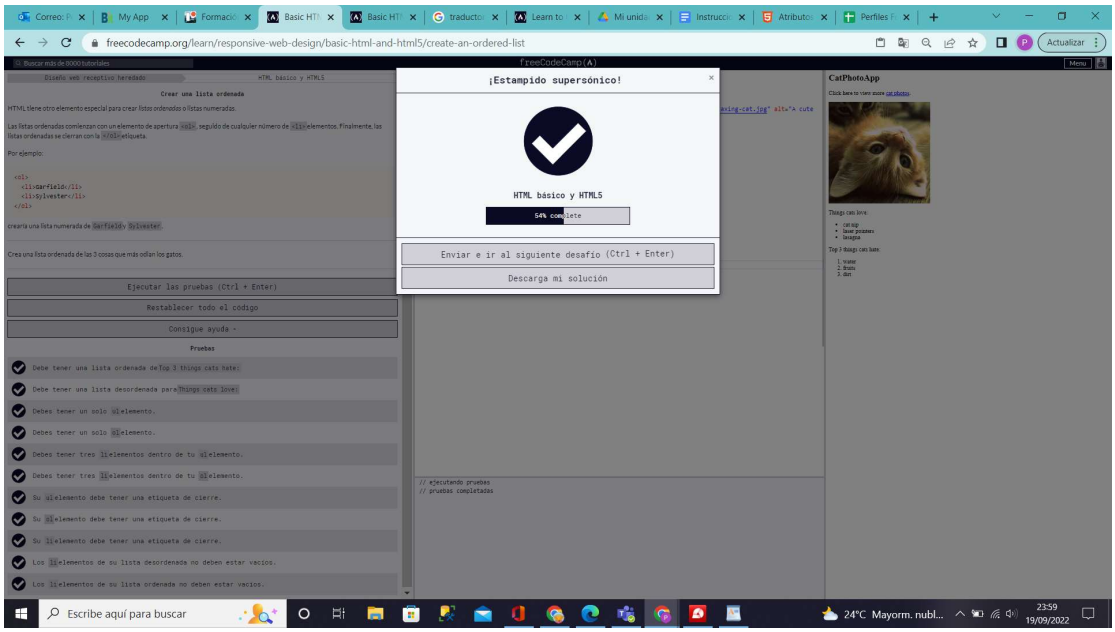

freecodecamp.org/learn/responsive-web-design/basic-html-and-html5/turn-an-image-into-a-link
freeCodeCamp(A)
Diseño web receptivo heredado
HTML básico y HTML5
Convierte una imagen en un enlace
Puede convertir elementos en enlaces anidándolos dentro de un a elemento.
Anide su imagen dentro de un a elemento. Aquí hay un ejemplo:
Recuerde usar # como propiedad a de su elemento href para convertirlo en un enlace muerto.
Coloque el elemento de imagen existente dentro de un elemento a (ancla).
Una vez que hayas hecho esto, pasa el cursor sobre tu imagen. El puntero normal de su cursor debe convertirse en el puntero de clic del enlace. La foto es ahora un enlace.
Ejecutar las pruebas (Ctrl + Enter)
Restablecer todo el código
Consigue ayuda -
Pruebas
El img elemento existente debe estar anidado dentro de un a elemento.
Su a elemento debe ser un enlace muerto con un href atributo establecido en #.
Cada uno de sus a elementos debe tener una etiqueta de cierre.
CatPhotoApp
Escribe aquí para buscar
25°C Nublado
19:49
19/09/2022

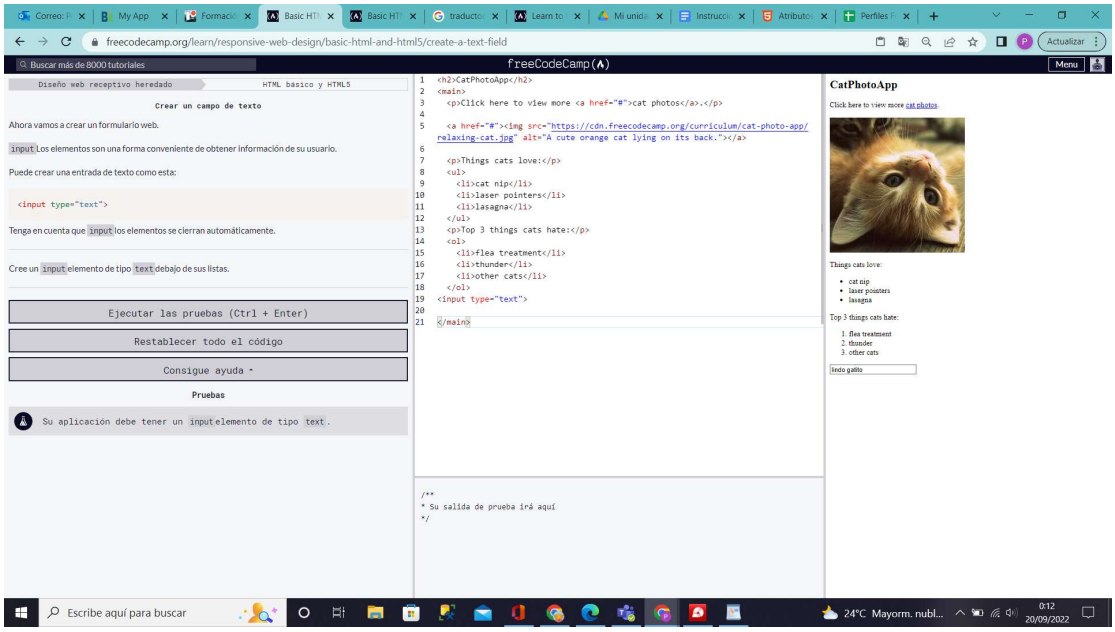

freecodecamp.org/learn/responsive-web-design/basic-html-and-html5/turn-an-image-into-a-link
¡Lo hiciste!
HTML básico y HTML5
46% complete
Enviar e ir al siguiente desafío (Ctrl + Enter)
Descarga mi solución
CatPhotoApp
Escribe aquí para buscar
25°C Nublado
19:49
19/09/2022

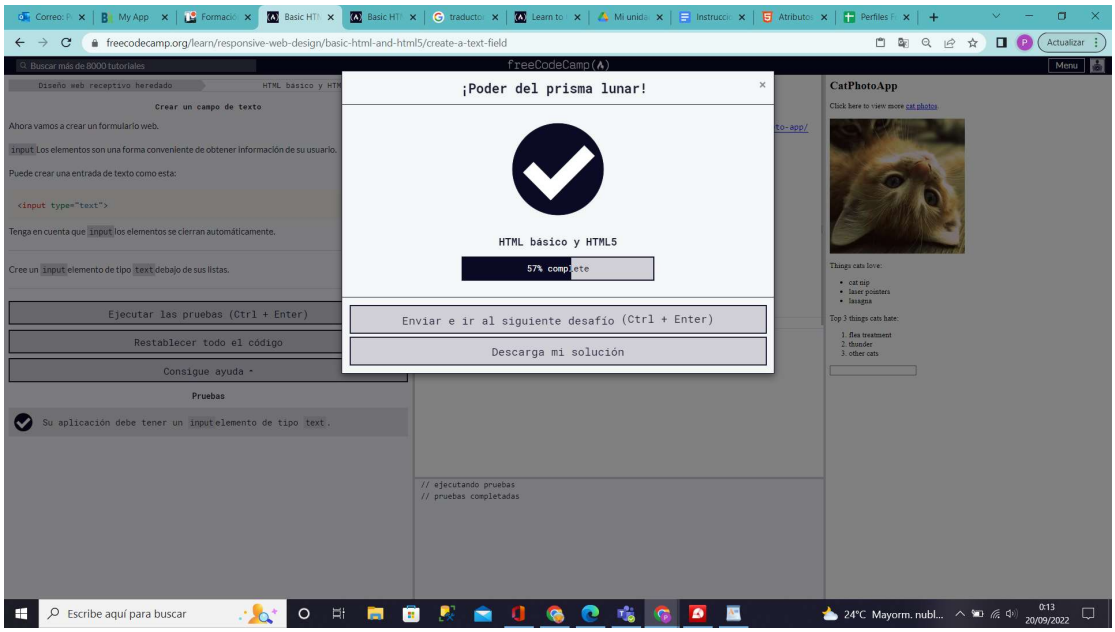

freecodecamp.org/learn/responsive-web-design/basic-html-and-html5/create-a-bulleted-unordered-list
freeCodeCamp
Crear una lista desordenada con viñetas
HTML tiene un elemento especial para crear listas desordenadas o listas de estilo de viñetas.
Las listas desordenadas comienzan con un <ul> elemento de apertura, seguido de cualquier número de <li> elementos. Finalmente, las listas desordenadas se cierran con un </ul>.
Por ejemplo:
<ul>
<li>milk</li>
<li>cheese</li>
</ul>
crearía una lista de estilo de viñetas de milk y cheese.
Elimine los dos últimos p elementos y cree una lista desordenada de tres cosas que les encantan a los gatos en la parte inferior de la página.
Ejecutar las pruebas (Ctrl + Enter)
Restablecer todo el código
Consigue ayuda
Pruebas
Crea un ul elemento.
Debes tener tres li elementos dentro de tu ul elemento.
Su ul elemento debe tener una etiqueta de cierre.
Sus li elementos deben tener etiquetas de cierre.
Sus li elementos no deben contener una cadena vacía o solo espacios en blanco.
CatPhotoApp
Click here to view more cat photos.
milk
cheese
fish

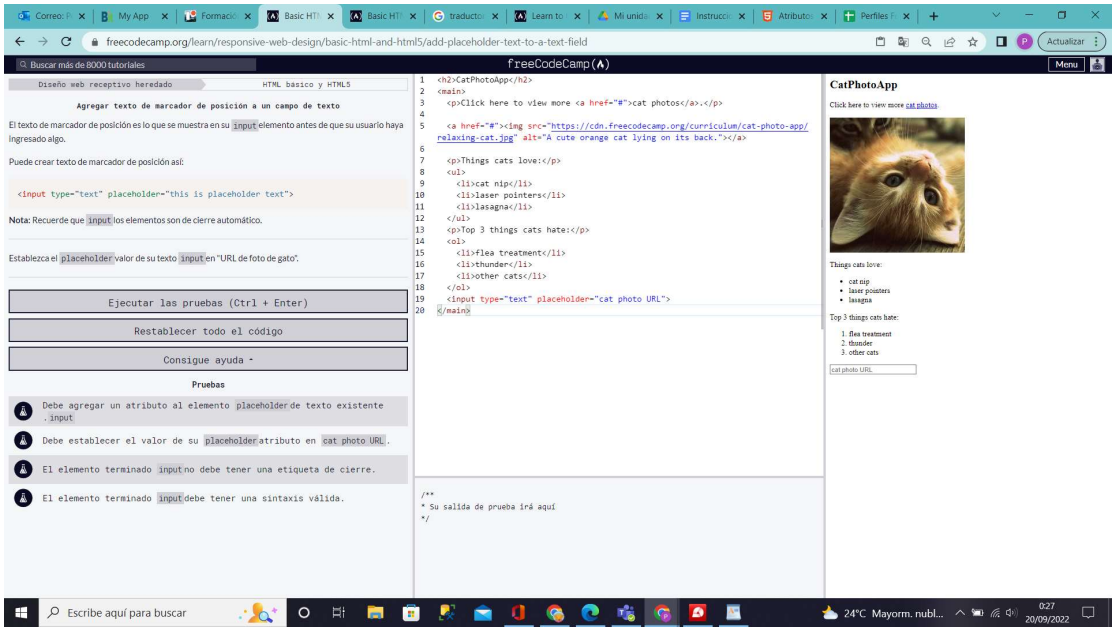

¡Más eficiente!
HTML básico y HTML5
50% complete
Enviar e ir al siguiente desafío (Ctrl + Enter)
Descarga mi solución

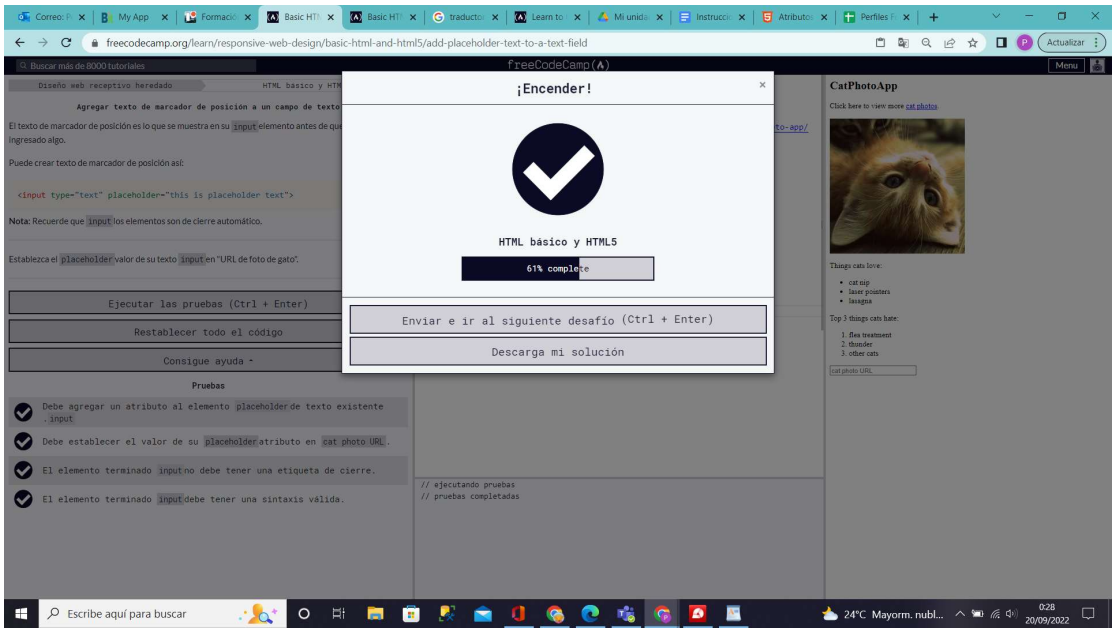

freecodecamp.org/learn/responsive-web-design/basic-html-and-html5/create-an-ordered-list
Crear una lista ordenada
Ejecutar las pruebas (Ctrl + Enter)
Restablecer todo el codigo
Consigue ayuda
Pruebas
CatPhotoApp

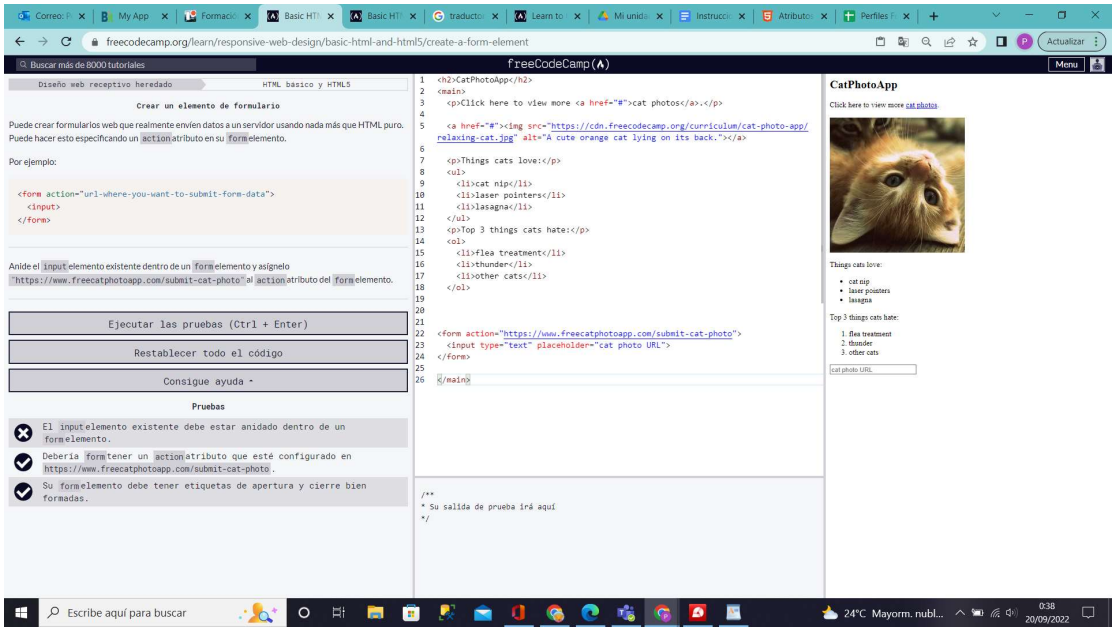

¡Estampido supersónico!
HTML básico y HTML5
54% complete
Enviar e ir al siguiente desafío (Ctrl + Enter)
Descarga mi solución

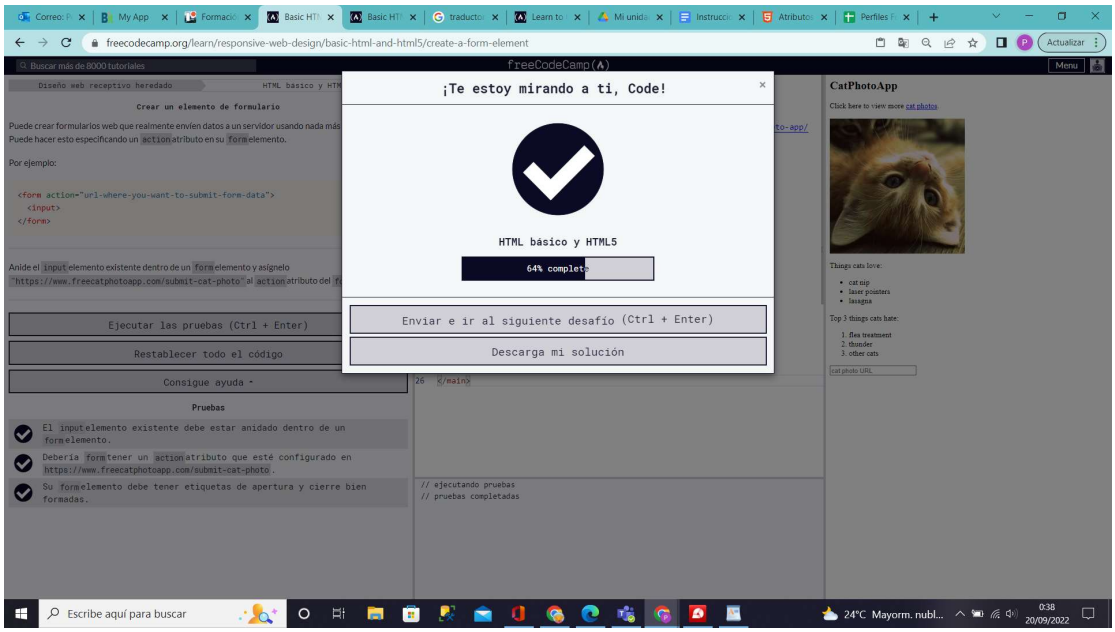

freecodecamp.org/learn/responsive-web-design/basic-html-and-html5/create-a-text-field
Buscar más de 8000 tutoriales
freeCodeCamp(A)
Menu
Diseño web receptivo heredado
HTML básico y HTML5
Crear un campo de texto
Ahora vamos a crear un formulario web.
input Los elementos son una forma conveniente de obtener información de su usuario.
Puede crear una entrada de texto como esta:
<input type="text">
Tenga en cuenta que input los elementos se cierran automáticamente.
Cree un input elemento de tipo text debajo de sus listas.
Ejecutar las pruebas (Ctrl + Enter)
Restablecer todo el código
Consigue ayuda
Pruebas
Su aplicación debe tener un input elemento de tipo text.
<h2>CatPhotoApp</h2>
<main>
Click here to view more <a href="#">cat photos</a>.
<a href="#"><img src="https://cdn.freecodecamp.org/curriculum/cat-photo-app/relaxing-cat.jpg" alt="A cute orange cat lying on its back."></a>
Things cats love:
<ul>
<li>cat nip</li>
<li>laser pointers</li>
<li>lasagna</li>
</ul>
Top 3 things cats hate:
<ol>
<li>flea treatment</li>
<li>thunder</li>
<li>other cats</li>
</ol>
<input type="text">
</main>
/**
* Su salida de prueba irá aquí
*/
CatPhotoApp
Click here to view more cat photos.
Things cats love:
cat nip
laser pointers
lasagna
Top 3 things cats hate:
1. flea treatment
2. thunder
3. other cats
lindo gatito
Escribe aquí para buscar
24°C Mayorm. nubl...
0:12
20/09/2022

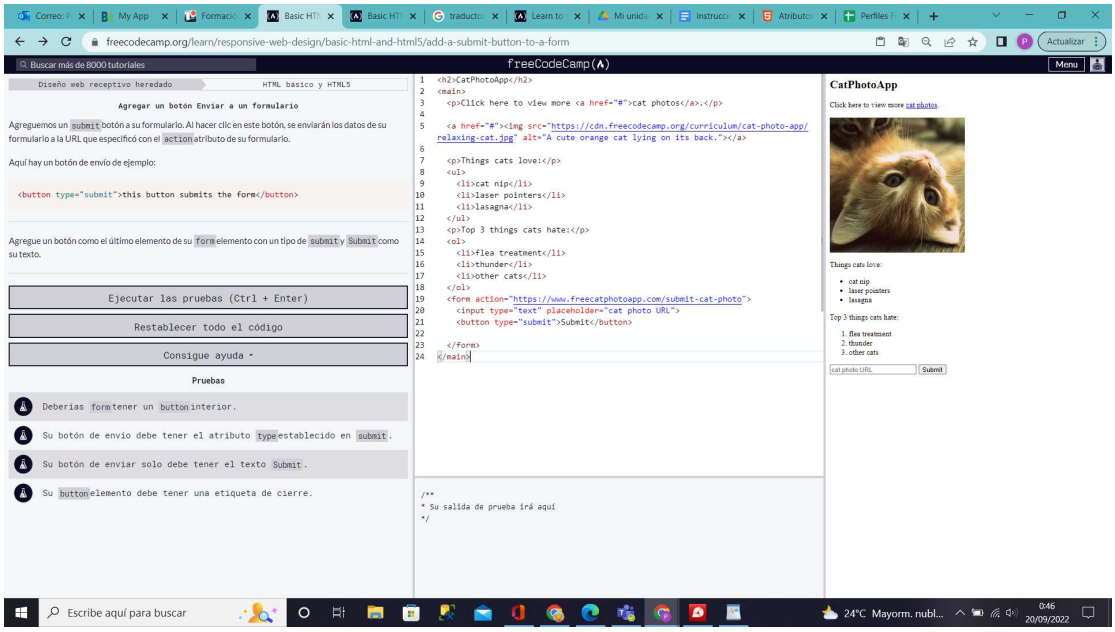

freecodecamp.org/learn/responsive-web-design/basic-html-and-html5/create-a-text-field
¡Poder del prisma lunar!
HTML básico y HTML5
57% completo
Enviar e ir al siguiente desafío (Ctrl + Enter)
Descarga mi solución
// ejecutando pruebas
// pruebas completadas
Escribe aquí para buscar
24°C Mayorm. nubl...
0:13
20/09/2022

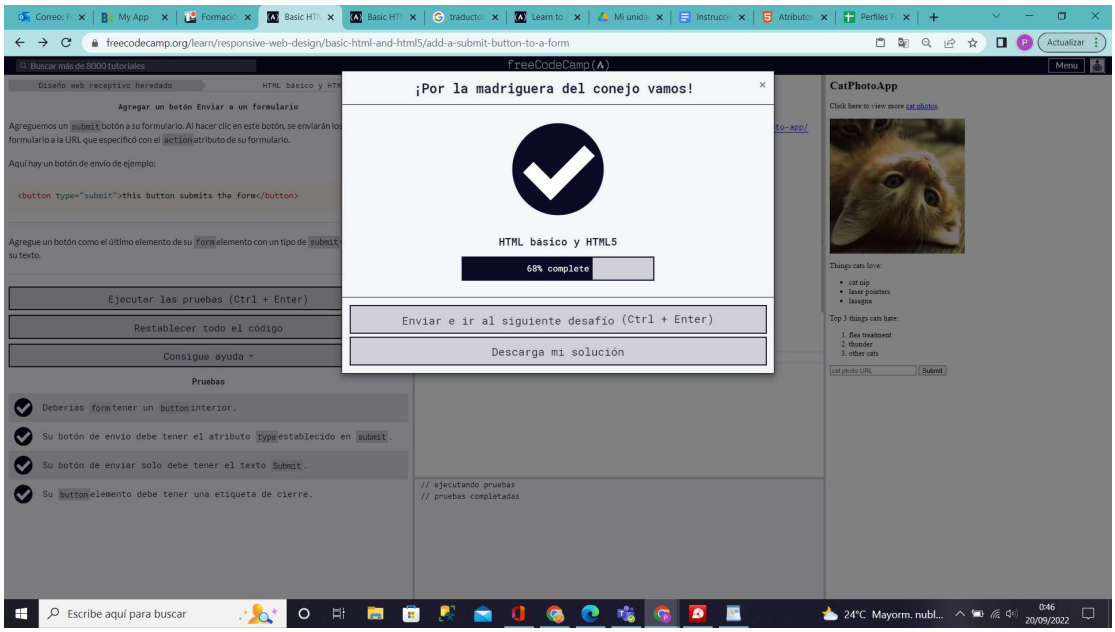

freecodecamp.org/learn/responsive-web-design/basic-html-and-html5/add-placeholder-text-to-a-text-field
freeCodeCamp
Diseño web receptivo heredado
HTML básico y HTML5
Agregar texto de marcador de posición a un campo de texto
El texto de marcador de posición es lo que se muestra en su input elemento antes de que su usuario haya ingresado algo.
Puede crear texto de marcador de posición así:
<input type="text" placeholder="this is placeholder text">
Nota: Recuerde que input los elementos son de cierre automático.
Establezca el placeholder valor de su texto input en "URL de foto de gato".
Ejecutar las pruebas (Ctrl + Enter)
Restablecer todo el código
Consigue ayuda
Pruebas
Debe agregar un atributo al elemento placeholder de texto existente .input
Debe establecer el valor de su placeholder atributo en cat photo URL.
El elemento terminado input no debe tener una etiqueta de cierre.
El elemento terminado input debe tener una sintaxis válida.
CatPhotoApp
Click here to view more cat photos.
Things cats love:
cat nip
laser pointers
lasagna
Top 3 things cats hate:
flea treatment
thunder
other cats
cat photo URL


¡Encender!
HTML básico y HTML5
61% completo
Enviar e ir al siguiente desafío (Ctrl + Enter)
Descarga mi solución


freecodecamp.org/learn/responsive-web-design/basic-html-and-html5/create-a-form-element
freeCodeCamp
Diseño web receptivo heredado
HTML básico y HTML5
Crear un elemento de formulario
Puede crear formularios web que realmente envíen datos a un servidor usando nada más que HTML puro. Puede hacer esto especificando un action atributo en su form elemento.
Por ejemplo:
<form action="url-where-you-want-to-submit-form-data">
<input>
</form>
Anide el input elemento existente dentro de un form elemento y asígnelo "https://www.freecatphotoapp.com/submit-cat-photo" al action atributo del form elemento.
Ejecutar las pruebas (Ctrl + Enter)
Restablecer todo el código
Consigue ayuda
Pruebas
El input elemento existente debe estar anidado dentro de un form elemento.
Deberia form tener un action atributo que esté configurado en https://www.freecatphotoapp.com/submit-cat-photo .
Su form elemento debe tener etiquetas de apertura y cierre bien formadas.
CatPhotoApp
Click here to view more cat photos.
Things cats love:
Top 3 things cats hate:


¡Te estoy mirando a ti, Code!
HTML básico y HTML5
64% complet
Enviar e ir al siguiente desafío (Ctrl + Enter)
Descarga mi solución





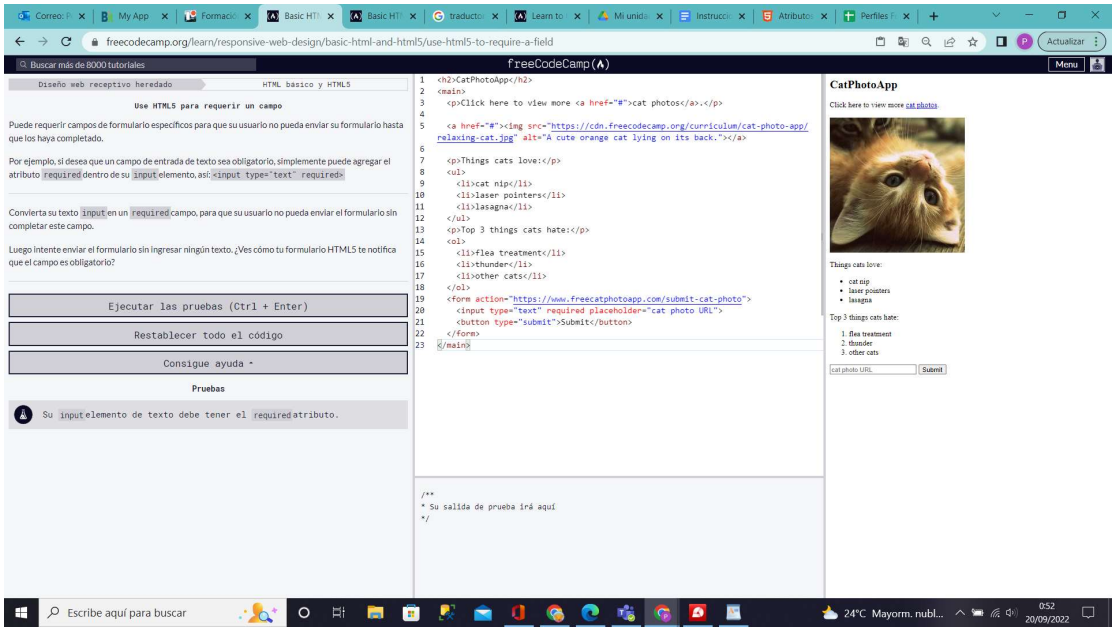
<h2>CatPhotoApp</h2>

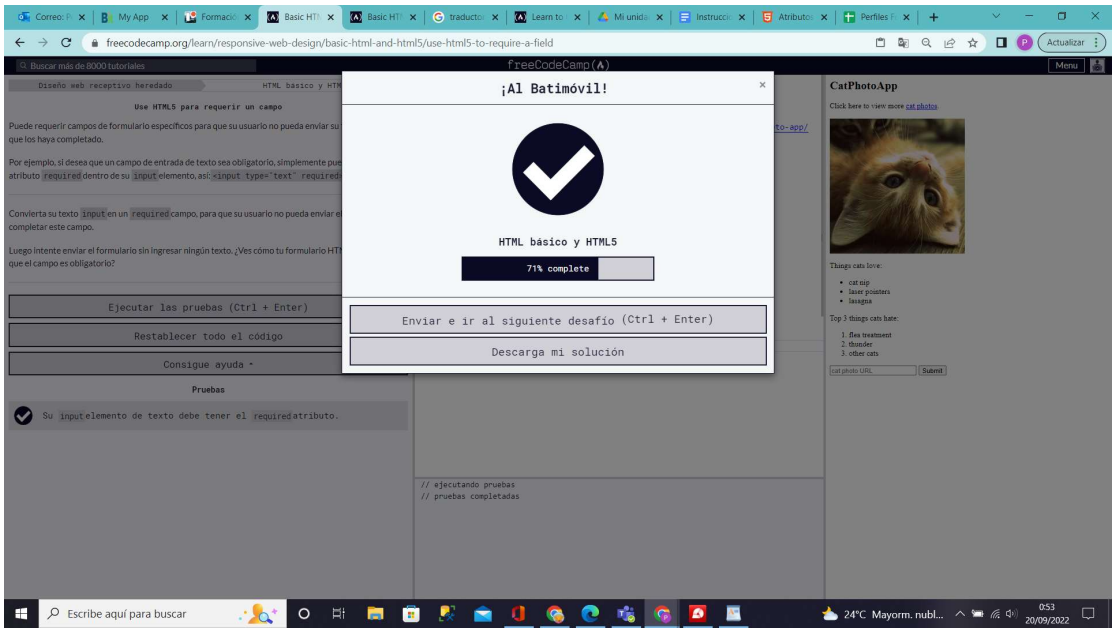
<main>

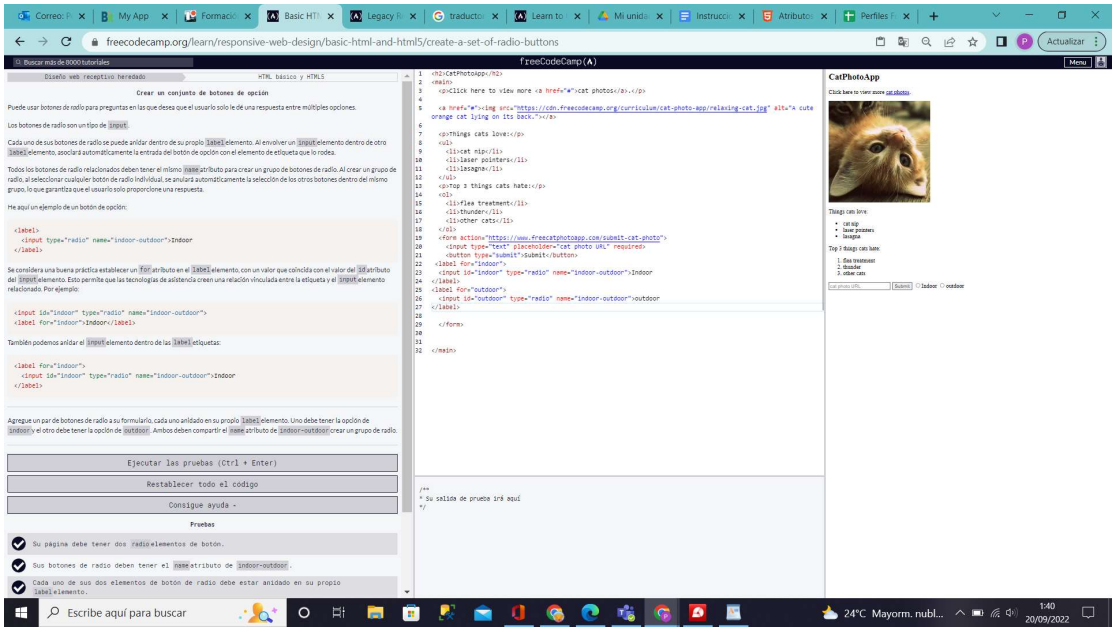
Click here to view more <a href="#">cat photos</a>.

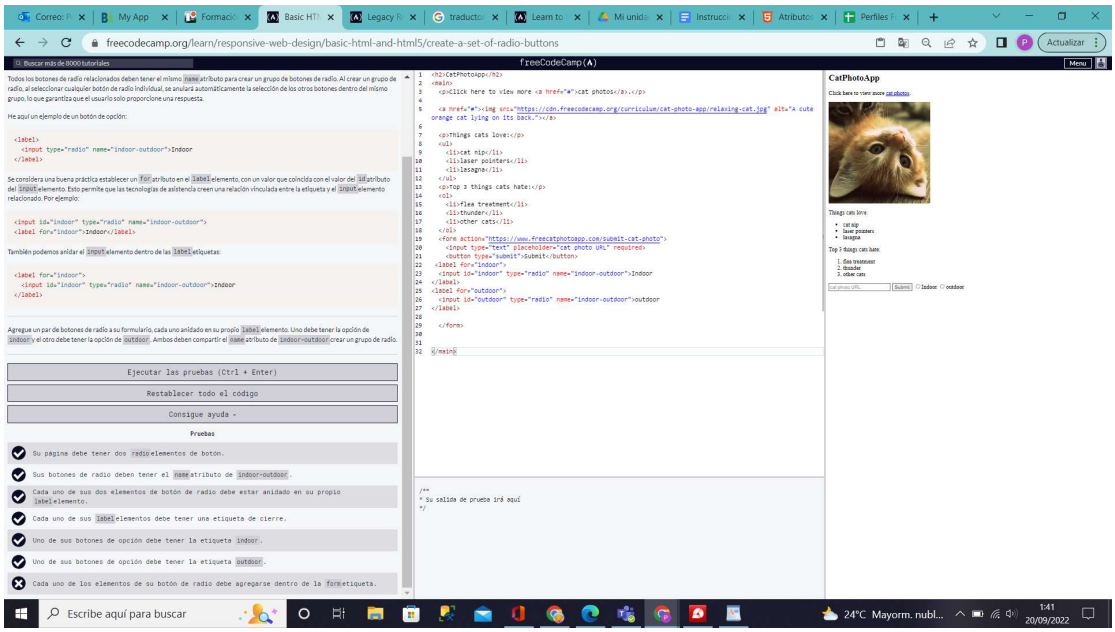
<a href="#"><img src="https://cdn.freecodecamp.org/curriculum/cat-photo-app/relaxing-cat.jpg" alt="A cute orange cat lying on its back."></a>

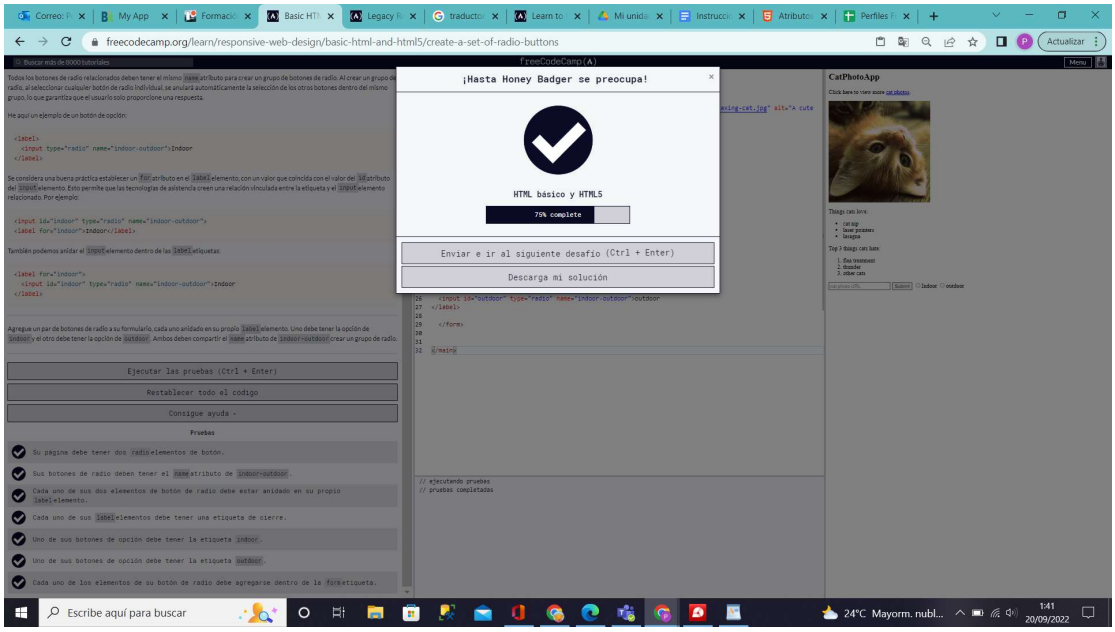
```
  <p>Things cats love:</p>
  <ul>
    <li>cat nip</li>
    <li>laser pointers</li>
    <li>lasagna</li>
  </ul>
  <p>Top 3 things cats hate:</p>
  <ol>
    <li>flea treatment</li>
    <li>thunder</li>
    <li>other cats</li>
  </ol>
  <form action="https://www.freecatphotoapp.com/submit-cat-photo">
    <input type="text" placeholder="cat photo URL">
    <button type="submit">Submit</button>
  </form>
</main>
```


freecodecamp.org/learn/responsive-web-design/basic-html-and-html5/use-html5-to-require-a-field
freeCodeCamp
Diseño web receptivo heredado
HTML básico y HTML5
Use HTML5 para requerir un campo
Puede requerir campos de formulario específicos para que su usuario no pueda enviar su formulario hasta que los haya completado.
Por ejemplo, si desea que un campo de entrada de texto sea obligatorio, simplemente puede agregar el atributo required dentro de su input elemento, así: <input type="text" required>
Convierta su texto input en un required campo, para que su usuario no pueda enviar el formulario sin completar este campo.
Luego intente enviar el formulario sin ingresar ningún texto. ¿Ves cómo tu formulario HTML5 te notifica que el campo es obligatorio?
Ejecutar las pruebas (Ctrl + Enter)
Restablecer todo el código
Consigue ayuda
Pruebas
Su input elemento de texto debe tener el required atributo.
CatPhotoApp
Click here to view more cat photos.
Things cats love:
cat nip
laser pointers
lasagna
Top 3 things cats hate:
flea treatment
thunder
other cats
Submit


¡Al Batimóvil!
HTML básico y HTML5
71% complete
Enviar e ir al siguiente desafío (Ctrl + Enter)
Descarga mi solución

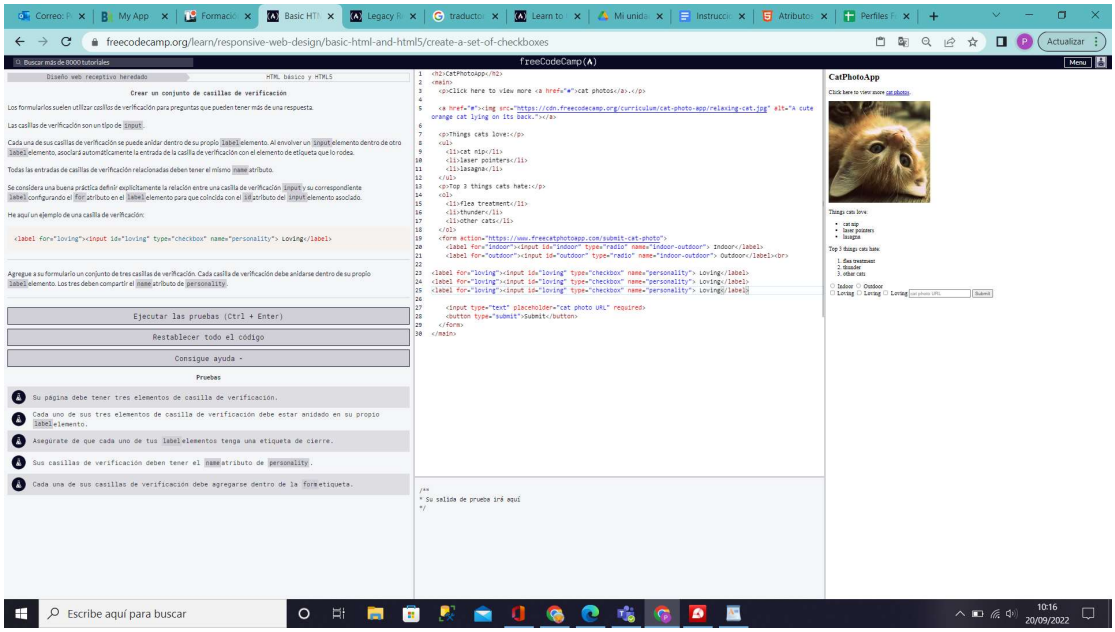

freecodecamp.org/learn/responsive-web-design/basic-html-and-html5/create-a-set-of-radio-buttons
freeCodeCamp(A)
Diseño web receptivo heredado
HTML básico y HTML5
Crear un conjunto de botones de opción
Puede usar botones de radio para preguntas en las que desea que el usuario solo le de una respuesta entre múltiples opciones.
Los botones de radio son un tipo de input.
Cada uno de sus botones de radio se puede anidar dentro de su propio label elemento. Al envolver un input elemento dentro de otro label elemento, asociará automáticamente la entrada del botón de opción con el elemento de etiqueta que lo rodea.
Todos los botones de radio relacionados deben tener el mismo name atributo para crear un grupo de botones de radio. Al crear un grupo de radio, al seleccionar cualquier botón de radio individual, se anulará automáticamente la selección de los otros botones dentro del mismo grupo, lo que garantiza que el usuario solo proporcione una respuesta.
He aquí un ejemplo de un botón de opción:
<label>
<input type="radio" name="indoor-outdoor">Indoor
</label>
Se considera una buena práctica establecer un for atributo en el label elemento, con un valor que coincida con el valor del id atributo del input elemento. Esto permite que las tecnologías de asistencia creen una relación vinculada entre la etiqueta y el input elemento relacionado. Por ejemplo:
<input id="indoor" type="radio" name="indoor-outdoor">
<label for="indoor">Indoor</label>
También podemos anidar el input elemento dentro de las label etiquetas:
<label for="indoor">
<input id="indoor" type="radio" name="indoor-outdoor">Indoor
</label>
Agregue un par de botones de radio a su formulario, cada uno anidado en su propio label elemento. Uno debe tener la opción de indoor y el otro debe tener la opción de outdoor. Ambos deben compartir el name atributo de indoor-outdoor crear un grupo de radio.
Ejecutar las pruebas (Ctrl + Enter)
Restablecer todo el código
Consigue ayuda -
Pruebas
Su pagina debe tener dos radio elementos de botón.
Sus botones de radio deben tener el name atributo de indoor-outdoor.
Cada uno de sus dos elementos de botón de radio debe estar anidado en su propio label elemento.
CatPhotoApp
Click here to view more cat photos.
Things cats love:
cat nip
laser pointers
lasagna
Top 3 things cats hate:
1. flea treatment
2. thunder
3. other cats
Cat photo URL
Submit
Indoor
outdoor
Escribe aquí para buscar
24°C Mayorm. nubl...
1:40
20/09/2022

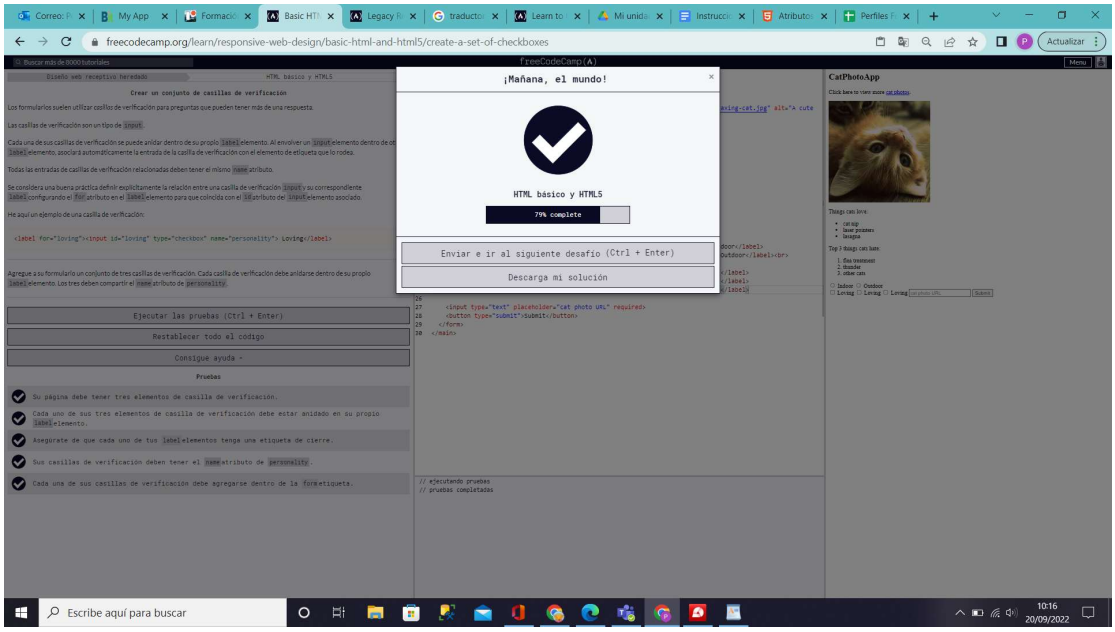

freecodecamp.org/learn/responsive-web-design/basic-html-and-html5/create-a-set-of-radio-buttons
freeCodeCamp(A)
Todos los botones de radio relacionados deben tener el mismo name atributo para crear un grupo de botones de radio. Al crear un grupo de radio, al seleccionar cualquier botón de radio individual, se anulará automáticamente la selección de los otros botones dentro del mismo grupo, lo que garantiza que el usuario solo proporcione una respuesta.
He aquí un ejemplo de un botón de opción:
Se considera una buena práctica establecer un for atributo en el label elemento, con un valor que coincida con el valor del id atributo del input elemento. Esto permite que las tecnologías de asistencia creen una relación vinculada entre la etiqueta y el input elemento relacionado. Por ejemplo:
También podemos anidar el input elemento dentro de las label etiquetas:
Agregue un par de botones de radio a su formulario, cada uno anidado en su propio label elemento. Uno debe tener la opción de indoor y el otro debe tener la opción de outdoor. Ambos deben compartir el name atributo de indoor-outdoor crear un grupo de radio.
Ejecutar las pruebas (Ctrl + Enter)
Restablecer todo el código
Consigue ayuda -
Pruebas
Su pagina debe tener dos radio elementos de botón.
Sus botones de radio deben tener el name atributo de indoor-outdoor.
Cada uno de sus dos elementos de botón de radio debe estar anidado en su propio label elemento.
Cada uno de sus label elementos debe tener una etiqueta de cierre.
Uno de sus botones de opción debe tener la etiqueta indoor.
Uno de sus botones de opción debe tener la etiqueta outdoor.
Cada uno de los elementos de su botón de radio debe agregarse dentro de la form etiqueta.
CatPhotoApp
Click here to view more cat photos.
Things cats love:
cat nip
laser pointers
lasagna
Top 3 things cats hate:
1. flea treatment
2. thunder
3. other cats
Cat photo URL
Submit
Indoor
outdoor
Escribe aquí para buscar
24°C Mayorm. nubl...
1:41
20/09/2022




```
<h2>CatPhotoApp</h2>

<main>

  <p>Click here to view more <a href="#">cat photos</a>.</p>


  <a  href="#"><img  src="https://cdn.freecodecamp.org/curriculum/cat-photo-app/relaxing-cat.jpg" alt="A cute orange cat lying on its back."></a>


  <p>Things cats love:</p>

  <ul>

    <li>cat nip</li>

    <li>laser pointers</li>

    <li>lasagna</li>

  </ul>

  <p>Top 3 things cats hate:</p>

  <ol>

    <li>flea treatment</li>

    <li>thunder</li>
```

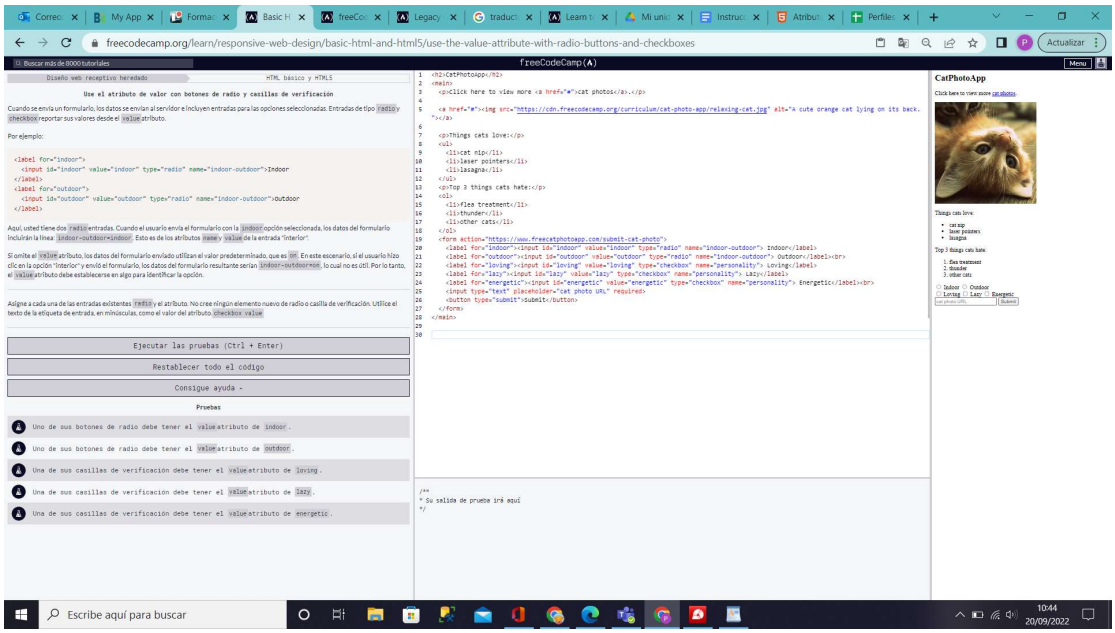
```
  <li>other cats</li>
 </ol>
 <form action="https://www.freecatphotoapp.com/submit-cat-photo">
  <label for="indoor"><input id="indoor" type="radio" name="indoor-outdoor">
Indoor</label>
  <label for="outdoor"><input id="outdoor" type="radio" name="indoor-outdoor">
Outdoor</label><br>
  <input type="text" placeholder="cat photo URL" required>
  <button type="submit">Submit</button>
 </form>
</main>
```

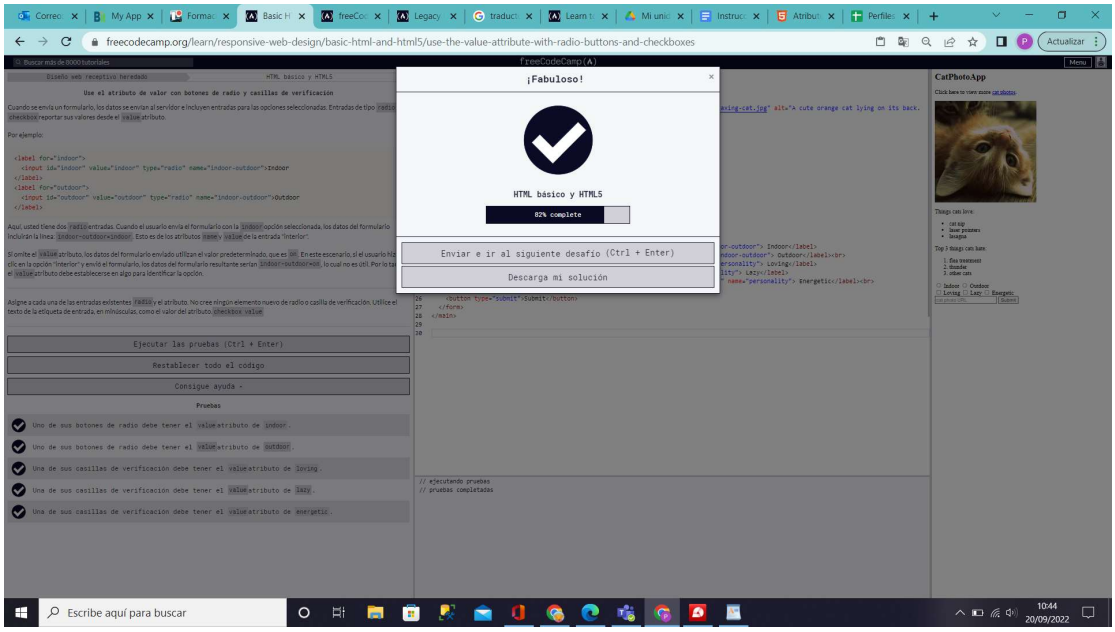





```
<h2>CatPhotoApp</h2>

<main>

  <p>Click here to view more <a href="#">cat photos</a>.</p>


  <a  href="#"><img  src="https://cdn.freecodecamp.org/curriculum/cat-photo-app/relaxing-cat.jpg" alt="A cute orange cat lying on its back."></a>


  <p>Things cats love:</p>

  <ul>

    <li>cat nip</li>

    <li>laser pointers</li>

    <li>lasagna</li>

  </ul>

  <p>Top 3 things cats hate:</p>

  <ol>

    <li>flea treatment</li>

    <li>thunder</li>
```

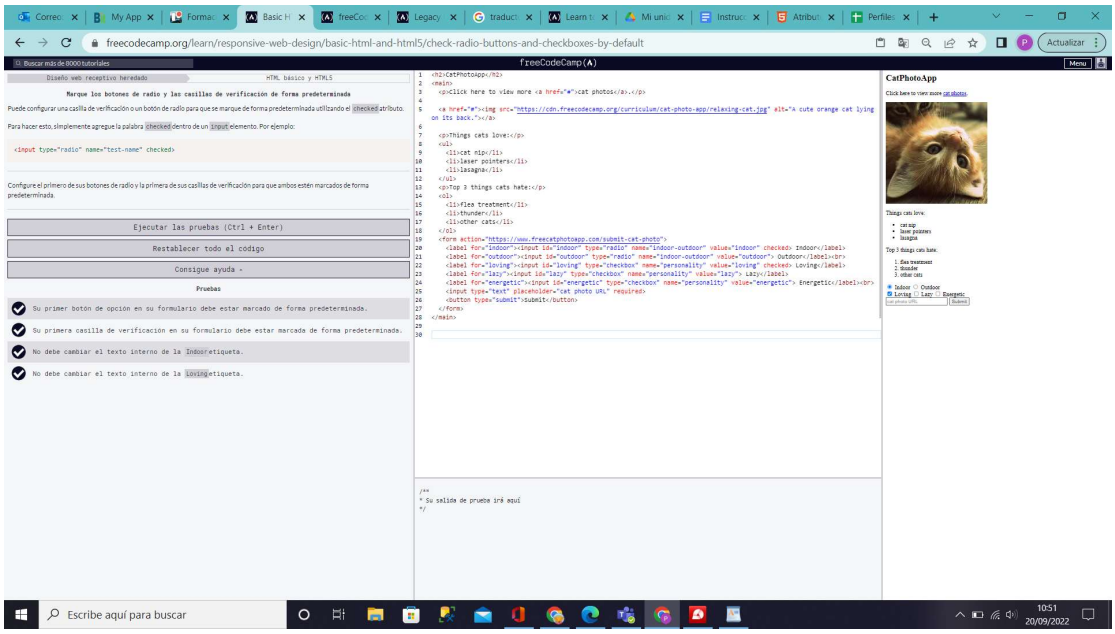
```
  <li>other cats</li>

 </ol>

 <form action="https://www.freecatphotoapp.com/submit-cat-photo">

  <label for="indoor"><input id="indoor" type="radio" name="indoor-outdoor">
Indoor</label>

  <label for="outdoor"><input id="outdoor" type="radio" name="indoor-outdoor">
Outdoor</label><br>

  <label for="loving"><input id="loving" type="checkbox" name="personality">
Loving</label>

  <label for="lazy"><input id="lazy" type="checkbox" name="personality"> Lazy</label>

  <label for="energetic"><input id="energetic" type="checkbox" name="personality">
Energetic</label><br>

  <input type="text" placeholder="cat photo URL" required>

  <button type="submit">Submit</button>

 </form>

</main>
```

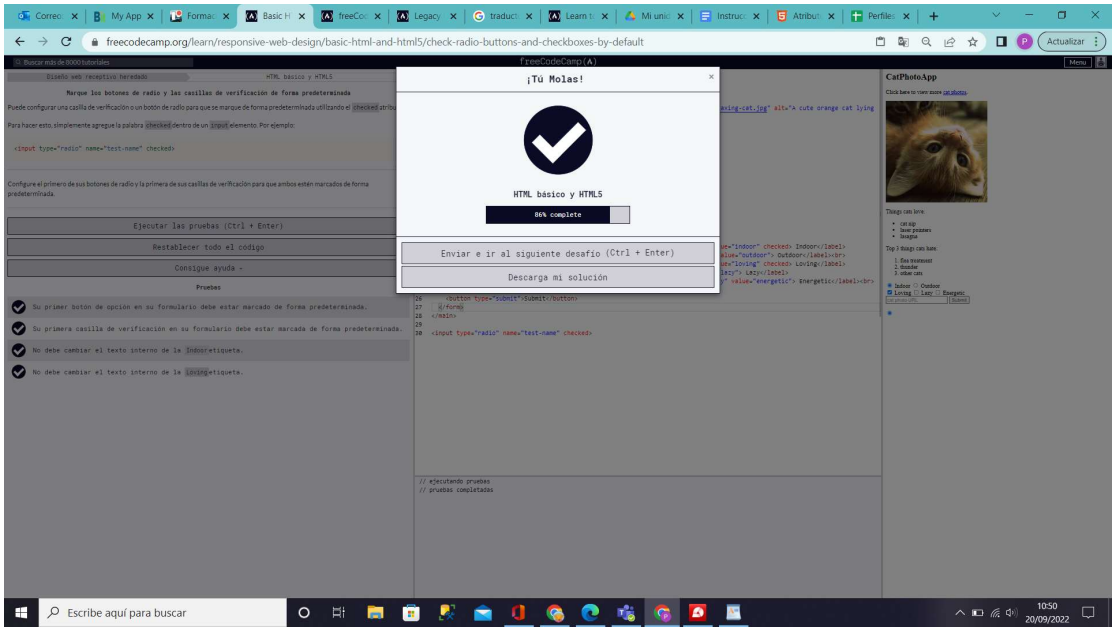





```
<h2>CatPhotoApp</h2>

<main>

  <p>Click here to view more <a href="#">cat photos</a>.</p>


  <a  href="#"><img  src="https://cdn.freecodecamp.org/curriculum/cat-photo-app/relaxing-cat.jpg" alt="A cute orange cat lying on its back."></a>


  <p>Things cats love:</p>

  <ul>

    <li>cat nip</li>

    <li>laser pointers</li>

    <li>lasagna</li>

  </ul>

  <p>Top 3 things cats hate:</p>

  <ol>

    <li>flea treatment</li>

    <li>thunder</li>
```

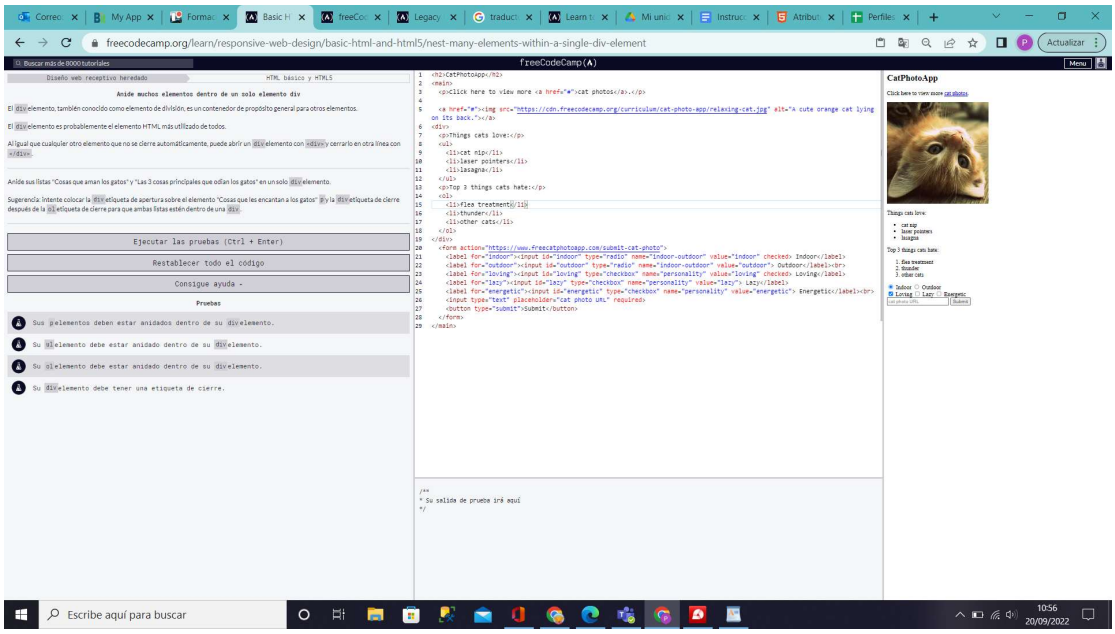
```
  <li>other cats</li>

 </ol>

 <form action="https://www.freecatphotoapp.com/submit-cat-photo">

  <label for="indoor"><input id="indoor" value="indoor" type="radio" name="indoor-outdoor"> Indoor</label>

  <label for="outdoor"><input id="outdoor" value="outdoor" type="radio" name="indoor-outdoor"> Outdoor</label><br>

  <label for="loving"><input id="loving" value="loving" type="checkbox" name="personality"> Loving</label>

  <label for="lazy"><input id="lazy" value="lazy" type="checkbox" name="personality"> Lazy</label>

  <label for="energetic"><input id="energetic" value="energetic" type="checkbox" name="personality"> Energetic</label><br>

  <input type="text" placeholder="cat photo URL" required>

  <button type="submit">Submit</button>

 </form>

</main>
```

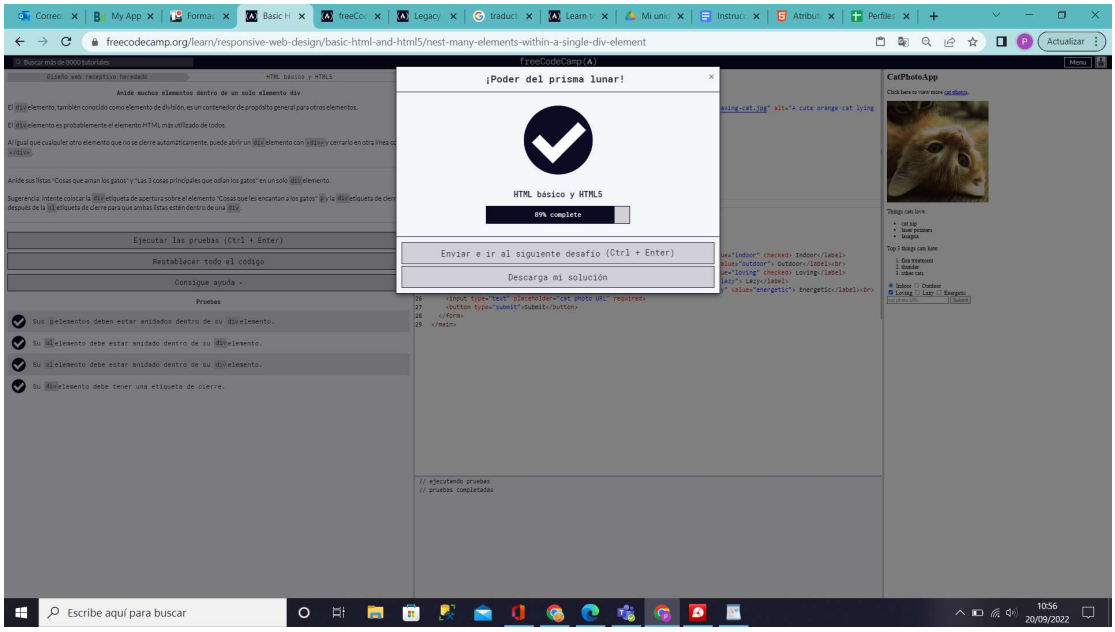


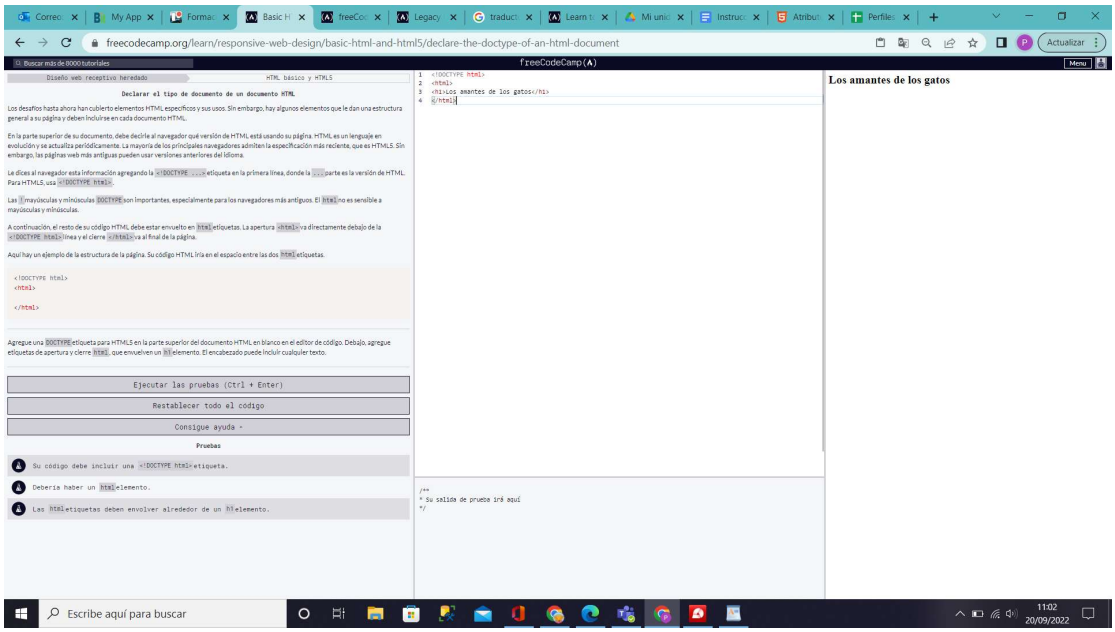


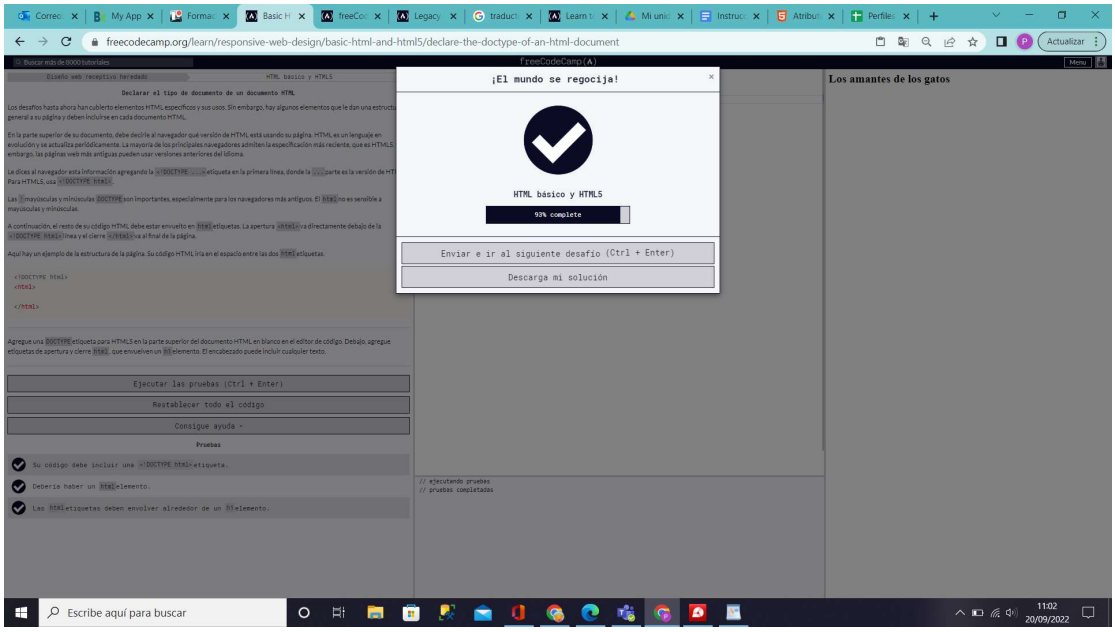

```
<h2>CatPhotoApp</h2>

<main>

  <p>Click here to view more <a href="#">cat photos</a>.</p>


  <a  href="#"><img  src="https://cdn.freecodecamp.org/curriculum/cat-photo-app/relaxing-cat.jpg" alt="A cute orange cat lying on its back."></a>


  <p>Things cats love:</p>

  <ul>

    <li>cat nip</li>

    <li>laser pointers</li>

    <li>lasagna</li>

  </ul>

  <p>Top 3 things cats hate:</p>

  <ol>

    <li>flea treatment</li>

    <li>thunder</li>
```

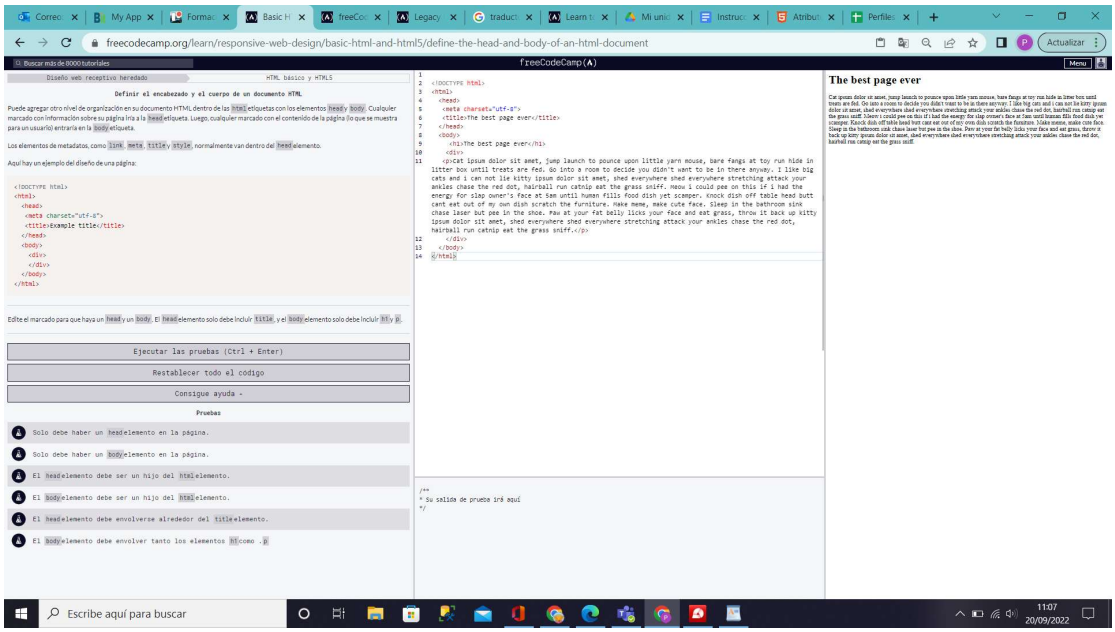
```
  <li>other cats</li>

 </ol>

 <form action="https://www.freecatphotoapp.com/submit-cat-photo">

  <label for="indoor"><input id="indoor" type="radio" name="indoor-outdoor" value="indoor" checked> Indoor</label>

  <label for="outdoor"><input id="outdoor" type="radio" name="indoor-outdoor" value="outdoor"> Outdoor</label><br>

  <label for="loving"><input id="loving" type="checkbox" name="personality" value="loving" checked> Loving</label>

  <label for="lazy"><input id="lazy" type="checkbox" name="personality" value="lazy"> Lazy</label>

  <label for="energetic"><input id="energetic" type="checkbox" name="personality" value="energetic"> Energetic</label><br>

  <input type="text" placeholder="cat photo URL" required>

  <button type="submit">Submit</button>

 </form>

</main>
```

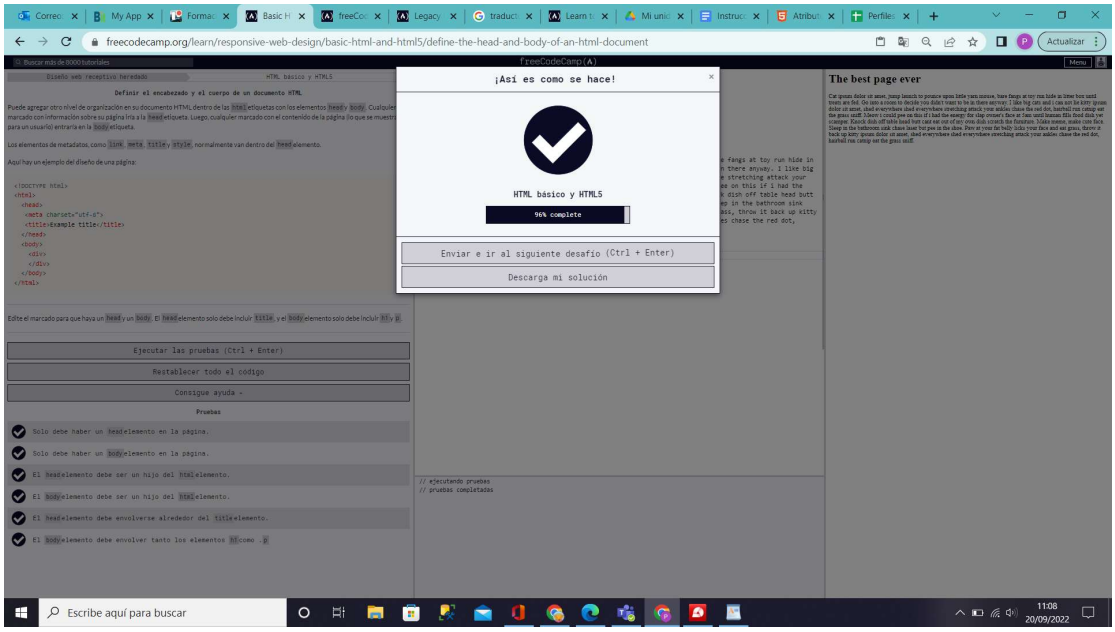



freecodecamp.org/learn/responsive-web-design/basic-html-and-html5/nest-many-elements-within-a-single-div-element
¡Poder del prisma lunar!
HTML básico y HTML5
89% complete
Enviar e ir al siguiente desafío (Ctrl + Enter)
Descarga mi solución
Ejecutar las pruebas (Ctrl + Enter)
Restablecer todo el código
Consigue ayuda
Pruebas
CatPhotoApp
Escribe aquí para buscar
10:56
20/09/2022


freecodecamp.org/learn/responsive-web-design/basic-html-and-html5/declare-the-doctype-of-an-html-document
Los amantes de los gatos
Ejecutar las pruebas (Ctrl + Enter)
Restablecer todo el código
Consigue ayuda
Pruebas
Escribe aquí para buscar
11:02
20/09/2022

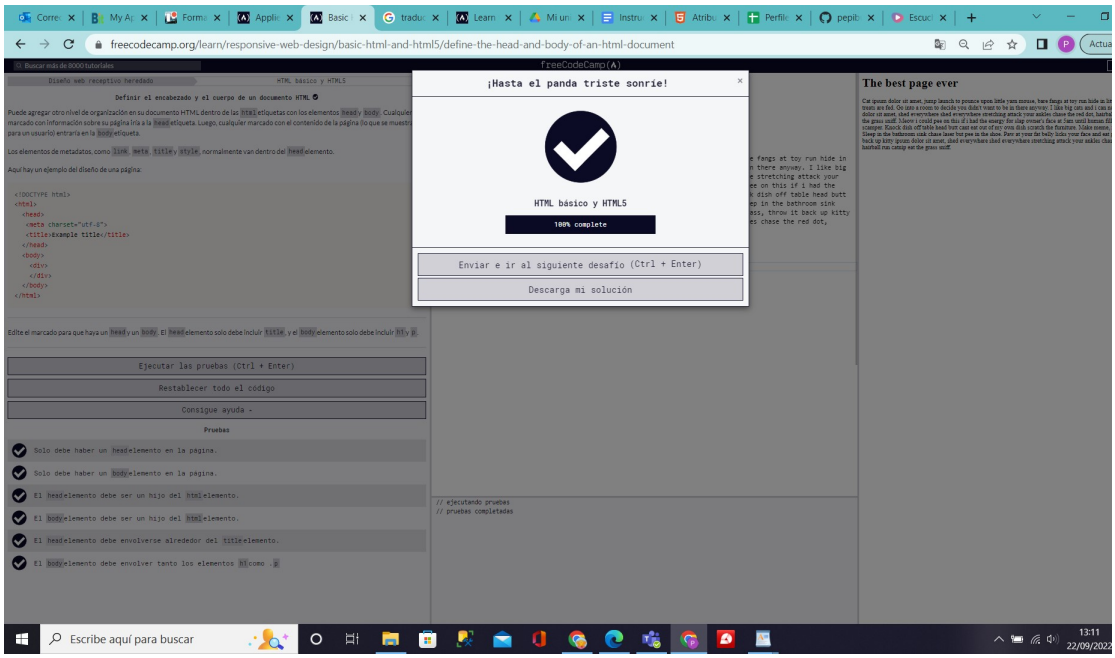

freecodecamp.org/learn/responsive-web-design/basic-html-and-html5/declare-the-doctype-of-an-html-document
¡El mundo se regocija!
HTML básico y HTML5
92% complete
Enviar e ir al siguiente desafío (Ctrl + Enter)
Descarga mi solución
Los amantes de los gatos
Escribe aquí para buscar
11:02
20/09/2022

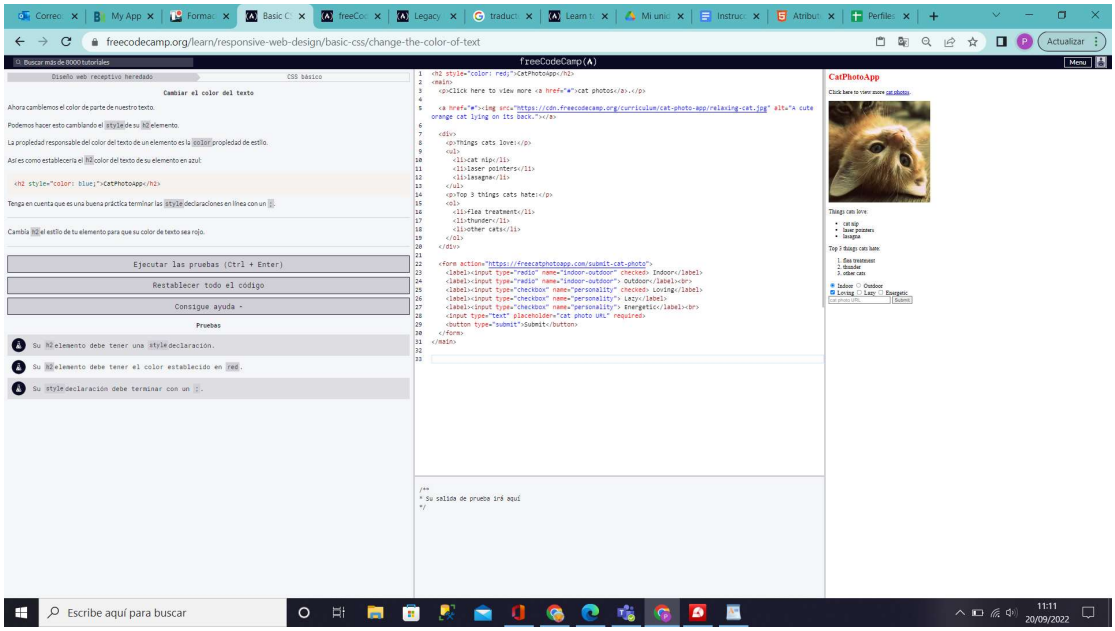

freecodecamp.org/learn/responsive-web-design/basic-html-and-html5/define-the-head-and-body-of-an-html-document
The best page ever
Escribe aquí para buscar
11:07
20/09/2022

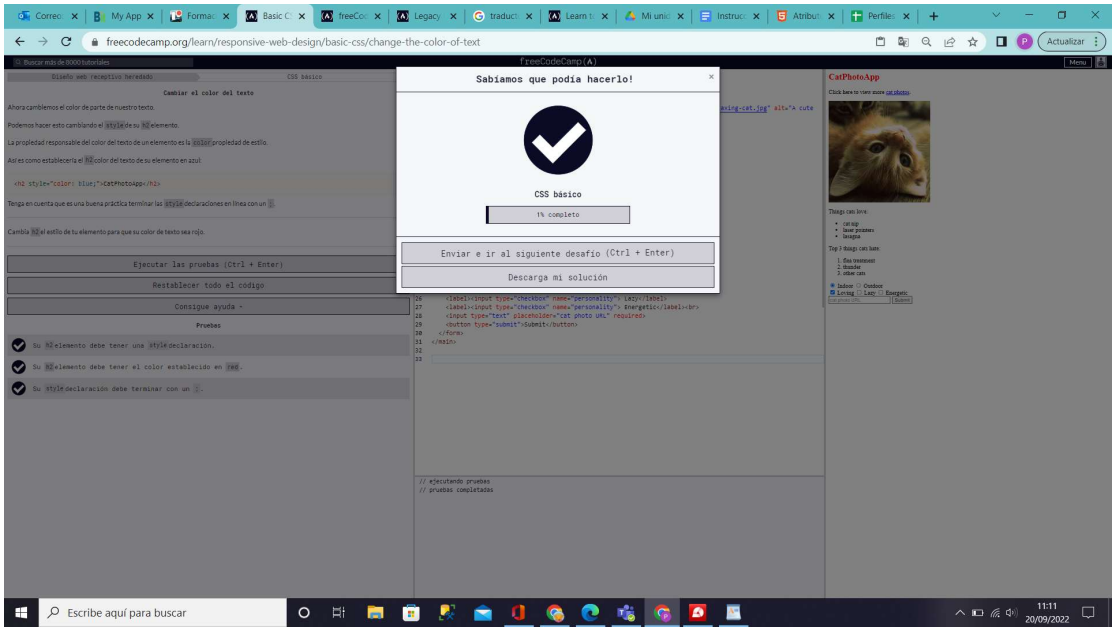



```
<!DOCTYPE html>

<html>

  <head>

   <meta charset="utf-8">

   <title>The best page ever</title>

  </head>

  <body>

    <h1>The best page ever</h1>

   <div>

   <p>Cat ipsum dolor sit amet, jump launch to pounce upon little yarn mouse, bare fangs at toy run hide in litter box until treats are fed. Go into a room to decide you didn't want to be in there anyway. I like big cats and i can not lie kitty ipsum dolor sit amet, shed everywhere shed everywhere stretching attack your ankles chase the red dot, hairball run catnip eat the grass sniff. Meow i could pee on this if i had the energy for slap owner's face at 5am until human fills food dish yet scamper. Knock dish off table head butt cant eat out of my own dish scratch the furniture. Make meme, make cute face. Sleep in the bathroom sink chase laser but pee in the shoe. Paw at your fat belly licks your face and eat grass, throw it back up kitty ipsum dolor sit amet, shed everywhere shed everywhere stretching attack your ankles chase the red dot, hairball run catnip eat the grass sniff.</p>

   </div>

  </body>
```

</html>


freecodecamp.org/learn/responsive-web-design/basic-html-and-html5/define-the-head-and-body-of-an-html-document
¡Hasta el panda triste sonríe!
HTML básico y HTML5
100% complete
Enviar e ir al siguiente desafío (Ctrl + Enter)
Descarga mi solución
The best page ever
Ejecutar las pruebas (Ctrl + Enter)
Restablecer todo el código
Consigue ayuda -
Pruebas
Escribe aquí para buscar


freecodecamp.org/learn/responsive-web-design/basic-css/change-the-color-of-text
Cambiar el color del texto
Ejecutar las pruebas (Ctrl + Enter)
Restablecer todo el código
Consigue ayuda -
Pruebas
CatPhotoApp
Escribe aquí para buscar

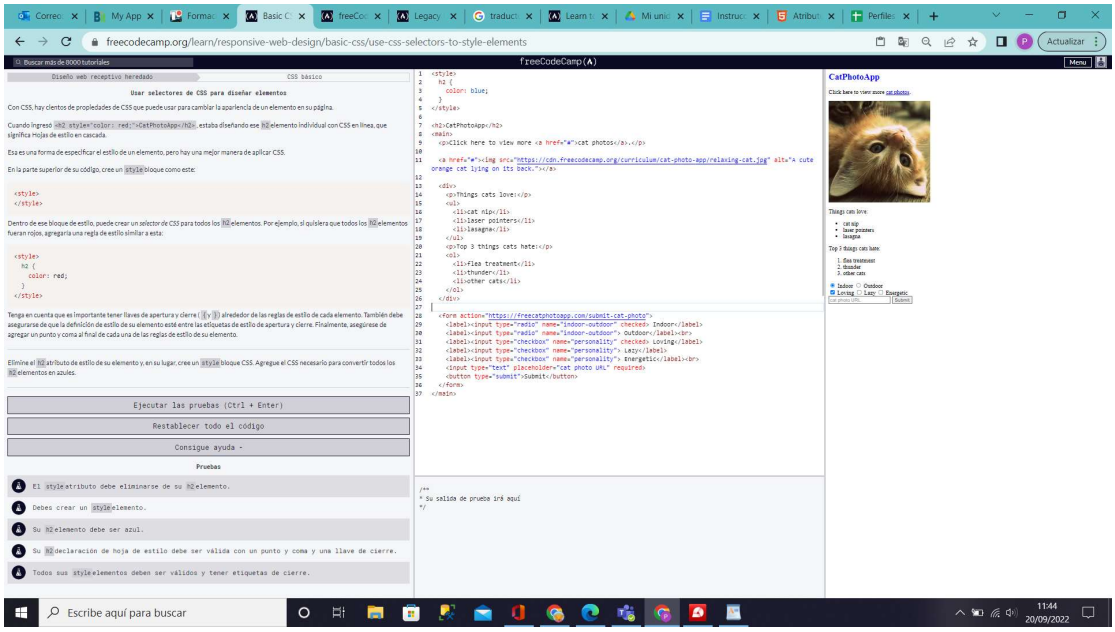


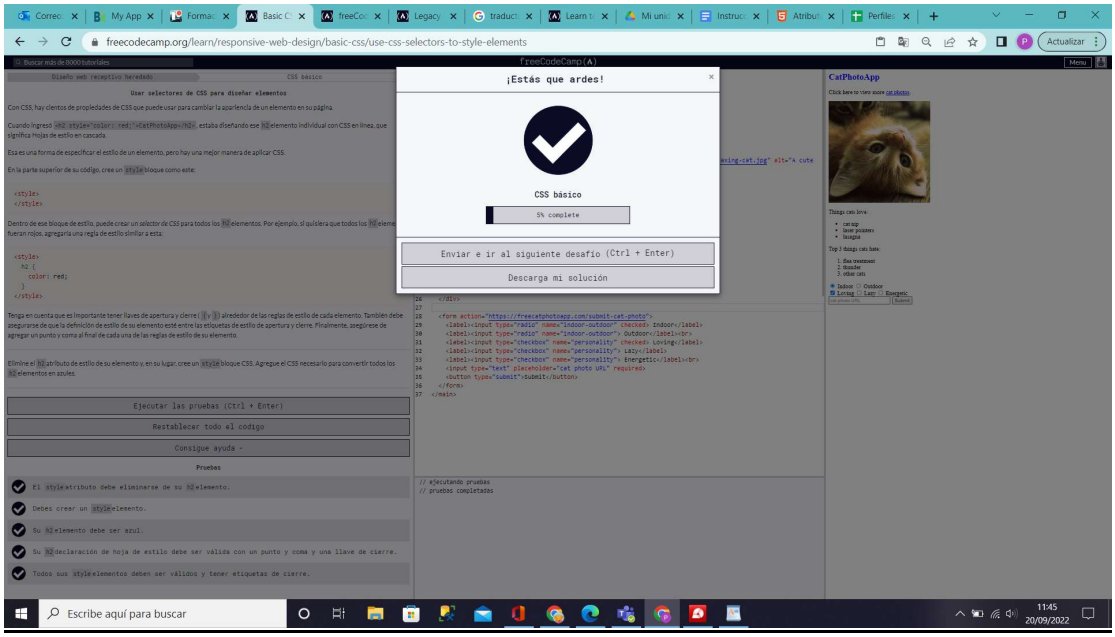
<h2 style="color: red;">CatPhotoApp</h2>

<u>**CSS**</u>: (En línea - dentro de html)

Las clases son **estilos reutilizables** que se pueden agregar a elementos HTML.

CSS: Hojas de estilo en cascada.

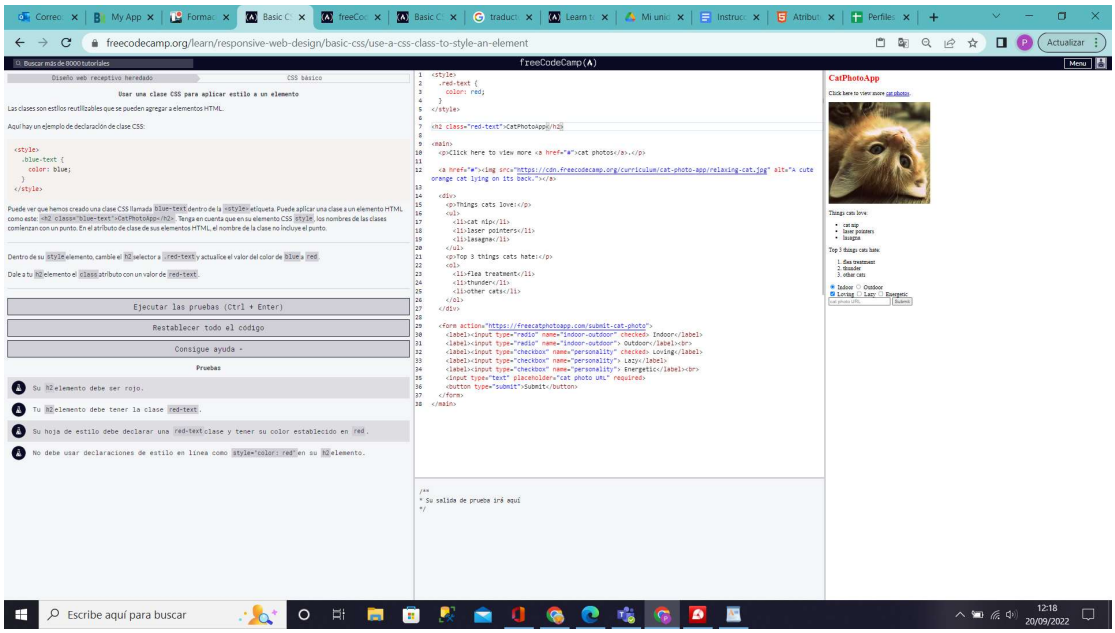


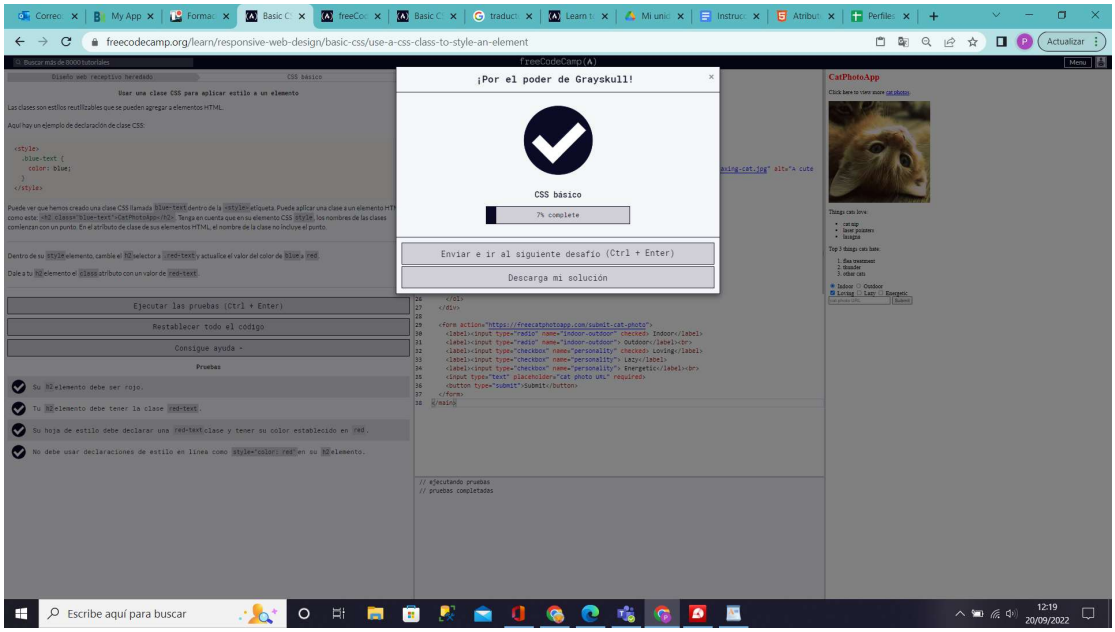



```
<style>
  h2 {
    color: blue;
  }
</style>
```

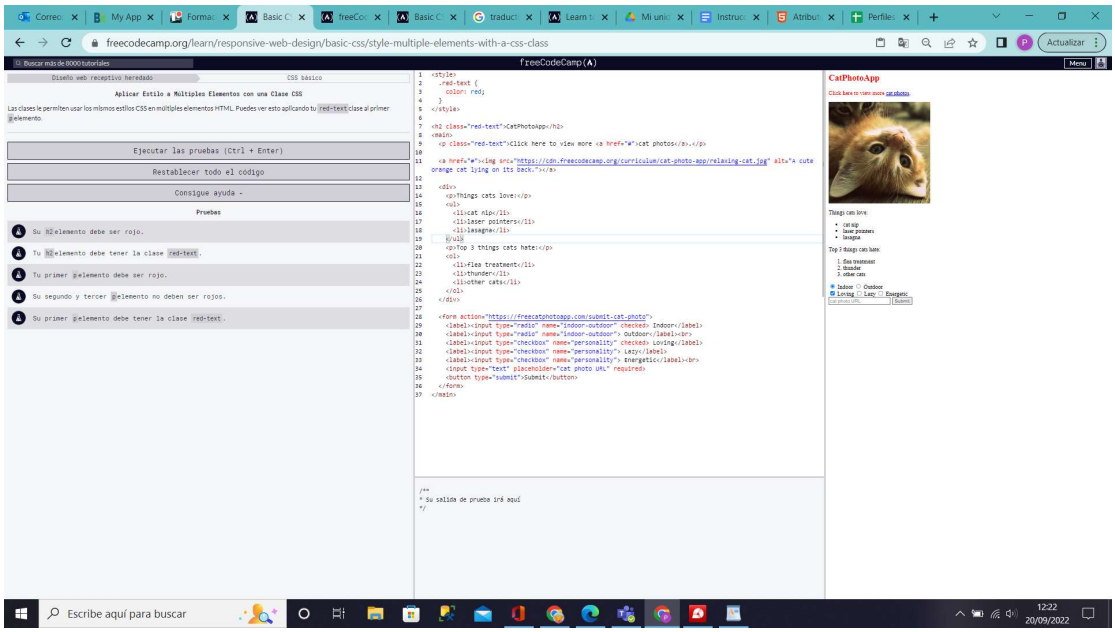
```
<h2>CatPhotoApp</h2>

<main>

  <p>Click here to view more <a href="#">cat photos</a>.</p>


  <a  href="#"><img  src="https://cdn.freecodecamp.org/curriculum/cat-photo-app/relaxing-cat.jpg" alt="A cute orange cat lying on its back."></a>


  <div>

    <p>Things cats love:</p>

    <ul>

      <li>cat nip</li>

      <li>laser pointers</li>

      <li>lasagna</li>

    </ul>

    <p>Top 3 things cats hate:</p>

    <ol>

      <li>flea treatment</li>

      <li>thunder</li>

      <li>other cats</li>

    </ol>

  </div>


  <form action="https://freecatphotoapp.com/submit-cat-photo">

    <label><input type="radio" name="indoor-outdoor" checked> Indoor</label>

    <label><input type="radio" name="indoor-outdoor"> Outdoor</label><br>

    <label><input type="checkbox" name="personality" checked> Loving</label>

    <label><input type="checkbox" name="personality"> Lazy</label>

    <label><input type="checkbox" name="personality"> Energetic</label><br>
```

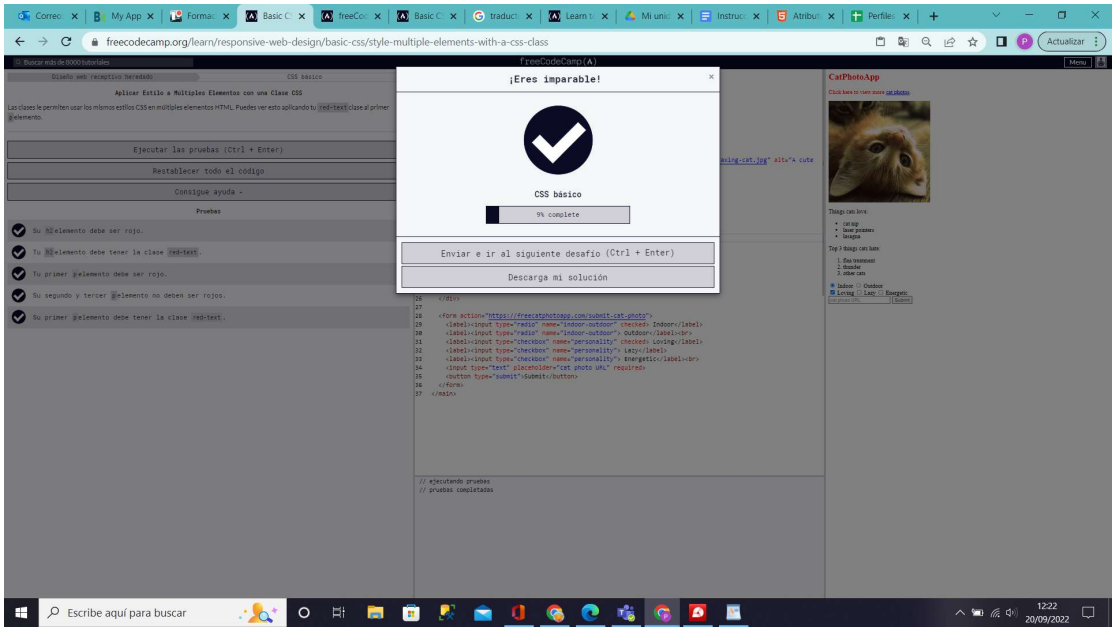
```
  <input type="text" placeholder="cat photo URL" required>

  <button type="submit">Submit</button>

 </form>

</main>
```





```
<style>
  .red-text {
    color: red;
  }
</style>
<h2 class="red-text">CatPhotoApp</h2>
```

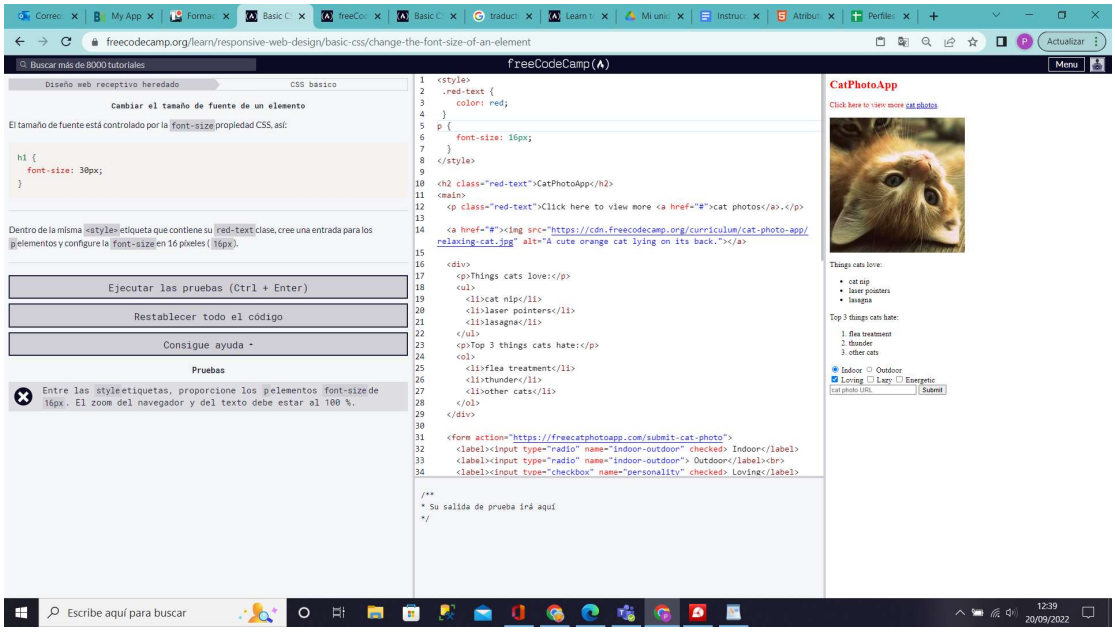


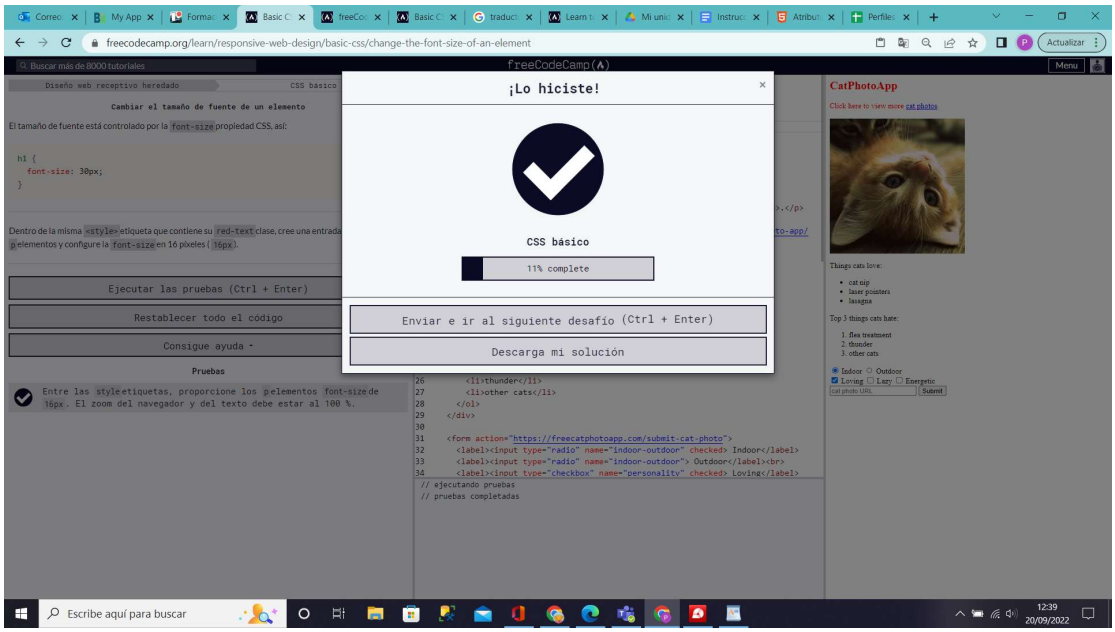



```
<style>
  .red-text {
    color: red;
  }
</style>
<h2 class="red-text">CatPhotoApp</h2>
<main>
  <p class="red-text">Click here to view more <a href="#">cat photos</a>.</p>
```

Cambiar el tamaño de fuente de un elemento

El tamaño de fuente está controlado por la **font-size** propiedad CSS, así:



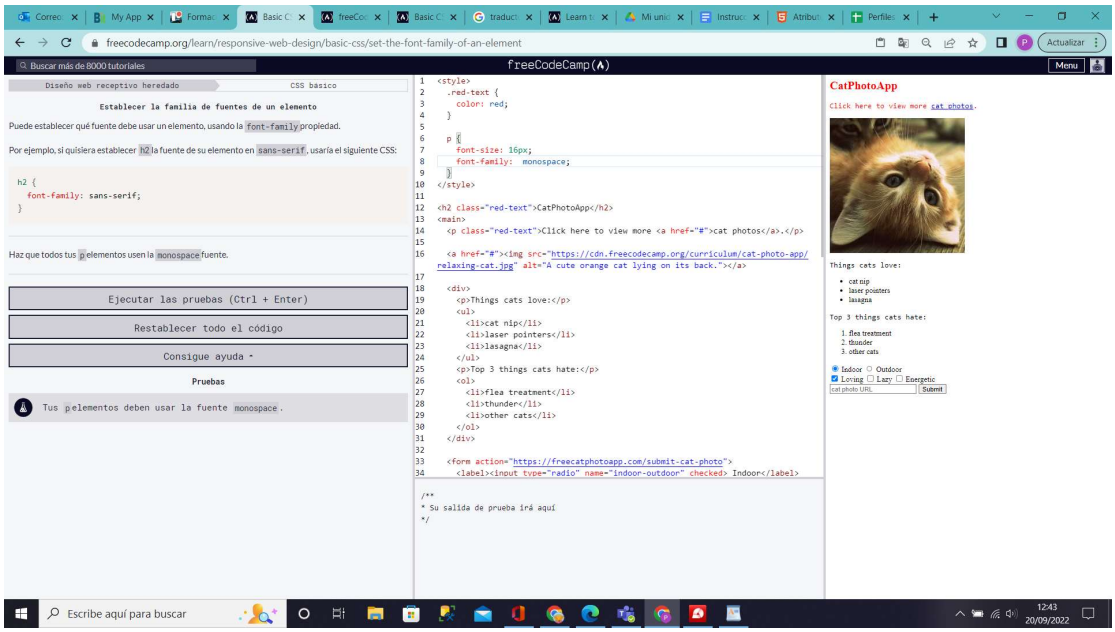


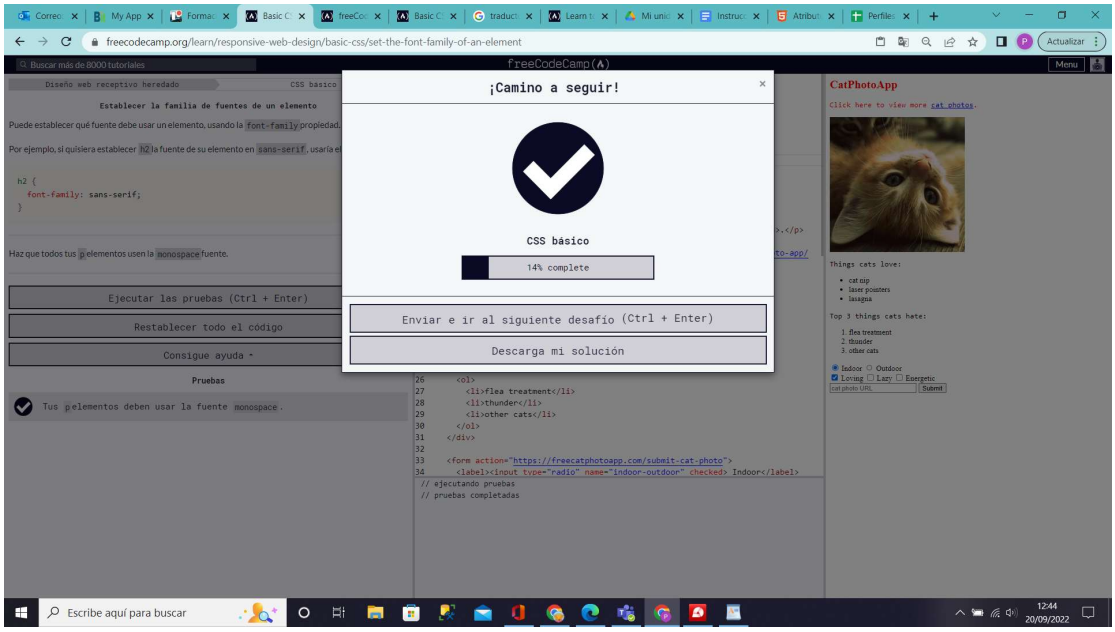

```
<style>
 .red-text {
   color: red;
 }
p {
```

```
  font-size: 16px;
 }
</style>
<h2 class="red-text">CatPhotoApp</h2>
<main>
 <p class="red-text">Click here to view more <a href="#">cat photos</a>.</p>
```

Establecer la familia de fuentes de un elemento

Puede establecer qué fuente debe usar un elemento, usando la **font-family** propiedad.






```
<style>
 .red-text {
  color: red;
 }
 p {
  font-size: 16px;
  font-family:  monospace;
 }
</style>
<h2 class="red-text">CatPhotoApp</h2>
<main>
 <p class="red-text">Click here to view more <a href="#">cat photos</a>.</p>
```

Importar una fuente de Google

Además de especificar fuentes comunes que se encuentran en la mayoría de los sistemas operativos, también podemos especificar fuentes web personalizadas no estándar para usar en nuestro sitio web. Hay muchas fuentes de fuentes web en Internet. Para este ejemplo, nos centraremos en la biblioteca de Google Fonts.

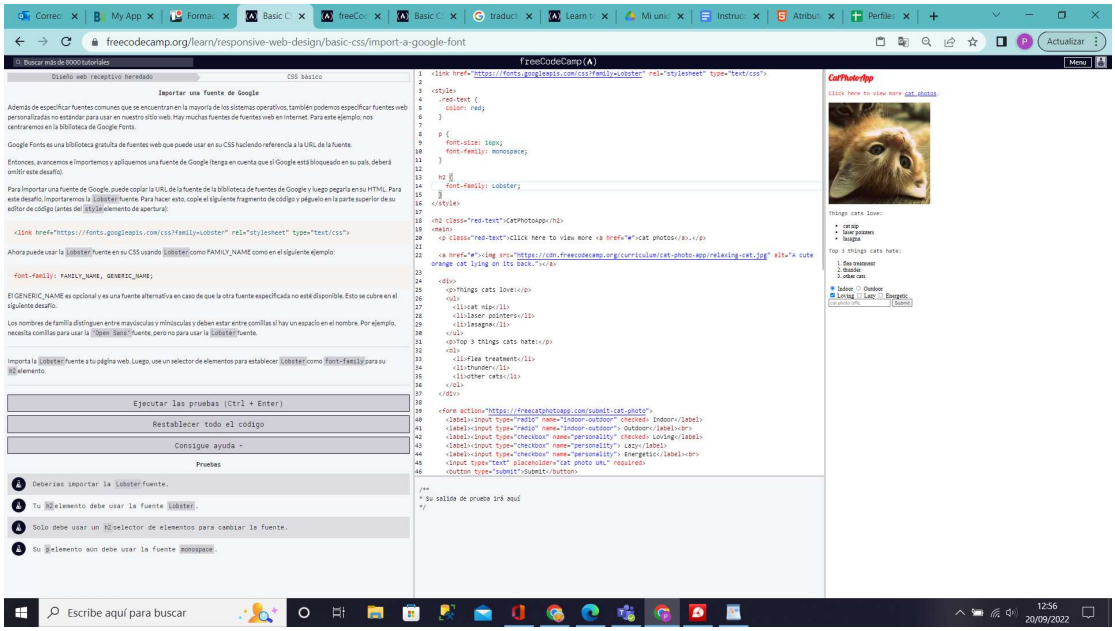
Google Fonts es una biblioteca gratuita de fuentes web que puede usar en su CSS haciendo referencia a la URL de la fuente.

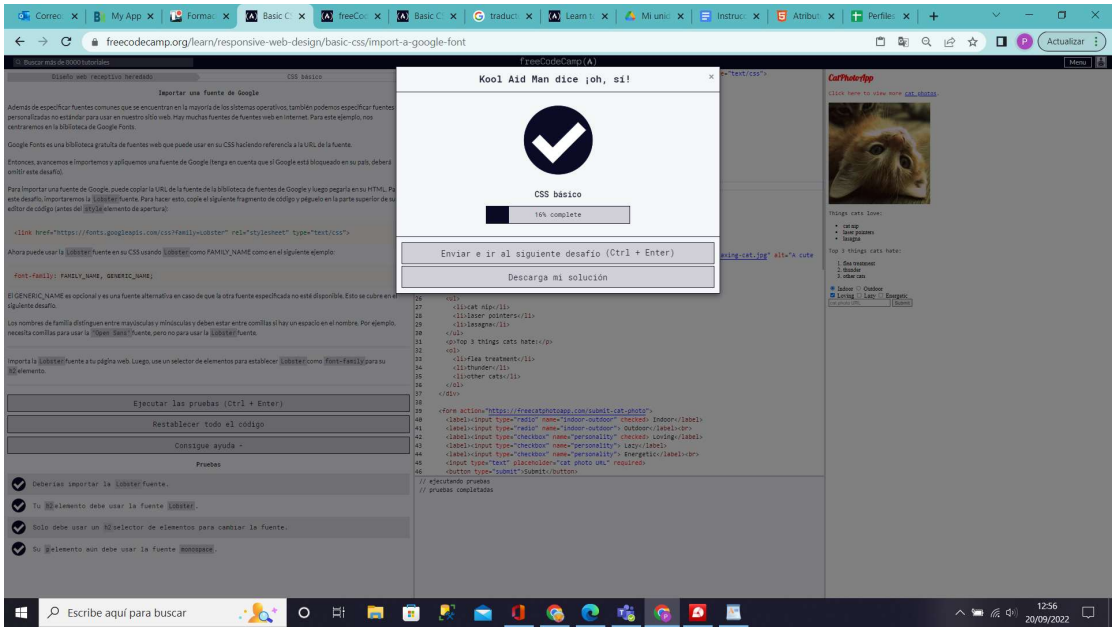
Entonces, avancemos e importemos y apliquemos una fuente de Google (tenga en cuenta que si Google está bloqueado en su país, deberá omitir este desafío).

Para importar una fuente de Google, puede copiar la URL de la fuente de la biblioteca de fuentes de Google y luego pegarla en su HTML. Para este desafío, importaremos la **Lobster**fuente.

Para hacer esto, copie el siguiente fragmento de código y péguelo en la parte superior de su editor de código (antes del styleelemento de apertura):

<link href="https://fonts.googleapis.com/css?family=Lobster" rel="stylesheet" type="text/css">






```
<link href="https://fonts.googleapis.com/css?family=Lobster" rel="stylesheet" type="text/css">

<style>

 .red-text {

  color: red;

 }

 p {

  font-size: 16px;

  font-family: monospace;

 }

 h2 {

  font-family: Lobster;

 }

</style>

<h2 class="red-text">CatPhotoApp</h2>

<main>

 <p class="red-text">Click here to view more <a href="#">cat photos</a>.</p>
```

Especifique cómo deben degradarse las fuentes

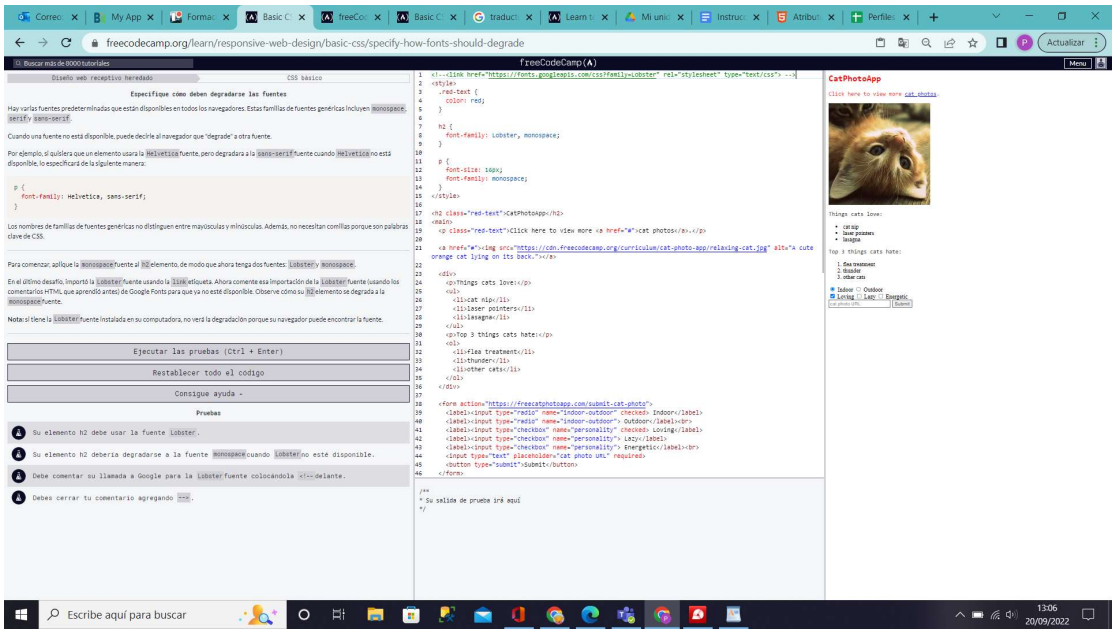
Hay varias fuentes predeterminadas que están disponibles en todos los navegadores. Estas familias de fuentes genéricas incluyen **monospace**, **serif** y **sans-serif**.

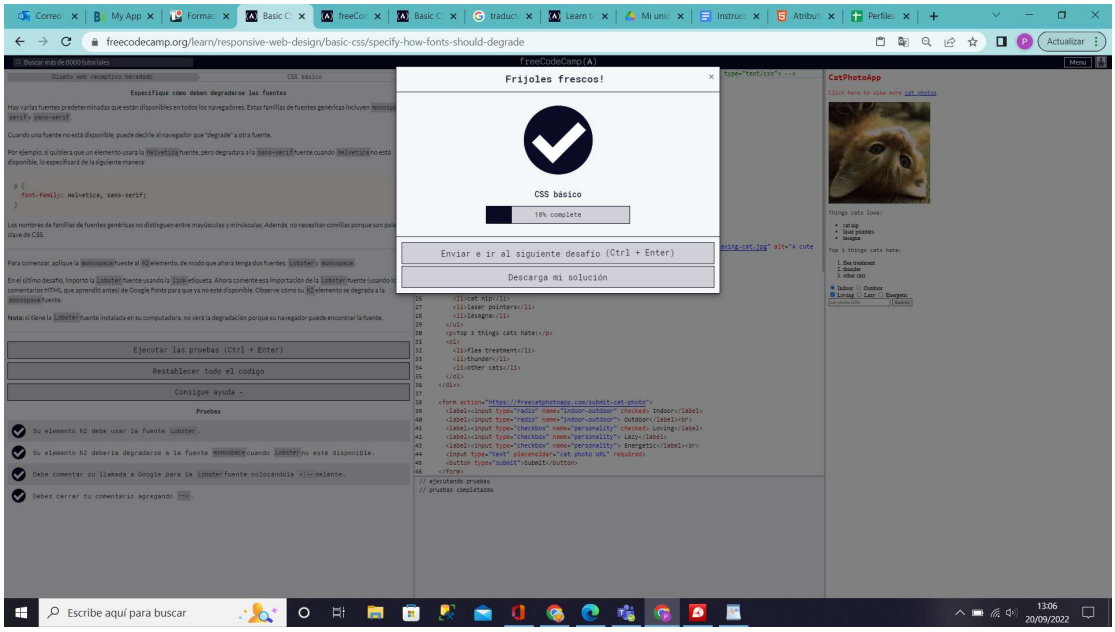
Cuando una fuente no está disponible, puede decirle al navegador que "degrade" a otra fuente.

Por ejemplo, si quisiera que un elemento usara la Helveticafuente, pero degradara a la sans-seriffuente cuando Helveticano está disponible, lo especificará de la siguiente manera:

```
p {
  font-family: Helvetica, sans-serif;
}
```

Los nombres de familias de fuentes genéricas no distinguen entre mayúsculas y minúsculas. Además, no necesitan comillas porque son palabras clave de CSS.





<!--<link href="https://fonts.googleapis.com/css?family=Lobster" rel="stylesheet" type="text/css"> -->

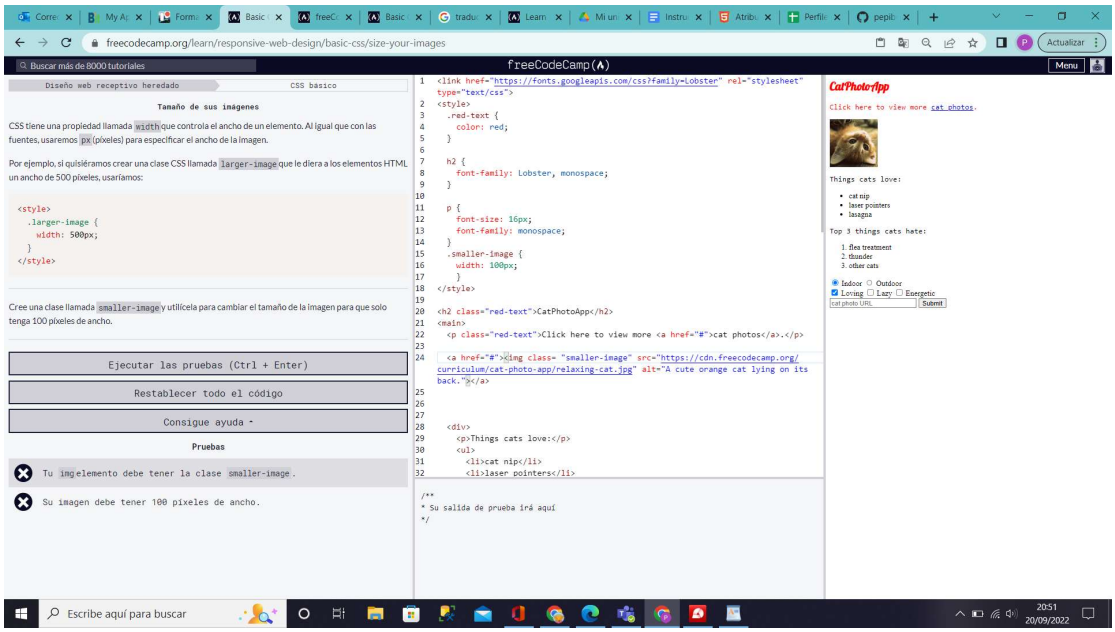
```
<style>
 .red-text {
  color: red;
 }
 h2 {
  font-family: Lobster, monospace;
 }
 p {
  font-size: 16px;
  font-family: monospace;
 }
</style>
<h2 class="red-text">CatPhotoApp</h2>
<main>
 <p class="red-text">Click here to view more <a href="#">cat photos</a>.</p>
```

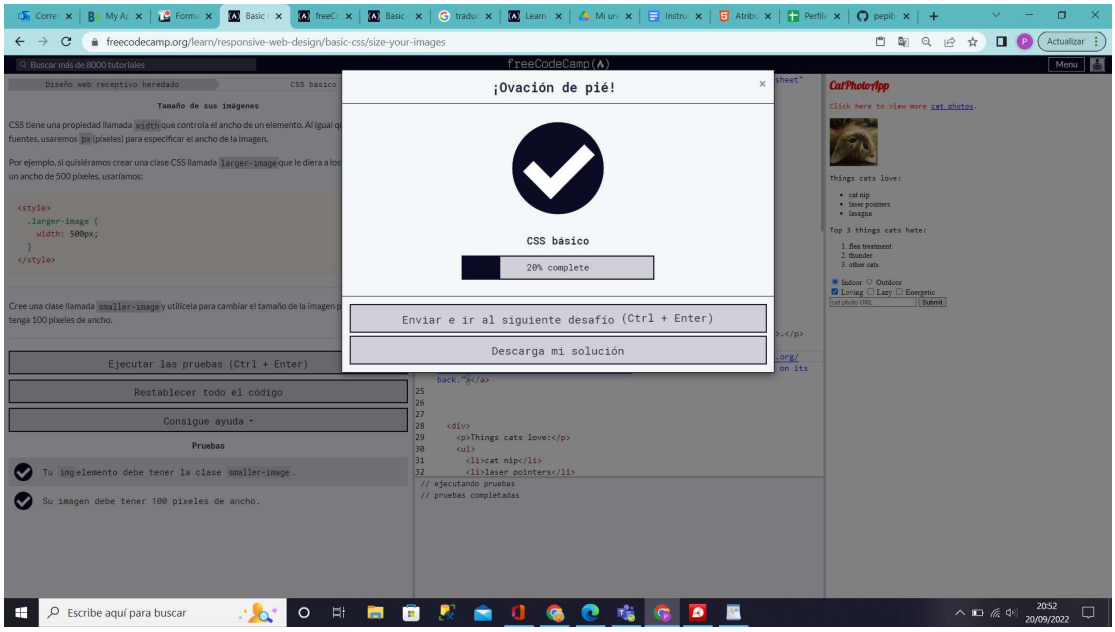
Tamaño de sus imágenes

CSS tiene una propiedad llamada **width** que controla el **ancho** de un elemento. Al igual que con las fuentes, usaremos **px**(píxeles) para especificar el ancho de la imagen.

Por ejemplo, si quisiéramos crear una clase CSS llamada larger-imageque le diera a los elementos HTML un ancho de 500 píxeles, usaríamos:

```
<style>
 .larger-image {
   width: 500px;
 }
</style>
```






```
<link href="https://fonts.googleapis.com/css?family=Lobster" rel="stylesheet" type="text/css">

<style>

 .red-text {

  color: red;

 }

 h2 {

  font-family: Lobster, monospace;

 }

 p {

  font-size: 16px;

  font-family: monospace;

 }

 .smaller-image {

  width: 100px;

  }

</style>
```

```
<h2 class="red-text">CatPhotoApp</h2>

<main>

  <p class="red-text">Click here to view more <a href="#">cat photos</a>.</p>

  <a href="#"><img class= "smaller-image" src="https://cdn.freecodecamp.org/curriculum/cat-photo-app/relaxing-cat.jpg" alt="A cute orange cat lying on its back."></a>
```

Agregue bordes alrededor de sus elementos

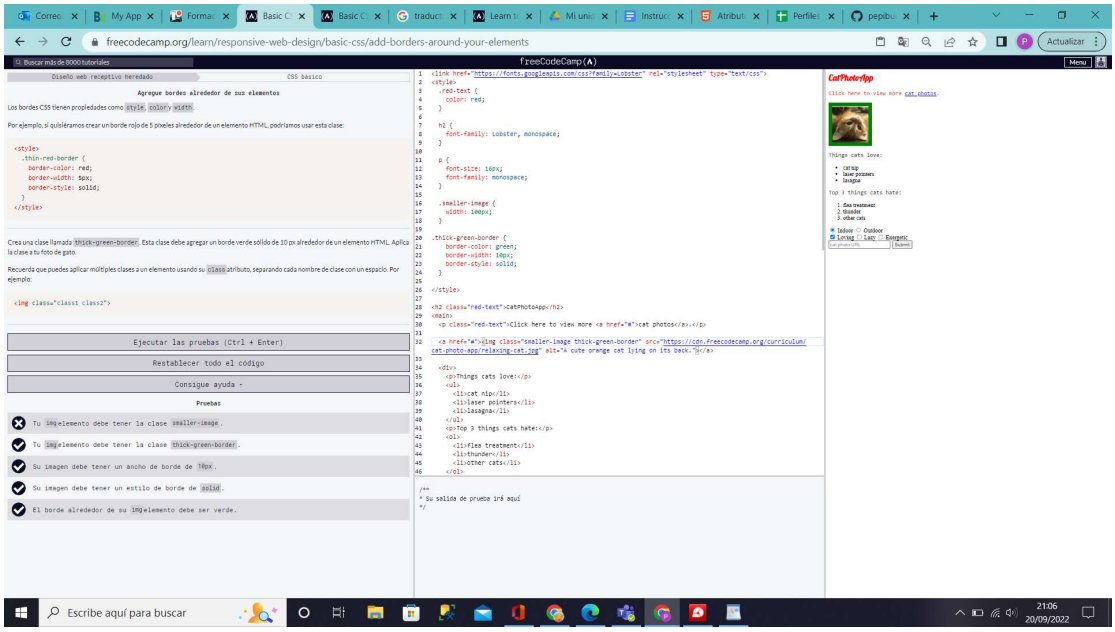
Los bordes CSS tienen propiedades como **style**, **color** y **width.**

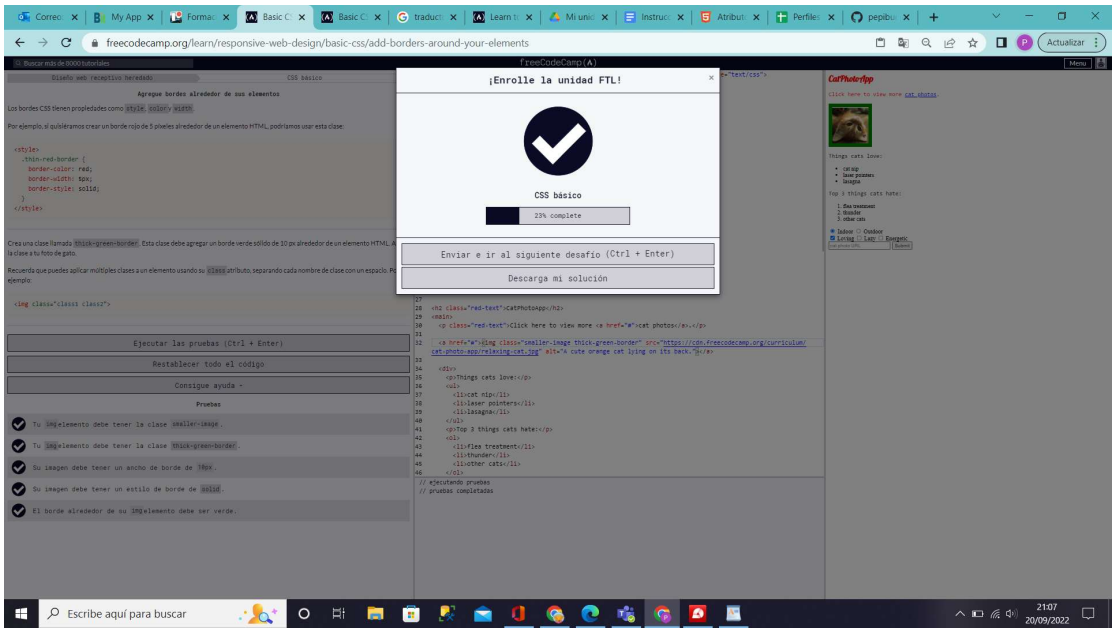
Por ejemplo, si quisiéramos crear un borde rojo de 5 píxeles alrededor de un elemento HTML, podríamos usar esta clase:

```
<style>

  .thin-red-border {

    border-color: red;

    border-width: 5px;

    border-style: solid;

  }

</style>
```

Recuerda que puedes aplicar múltiples clases a un elemento usando su classatributo, separando cada nombre de clase con un espacio. Por ejemplo:

```
<img class="class1 class2">
```





<link href="https://fonts.googleapis.com/css?family=Lobster" rel="stylesheet" type="text/css">

<style>

.red-text {

color: red;

}

```
 h2 {
  font-family: Lobster, monospace;
 }
 p {
  font-size: 16px;
  font-family: monospace;
 }
 .smaller-image {
  width: 100px;
 }
.thick-green-border {
  border-color: green;
  border-width: 10px;
  border-style: solid;
 }
</style>
<h2 class="red-text">CatPhotoApp</h2>
<main>
 <p class="red-text">Click here to view more <a href="#">cat photos</a>.</p>
```

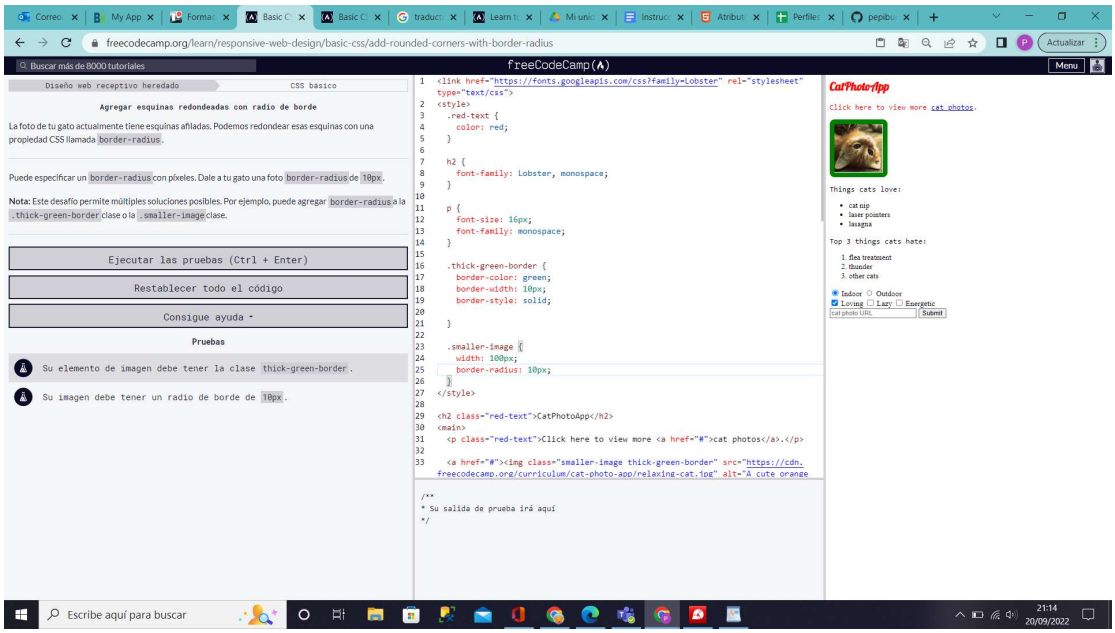
<a href="#"><img class="smaller-image **thick-green-border**" src="https://cdn.freecodecamp.org/curriculum/cat-photo-app/relaxing-cat.jpg" alt="A cute orange cat lying on its back."></a>

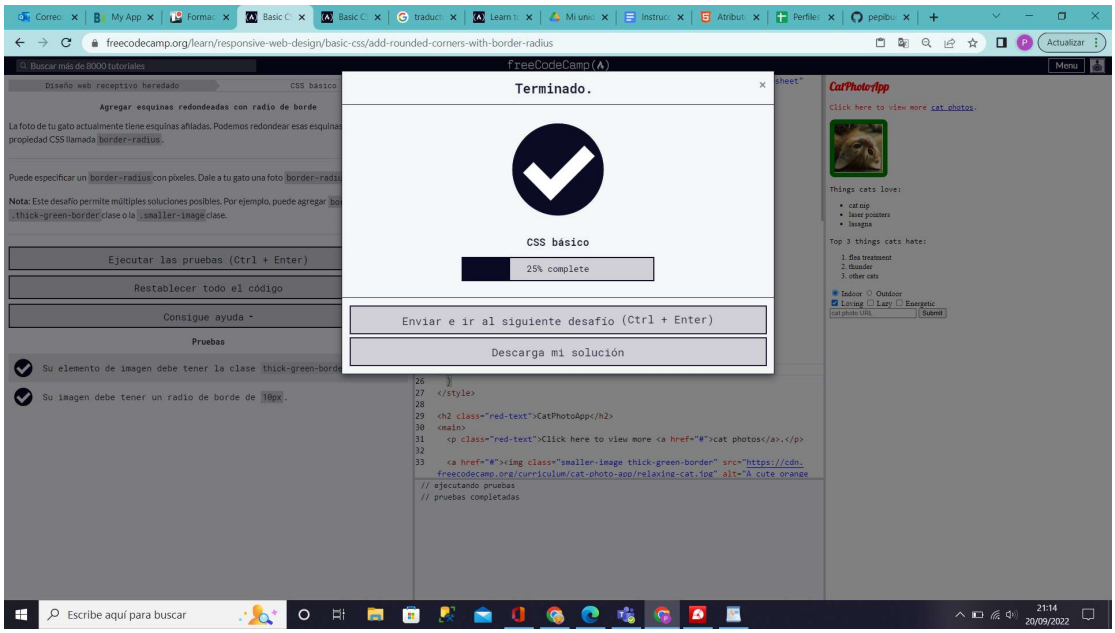
Agregar esquinas redondeadas con radio de borde

La foto de tu gato actualmente tiene esquinas afiladas. Podemos redondear esas esquinas con una propiedad CSS llamada **border-radius**.

Puede especificar un border-radius con píxeles. Dale a tu gato una foto border-radiusde **10px**.

Nota: Este desafío permite múltiples soluciones posibles. Por ejemplo, puede agregar border-radiusa la .thick-green-borderclase o la .smaller-imageclase.





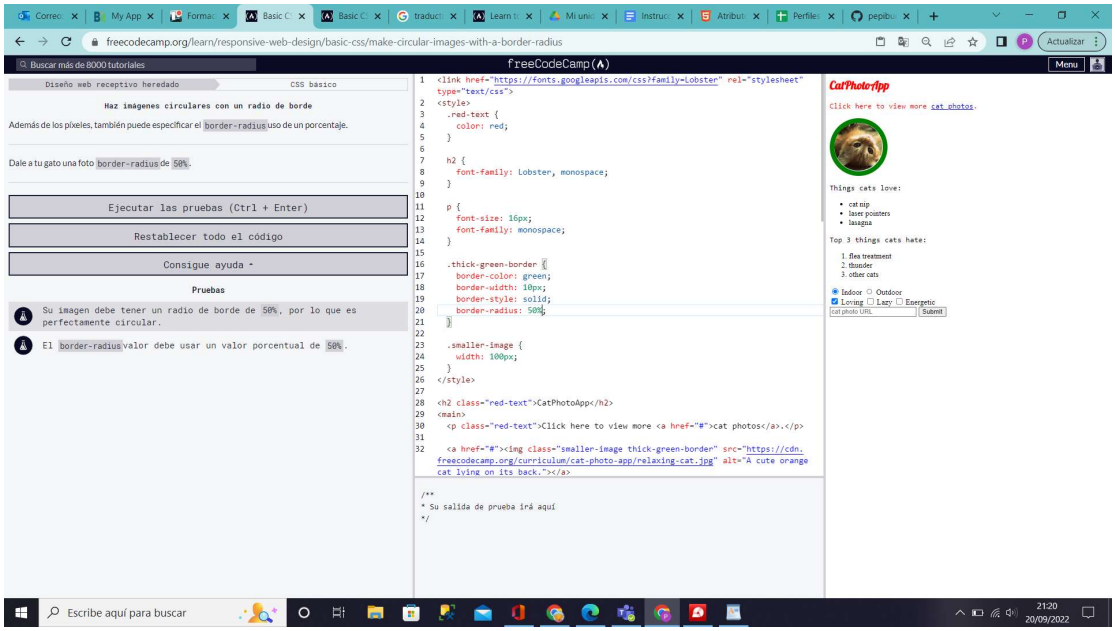
<link href="https://fonts.googleapis.com/css?family=Lobster" rel="stylesheet" type="text/css">

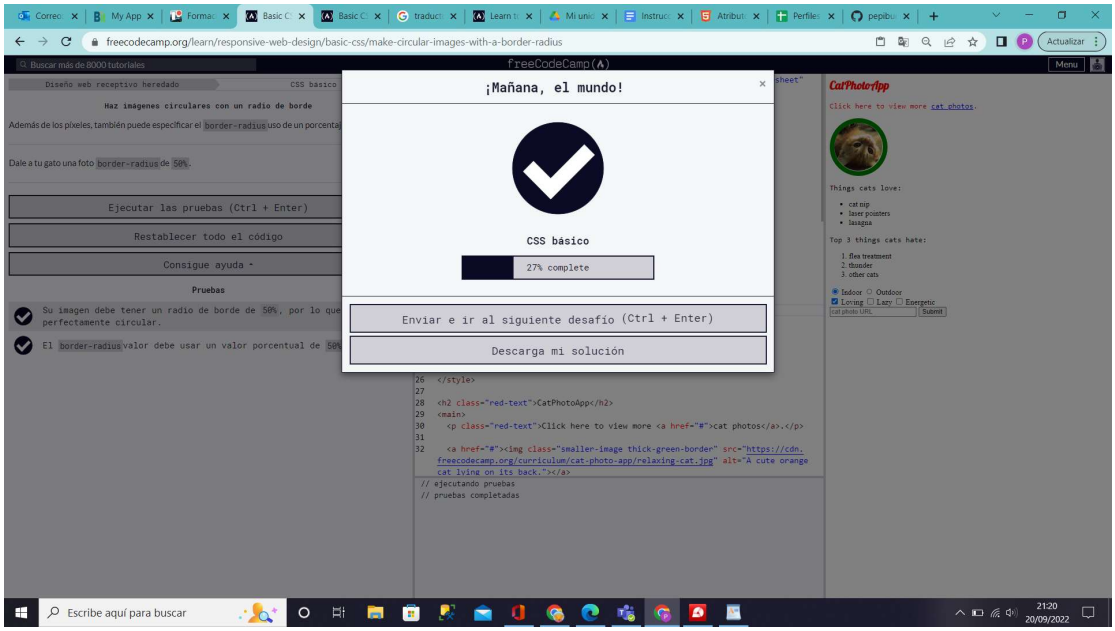
```
<style>
 .red-text {
  color: red;
 }
 h2 {
  font-family: Lobster, monospace;
 }
 p {
  font-size: 16px;
  font-family: monospace;
 }
 .thick-green-border {
  border-color: green;
  border-width: 10px;
  border-style: solid;
 }
 .smaller-image {
  width: 100px;
  border-radius: 10px;
 }
</style>
<h2 class="red-text">CatPhotoApp</h2>
<main>
 <p class="red-text">Click here to view more <a href="#">cat photos</a>.</p>
 <a href="#"><img class="smaller-image thick-green-border" src="https://cdn.freecodecamp.org/curriculum/cat-photo-app/relaxing-cat.jpg" alt="A cute orange cat lying on its back."></a>
```

Haz imágenes circulares con un radio de borde

Además de los píxeles, también puede especificar el **border-radius** uso de un **porcentaje**.

Dale a tu gato una foto border-radius de 50**%**.





<link href="https://fonts.googleapis.com/css?family=Lobster" rel="stylesheet"

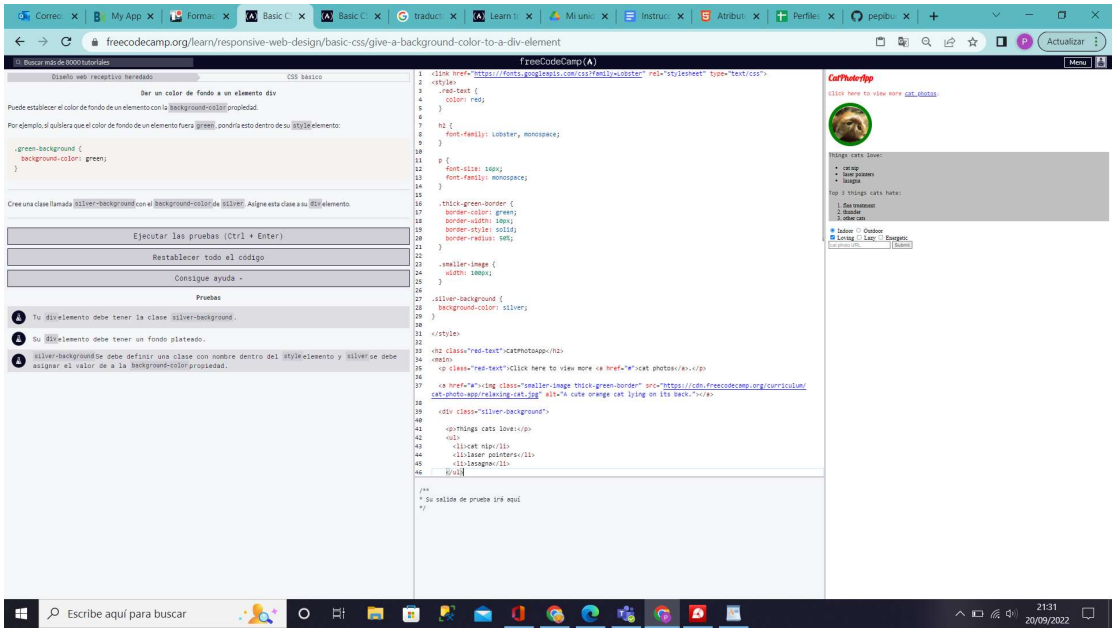
```
type="text/css">
<style>
 .red-text {
  color: red;
 }
 h2 {
  font-family: Lobster, monospace;
 }
 p {
  font-size: 16px;
  font-family: monospace;
 }
 .thick-green-border {
  border-color: green;
  border-width: 10px;
  border-style: solid;
  border-radius: 50%;
 }
 .smaller-image {
  width: 100px;
 }
</style>
<h2 class="red-text">CatPhotoApp</h2>
<main>
 <p class="red-text">Click here to view more <a href="#">cat photos</a>.</p>
 <a href="#"><img class="smaller-image thick-green-border" src="https://cdn.freecodecamp.org/curriculum/cat-photo-app/relaxing-cat.jpg" alt="A cute orange cat lying on its back."></a>
```

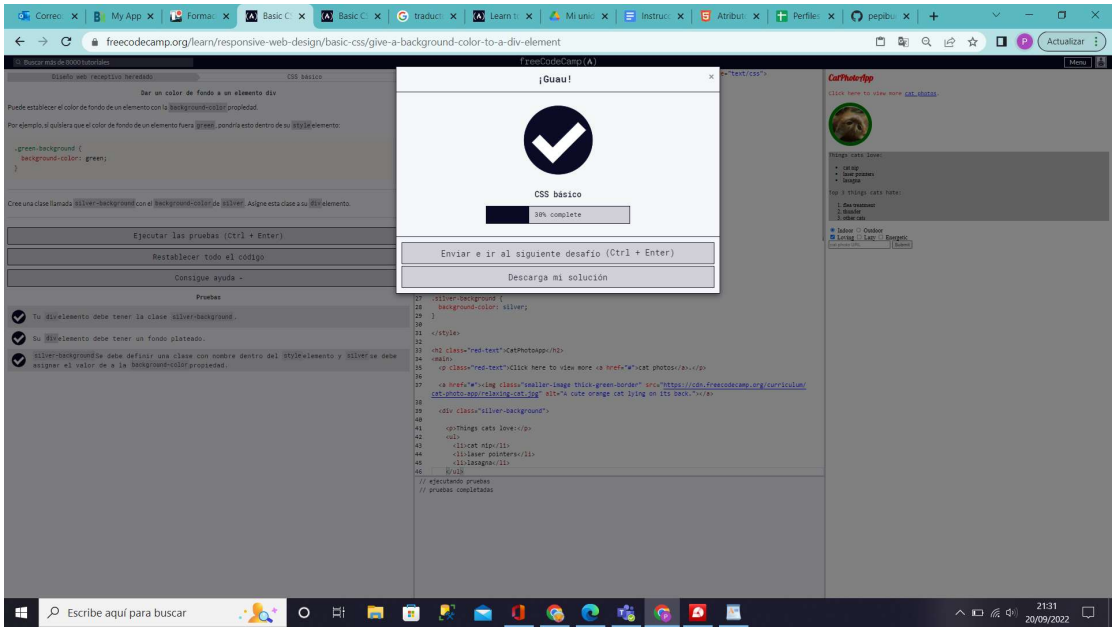
Dar un color de fondo a un elemento div

Puede establecer el color de fondo de un elemento con la **background-color** propiedad.

Por ejemplo, si quisiera que el color de fondo de un elemento fuera **green**, pondría esto dentro de su **style** elemento:

```
.green-background {
  background-color: green;
}
```






```
<link href="https://fonts.googleapis.com/css?family=Lobster" rel="stylesheet" type="text/css">

<style>

 .red-text {

  color: red;

 }

 h2 {

  font-family: Lobster, monospace;

 }

 p {

  font-size: 16px;

  font-family: monospace;

 }

 .thick-green-border {

  border-color: green;

  border-width: 10px;

  border-style: solid;
```

```
    border-radius: 50%;
  }
  .smaller-image {
    width: 100px;
  }
.silver-background {
  background-color: silver;
}
</style>
<h2 class="red-text">CatPhotoApp</h2>
<main>
  <p class="red-text">Click here to view more <a href="#">cat photos</a>.</p>
  <a href="#"><img class="smaller-image thick-green-border" src="https://cdn.freecodecamp.org/curriculum/cat-photo-app/relaxing-cat.jpg" alt="A cute orange cat lying on its back."></a>
  <div class="silver-background">
    <p>Things cats love:</p>
    <ul>
      <li>cat nip</li>
      <li>laser pointers</li>
      <li>lasagna</li>
    </ul>
    <p>Top 3 things cats hate:</p>
    <ol>
      <li>flea treatment</li>
      <li>thunder</li>
      <li>other cats</li>
    </ol>
  </div>
```

Establecer la identificación de un elemento

Además de las clases, cada elemento HTML también puede tener un **id** atributo.

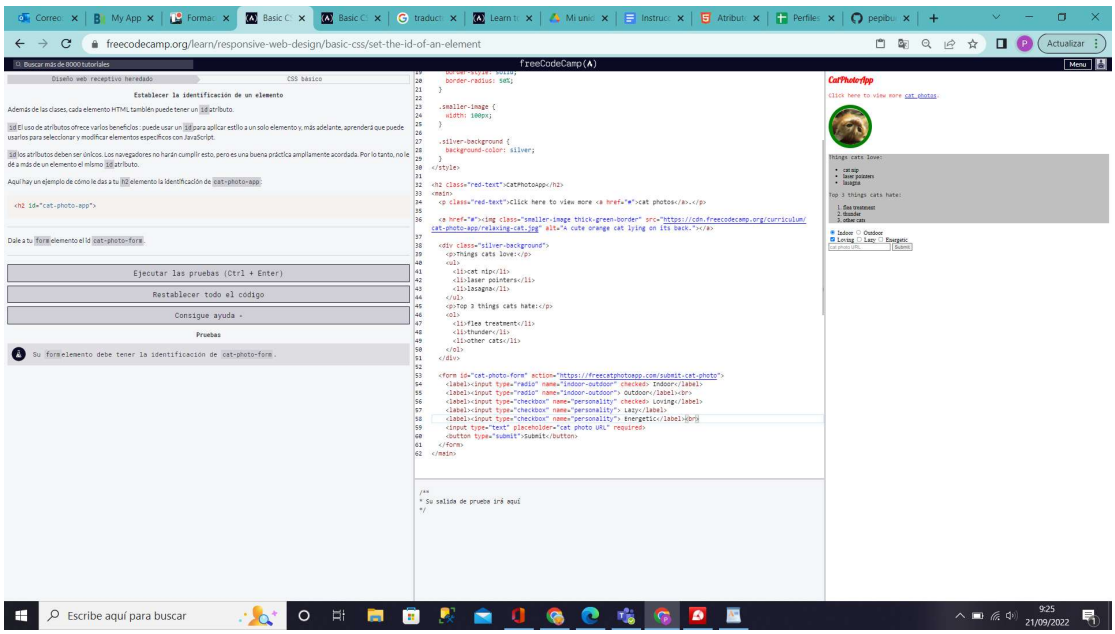
idEl uso de atributos ofrece varios beneficios : puede usar un idpara aplicar estilo a un solo elemento y, más adelante, aprenderá que puede usarlos para seleccionar y modificar elementos específicos con JavaScript.

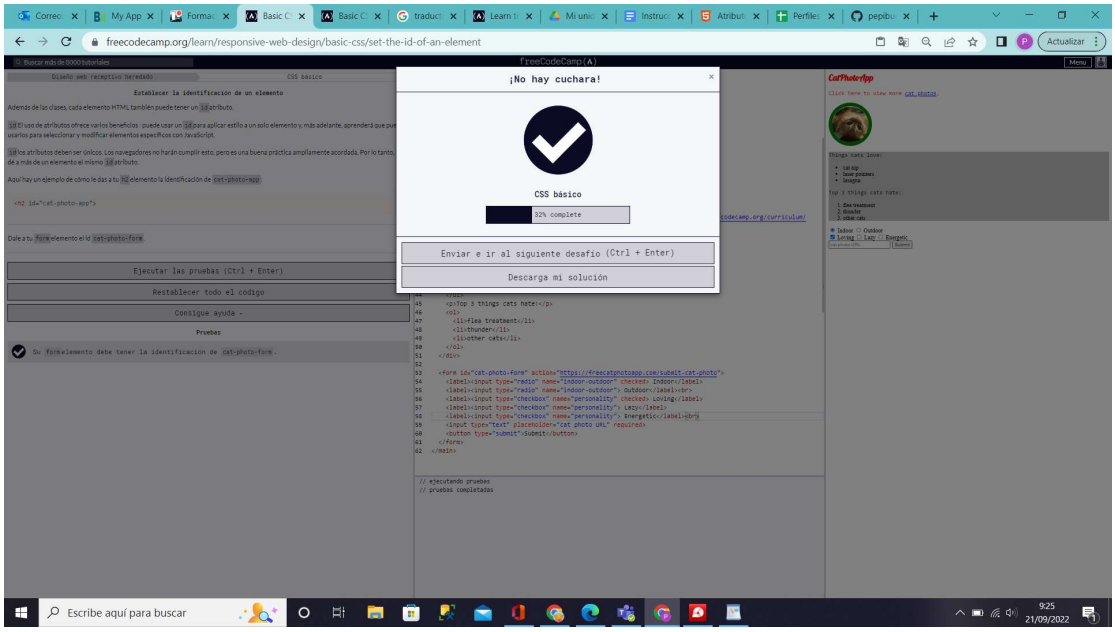
idlos atributos deben ser únicos. Los navegadores no harán cumplir esto, pero es una buena práctica ampliamente acordada. Por lo tanto, no le dé a más de un elemento el mismo idatributo.

Aquí hay un ejemplo de cómo le das a tu h2elemento la identificación de cat-photo-app:

<h2 id="cat-photo-app">

Dale a tu formelemento el id cat-photo-form.






```
<h2 class="red-text">CatPhotoApp</h2>

<main>

 <p class="red-text">Click here to view more <a href="#">cat photos</a>.</p>

 <a href="#"><img class="smaller-image thick-green-border" src="https://cdn.freecodecamp.org/curriculum/cat-photo-app/relaxing-cat.jpg" alt="A cute orange cat lying on its back."></a>

 <div class="silver-background">

  <p>Things cats love:</p>

  <ul>

   <li>cat nip</li>

   <li>laser pointers</li>

   <li>lasagna</li>

  </ul>

  <p>Top 3 things cats hate:</p>

  <ol>

   <li>flea treatment</li>

   <li>thunder</li>
```

```
    <li>other cats</li>
  </ol>
 </div>
 <form id="cat-photo-form" action="https://freecatphotoapp.com/submit-cat-photo">
  <label><input type="radio" name="indoor-outdoor" checked> Indoor</label>
  <label><input type="radio" name="indoor-outdoor"> Outdoor</label><br>
  <label><input type="checkbox" name="personality" checked> Loving</label>
  <label><input type="checkbox" name="personality"> Lazy</label>
  <label><input type="checkbox" name="personality"> Energetic</label><br>
  <input type="text" placeholder="cat photo URL" required>
  <button type="submit">Submit</button>
 </form>
</main>
```

Use un atributo id para aplicar estilo a un elemento

Una cosa interesante acerca de **id** los atributos es que, al igual que las clases, puede diseñarlos usando CSS.

Sin embargo, an id no es reutilizable y solo debe aplicarse a un elemento. An también tiene una mayor especificidad (importancia) que una clase, por lo que si ambos se aplican al mismo elemento y tienen estilos en conflicto, se aplicarán **id** los estilos de **.id**

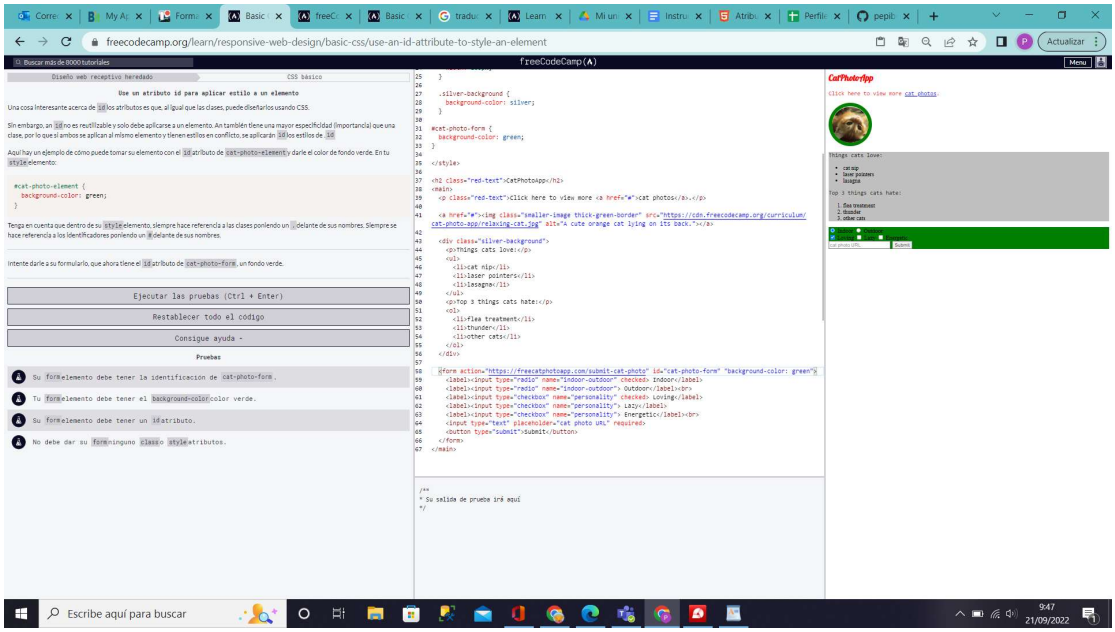
Aquí hay un ejemplo de cómo puede tomar su elemento con el id atributo de cat-photo-elementy darle el color de fondo verde. En tu style elemento:

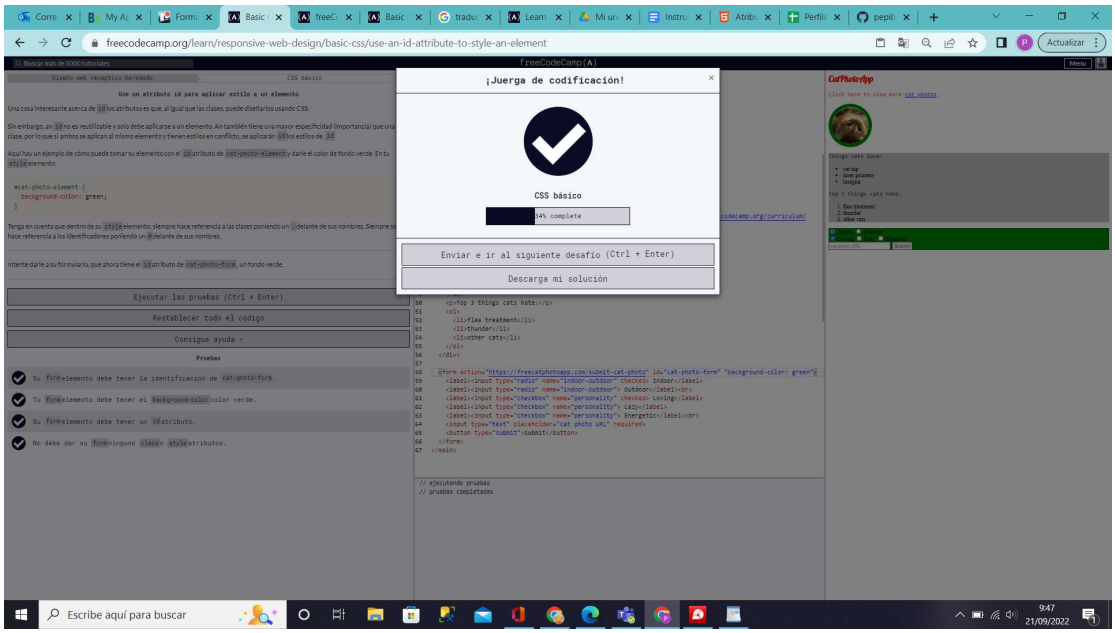
```
#cat-photo-element {
 background-color: green;
}
```

Tenga en cuenta que dentro de su **style** elemento, siempre hace referencia a las clases poniendo un **.** delante de sus nombres. Siempre se hace referencia a los identificadores

poniendo un **#** delante de sus nombres.

Intente darle a su formulario, que ahora tiene el **id** atributo de **cat-photo-form**, un fondo **verde**.





.silver-background {

```
  background-color: silver;
 }
#cat-photo-form {
 background-color: green;
}
</style>
<h2 class="red-text">CatPhotoApp</h2>
<main>
 <p class="red-text">Click here to view more <a href="#">cat photos</a>.</p>
 <a href="#"><img class="smaller-image thick-green-border" src="https://cdn.freecodecamp.org/curriculum/cat-photo-app/relaxing-cat.jpg" alt="A cute orange cat lying on its back."></a>
 <div class="silver-background">
  <p>Things cats love:</p>
  <ul>
   <li>cat nip</li>
   <li>laser pointers</li>
   <li>lasagna</li>
  </ul>
  <p>Top 3 things cats hate:</p>
  <ol>
   <li>flea treatment</li>
   <li>thunder</li>
   <li>other cats</li>
  </ol>
 </div>
 <form action="https://freecatphotoapp.com/submit-cat-photo" id="cat-photo-form" "background-color: green">
  <label><input type="radio" name="indoor-outdoor" checked> Indoor</label>
  <label><input type="radio" name="indoor-outdoor"> Outdoor</label><br>
```

```
  <label><input type="checkbox" name="personality" checked> Loving</label>

  <label><input type="checkbox" name="personality"> Lazy</label>

  <label><input type="checkbox" name="personality"> Energetic</label><br>

  <input type="text" placeholder="cat photo URL" required>

  <button type="submit">Submit</button>

 </form>

</main>
```

Ajustar el relleno de un elemento

Ahora dejemos de lado nuestra aplicación Cat Photo por un momento **y aprendamos más sobre el estilo de HTML.**

Es posible que ya haya notado esto, pero todos los elementos HTML son esencialmente pequeños rectángulos.

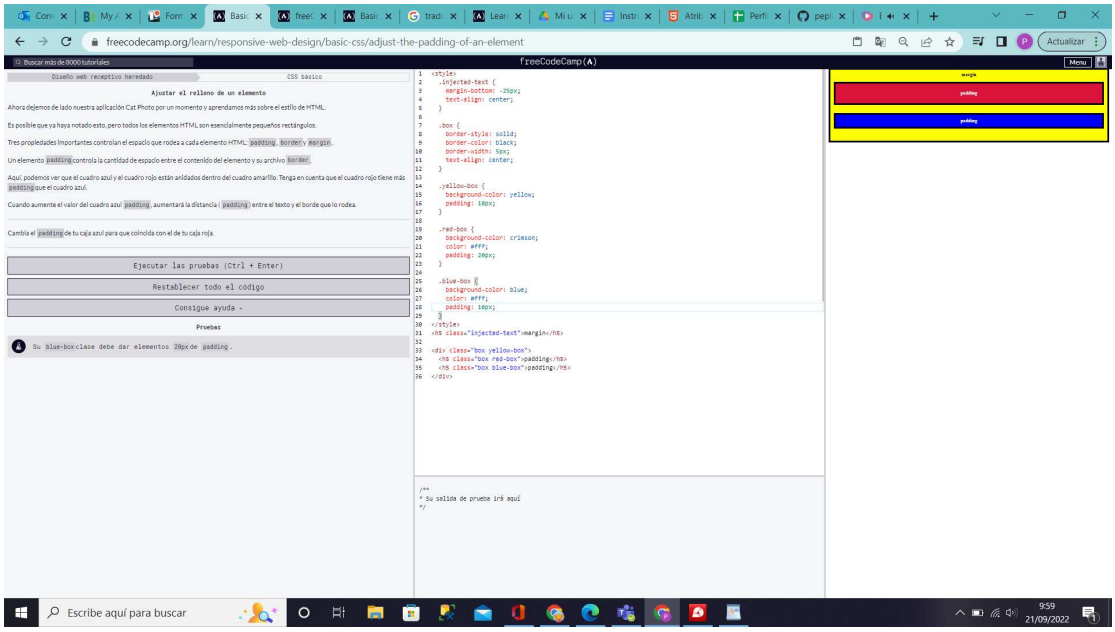
Tres propiedades importantes controlan el espacio que rodea a cada elemento HTML: **padding, border** y **margin.**

Un elemento paddingcontrola la cantidad de espacio entre el contenido del elemento y su archivo border.

Aquí, podemos ver que el cuadro azul y el cuadro rojo están anidados dentro del cuadro amarillo. Tenga en cuenta que el cuadro rojo tiene más paddingque el cuadro azul.

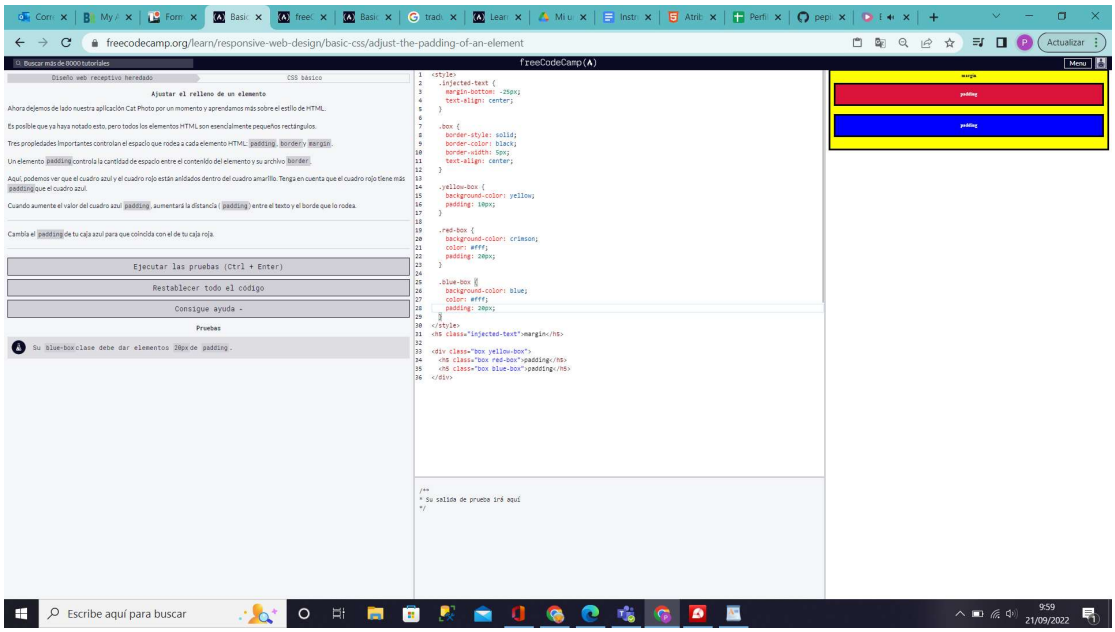
Cuando aumente el valor del cuadro azul padding, aumentará la distancia (padding) entre el texto y el borde que lo rodea.

Cambia el paddingde tu caja azul para que coincida con el de tu caja roja.


freecodecamp.org/learn/responsive-web-design/basic-css/adjust-the-padding-of-an-element
Actualizar
Buscar más de 8000 tutoriales
freeCodeCamp(A)
Menu
Diseño web receptivo heredado
CSS básico
Ajustar el relleno de un elemento
Ahora dejemos de lado nuestra aplicación Cat Photo por un momento y aprendamos más sobre el estilo de HTML.
Es posible que ya haya notado esto, pero todos los elementos HTML son esencialmente pequeños rectángulos.
Tres propiedades importantes controlan el espacio que rodea a cada elemento HTML. padding, border y margin.
Un elemento padding controla la cantidad de espacio entre el contenido del elemento y su archivo border.
Aquí, podemos ver que el cuadro azul y el cuadro rojo están anidados dentro del cuadro amarillo. Tenga en cuenta que el cuadro rojo tiene más padding que el cuadro azul.
Cuando aumente el valor del cuadro azul padding, aumentará la distancia (padding) entre el texto y el borde que lo rodea.
Cambia el padding de tu caja azul para que coincida con el de tu caja roja.
Ejecutar las pruebas (Ctrl + Enter)
Restablecer todo el código
Consigue ayuda -
Pruebas
Su blue-box clase debe dar elementos 20px de padding .
<style>
.injected-text {
margin-bottom: -25px;
text-align: center;
}
.box {
border-style: solid;
border-color: black;
border-width: 5px;
text-align: center;
}
.yellow-box {
background-color: yellow;
padding: 10px;
}
.red-box {
background-color: crimson;
color: #fff;
padding: 20px;
}
.blue-box {
background-color: blue;
color: #fff;
padding: 10px;
}
</style>
<h5 class="injected-text">margin</h5>

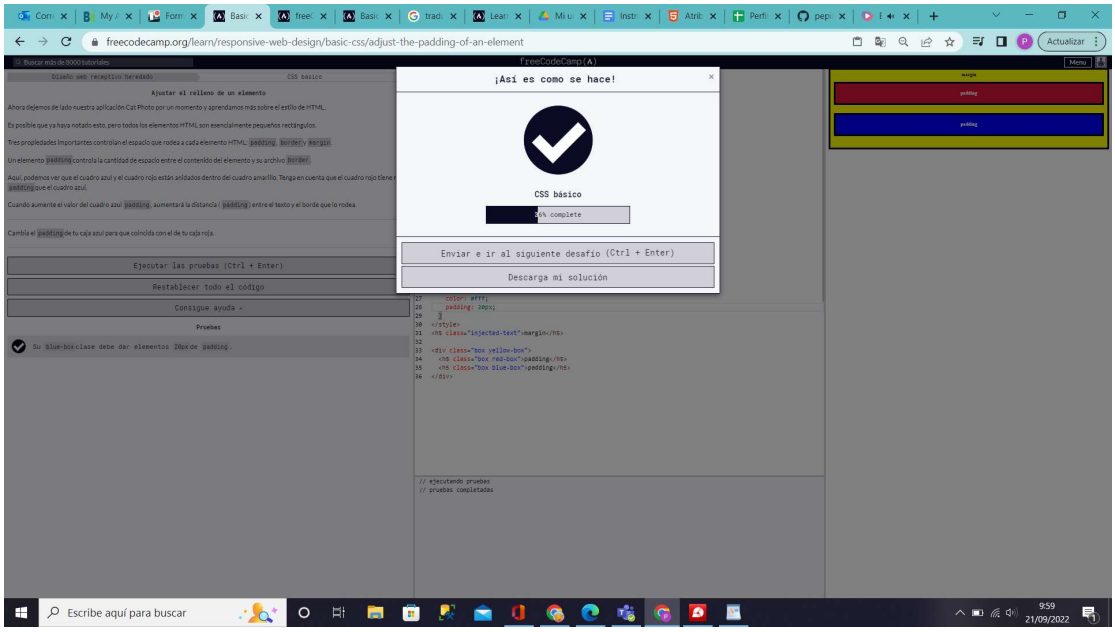
<h5 class="box red-box">padding</h5>
<h5 class="box blue-box">padding</h5>

/**
* Su salida de prueba irá aquí
*/
margin
padding
padding
Escribe aquí para buscar
9:59
21/09/2022


freecodecamp.org/learn/responsive-web-design/basic-css/adjust-the-padding-of-an-element
Actualizar
Buscar más de 8000 tutoriales
freeCodeCamp(A)
Menu
Diseño web receptivo heredado
CSS básico
Ajustar el relleno de un elemento
Ahora dejemos de lado nuestra aplicación Cat Photo por un momento y aprendamos más sobre el estilo de HTML.
Es posible que ya haya notado esto, pero todos los elementos HTML son esencialmente pequeños rectángulos.
Tres propiedades importantes controlan el espacio que rodea a cada elemento HTML. padding, border y margin.
Un elemento padding controla la cantidad de espacio entre el contenido del elemento y su archivo border.
Aquí, podemos ver que el cuadro azul y el cuadro rojo están anidados dentro del cuadro amarillo. Tenga en cuenta que el cuadro rojo tiene más padding que el cuadro azul.
Cuando aumente el valor del cuadro azul padding, aumentará la distancia (padding) entre el texto y el borde que lo rodea.
Cambia el padding de tu caja azul para que coincida con el de tu caja roja.
Ejecutar las pruebas (Ctrl + Enter)
Restablecer todo el código
Consigue ayuda -
Pruebas
Su blue-box clase debe dar elementos 20px de padding .
<style>
.injected-text {
margin-bottom: -25px;
text-align: center;
}
.box {
border-style: solid;
border-color: black;
border-width: 5px;
text-align: center;
}
.yellow-box {
background-color: yellow;
padding: 10px;
}
.red-box {
background-color: crimson;
color: #fff;
padding: 20px;
}
.blue-box {
background-color: blue;
color: #fff;
padding: 20px;
}
</style>
<h5 class="injected-text">margin</h5>

<h5 class="box red-box">padding</h5>
<h5 class="box blue-box">padding</h5>

/**
* Su salida de prueba irá aquí
*/
margin
padding
padding
Escribe aquí para buscar
9:59
21/09/2022




```
<style>
 .injected-text {
   margin-bottom: -25px;
   text-align: center;
 }
 .box {
   border-style: solid;
   border-color: black;
   border-width: 5px;
   text-align: center;
 }
 .yellow-box {
   background-color: yellow;
   padding: 10px;
 }
 .red-box {
   background-color: crimson;
```

```
    color: #fff;
    padding: 20px;
  }
  .blue-box {
    background-color: blue;
    color: #fff;
    padding: 20px;
  }
</style>
<h5 class="injected-text">margin</h5>
<div class="box yellow-box">
  <h5 class="box red-box">padding</h5>
  <h5 class="box blue-box">padding</h5>
</div>
```

Ajustar el margen de un elemento

Un elemento **margin** controla la cantidad de espacio entre un elemento **border** y los elementos que lo rodean.

Aquí, podemos ver que el cuadro azul y el cuadro rojo están anidados dentro del cuadro amarillo. Tenga en cuenta que el cuadro rojo tiene un tamaño más grande marginque el cuadro azul, lo que hace que parezca más pequeño.

Cuando aumenta el cuadro azul margin, aumentará la distancia entre su borde y los elementos circundantes.

Cambie el margindel cuadro azul para que coincida con el del cuadro rojo.

freecodecamp.org/learn/responsive-web-design/basic-css/adjust-the-margin-of-an-element

Diseño web receptivo heredado — CSS básico

Ajustar el margen de un elemento

Un elemento `margin` controla la cantidad de espacio entre un elemento `border` y los elementos que lo rodean.

Aquí, podemos ver que el cuadro azul y el cuadro rojo están anidados dentro del cuadro amarillo. Tenga en cuenta que el cuadro rojo tiene un tamaño más grande `margin` que el cuadro azul, lo que hace que parezca más pequeño.

Cuando aumenta el cuadro azul `margin`, aumentará la distancia entre su borde y los elementos circundantes.

Cambie el `margin` del cuadro azul para que coincida con el del cuadro rojo.

Ejecutar las pruebas (Ctrl + Enter)

Restablecer todo el código

Consigue ayuda -

Pruebas

Su `blue-box` clase debe dar elementos `20px` de `margin`.

```
<style>
  .injected-text {
    margin-bottom: -25px;
    text-align: center;
  }

  .box {
    border-style: solid;
    border-color: black;
    border-width: 5px;
    text-align: center;
  }

  .yellow-box {
    background-color: yellow;
    padding: 10px;
  }

  .red-box {
    background-color: crimson;
    color: #fff;
    padding: 20px;
    margin: 20px;
  }

  .blue-box {
    background-color: blue;
    color: #fff;
    padding: 20px;
    margin: 20px;
  }
</style>
<h5 class="injected-text">margin</h5>

<div class="box yellow-box">
  <h5 class="box red-box">padding</h5>
  <h5 class="box blue-box">padding</h5>
</div>
```

```
/**
* Su salida de prueba irá aquí
*/
```

freecodecamp.org/learn/responsive-web-design/basic-css/adjust-the-margin-of-an-element

Diseño web receptivo heredado — CSS básico

Ajustar el margen de un elemento

Un elemento `margin` controla la cantidad de espacio entre un elemento `border` y los elementos que lo rodean.

Aquí, podemos ver que el cuadro azul y el cuadro rojo están anidados dentro del cuadro amarillo. Tenga en cuenta que el cuadro rojo tiene un tamaño más grande `margin` que el cuadro azul, lo que hace que parezca más pequeño.

Cuando aumenta el cuadro azul `margin`, aumentará la distancia entre su borde y los elementos circundantes.

Cambie el `margin` del cuadro azul para que coincida con el del cuadro rojo.

Ejecutar las pruebas (Ctrl + Enter)

Restablecer todo el código

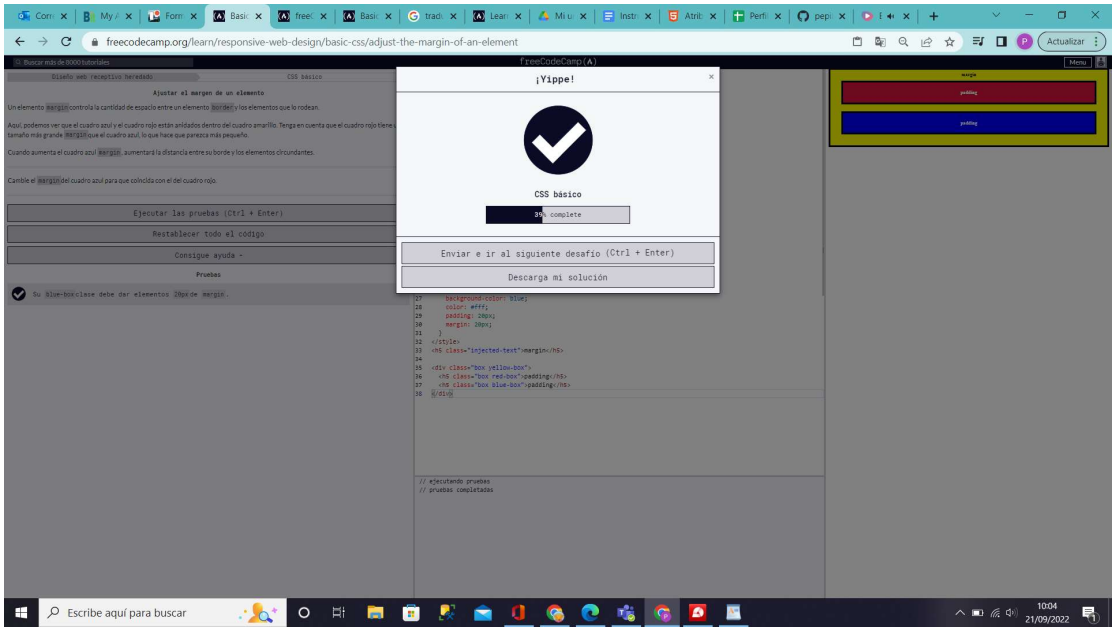
Consigue ayuda -

Pruebas

Su `blue-box` clase debe dar elementos `20px` de `margin`.

```
<style>
  .injected-text {
    margin-bottom: -25px;
    text-align: center;
  }

  .box {
    border-style: solid;
    border-color: black;
    border-width: 5px;
    text-align: center;
  }

  .yellow-box {
    background-color: yellow;
    padding: 10px;
  }

  .red-box {
    background-color: crimson;
    color: #fff;
    padding: 20px;
    margin: 20px;
  }

  .blue-box {
    background-color: blue;
    color: #fff;
    padding: 20px;
    margin: 20px;
  }
</style>
<h5 class="injected-text">margin</h5>

<div class="box yellow-box">
  <h5 class="box red-box">padding</h5>
  <h5 class="box blue-box">padding</h5>
</div>
```

```
/**
* Su salida de prueba irá aquí
*/
```




```
<style>
 .injected-text {
   margin-bottom: -25px;
   text-align: center;
 }
 .box {
   border-style: solid;
   border-color: black;
   border-width: 5px;
   text-align: center;
 }
 .yellow-box {
   background-color: yellow;
   padding: 10px;
 }
 .red-box {
   background-color: crimson;
```

```
    color: #fff;
    padding: 20px;
    margin: 20px;
  }
  .blue-box {
    background-color: blue;
    color: #fff;
    padding: 20px;
    margin: 20px;
  }
</style>
<h5 class="injected-text">margin</h5>
<div class="box yellow-box">
  <h5 class="box red-box">padding</h5>
  <h5 class="box blue-box">padding</h5>
</div>
```

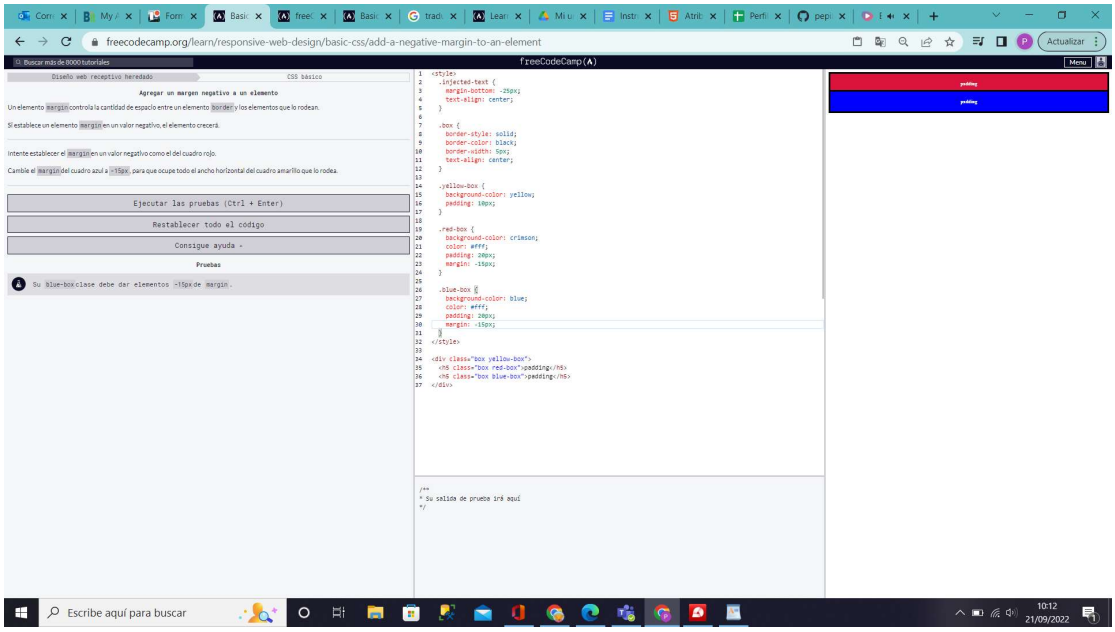
Agregar un margen negativo a un elemento

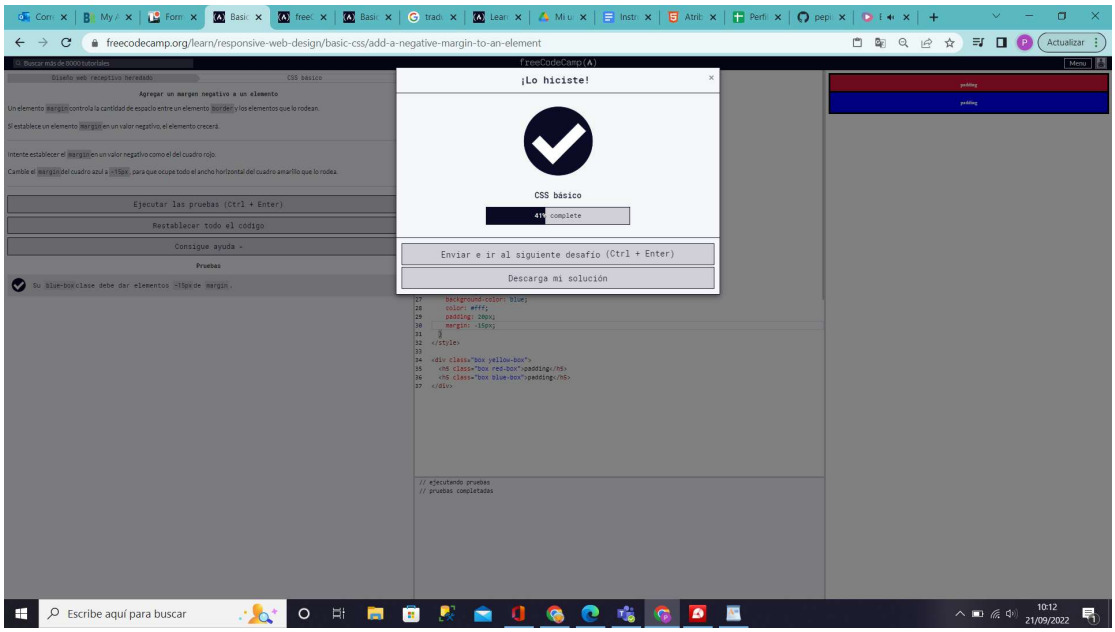
Un elemento **margin** controla la cantidad de espacio entre un elemento **border** y los elementos que lo rodean.

Si establece un elemento margin en un valor negativo, el elemento crecerá.

Intente establecer el marginen un valor negativo como el del cuadro rojo.

Cambie el margin del cuadro azul a **-15px**, para que ocupe todo el ancho horizontal del cuadro amarillo que lo rodea.






```
<style>
 .injected-text {
   margin-bottom: -25px;
   text-align: center;
 }
```

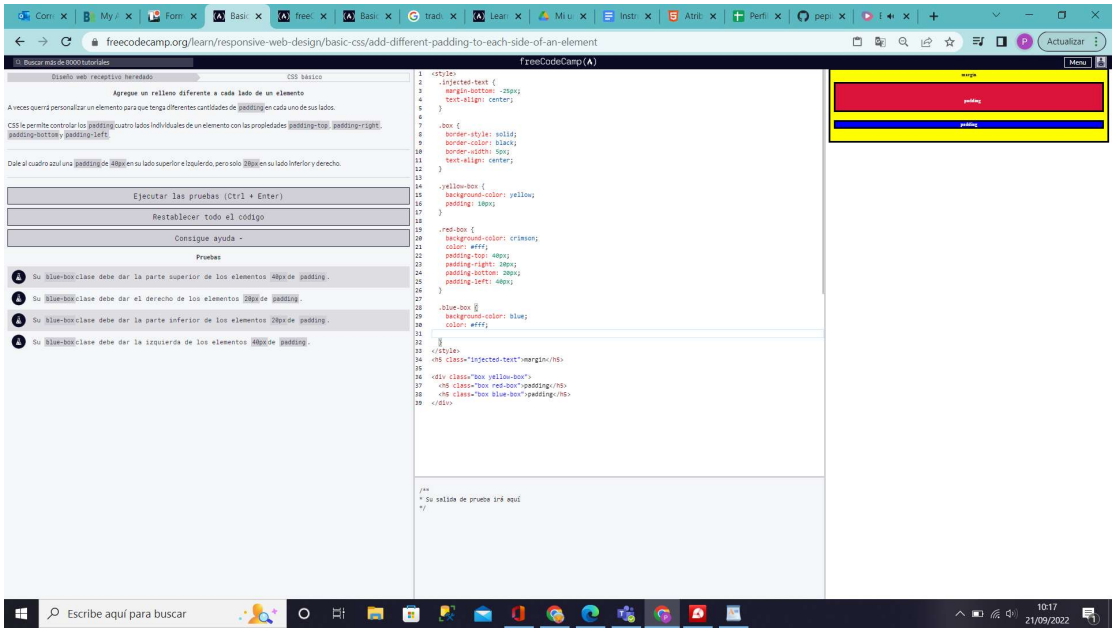
```
  .box {
    border-style: solid;
    border-color: black;
    border-width: 5px;
    text-align: center;
  }
  .yellow-box {
    background-color: yellow;
    padding: 10px;
  }
  .red-box {
    background-color: crimson;
    color: #fff;
    padding: 20px;
    margin: -15px;
  }
  .blue-box {
    background-color: blue;
    color: #fff;
    padding: 20px;
    margin: -15px;
  }
</style>
<div class="box yellow-box">
  <h5 class="box red-box">padding</h5>
  <h5 class="box blue-box">padding</h5>
</div>
```

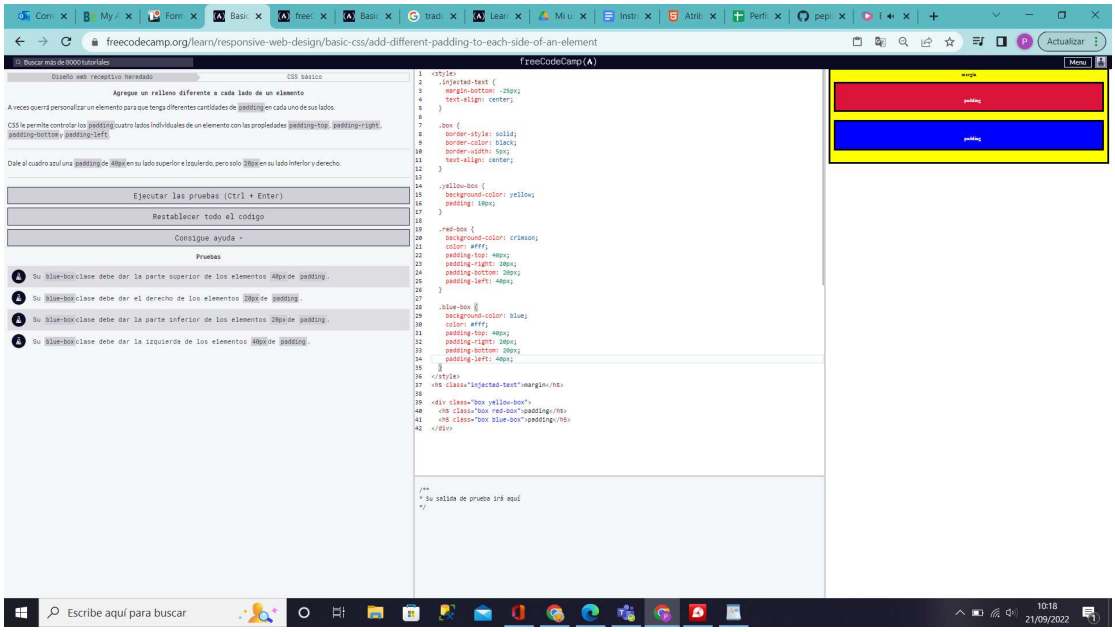
Agregue un relleno diferente a cada lado de un elemento

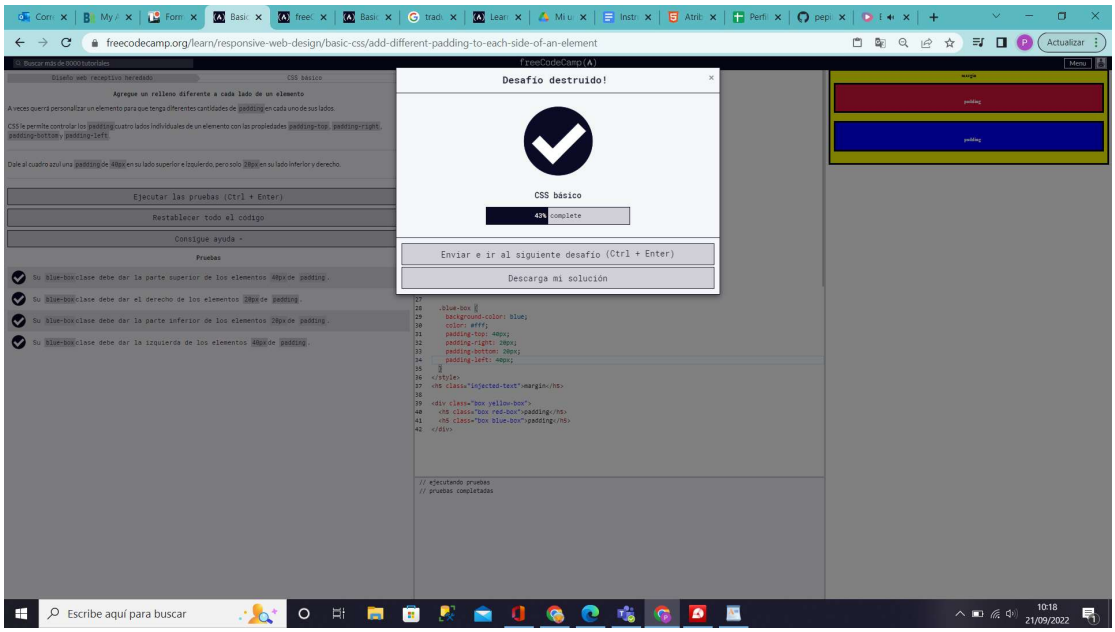
A veces querrá personalizar un elemento para que tenga diferentes cantidades de **padding** en cada uno de sus lados.

CSS le permite controlar los **padding cuatro lados** individuales de un elemento con las propiedades **padding-top, padding-right, padding-bottom** y **padding-left**.

Dale al cuadro azul una padding de 40px en su lado superior e izquierdo, pero solo 20px en su lado inferior y derecho.








```
<style>
 .injected-text {
   margin-bottom: -25px;
   text-align: center;
 }
```

```
 .box {
   border-style: solid;
   border-color: black;
   border-width: 5px;
   text-align: center;
 }
 .yellow-box {
   background-color: yellow;
   padding: 10px;
 }
 .red-box {
   background-color: crimson;
   color: #fff;
   padding-top: 40px;
   padding-right: 20px;
   padding-bottom: 20px;
   padding-left: 40px;
 }
 .blue-box {
   background-color: blue;
   color: #fff;
   padding-top: 40px;
   padding-right: 20px;
   padding-bottom: 20px;
   padding-left: 40px;
 }
</style>
<h5 class="injected-text">margin</h5>
```

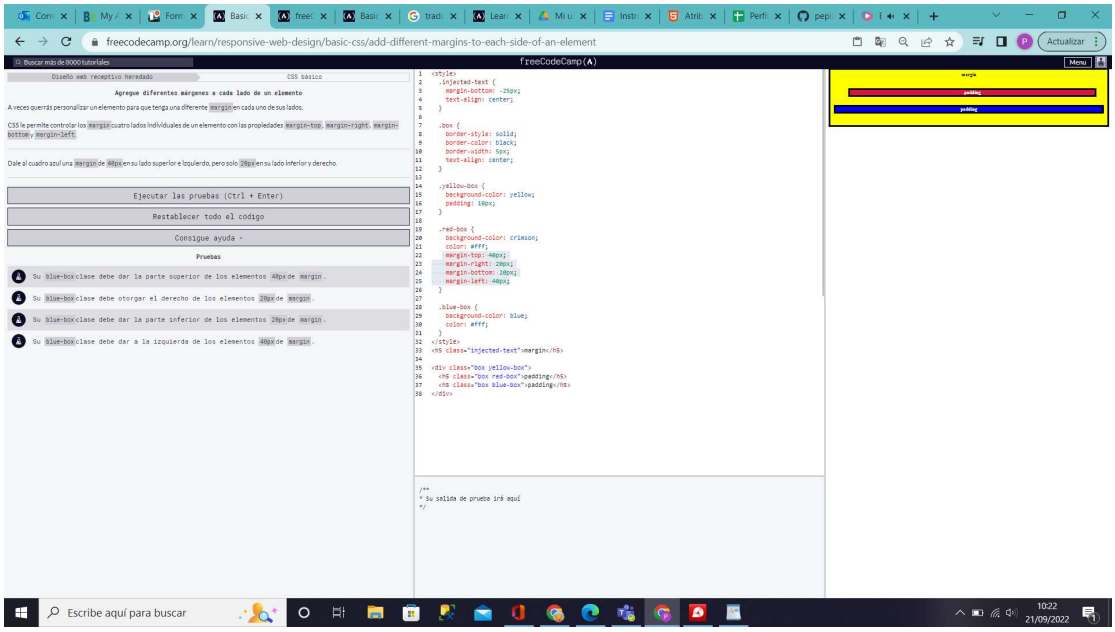
```
<div class="box yellow-box">
  <h5 class="box red-box">padding</h5>
  <h5 class="box blue-box">padding</h5>
</div>
```

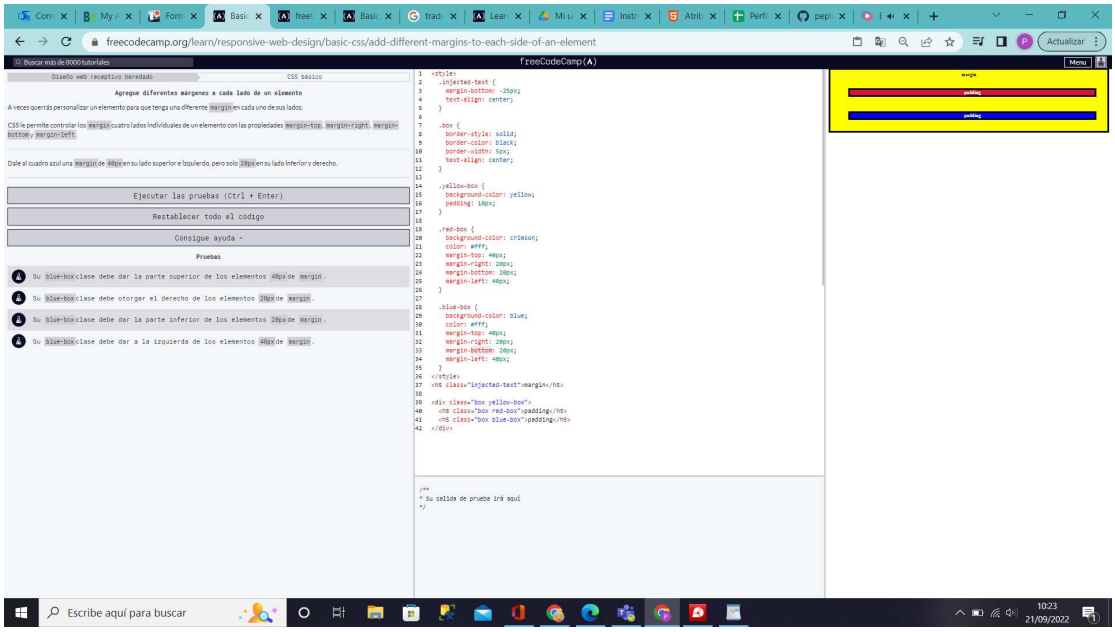
Agregue diferentes márgenes a cada lado de un elemento

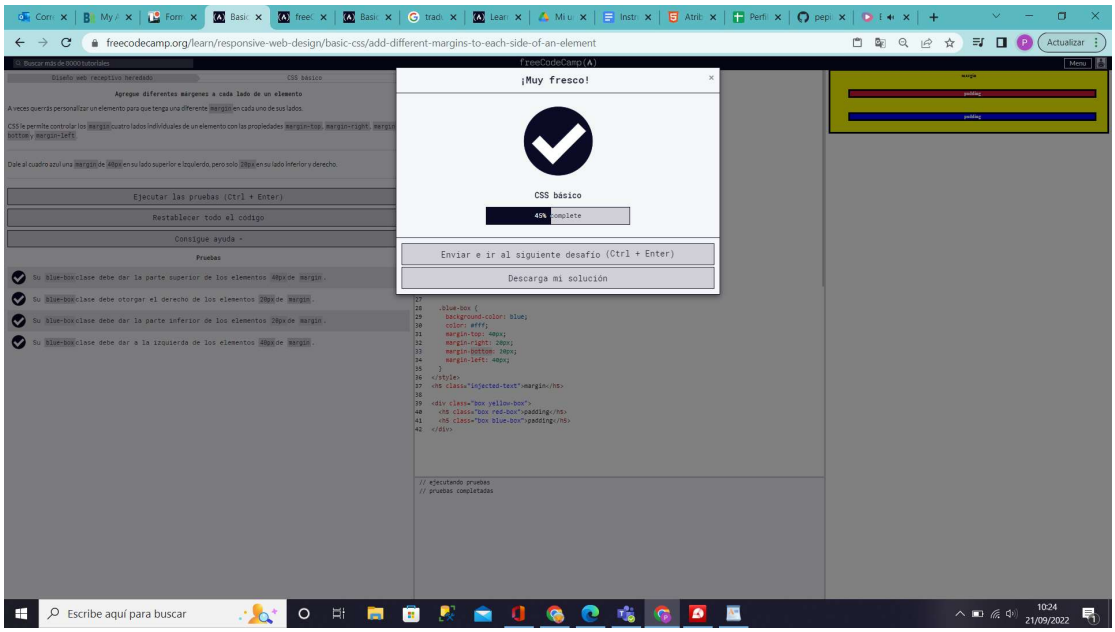
A veces querrás personalizar un elemento para que tenga una diferente **margin** en cada uno de sus lados.

CSS le permite controlar los margin cuatro lados individuales de un elemento con las propiedades margin-top, margin-right, margin-bottom y margin-left.

Dale al cuadro azul una marginde 40px en su lado superior e izquierdo, pero solo 20px en su lado inferior y derecho.








```
<style>
 .injected-text {
   margin-bottom: -25px;
   text-align: center;
 }
```

```
 .box {
   border-style: solid;
   border-color: black;
   border-width: 5px;
   text-align: center;
 }
 .yellow-box {
   background-color: yellow;
   padding: 10px;
 }
 .red-box {
   background-color: crimson;
   color: #fff;
   margin-top: 40px;
   margin-right: 20px;
   margin-bottom: 20px;
   margin-left: 40px;
 }
 .blue-box {
   background-color: blue;
   color: #fff;
   margin-top: 40px;
   margin-right: 20px;
   margin-bottom: 20px;
   margin-left: 40px;
 }
</style>
<h5 class="injected-text">margin</h5>
```

```
<div class="box yellow-box">
  <h5 class="box red-box">padding</h5>
  <h5 class="box blue-box">padding</h5>
</div>
```

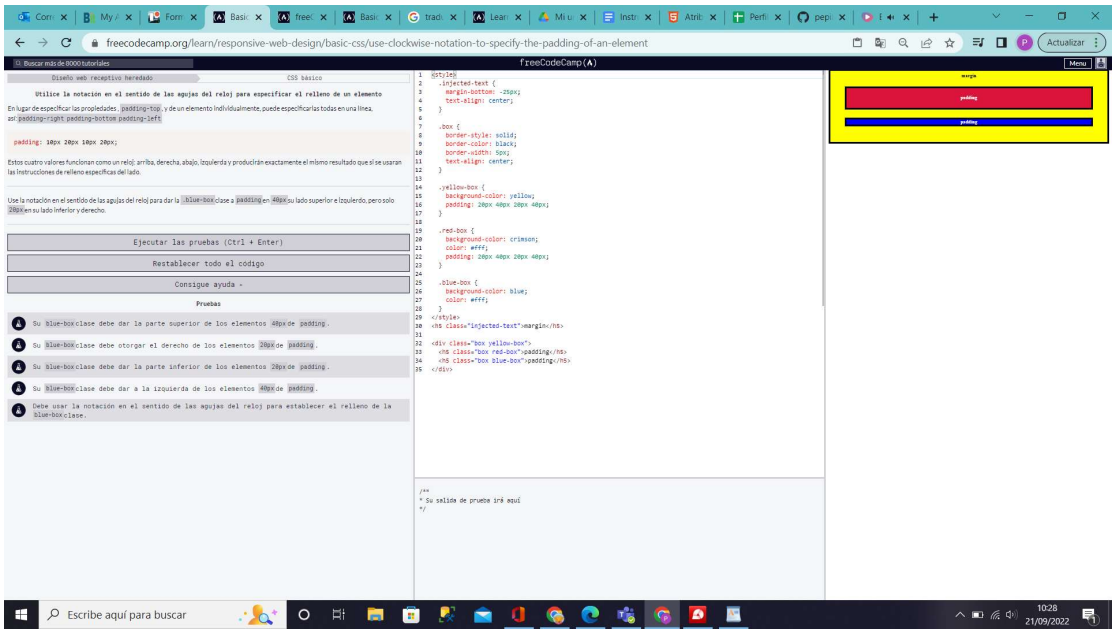
Utilice la notación en el sentido de las agujas del reloj para especificar el relleno de un elemento

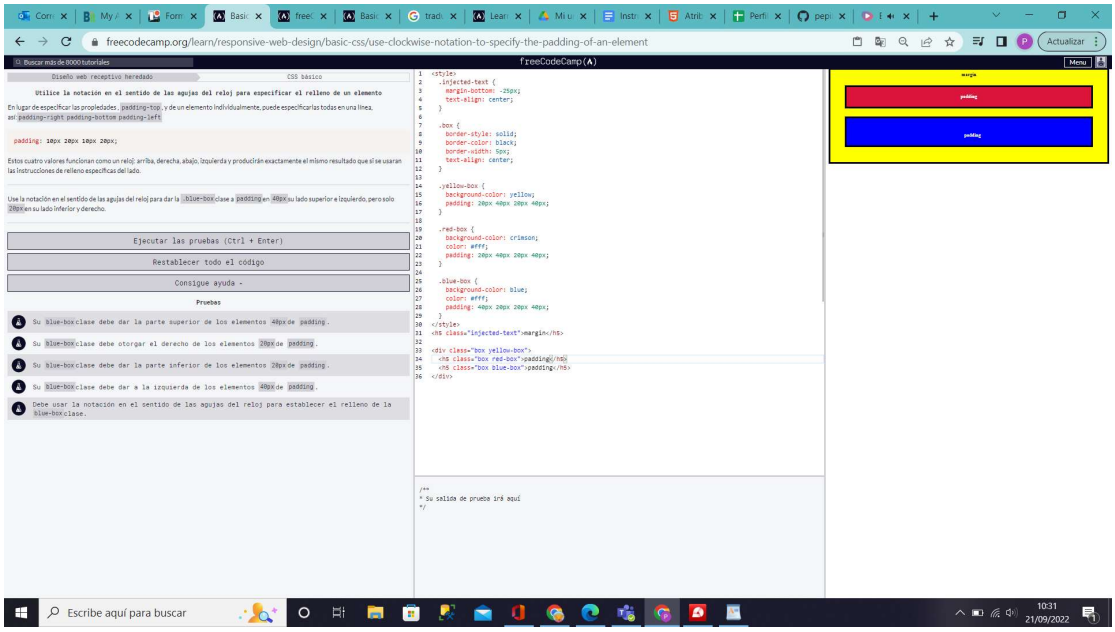
En lugar de especificar las propiedades , **padding-top**, y de un elemento individualmente, puede especificarlas todas en una línea, así: **padding-right padding-bottom padding-left**

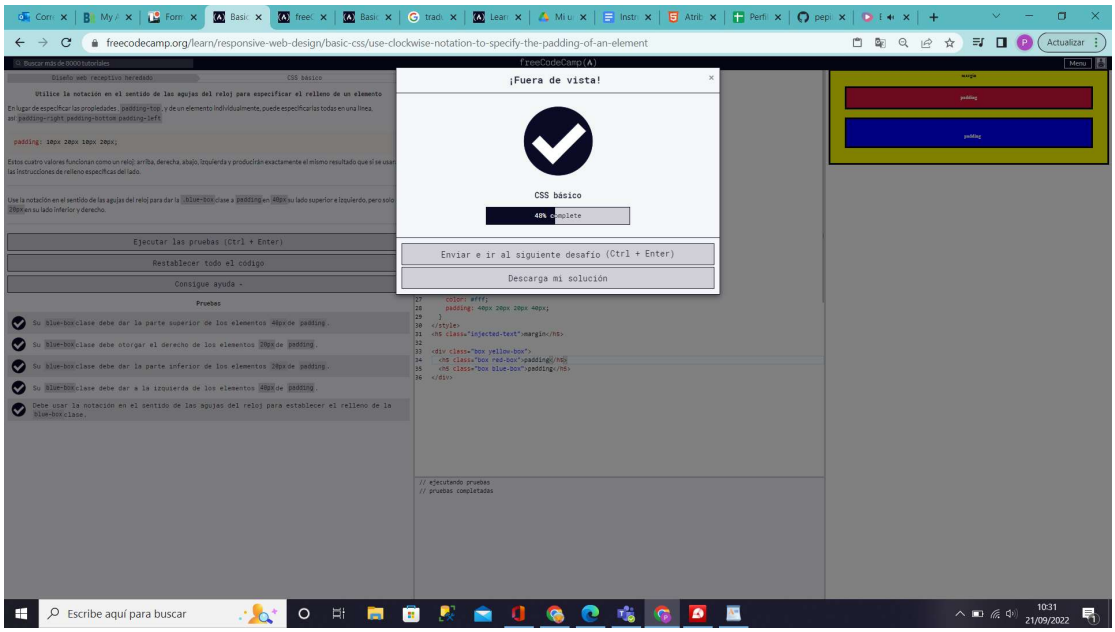
padding: 10px 20px 10px 20px;

Estos cuatro valores funcionan como un reloj: **arriba, derecha, abajo, izquierda** y producirán exactamente el mismo resultado que si se usaran las instrucciones de relleno específicas del lado.

Use la notación en el **sentido de las agujas del reloj** para dar la .blue-boxclase a paddingen 40pxsu lado superior e izquierdo, pero solo 20pxen su lado inferior y derecho.








```
<style>
 .injected-text {
   margin-bottom: -25px;
   text-align: center;
 }
```

```
  .box {
    border-style: solid;
    border-color: black;
    border-width: 5px;
    text-align: center;
  }
  .yellow-box {
    background-color: yellow;
    padding: 20px 40px 20px 40px;
  }
  .red-box {
    background-color: crimson;
    color: #fff;
    padding: 20px 40px 20px 40px;
  }
  .blue-box {
    background-color: blue;
    color: #fff;
    padding: 40px 20px 20px 40px;
  }
</style>
<h5 class="injected-text">margin</h5>
<div class="box yellow-box">
  <h5 class="box red-box">padding</h5>
  <h5 class="box blue-box">padding</h5>
</div>
```

Utilice la notación en el sentido de las agujas del reloj para

especificar el margen de un elemento

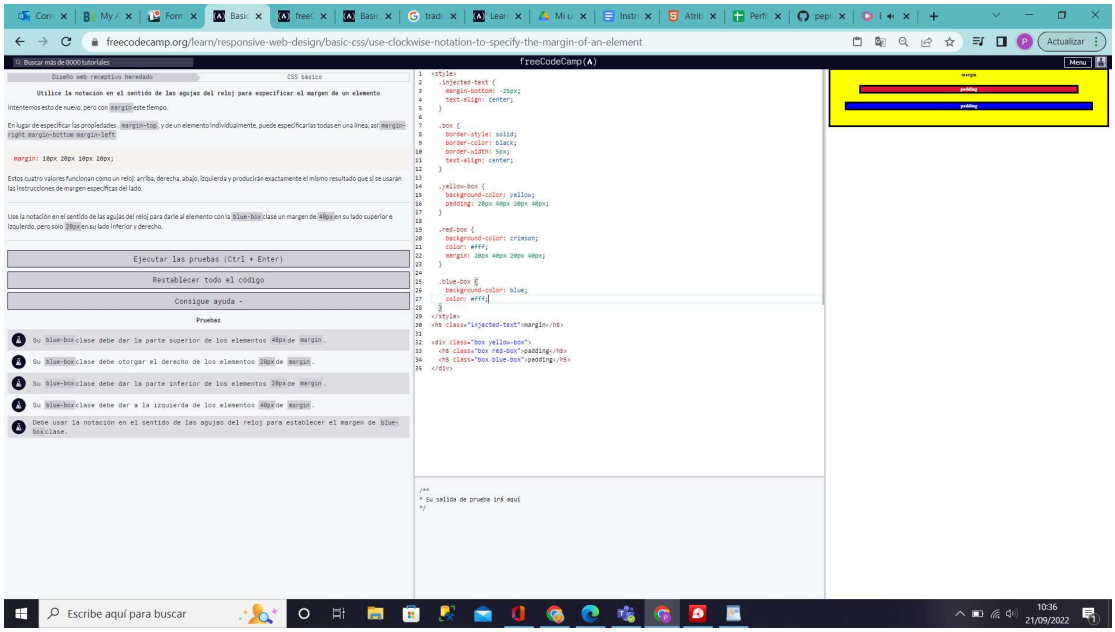
Intentemos esto de nuevo, pero con **margin** este tiempo.

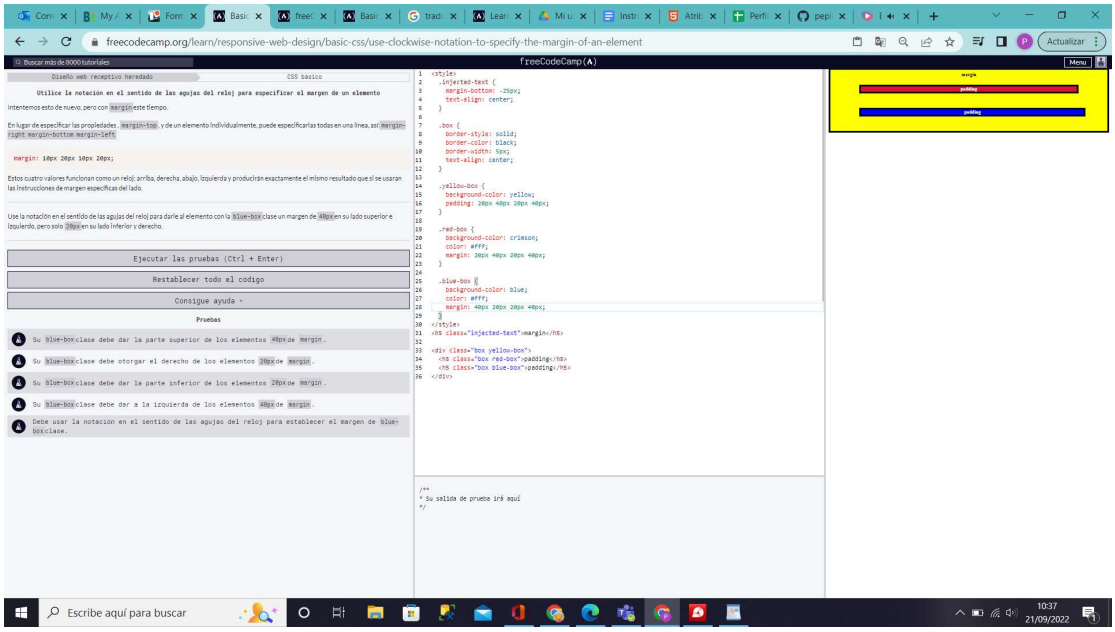
En lugar de especificar las propiedades , **margin-top**, y de un elemento individualmente, **puede** especificarlas **todas en una línea**, así:

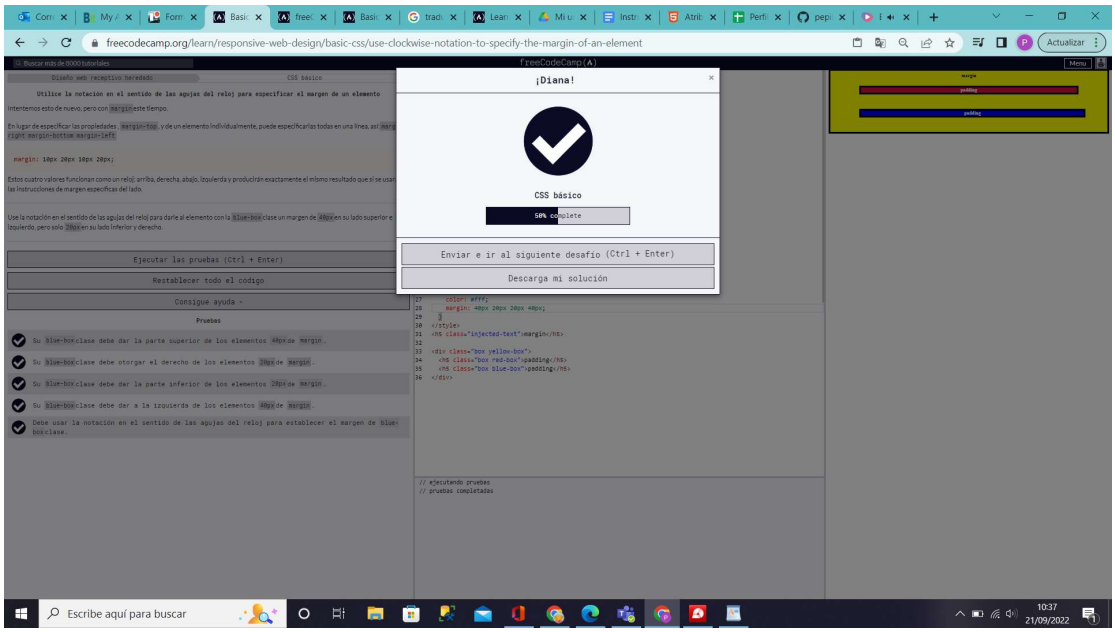
margin-right margin-bottom margin-left

margin: 10px 20px 10px 20px;

Estos cuatro valores funcionan **como un reloj**: arriba, derecha, abajo, izquierda y producirán exactamente el mismo resultado que si se usaran las instrucciones de margen específicas del lado.








```
<style>
 .injected-text {
   margin-bottom: -25px;
   text-align: center;
 }
```

```
  .box {
    border-style: solid;
    border-color: black;
    border-width: 5px;
    text-align: center;
  }
  .yellow-box {
    background-color: yellow;
    padding: 20px 40px 20px 40px;
  }
  .red-box {
    background-color: crimson;
    color: #fff;
    margin: 20px 40px 20px 40px;
  }
  .blue-box {
    background-color: blue;
    color: #fff;
    margin: 40px 20px 20px 40px;
  }
</style>
<h5 class="injected-text">margin</h5>
<div class="box yellow-box">
  <h5 class="box red-box">padding</h5>
  <h5 class="box blue-box">padding</h5>
</div>
```

Usar selectores de atributos para diseñar elementos

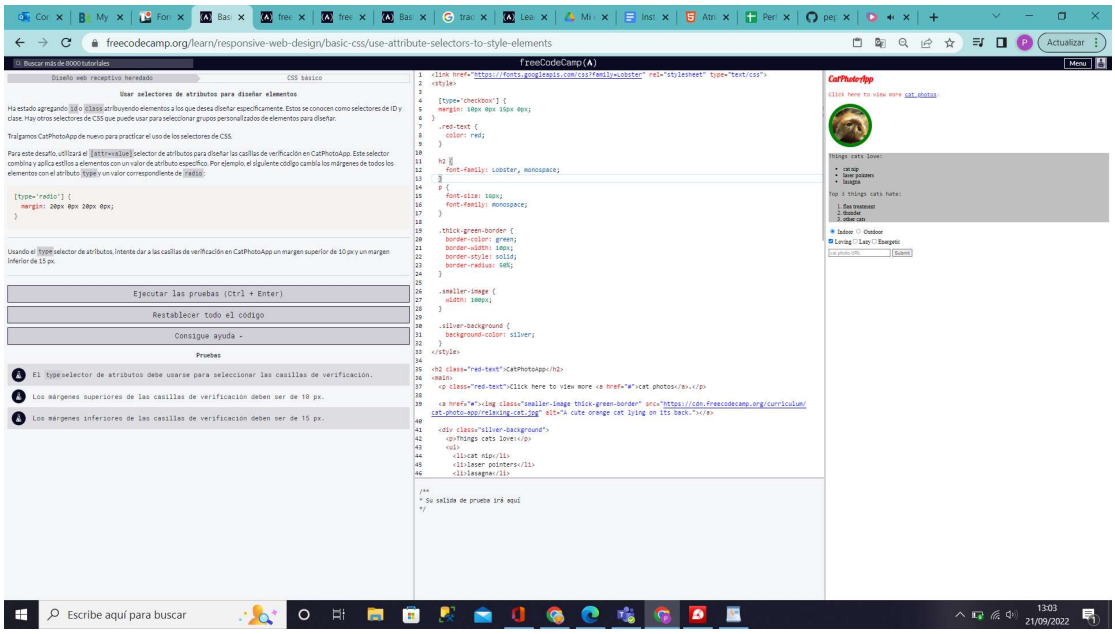
Ha estado agregando **id** o **class** atribuyendo elementos a los que desea diseñar específicamente. Estos se conocen como **selectores de ID y clase**. **Hay otros selectores** de CSS que puede usar para **seleccionar grupos personalizados** de elementos para diseñar.

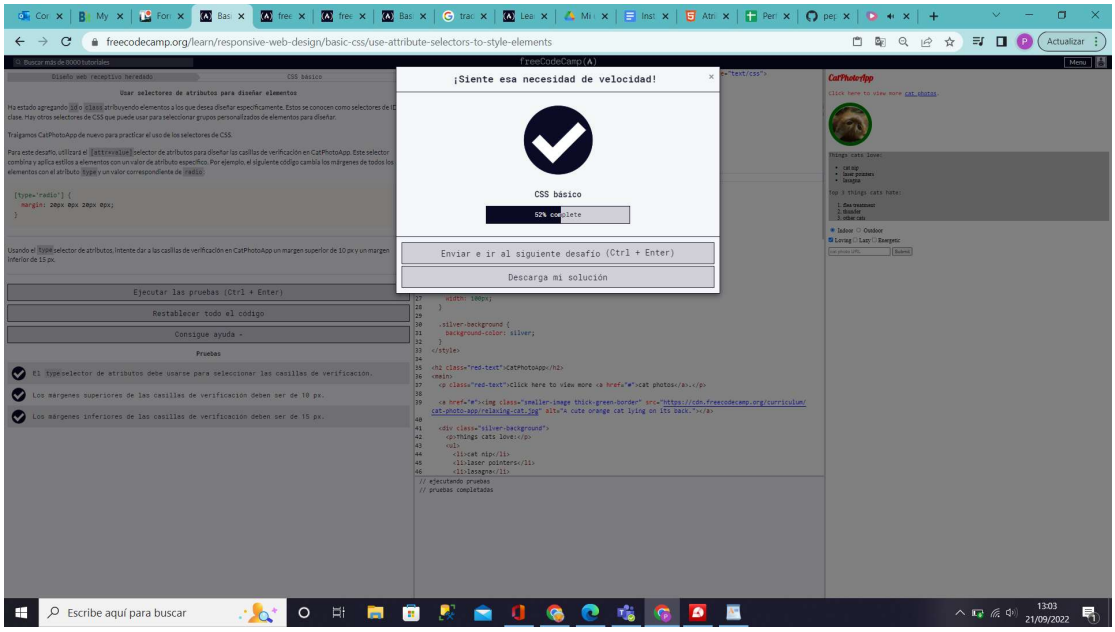
Traigamos CatPhotoApp de nuevo para practicar el uso de los selectores de CSS.

Para este desafío, utilizará el **[attr=value]** selector de atributos para diseñar las casillas de verificación en CatPhotoApp. Este selector combina y aplica estilos a elementos con un valor de atributo específico. Por ejemplo, el siguiente código cambia los márgenes de todos los elementos con el atributo **type** y un valor correspondiente de **radio**:

```
[type='radio'] {

  margin: 20px 0px 20px 0px;

}
```

Usando el typeselector de atributos, intente dar a las casillas de verificación en CatPhotoApp un margen superior de 10 px y un margen inferior de 15 px.






```
<link href="https://fonts.googleapis.com/css?family=Lobster" rel="stylesheet" type="text/css">

<style>

 [type='checkbox'] {

 margin: 10px 0px 15px 0px;

}

 .red-text {

  color: red;

 }

 h2 {

  font-family: Lobster, monospace;

 }

 p {

  font-size: 16px;

  font-family: monospace;

 }

 .thick-green-border {
```

```
    border-color: green;
    border-width: 10px;
    border-style: solid;
    border-radius: 50%;
  }
  .smaller-image {
    width: 100px;
  }
  .silver-background {
    background-color: silver;
  }
</style>
<h2 class="red-text">CatPhotoApp</h2>
<main>
  <p class="red-text">Click here to view more <a href="#">cat photos</a>.</p>
  <a href="#"><img class="smaller-image thick-green-border" src="https://cdn.freecodecamp.org/curriculum/cat-photo-app/relaxing-cat.jpg" alt="A cute orange cat lying on its back."></a>
  <div class="silver-background">
    <p>Things cats love:</p>
    <ul>
      <li>cat nip</li>
      <li>laser pointers</li>
      <li>lasagna</li>
    </ul>
    <p>Top 3 things cats hate:</p>
    <ol>
      <li>flea treatment</li>
      <li>thunder</li>
```

```
    <li>other cats</li>
  </ol>
 </div>
 <form action="https://freecatphotoapp.com/submit-cat-photo" id="cat-photo-form">
  <label><input type="radio" name="indoor-outdoor" checked> Indoor</label>
  <label><input type="radio" name="indoor-outdoor"> Outdoor</label><br>
  <label><input type="checkbox" name="personality" checked> Loving</label>
  <label><input type="checkbox" name="personality"> Lazy</label>
  <label><input type="checkbox" name="personality"> Energetic</label><br>
  <input type="text" placeholder="cat photo URL" required>
  <button type="submit">Submit</button>
 </form>
</main>
```

Comprender las unidades absolutas frente a las relativas

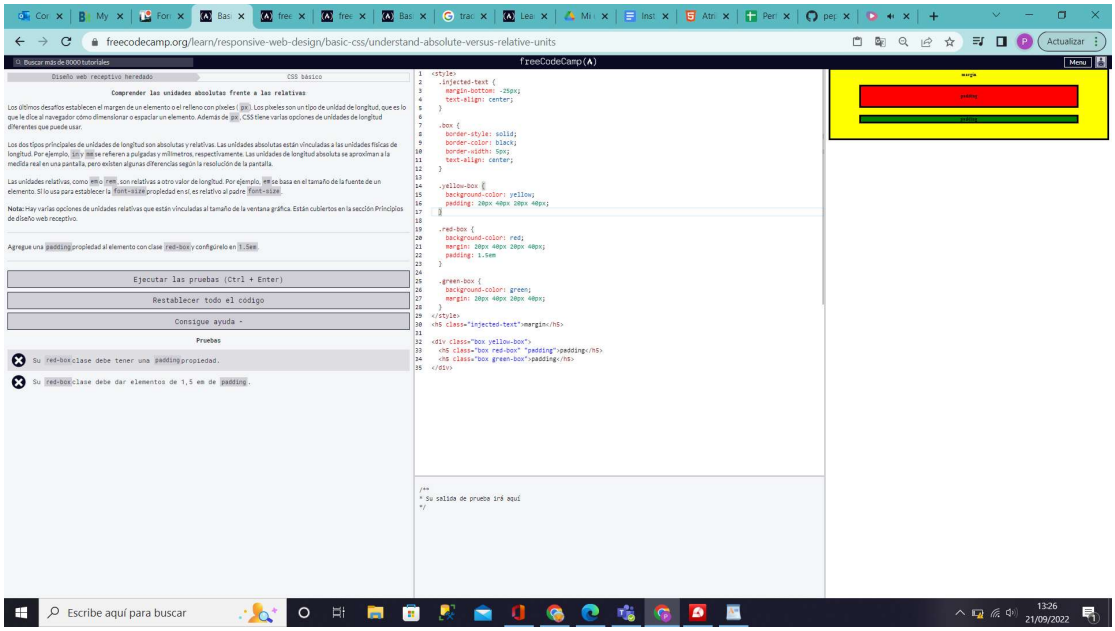
Los últimos desafíos establecen el margen de un elemento o el relleno con píxeles (**px**). Los píxeles **son un tipo de unidad de longitud**, que es lo que le dice al navegador cómo dimensionar o espaciar un elemento. Además de px, CSS tiene varias opciones de unidades de longitud diferentes que puede usar.

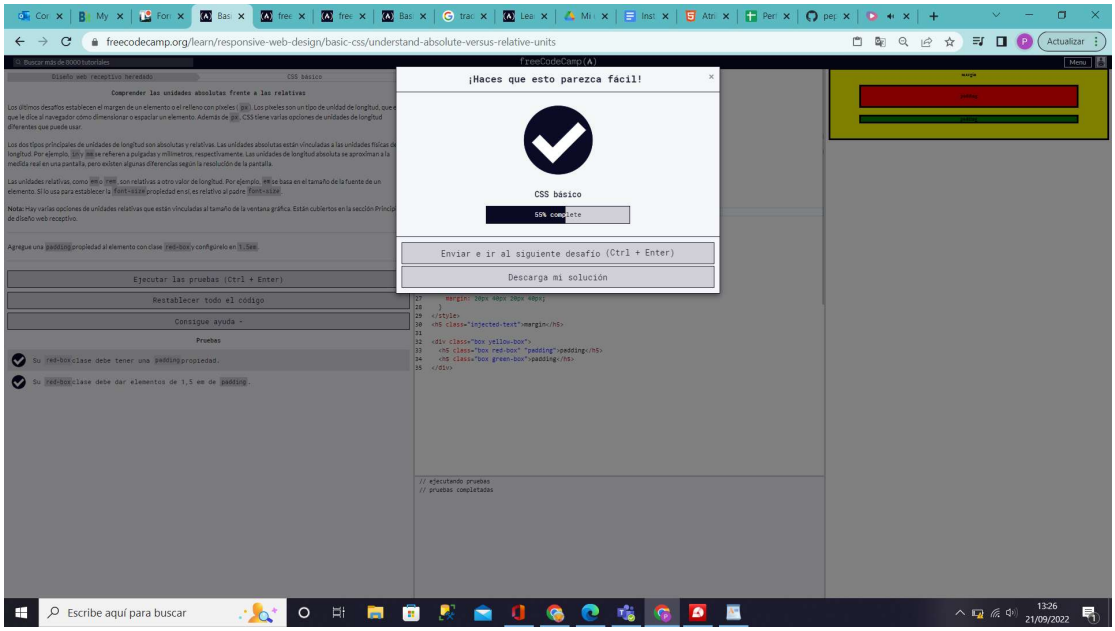
Los dos tipos principales de unidades de longitud son **absolutas y relativas**. Las unidades absolutas están vinculadas a las unidades físicas de longitud. Por ejemplo, **in** y **mm** se refieren a **pulgadas y milímetros**, respectivamente. Las unidades de **longitud absoluta** se aproximan a la medida real en una pantalla, pero existen algunas diferencias según la resolución de la pantalla.

Las **unidades relativas**, como **em** o **rem**, son relativas a otro valor de longitud. Por ejemplo, em se basa en el tamaño de la fuente de un elemento. Si lo usa para establecer la font-size propiedad en sí, es relativo al padre **font-size**.

Nota: Hay varias opciones de unidades relativas que están vinculadas al tamaño de la ventana gráfica. Están cubiertos en la sección Principios de diseño web receptivo.

Agregue una **padding** propiedad al elemento con clase **red-boxy** configúrelo en **1.5em**.






```
<style>

 .injected-text {
```

```
    margin-bottom: -25px;
    text-align: center;
  }
  .box {
    border-style: solid;
    border-color: black;
    border-width: 5px;
    text-align: center;
  }
  .yellow-box {
    background-color: yellow;
    padding: 20px 40px 20px 40px;
  }
  .red-box {
    background-color: red;
    margin: 20px 40px 20px 40px;
    padding: 1.5em
  }
  .green-box {
    background-color: green;
    margin: 20px 40px 20px 40px;
  }
</style>
<h5 class="injected-text">margin</h5>
<div class="box yellow-box">
  <h5 class="box red-box" "padding">padding</h5>
  <h5 class="box green-box">padding</h5>
</div>
```

Aplicar estilo al elemento del cuerpo HTML

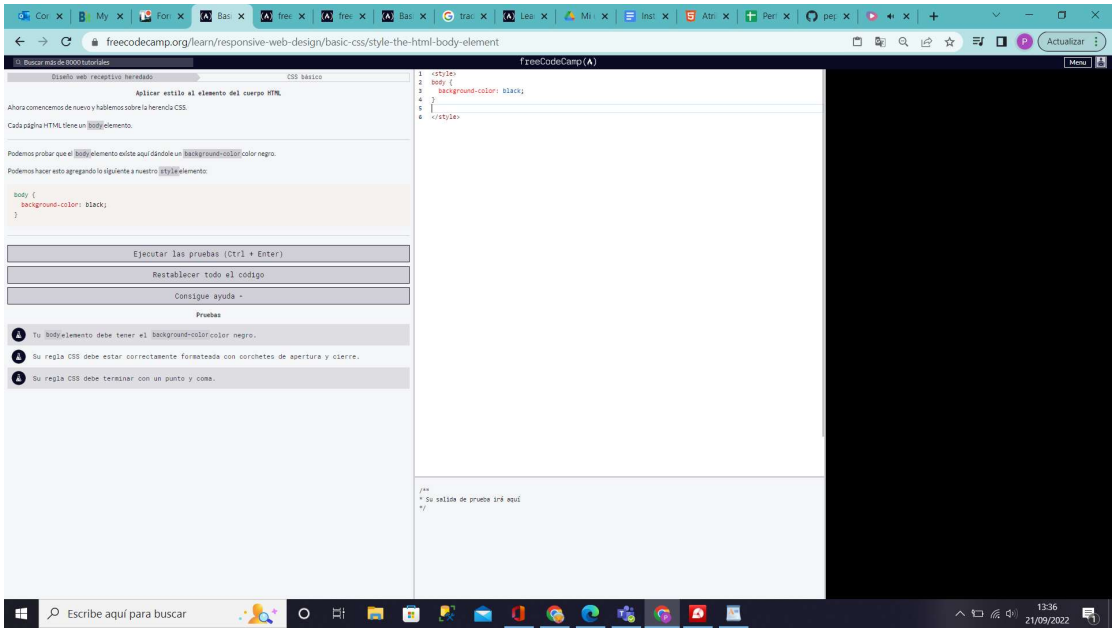
Ahora comencemos de nuevo y hablemos sobre la herencia CSS.

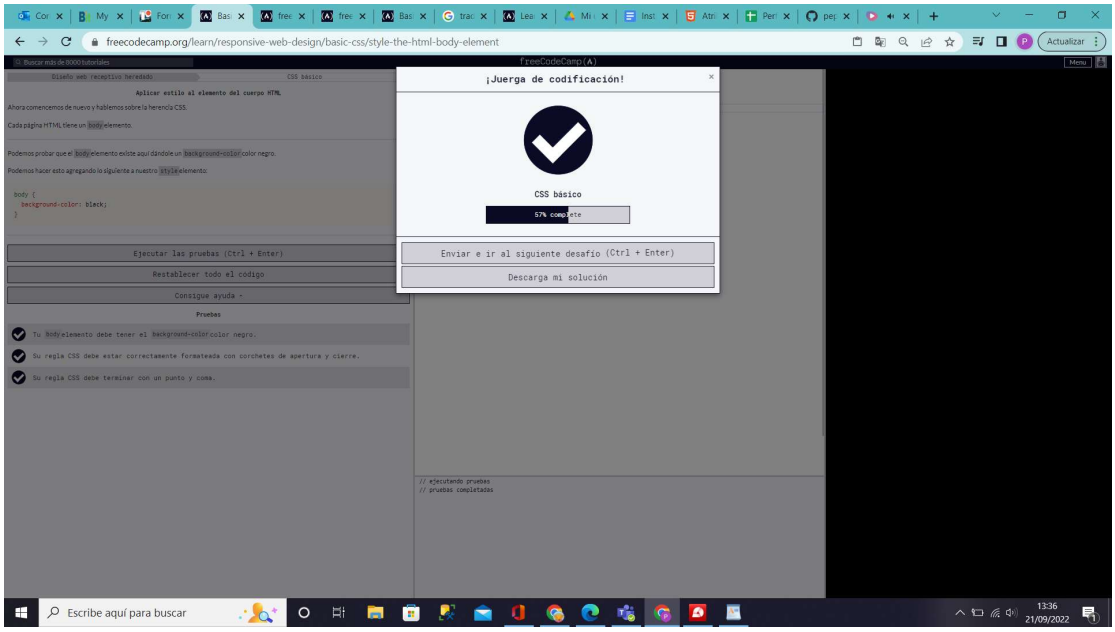
Cada página HTML tiene un **body** elemento.

Podemos probar que el bodyelemento existe aquí dándole un **background-color** color negro.

Podemos hacer esto agregando lo siguiente a nuestro **style** elemento:

```
body {
  background-color: black;
}
```





<style>

body {

background-color: black;

}

</style>

Heredar estilos del elemento del cuerpo

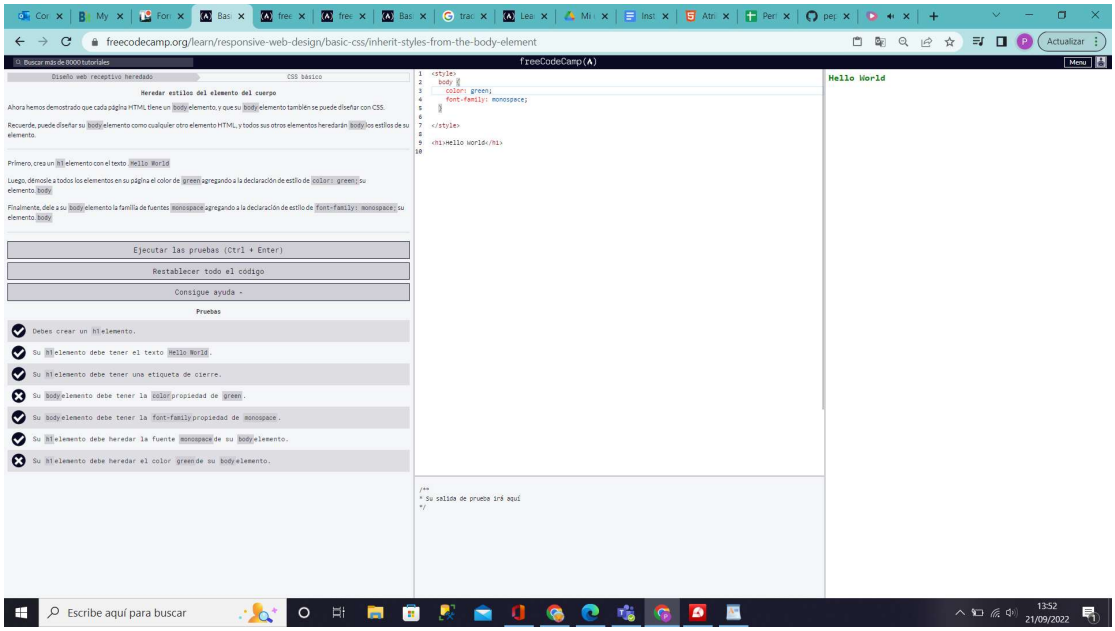
Ahora hemos demostrado que cada página HTML tiene un **body** elemento, y que su bodyelemento también se puede diseñar con CSS.

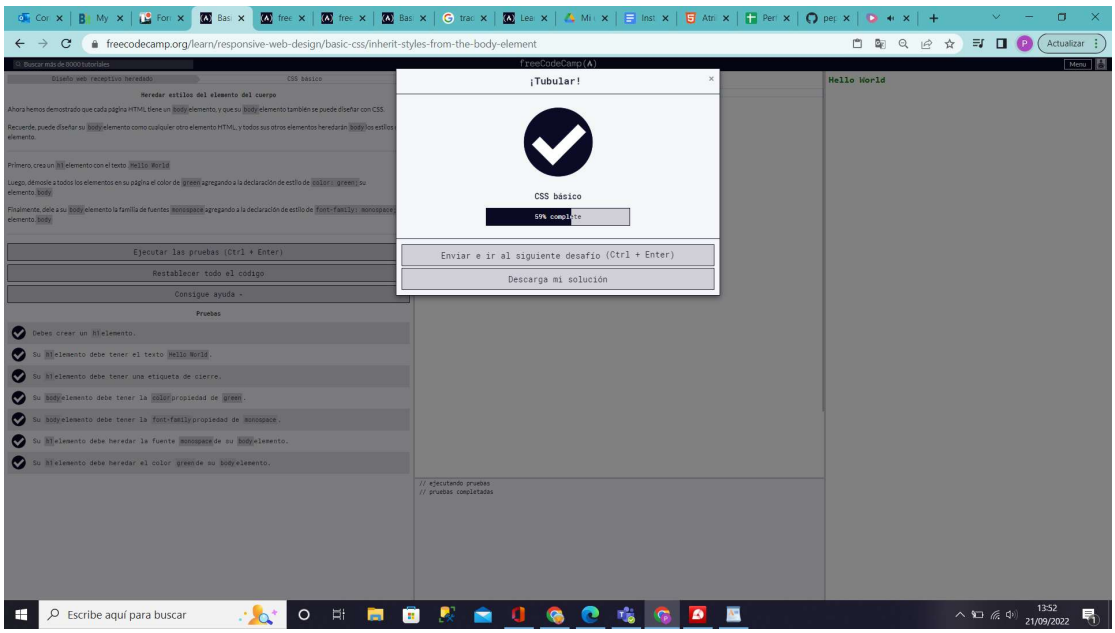
Recuerde, puede diseñar su body elemento como cualquier otro elemento HTML, y todos **sus otros elementos heredarán body** los **estilos** de su elemento.

Primero, crea un h1 elemento con el texto .**Hello World**

Luego, démosle a todos los elementos en su página el color de **green** agregando a la declaración de estilo de **color: green**;su elemento.body

Finalmente, dele a su bodyelemento la familia de fuentes **monospace** agregando a la declaración de estilo de **font-family: monospace**;su elemento.body





<style>

body {

```
  color: green;
  font-family: monospace;
 }
</style>
<h1>Hello World</h1>
```

Prioriza un estilo sobre otro

A veces, sus elementos HTML recibirán varios estilos que entran en conflicto entre sí.

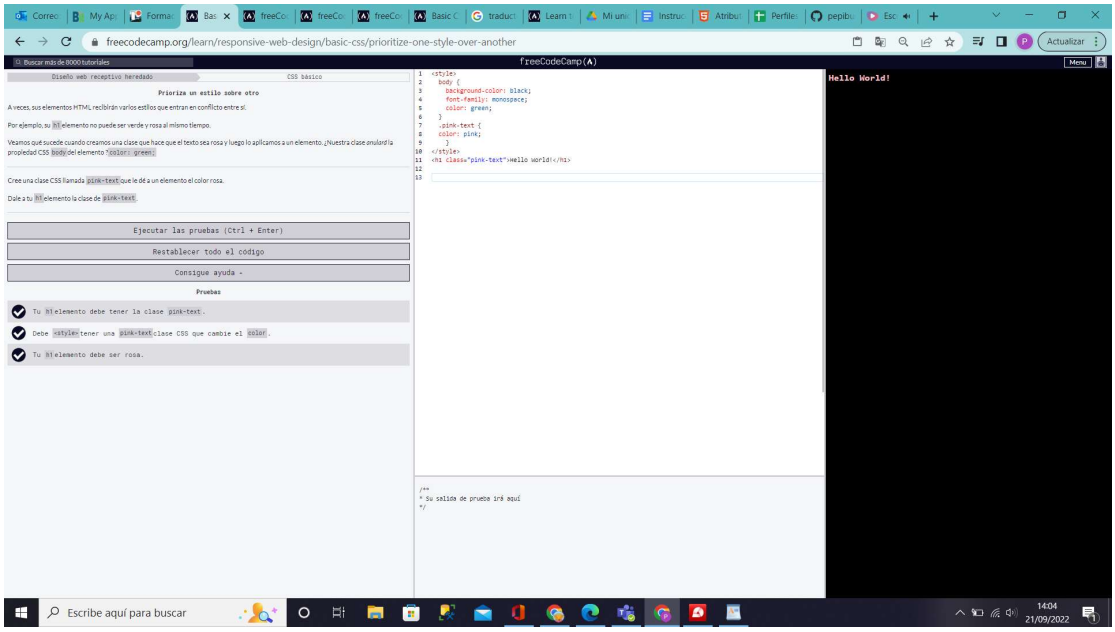
Por ejemplo, su **h1** elemento no puede ser verde y rosa al mismo tiempo.

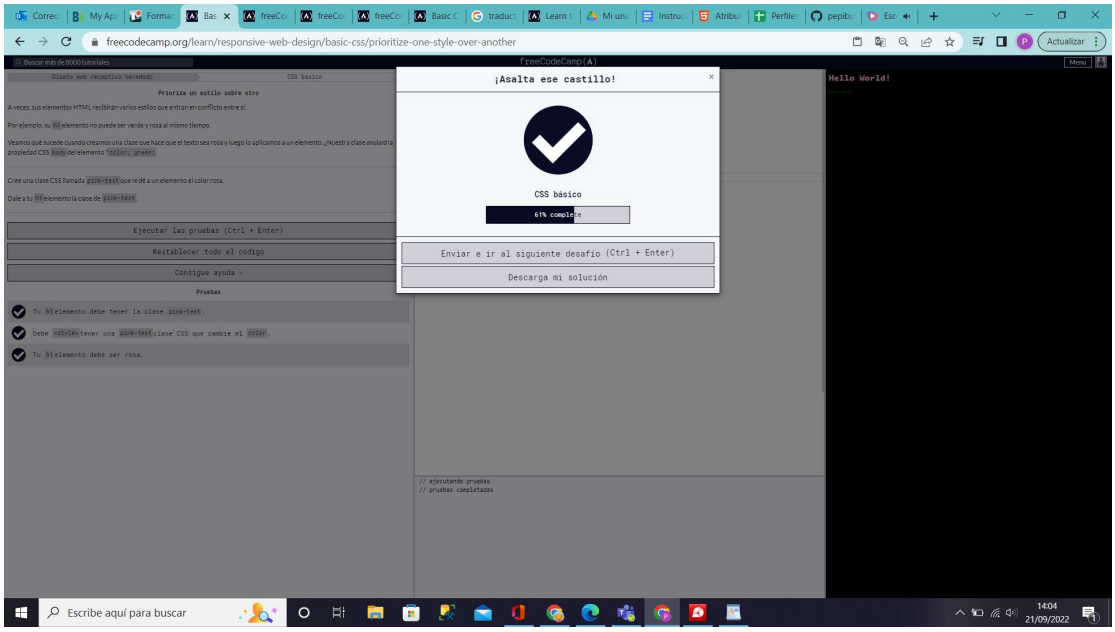
Veamos qué sucede cuando creamos una clase que hace que el texto sea rosa y luego lo aplicamos a un elemento. ¿Nuestra clase anulará la propiedad CSS **body** del elemento ?

color: green;

Cree una clase CSS llamada **pink-text** que le dé a un elemento el color rosa.

Dale a tu **h1** elemento la clase de **pink-text**.






```
<style>
  body {
    background-color: black;
    font-family: monospace;
    color: green;
  }
  .pink-text {
  color: pink;
    }
</style>
<h1 class="pink-text">Hello World!</h1>
```

Anular estilos en CSS subsiguientes

¡ Nuestra **pink-text** clase anuló **body** la declaración CSS de nuestro elemento!

Acabamos de demostrar que nuestras clases anularán el body CSS del elemento. Entonces, la siguiente pregunta lógica es, ¿qué podemos hacer para anular nuestra pink-text clase?

Cree una clase CSS adicional llamada **blue-text** que le dé a un elemento el color azul. Asegúrate de que esté debajo de tu **pink-text** declaración de clase.

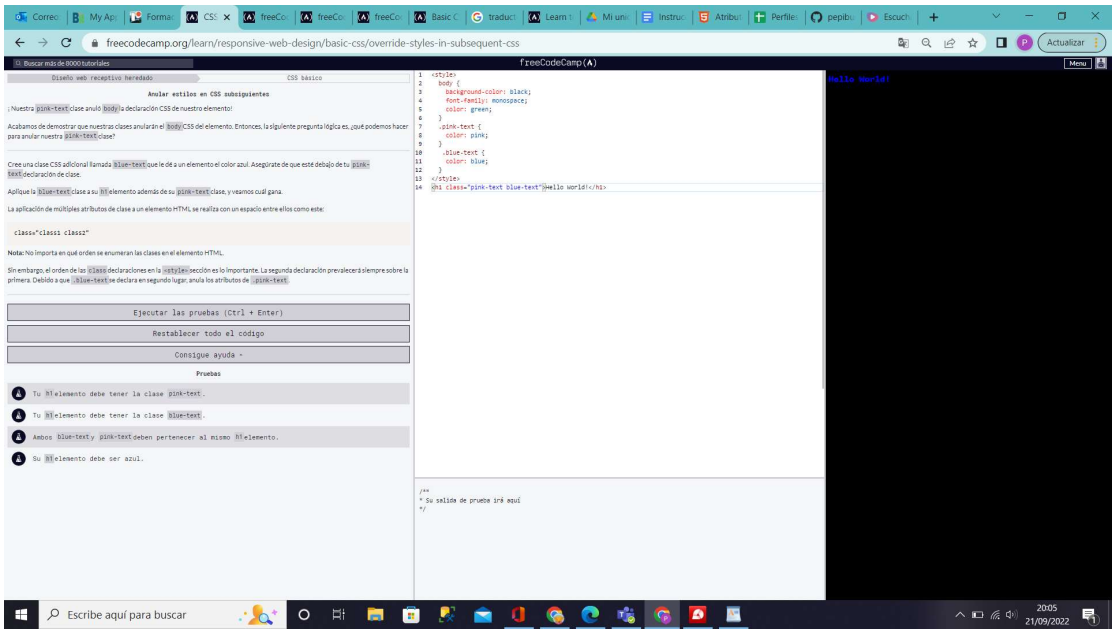
Aplique la blue-text clase a su h1elemento además de su pink-text clase, y veamos cuál gana.

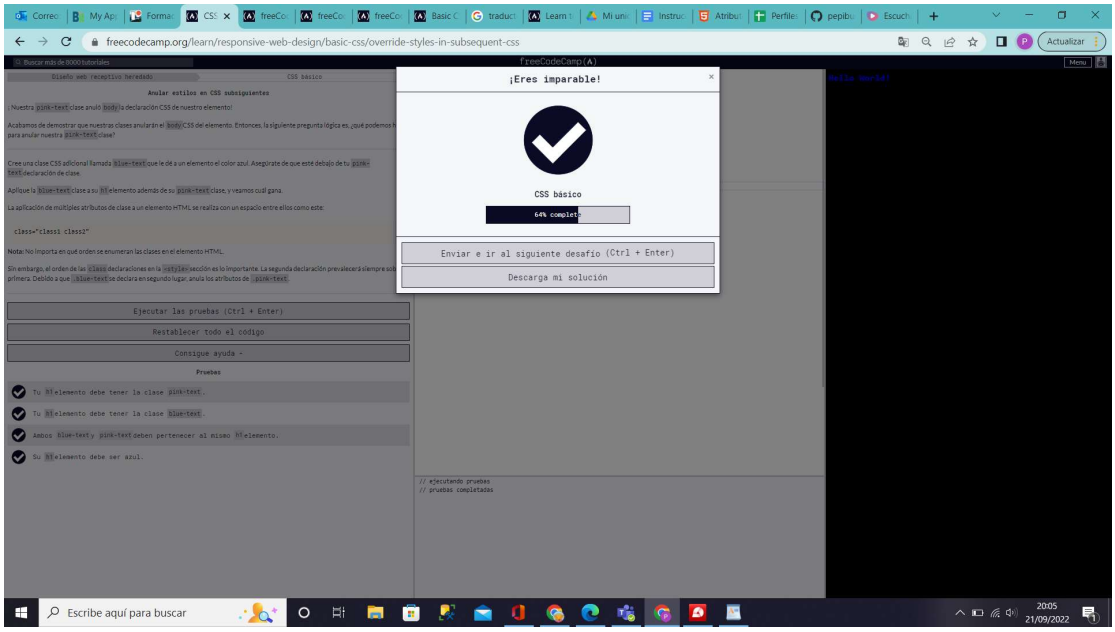
La aplicación de múltiples atributos de clase a un elemento HTML se realiza con un espacio entre ellos como este:

class="class1 class2"

Nota: No importa en qué orden se enumeran las clases en el elemento HTML.

Sin embargo, el orden de las class declaraciones en la **<style>** sección es lo importante. **La segunda declaración prevalecerá siempre sobre la primera.** Debido a que **.blue-text** se declara en segundo lugar, anula los atributos de **.pink-text.**






```
<style>
 body {
   background-color: black;
   font-family: monospace;
   color: green;
 }
 .pink-text {
   color: pink;
 }
  .blue-text {
   color: blue;
 }
</style>
<h1 class="pink-text blue-text">Hello World!</h1>
```

Anular declaraciones de clase por atributos de ID de estilo

Acabamos de demostrar que los navegadores leen CSS de arriba a abajo en el orden de su declaración. Eso significa que, en caso de conflicto, el navegador utilizará la última

declaración de CSS. Tenga en cuenta que incluso si hubiéramos puesto **blue-text** antes en las clases de nuestro **h1** elemento, ¡todavía miraría el orden de declaración y no el orden de su uso!

Pero aún no hemos terminado. Hay otras formas de anular CSS. ¿Recuerdas los atributos de identificación?

Anulemos sus clases y hagamos que su pink-textelemento sea de color naranja, dándole al elemento una identificación y luego dándole un estilo a esa identificación.blue-texth1h1

Dale a tu h1 elemento el idatributo de orange-text. Recuerde, los estilos de identificación se ven así:

<h1 id="orange-text">

Deje las clases **blue-text** y en su elemento.**pink-text h1**

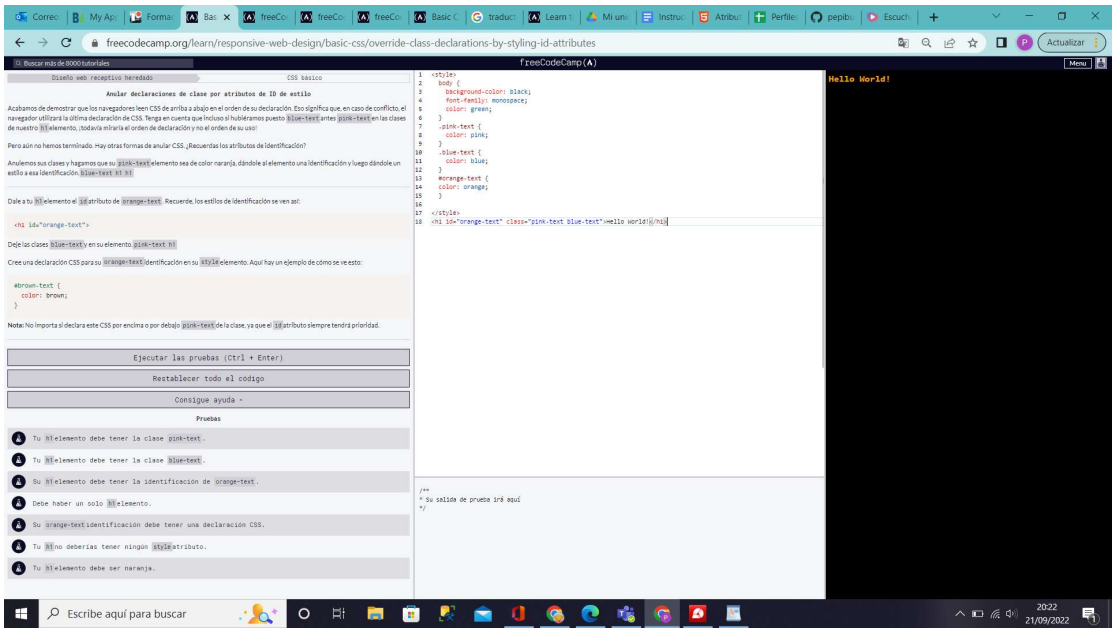
Cree una declaración CSS para su **orange-text** identificación en su **style** elemento. Aquí hay un ejemplo de cómo se ve esto:

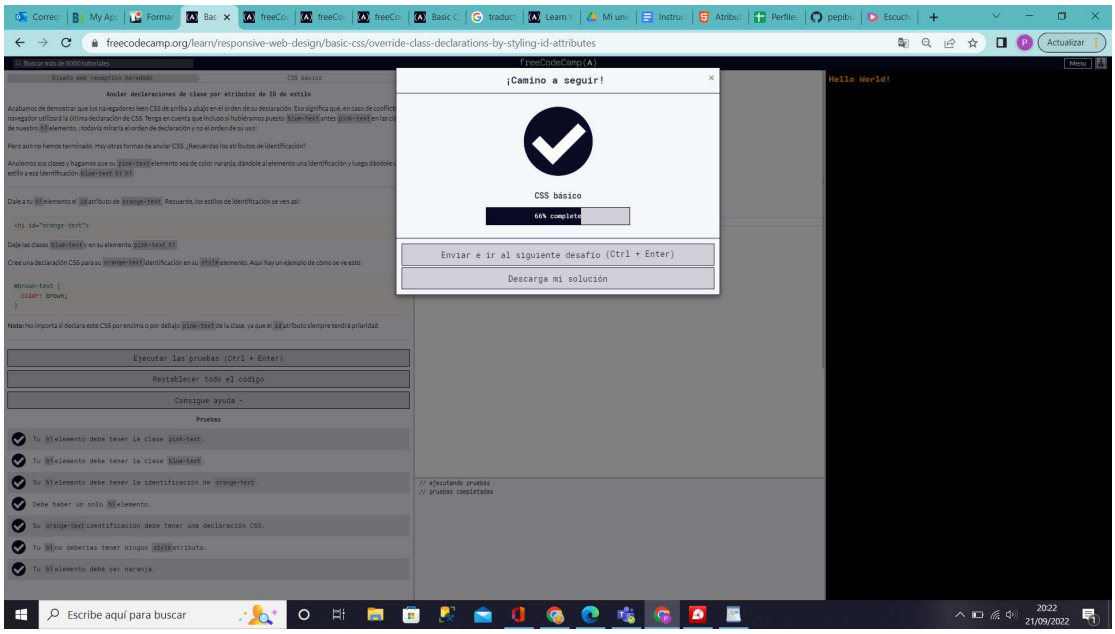
#brown-text {

color: brown;

}

Nota: No importa si declara este CSS por encima o por debajo **pink-text** de la clase, ya que el **id atributo siempre tendrá prioridad.**






```
<style>
  body {
    background-color: black;
    font-family: monospace;
    color: green;
  }
  .pink-text {
    color: pink;
  }
  .blue-text {
    color: blue;
  }
  #orange-text {
  color: orange;
  }
</style>
```

<h1 **id="orange-text"** class="pink-text blue-text">Hello World!</h1>

Anular declaraciones de clase con estilos en línea

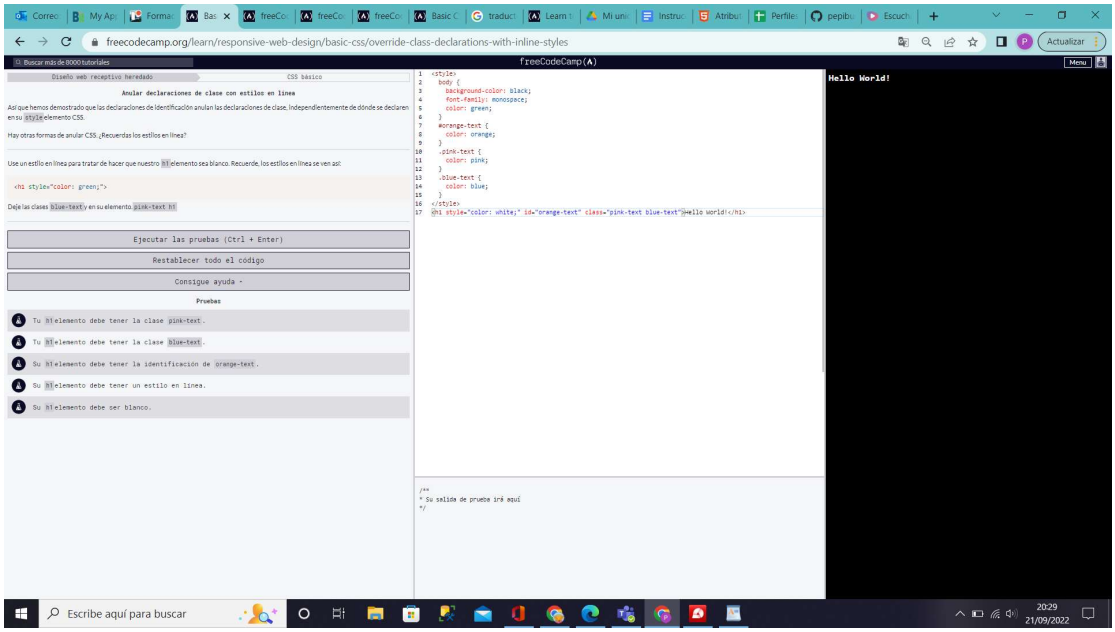
Así que hemos demostrado que las declaraciones de identificación anulan las declaraciones de clase, independientemente de dónde se declaren en su **style** elemento CSS.

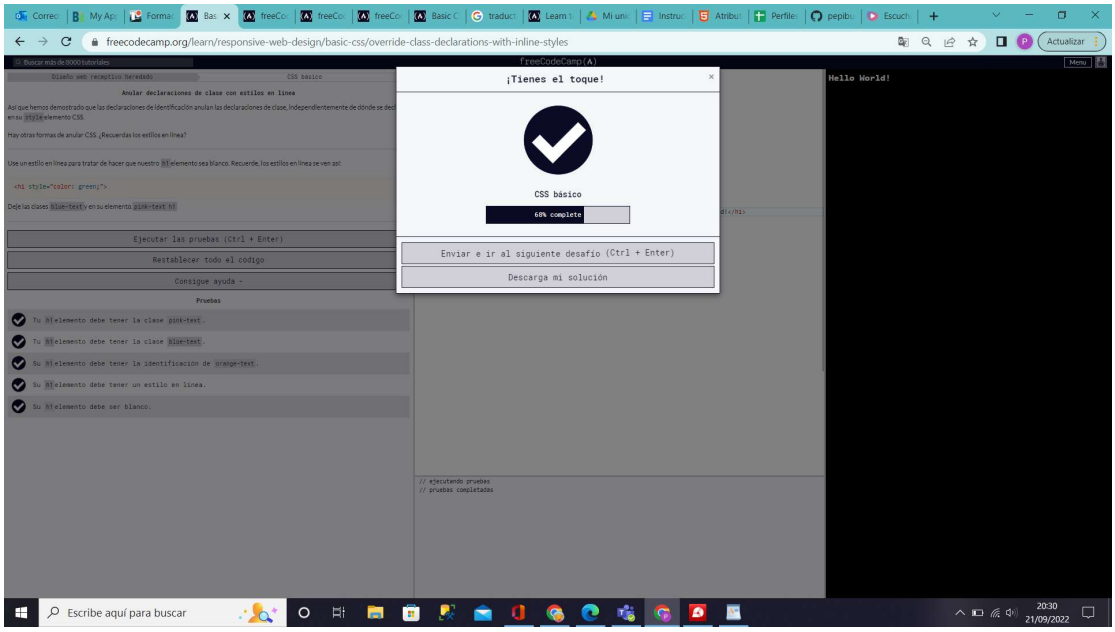
Hay otras formas de anular CSS. ¿Recuerdas los estilos en línea?

Use un estilo en línea para tratar de hacer que nuestro **h1** elemento sea blanco. Recuerde, **<u>los estilos en línea se ven así:</u>**

<h1 style="color: green;">

Deje las clases **blue-text** y en su elemento. **pink-text h1**






```
<style>
 body {
   background-color: black;
   font-family: monospace;
   color: green;
 }
 #orange-text {
   color: orange;
 }
 .pink-text {
   color: pink;
 }
 .blue-text {
   color: blue;
 }
</style>
<h1 style="color: white;" id="orange-text" class="pink-text blue-text">Hello World!</h1>
```

Anule todos los demás estilos usando Importante

¡Hurra! Acabamos de demostrar que **los estilos en línea anularán todas las declaraciones** CSS **en su style elemento**.

Pero espera. Hay una última forma de anular CSS. Este es el método más poderoso de todos. Pero antes de hacerlo, hablemos de por qué querrías anular CSS.

En muchas situaciones, utilizará bibliotecas CSS. Estos pueden anular accidentalmente su propio CSS. Entonces, cuando necesite estar absolutamente seguro de que un elemento tiene un CSS específico, puede usar **!important**.

Regresemos a nuestra pink-text declaración de clase. **Recuerde que nuestra pink-text clase fue anulada por declaraciones de clase posteriores, declaraciones de identificación y estilos en línea.**

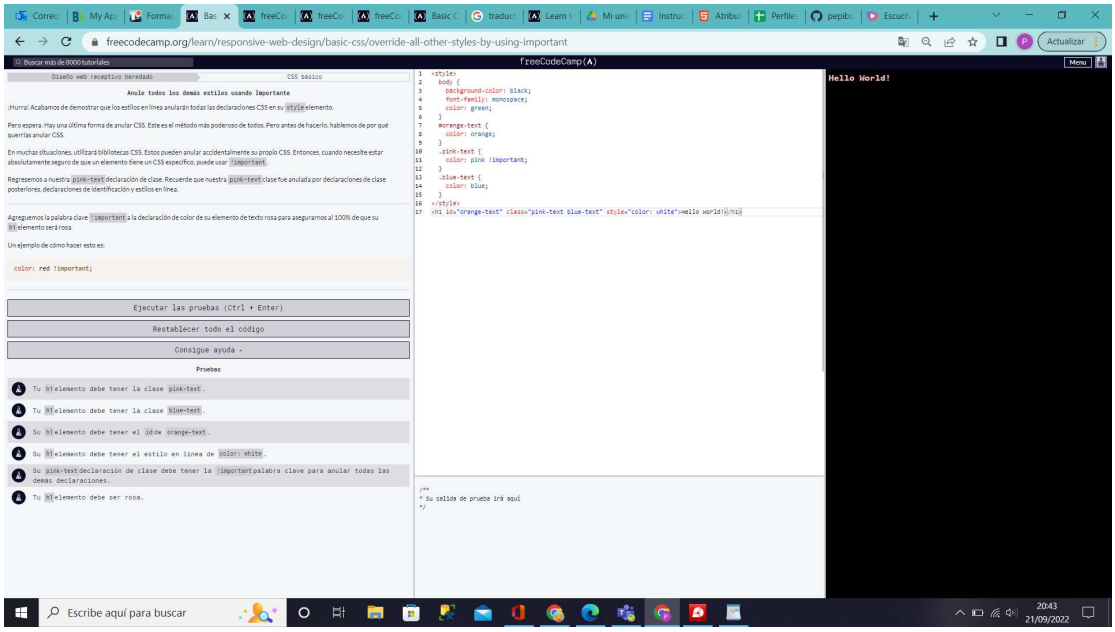
Agreguemos la **palabra clave !important** a la declaración de color de su elemento de texto rosa para asegurarnos al 100% de que su h1 elemento será rosa.

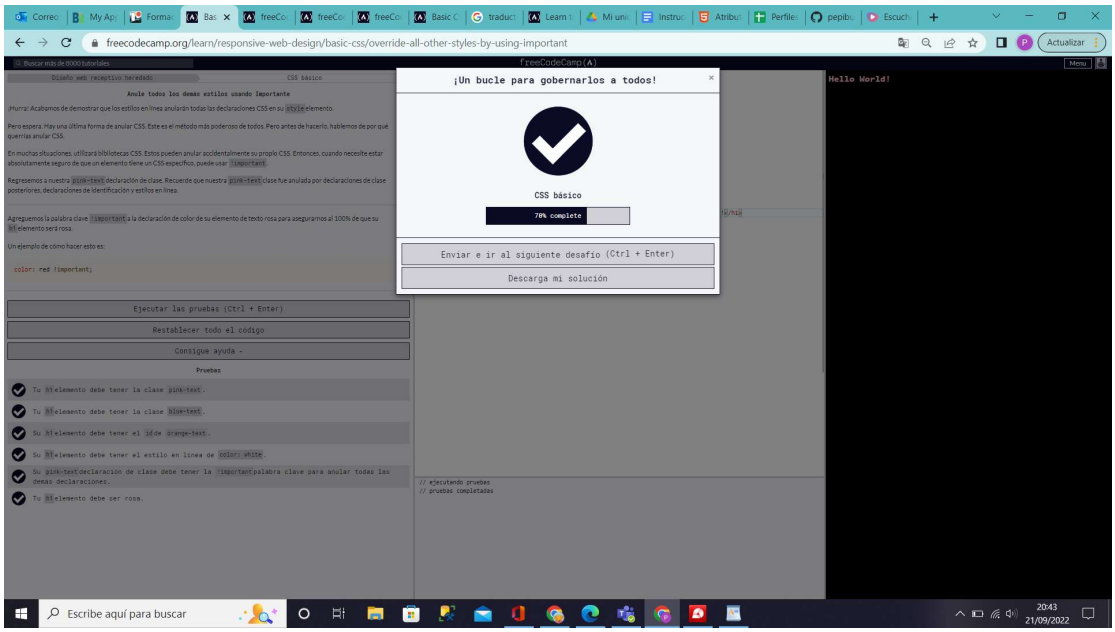
Un ejemplo de cómo hacer esto es:

color: red !important;

ORDEN DE PREVALENCIA DE DECLARACIONES:

1. PALABRA CLAVE **!important**
2. ESTILOS EN LÍNEA.
3. DECLARACIÓN DE IDENTIFICACIÓN (ID).
4. DECLARACIÓN DE CLASE (CLASS) POSTERIOR.
5. DECLARACIÓN DE CLASE (CLASS).






```
<style>
  body {
    background-color: black;
    font-family: monospace;
    color: green;
```

```
 }
 #orange-text {
  color: orange;
 }
 .pink-text {
  color: pink !important;
 }
 .blue-text {
  color: blue;
 }
</style>
<h1 id="orange-text" class="pink-text blue-text" style="color: white">Hello World!</h1>
```

Usar código hexadecimal para colores específicos

¿Sabías que hay **otras formas de representar colores** en CSS? Una de estas formas se llama **código hexadecimal** o código hexadecimal para abreviar.

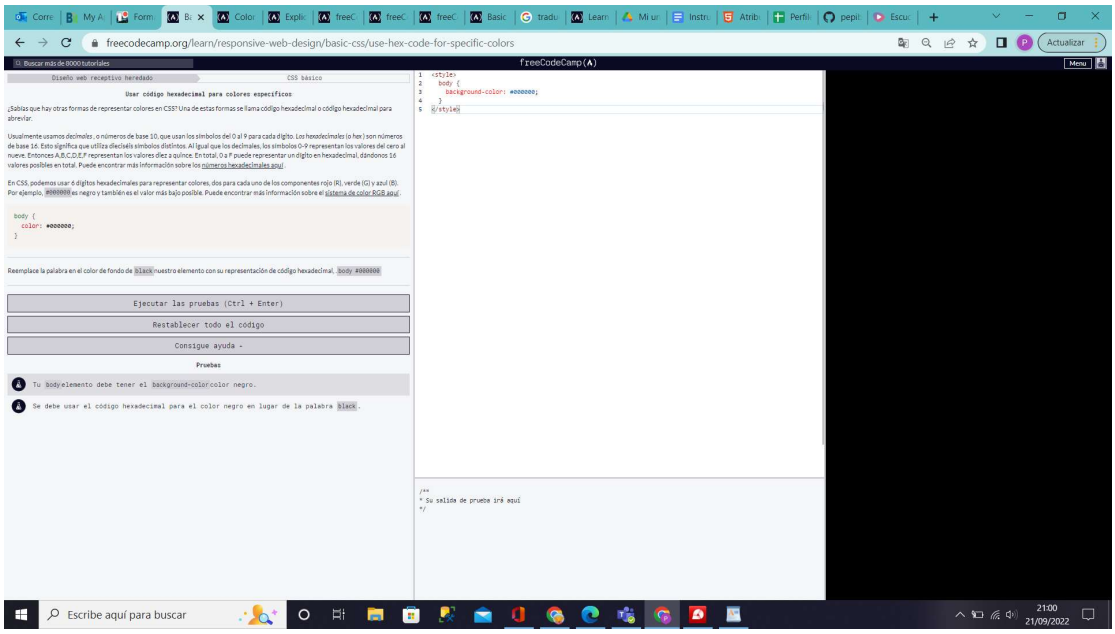
Usualmente usamos decimales , o números de base 10, que usan los símbolos del **0 al 9** para cada dígito. Los hexadecimales (o hex) son números de base 16. Esto significa que utiliza dieciséis símbolos distintos. Al igual que los decimales, los símbolos 0-9 representan los valores del cero al nueve. Entonces **A,B,C,D,E,F** representan los **valores diez a quince**. En total, **0 a F** puede representar un dígito en hexadecimal, dándonos **16 valores posibles** en total. Puede encontrar más información sobre los números hexadecimales aquí .(https://www.freecodecamp.org/news/hexadecimal-number-system/)

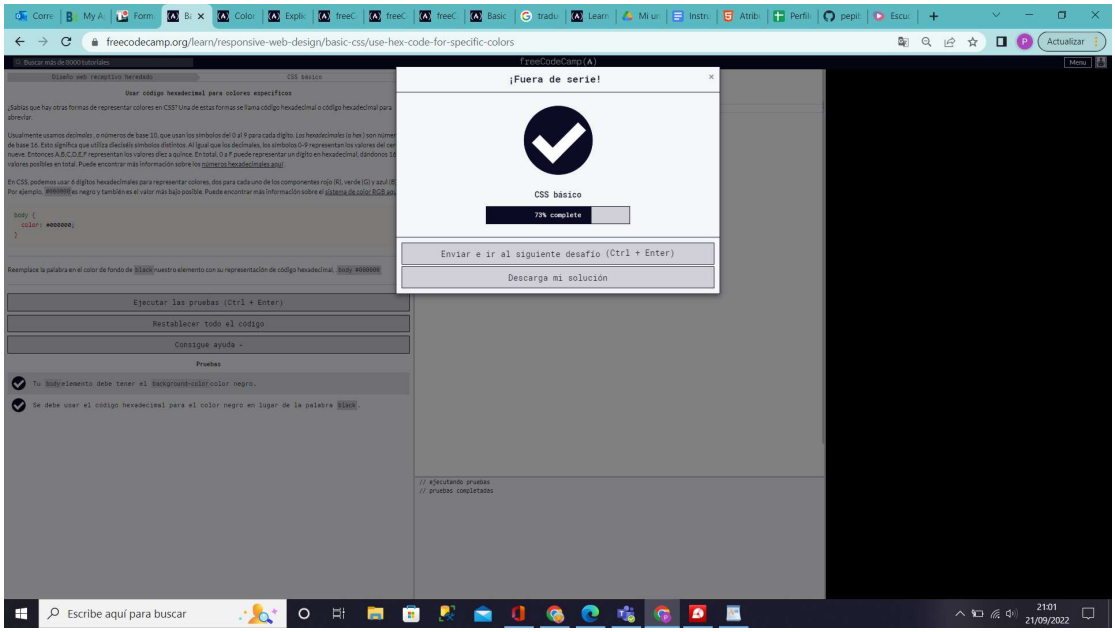
En CSS, podemos usar **6 dígitos** hexadecimales para representar colores, **dos para cada uno** de los componentes rojo **(R), verde (G) y azul (B)**. Por ejemplo, **#000000** es negro y también es el valor más bajo posible. Puede encontrar más información sobre el sistema de color RGB aquí .(https://www.freecodecamp.org/news/rgb-color-html-and-css-guide/#whatisthergbcolormodel)

```
body {
  color: #000000;
}
```

Reemplace la palabra en el color de fondo de **black** nuestro elemento con su

representación de código hexadecimal, .**body#000000**






```
<style>
  body {
    background-color: #000000;
```

```
 }
</style>
```

Usar código hexadecimal para mezclar colores

Para repasar, los códigos hexadecimales usan 6 dígitos hexadecimales para representar los colores, dos para los componentes rojo (R), verde (G) y azul (B).

A partir de estos tres colores puros (rojo, verde y azul), podemos variar las cantidades de cada uno para crear más de 16 millones de colores.

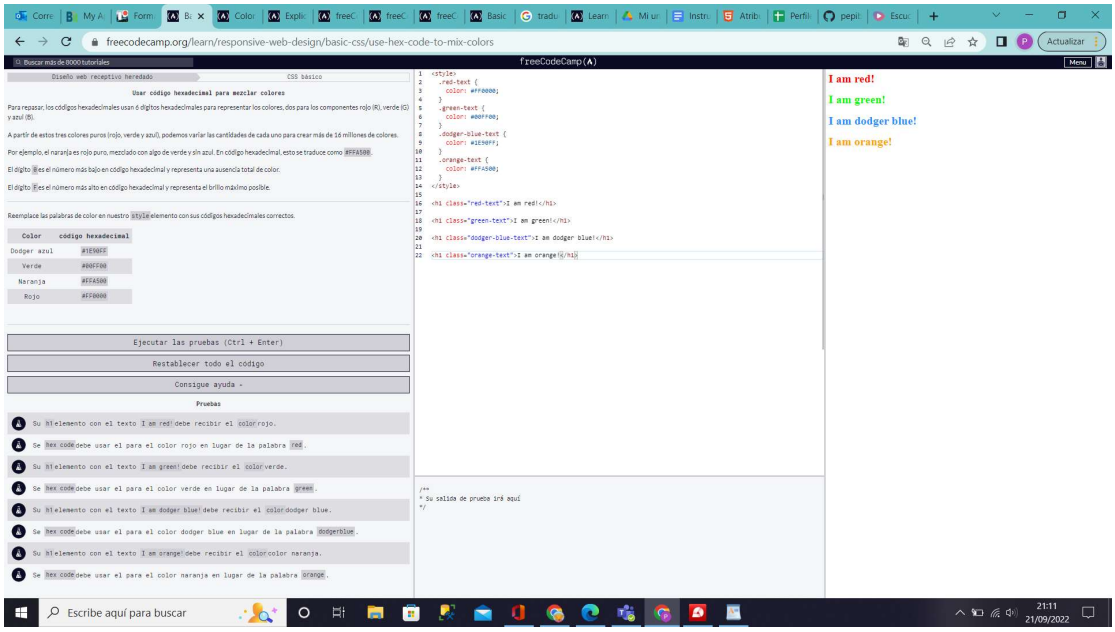
Por ejemplo, el naranja es rojo puro, mezclado con algo de verde y sin azul. En código hexadecimal, esto se traduce como **#FFA500**.

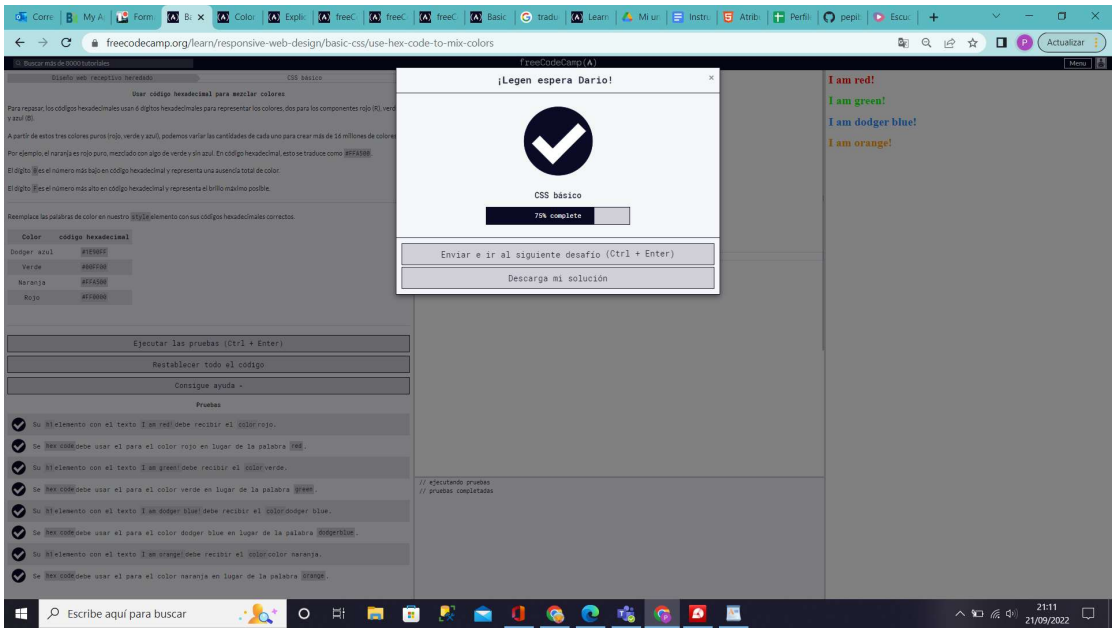
El dígito **0** es el **número más bajo** en código hexadecimal y representa una **ausencia total de color.**

El dígito **F** es el **número más alto** en código hexadecimal y representa el **brillo máximo posible.**

Reemplace las palabras de color en nuestro **style** elemento con sus códigos hexadecimales correctos.

Color	código hexadecimal
Dodger azul	#1E90FF
Verde	#00FF00
Naranja	#FFA500
Rojo	#FF0000






```
<style>
  .red-text {
    color: #FF0000;
  }
  .green-text {
```

```
  color: #00FF00;
 }
 .dodger-blue-text {
  color: #1E90FF;
 }
 .orange-text {
  color: #FFA500;
 }
</style>
<h1 class="red-text">I am red!</h1>
<h1 class="green-text">I am green!</h1>
<h1 class="dodger-blue-text">I am dodger blue!</h1>
<h1 class="orange-text">I am orange!</h1>
```

Usar código hexadecimal abreviado

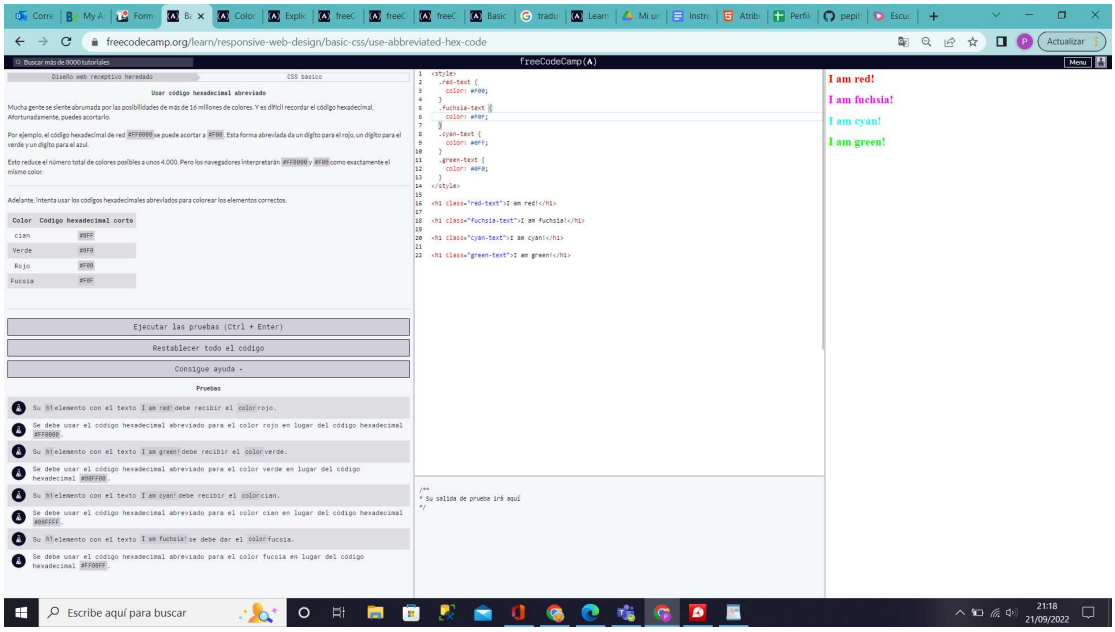
Mucha gente se siente abrumada por las posibilidades de más de 16 millones de colores. Y es difícil recordar el código hexadecimal. Afortunadamente, puedes acortarlo.

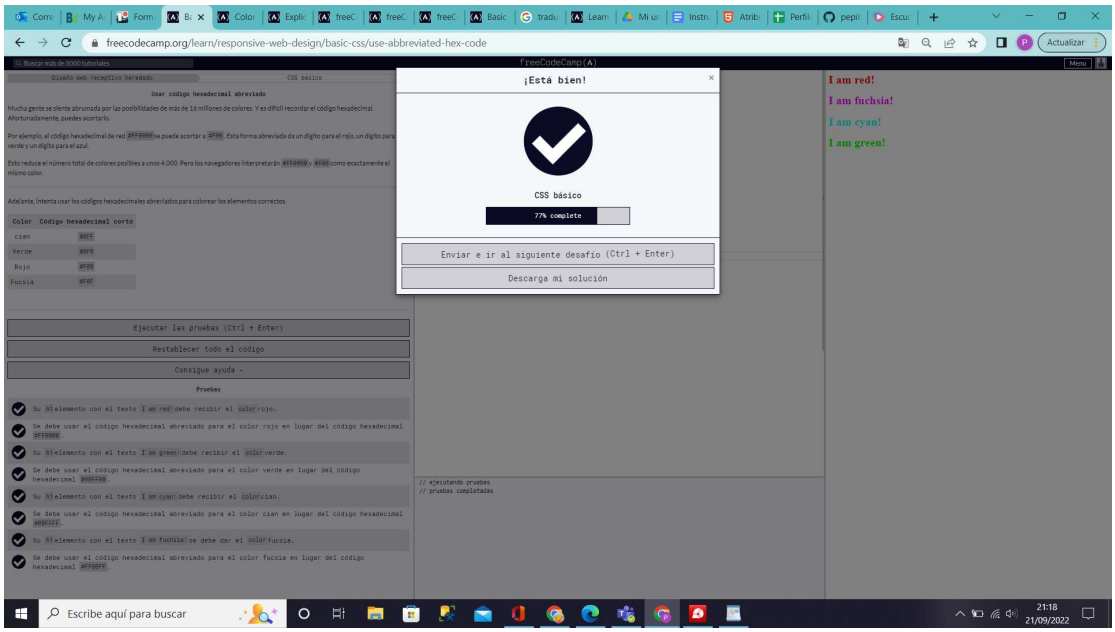
Por ejemplo, el código hexadecimal de red **#FF0000** se puede acortar a **#F00**. Esta forma abreviada da un dígito para el rojo, un dígito para el verde y un dígito para el azul.

Esto **reduce el número total** de colores posibles **a unos 4.000**. **Pero los navegadores interpretarán #FF0000** y #F00 como exactamente el mismo color.

Adelante, intenta usar los códigos hexadecimales abreviados para colorear los elementos correctos.

Color	Código hexadecimal corto
cian	#0FF
Verde	#0F0
Rojo	#F00
Fucsia	#F0F






```
<style>
 .red-text {
  color: #F00;
 }
 .fuchsia-text {
```

```
  color: #F0F;
 }
 .cyan-text {
  color: #0FF;
 }
 .green-text {
  color: #0F0;
 }
</style>
<h1 class="red-text">I am red!</h1>
<h1 class="fuchsia-text">I am fuchsia!</h1>
<h1 class="cyan-text">I am cyan!</h1>
<h1 class="green-text">I am green!</h1>
```

Usar valores RGB para colorear elementos

Otra forma de representar los **colores** en CSS es **mediante** el uso de **RGB** valores.

El RGB valor para el **negro** se ve así:

rgb(0, 0, 0)

El RGB valor para el **blanco** se ve así:

rgb(255, 255, 255)

En lugar de usar seis dígitos hexadecimales como lo hace con el código hexadecimal, RGBespecifique el brillo de cada color con **un número entre 0 y 255.**

Si haces los cálculos, los dos dígitos de un color equivalen a 16 por 16, lo que nos da 256 valores totales. Entonces RGB, que comienza a contar desde cero, tiene exactamente el mismo número de valores posibles que el código hexadecimal.

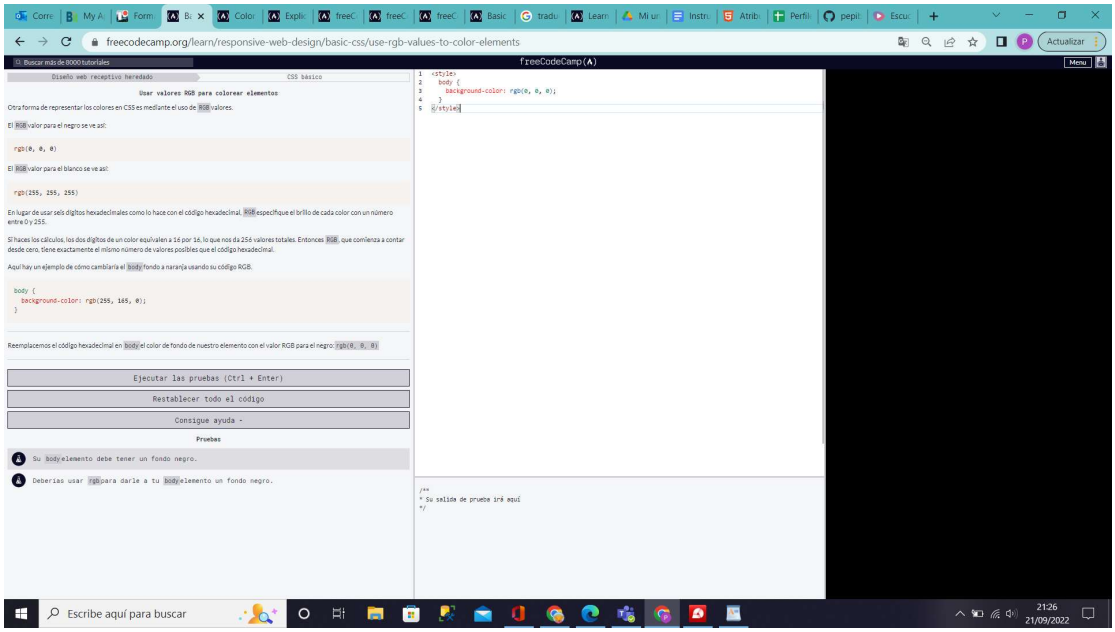
Aquí hay un ejemplo de cómo cambiaría el **body** fondo a naranja usando su código RGB.

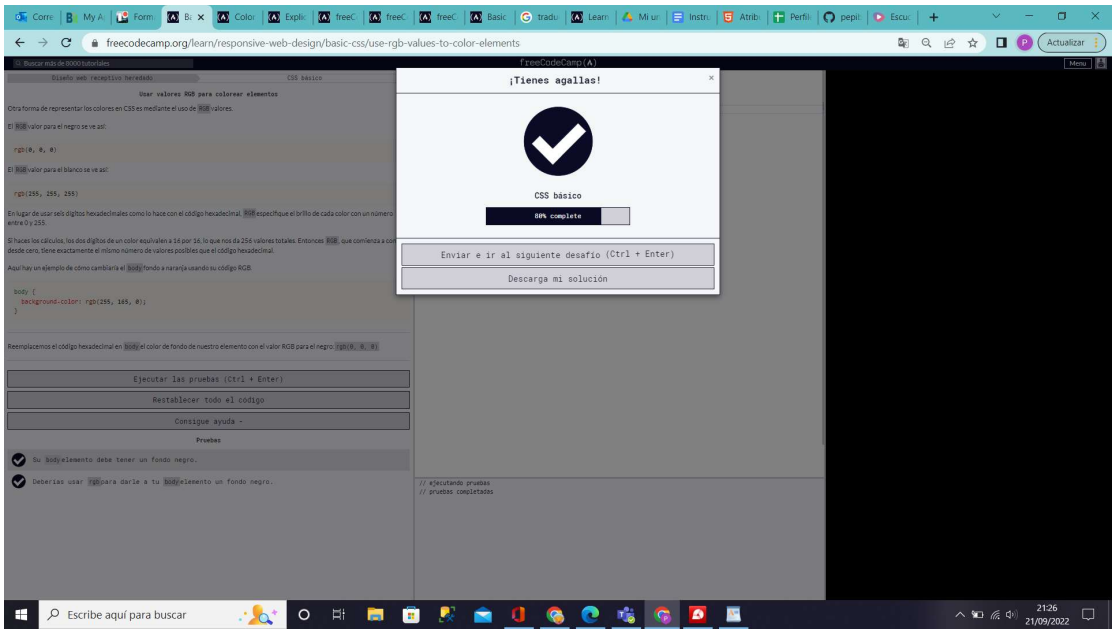
body {

background-color: rgb(255, 165, 0);

}

Reemplacemos el código hexadecimal en body el color de fondo de nuestro elemento con el valor RGB para el negro: **rgb(0, 0, 0)**





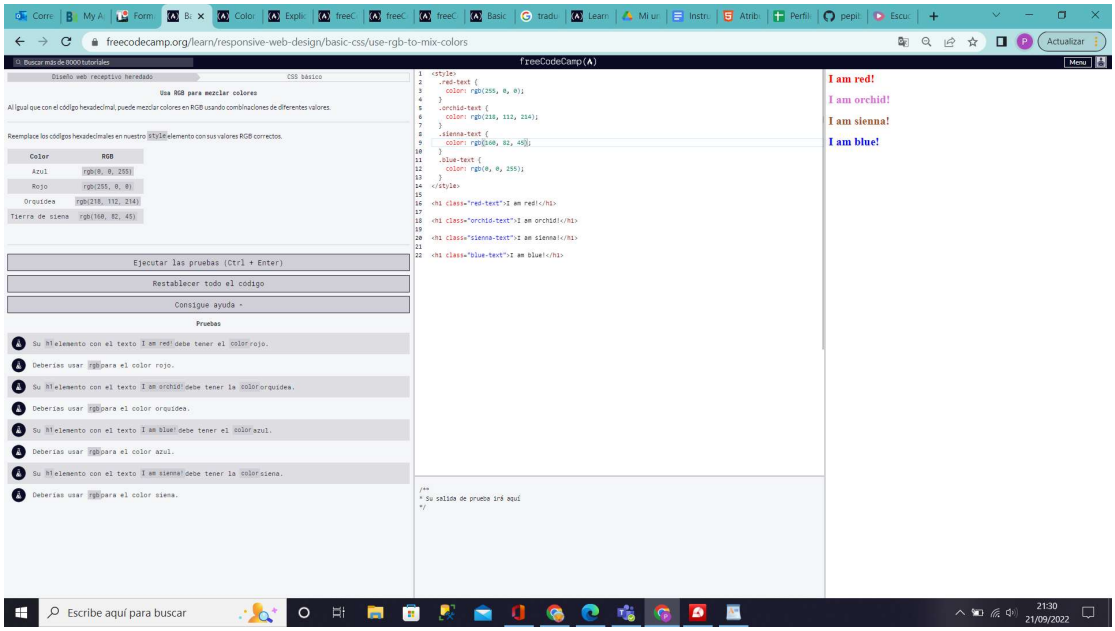
```
<style>
  body {
    background-color: rgb(0, 0, 0);
  }
</style>
```

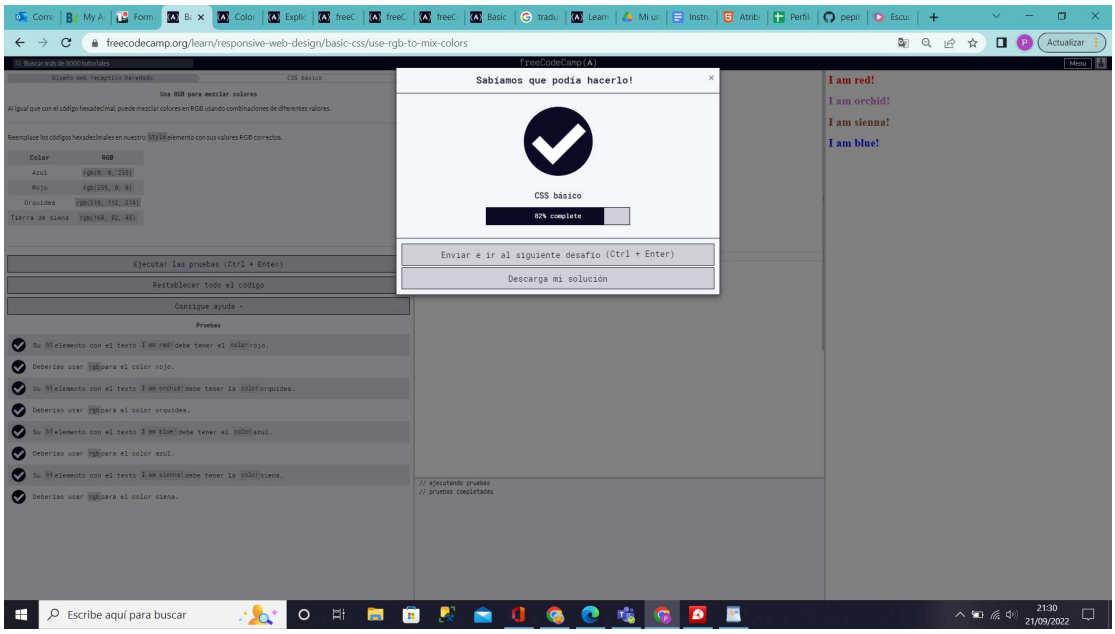
Usa RGB para mezclar colores

Al igual que con el código hexadecimal, **puede mezclar colores en RGB** usando combinaciones de diferentes valores.

Reemplace los códigos hexadecimales en nuestro style elemento con sus valores RGB correctos.

Color	RGB
Azul	rgb(0, 0, 255)
Rojo	rgb(255, 0, 0)
Orquídea	rgb(218, 112, 214)
Tierra de siena	rgb(160, 82, 45)






```
<style>
  .red-text {
    color: rgb(255, 0, 0);
  }
  .orchid-text {
    color: rgb(218, 112, 214);
  }
  .sienna-text {
    color: rgb(160, 82, 45);
  }
  .blue-text {
    color: rgb(0, 0, 255);
  }
</style>
<h1 class="red-text">I am red!</h1>
<h1 class="orchid-text">I am orchid!</h1>
```

<h1 class="sienna-text">I am sienna!</h1>

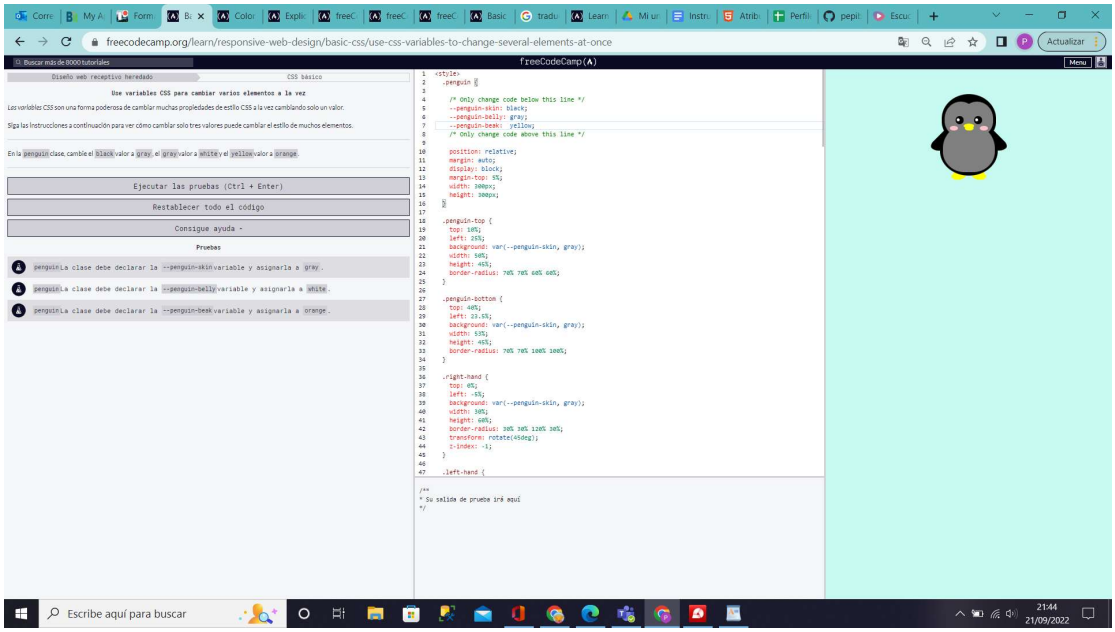
<h1 class="blue-text">I am blue!</h1>

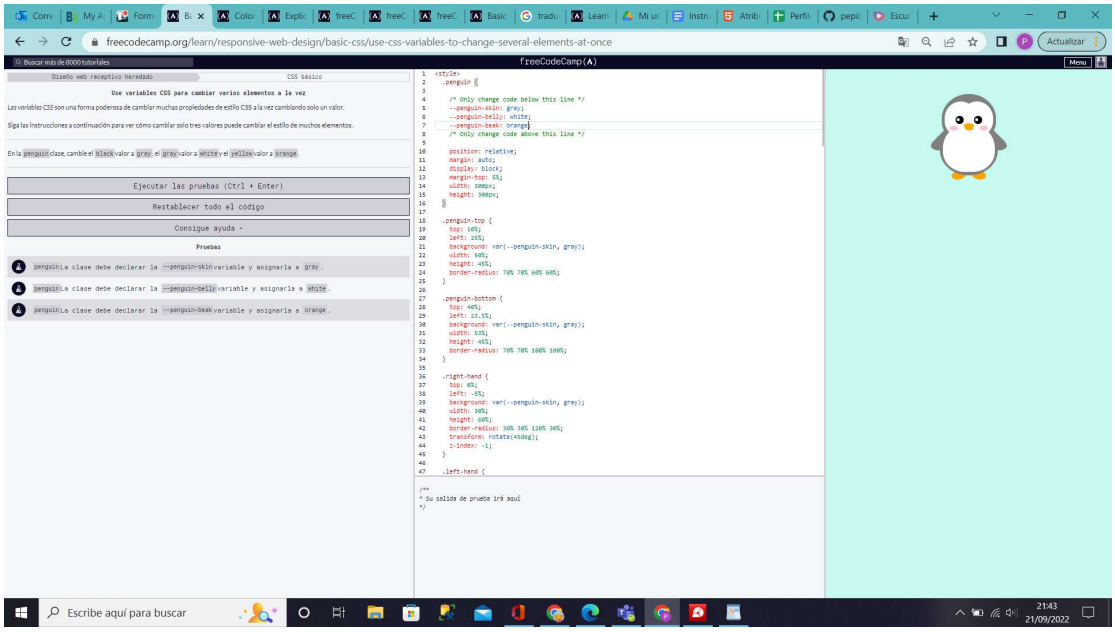
Use variables CSS para cambiar varios elementos a la vez

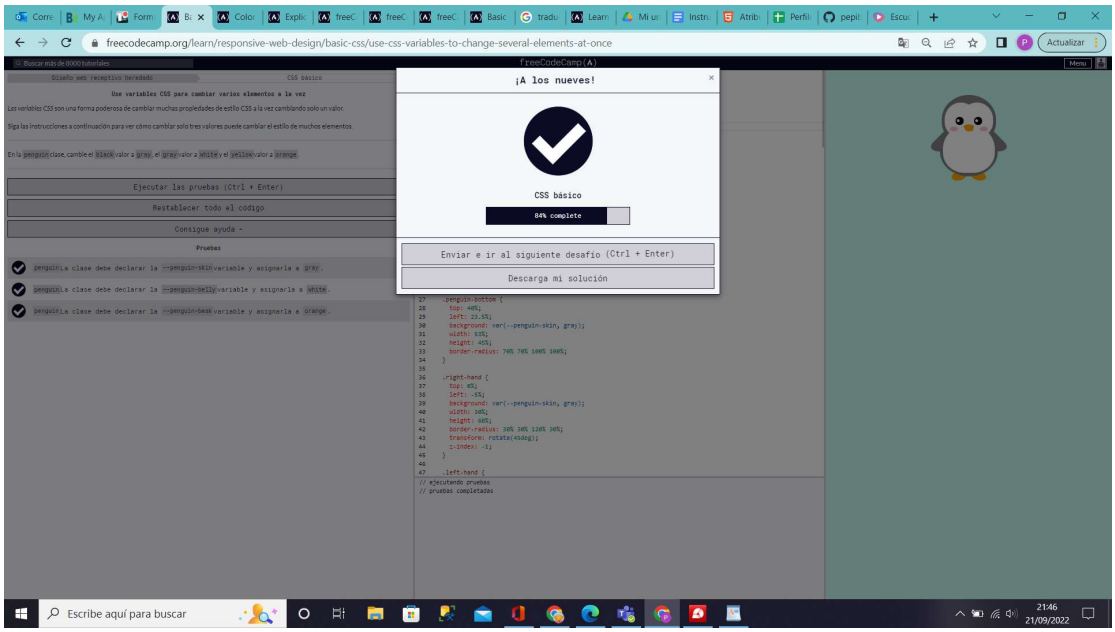
Las variables CSS son una forma poderosa de cambiar muchas propiedades de estilo CSS a la vez cambiando solo un valor.

Siga las instrucciones a continuación para ver cómo **cambiar solo tres valores** puede **cambiar el estilo de muchos elementos**.

En la **penguin** clase, cambie el **black** valor a **gray**, el gray valor a **white** y el **yellow** valor a **orange**.








```
<style>
 .penguin {
/*Solo cambie el código debajo de esta línea*/
/* Only change code below this line */
   --penguin-skin: gray;
```

```
  --penguin-belly: white;
  --penguin-beak:  orange;
/*Solo cambie el código por encima de esta línea*/
 /* Only change code above this line */
  position: relative;
  margin: auto;
  display: block;
  margin-top: 5%;
  width: 300px;
  height: 300px;
 }
 .penguin-top {
  top: 10%;
  left: 25%;
  background: var(--penguin-skin, gray);
  width: 50%;
  height: 45%;
  border-radius: 70% 70% 60% 60%;
 }
 .penguin-bottom {
  top: 40%;
  left: 23.5%;
  background: var(--penguin-skin, gray);
  width: 53%;
  height: 45%;
  border-radius: 70% 70% 100% 100%;
 }
 .right-hand {
```

```
  top: 0%;
  left: -5%;
  background: var(--penguin-skin, gray);
  width: 30%;
  height: 60%;
  border-radius: 30% 30% 120% 30%;
  transform: rotate(45deg);
  z-index: -1;
}
.left-hand {
  top: 0%;
  left: 75%;
  background: var(--penguin-skin, gray);
  width: 30%;
  height: 60%;
  border-radius: 30% 30% 30% 120%;
  transform: rotate(-45deg);
  z-index: -1;
}
.right-cheek {
  top: 15%;
  left: 35%;
  background: var(--penguin-belly, white);
  width: 60%;
  height: 70%;
  border-radius: 70% 70% 60% 60%;
}
.left-cheek {
```

```
  top: 15%;
  left: 5%;
  background: var(--penguin-belly, white);
  width: 60%;
  height: 70%;
  border-radius: 70% 70% 60% 60%;
}
.belly {
  top: 60%;
  left: 2.5%;
  background: var(--penguin-belly, white);
  width: 95%;
  height: 100%;
  border-radius: 120% 120% 100% 100%;
}
.right-feet {
  top: 85%;
  left: 60%;
  background: var(--penguin-beak, orange);
  width: 15%;
  height: 30%;
  border-radius: 50% 50% 50% 50%;
  transform: rotate(-80deg);
  z-index: -2222;
}
.left-feet {
  top: 85%;
  left: 25%;
```

```
  background: var(--penguin-beak, orange);
  width: 15%;
  height: 30%;
  border-radius: 50% 50% 50% 50%;
  transform: rotate(80deg);
  z-index: -2222;
}
.right-eye {
  top: 45%;
  left: 60%;
  background: black;
  width: 15%;
  height: 17%;
  border-radius: 50%;
}
.left-eye {
  top: 45%;
  left: 25%;
  background: black;
  width: 15%;
  height: 17%;
  border-radius: 50%;
}
.sparkle {
  top: 25%;
  left: 15%;
  background: white;
  width: 35%;
```

```
  height: 35%;
  border-radius: 50%;
}
.blush-right {
  top: 65%;
  left: 15%;
  background: pink;
  width: 15%;
  height: 10%;
  border-radius: 50%;
}
.blush-left {
  top: 65%;
  left: 70%;
  background: pink;
  width: 15%;
  height: 10%;
  border-radius: 50%;
}
.beak-top {
  top: 60%;
  left: 40%;
  background: var(--penguin-beak, orange);
  width: 20%;
  height: 10%;
  border-radius: 50%;
}
.beak-bottom {
```

```
    top: 65%;
    left: 42%;
    background: var(--penguin-beak, orange);
    width: 16%;
    height: 10%;
    border-radius: 50%;
  }
  body {
    background:#c6faf1;
  }
  .penguin * {
    position: absolute;
  }
</style>
<div class="penguin">
  <div class="penguin-bottom">
    <div class="right-hand"></div>
    <div class="left-hand"></div>
    <div class="right-feet"></div>
    <div class="left-feet"></div>
  </div>
  <div class="penguin-top">
    <div class="right-cheek"></div>
    <div class="left-cheek"></div>
    <div class="belly"></div>
    <div class="right-eye">
      <div class="sparkle"></div>
    </div>
```

```
    <div class="left-eye">
      <div class="sparkle"></div>
    </div>
    <div class="blush-right"></div>
    <div class="blush-left"></div>
    <div class="beak-top"></div>
    <div class="beak-bottom"></div>
  </div>
</div>
```

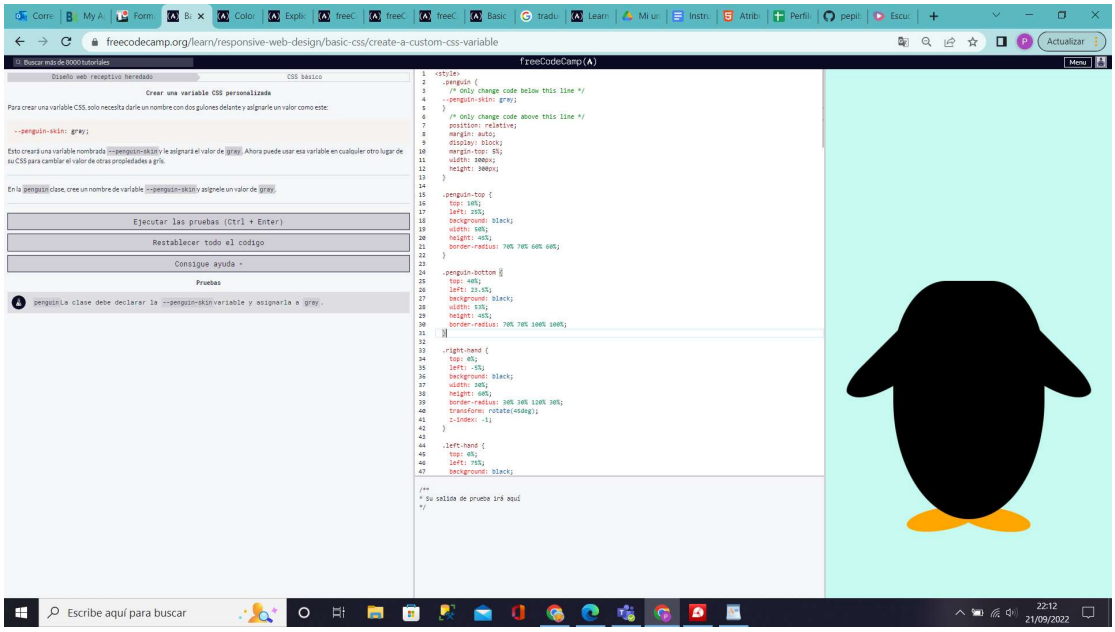
Crear una variable CSS personalizada

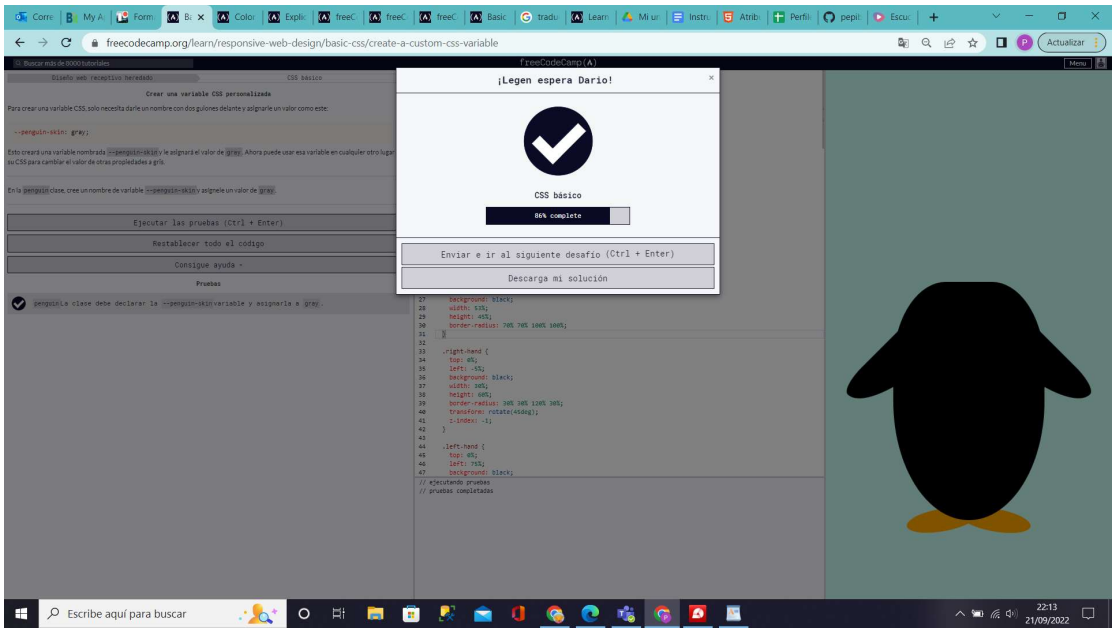
Para **crear una variable CSS**, solo necesita darle **un nombre con dos guiones delante** y **asignarle un valor** como este:

--penguin-skin: gray;

Esto creará una variable nombrada **--penguin-skin** y le asignará el valor de **gray**. Ahora puede usar esa variable en cualquier otro lugar de su CSS para cambiar el valor de otras propiedades a **gris.**

En la penguin clase, cree un nombre de variable --penguin-skin y asígnele un valor de gray.


freecodecamp.org/learn/responsive-web-design/basic-css/create-a-custom-css-variable
Actualizar
Buscar más de 8000 tutoriales
freeCodeCamp(A)
Menu
Diseño web receptivo heredado
CSS básico
Crear una variable CSS personalizada
Para crear una variable CSS, solo necesita darle un nombre con dos guiones delante y asignarle un valor como este:
--penguin-skin: gray;
Esto creará una variable nombrada --penguin-skin y le asignará el valor de gray. Ahora puede usar esa variable en cualquier otro lugar de su CSS para cambiar el valor de otras propiedades a gris.
En la penguin clase, cree un nombre de variable --penguin-skin y asígnele un valor de gray.
Ejecutar las pruebas (Ctrl + Enter)
Restablecer todo el código
Consigue ayuda
Pruebas
penguinLa clase debe declarar la --penguin-skinvariable y asignarla a gray.
/**
* Su salida de prueba irá aquí
*/
Escribe aquí para buscar
22:12
21/09/2022


freecodecamp.org/learn/responsive-web-design/basic-css/create-a-custom-css-variable
Actualizar
¡Legen espera Dario!
CSS básico
86% complete
Enviar e ir al siguiente desafío (Ctrl + Enter)
Descarga mi solución
Ejecutar las pruebas (Ctrl + Enter)
Restablecer todo el código
Consigue ayuda
Pruebas
// ejecutando pruebas
// pruebas completadas
Escribe aquí para buscar
22:13
21/09/2022

```
<style>
  .penguin {
/*Solo cambie el código debajo de esta línea*/
/* Only change code below this line */
  --penguin-skin: gray;
  }
/*Solo cambie el código por encima de esta línea*/
/* Only change code above this line */
    position: relative;
    margin: auto;
    display: block;
    margin-top: 5%;
    width: 300px;
    height: 300px;
  }
  .penguin-top {
    top: 10%;
    left: 25%;
    background: black;
    width: 50%;
    height: 45%;
    border-radius: 70% 70% 60% 60%;
  }
  .penguin-bottom {
    top: 40%;
    left: 23.5%;
    background: black;
    width: 53%;
```

```
  height: 45%;
  border-radius: 70% 70% 100% 100%;
}
.right-hand {
  top: 0%;
  left: -5%;
  background: black;
  width: 30%;
  height: 60%;
  border-radius: 30% 30% 120% 30%;
  transform: rotate(45deg);
  z-index: -1;
}
.left-hand {
  top: 0%;
  left: 75%;
  background: black;
  width: 30%;
  height: 60%;
  border-radius: 30% 30% 30% 120%;
  transform: rotate(-45deg);
  z-index: -1;
}
.right-cheek {
  top: 15%;
  left: 35%;
  background: white;
  width: 60%;
```

```
  height: 70%;
  border-radius: 70% 70% 60% 60%;
}
.left-cheek {
  top: 15%;
  left: 5%;
  background: white;
  width: 60%;
  height: 70%;
  border-radius: 70% 70% 60% 60%;
}
.belly {
  top: 60%;
  left: 2.5%;
  background: white;
  width: 95%;
  height: 100%;
  border-radius: 120% 120% 100% 100%;
}
.right-feet {
  top: 85%;
  left: 60%;
  background: orange;
  width: 15%;
  height: 30%;
  border-radius: 50% 50% 50% 50%;
  transform: rotate(-80deg);
  z-index: -2222;
```

```
}
.left-feet {
  top: 85%;
  left: 25%;
  background: orange;
  width: 15%;
  height: 30%;
  border-radius: 50% 50% 50% 50%;
  transform: rotate(80deg);
  z-index: -2222;
}
.right-eye {
  top: 45%;
  left: 60%;
  background: black;
  width: 15%;
  height: 17%;
  border-radius: 50%;
}
.left-eye {
  top: 45%;
  left: 25%;
  background: black;
  width: 15%;
  height: 17%;
  border-radius: 50%;
}
.sparkle {
```

```
  top: 25%;
  left: 15%;
  background: white;
  width: 35%;
  height: 35%;
  border-radius: 50%;
}
.blush-right {
  top: 65%;
  left: 15%;
  background: pink;
  width: 15%;
  height: 10%;
  border-radius: 50%;
}
.blush-left {
  top: 65%;
  left: 70%;
  background: pink;
  width: 15%;
  height: 10%;
  border-radius: 50%;
}
.beak-top {
  top: 60%;
  left: 40%;
  background: orange;
  width: 20%;
```

```
    height: 10%;
    border-radius: 50%;
  }
  .beak-bottom {
    top: 65%;
    left: 42%;
    background: orange;
    width: 16%;
    height: 10%;
    border-radius: 50%;
  }
  body {
    background:#c6faf1;
  }
  .penguin * {
    position: absolute;
  }
</style>
<div class="penguin" "--penguin-skin">
  <div class="penguin-bottom">
    <div class="right-hand"></div>
    <div class="left-hand"></div>
    <div class="right-feet"></div>
    <div class="left-feet"></div>
  </div>
  <div class="penguin-top">
    <div class="right-cheek"></div>
    <div class="left-cheek"></div>
```

```
    <div class="belly"></div>
    <div class="right-eye">
      <div class="sparkle"></div>
    </div>
    <div class="left-eye">
      <div class="sparkle"></div>
    </div>
    <div class="blush-right"></div>
    <div class="blush-left"></div>
    <div class="beak-top"></div>
    <div class="beak-bottom"></div>
  </div>
</div>
```

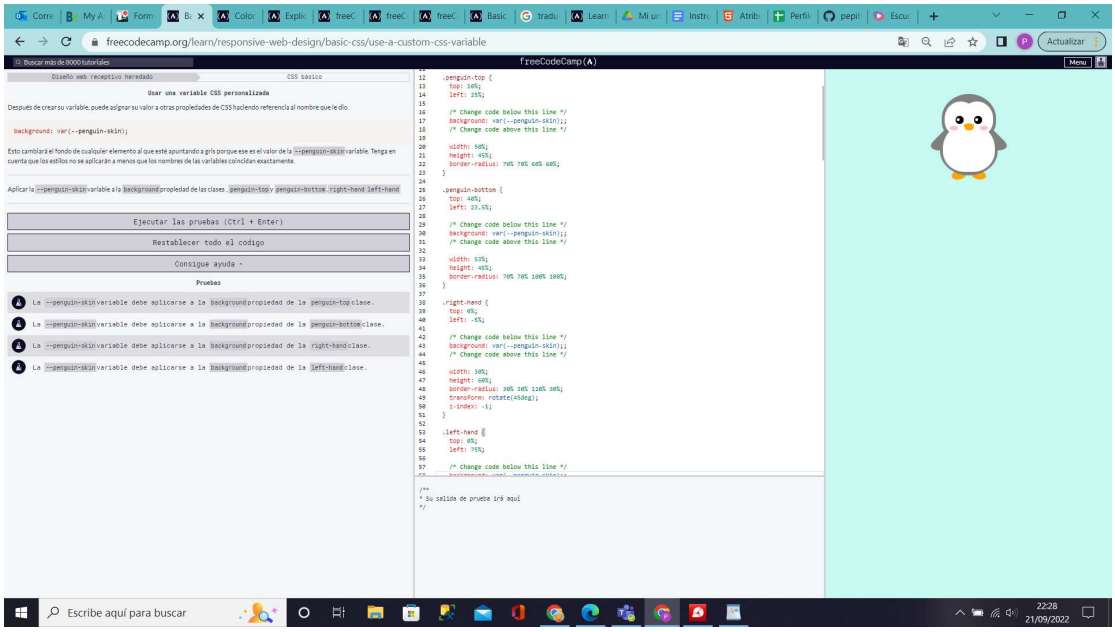
Usar una variable CSS personalizada

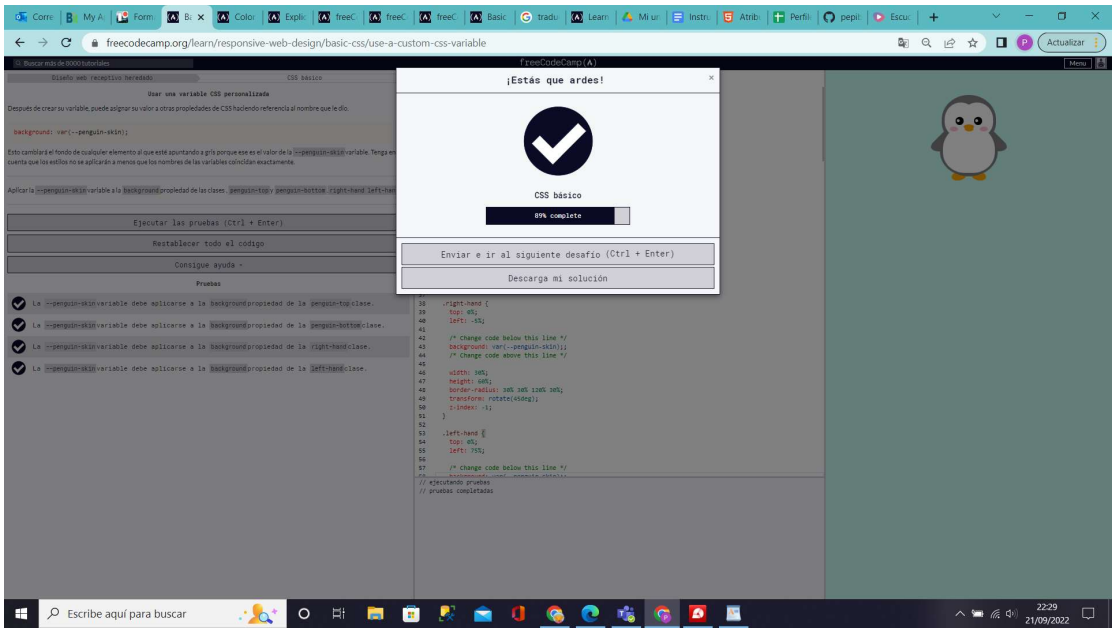
Después de crear su variable, puede asignar su valor a otras propiedades de CSS haciendo referencia al nombre que le dio.

background: var(--penguin-skin);

Esto cambiará el fondo de cualquier elemento al que esté apuntando a gris porque ese es el valor de la **--penguin-skin** variable. Tenga en cuenta que los estilos no se aplicarán a menos que los nombres de las variables coincidan exactamente.

Aplicar la **--penguin-skin** variable a la background propiedad de las clases , **penguin-topy penguin-bottom**.**right-handleft-hand**






```
<style>
 .penguin {
   --penguin-skin: gray;
   position: relative;
   margin: auto;
```

```
  display: block;
  margin-top: 5%;
  width: 300px;
  height: 300px;
}
.penguin-top {
  top: 10%;
  left: 25%;
  /* Change code below this line */ /*Cambiar el código debajo de esta línea*/
  background: var(--penguin-skin);
  /* Change code above this line */
  width: 50%;
  height: 45%;
  border-radius: 70% 70% 60% 60%;
}
.penguin-bottom {
  top: 40%;
  left: 23.5%;
  /* Change code below this line */ /*Cambiar el código debajo de esta línea*/
  background: var(--penguin-skin);
  /* Change code above this line */
  width: 53%;
  height: 45%;
  border-radius: 70% 70% 100% 100%;
}
.right-hand {
  top: 0%;
  left: -5%;
```

```
  /* Change code below this line */ /*Cambiar el código debajo de esta línea*/
  background: var(--penguin-skin);
  /* Change code above this line */
  width: 30%;
  height: 60%;
  border-radius: 30% 30% 120% 30%;
  transform: rotate(45deg);
  z-index: -1;
}
.left-hand {
  top: 0%;
  left: 75%;
  /* Change code below this line */ /*Cambiar el código debajo de esta línea*/
  background: var(--penguin-skin);
  /* Change code above this line */
  width: 30%;
  height: 60%;
  border-radius: 30% 30% 30% 120%;
  transform: rotate(-45deg);
  z-index: -1;
}
.right-cheek {
  top: 15%;
  left: 35%;
  background: white;
  width: 60%;
  height: 70%;
  border-radius: 70% 70% 60% 60%;
```

```
}
.left-cheek {
  top: 15%;
  left: 5%;
  background: white;
  width: 60%;
  height: 70%;
  border-radius: 70% 70% 60% 60%;
}
.belly {
  top: 60%;
  left: 2.5%;
  background: white;
  width: 95%;
  height: 100%;
  border-radius: 120% 120% 100% 100%;
}
.right-feet {
  top: 85%;
  left: 60%;
  background: orange;
  width: 15%;
  height: 30%;
  border-radius: 50% 50% 50% 50%;
  transform: rotate(-80deg);
  z-index: -2222;
}
.left-feet {
```

```
  top: 85%;
  left: 25%;
  background: orange;
  width: 15%;
  height: 30%;
  border-radius: 50% 50% 50% 50%;
  transform: rotate(80deg);
  z-index: -2222;
}
.right-eye {
  top: 45%;
  left: 60%;
  background: black;
  width: 15%;
  height: 17%;
  border-radius: 50%;
}
.left-eye {
  top: 45%;
  left: 25%;
  background: black;
  width: 15%;
  height: 17%;
  border-radius: 50%;
}
.sparkle {
  top: 25%;
  left: 15%;
```

```
  background: white;
  width: 35%;
  height: 35%;
  border-radius: 50%;
}
.blush-right {
  top: 65%;
  left: 15%;
  background: pink;
  width: 15%;
  height: 10%;
  border-radius: 50%;
}
.blush-left {
  top: 65%;
  left: 70%;
  background: pink;
  width: 15%;
  height: 10%;
  border-radius: 50%;
}
.beak-top {
  top: 60%;
  left: 40%;
  background: orange;
  width: 20%;
  height: 10%;
  border-radius: 50%;
```

```
  }
  .beak-bottom {
    top: 65%;
    left: 42%;
    background: orange;
    width: 16%;
    height: 10%;
    border-radius: 50%;
  }
  body {
    background:#c6faf1;
  }
  .penguin * {
    position: absolute;
  }
</style>
<div class="penguin">
  <div class="penguin-bottom">
    <div class="right-hand"></div>
    <div class="left-hand"></div>
    <div class="right-feet"></div>
    <div class="left-feet"></div>
  </div>
  <div class="penguin-top">
    <div class="right-cheek"></div>
    <div class="left-cheek"></div>
    <div class="belly"></div>
    <div class="right-eye">
```

```
    <div class="sparkle"></div>
   </div>
   <div class="left-eye">
    <div class="sparkle"></div>
   </div>
   <div class="blush-right"></div>
   <div class="blush-left"></div>
   <div class="beak-top"></div>
   <div class="beak-bottom"></div>
  </div>
</div>
```

Adjunte un valor alternativo a una variable CSS

Al usar su variable como un valor de propiedad CSS, **puede adjuntar un valor alternativo** al que su navegador volverá **si la variable dada no es válida**.

Nota: este respaldo no se usa para aumentar la compatibilidad del navegador y no funcionará en los navegadores IE. Más bien, se usa para que el navegador tenga un color para mostrar si no puede encontrar su variable.

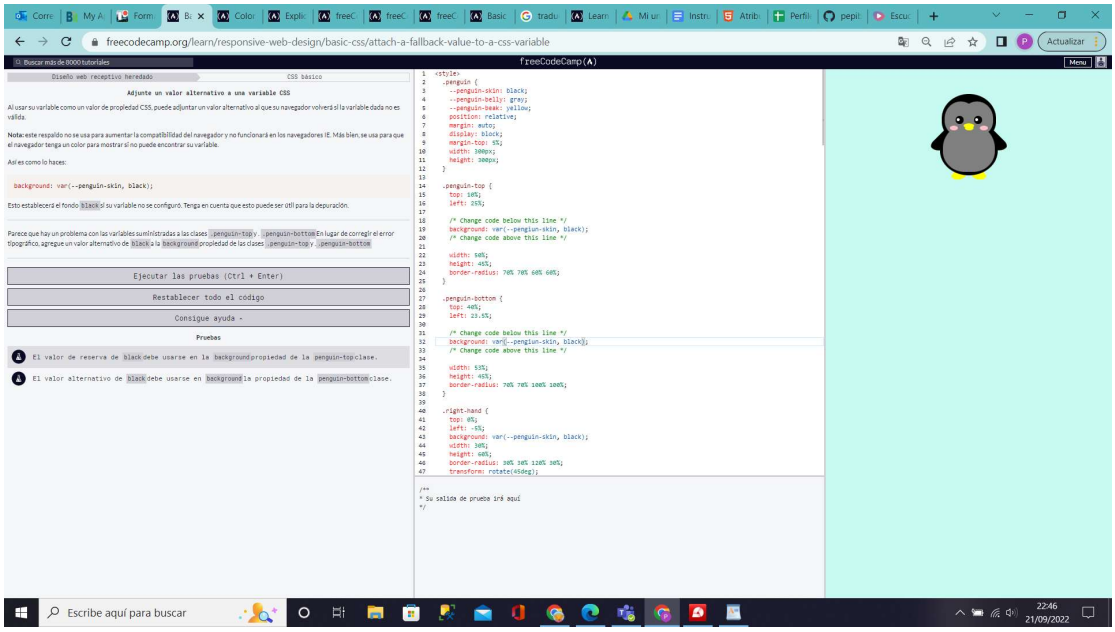
Así es como lo haces:

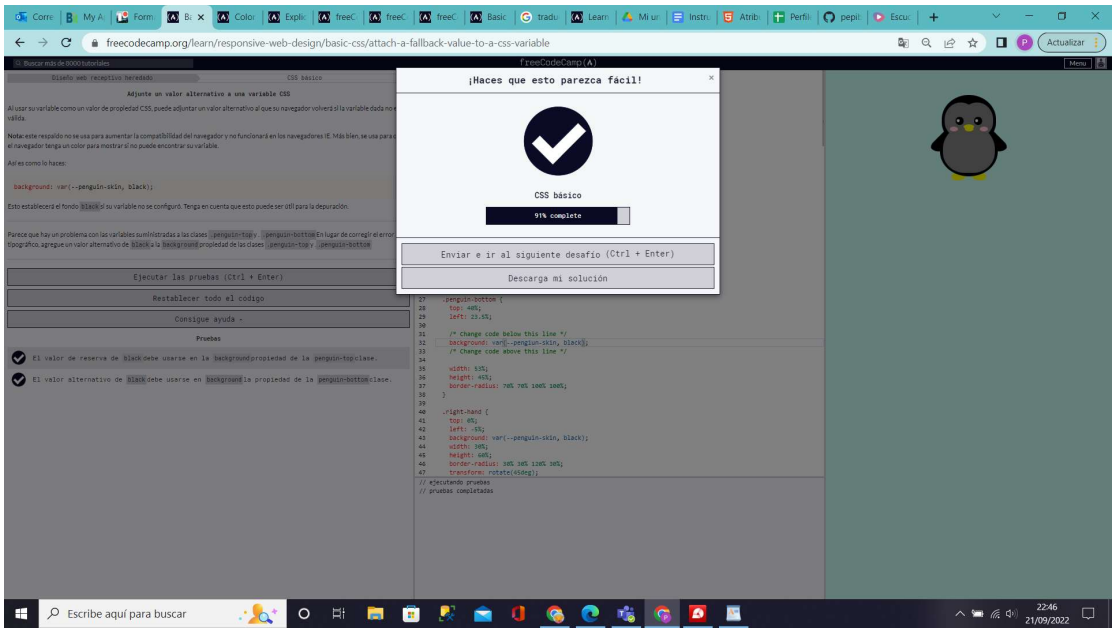
background: var(--penguin-skin, black);

Esto establecerá el **fondo black** si su variable no se configuró. Tenga en cuenta que **esto puede ser útil para la depuración.**

Parece que hay un problema con las variables suministradas a las clases **.penguin-top** y . **.penguin-bottom**

En lugar de corregir el error tipográfico, **agregue un valor alternativo** de **black** a la **background** propiedad de las clases **.penguin-topy** . **.penguin-bottom**






```
<style>
 .penguin {
   --penguin-skin: black;
   --penguin-belly: gray;
   --penguin-beak: yellow;
```

```
  position: relative;
  margin: auto;
  display: block;
  margin-top: 5%;
  width: 300px;
  height: 300px;
}
.penguin-top {
  top: 10%;
  left: 25%;
  /* Change code below this line */
  background: var(--pengiun-skin, black);
  /* Change code above this line */
  width: 50%;
  height: 45%;
  border-radius: 70% 70% 60% 60%;
}
.penguin-bottom {
  top: 40%;
  left: 23.5%;
  /* Change code below this line */
  background: var(--pengiun-skin, black);
  /* Change code above this line */
  width: 53%;
  height: 45%;
  border-radius: 70% 70% 100% 100%;
}
```

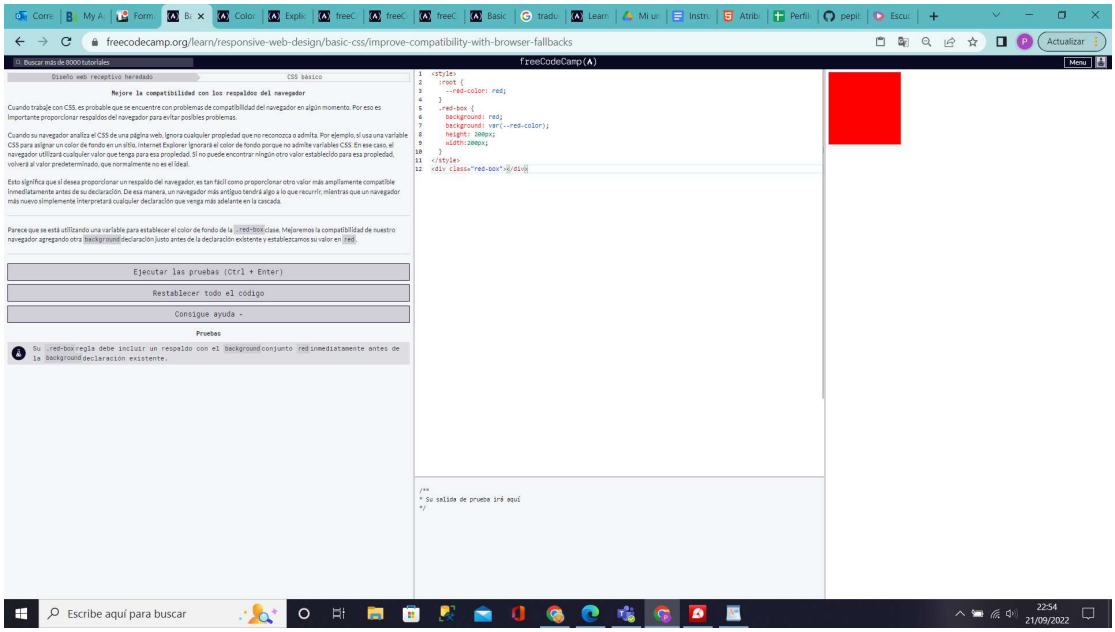
Mejore la compatibilidad con los respaldos del navegador

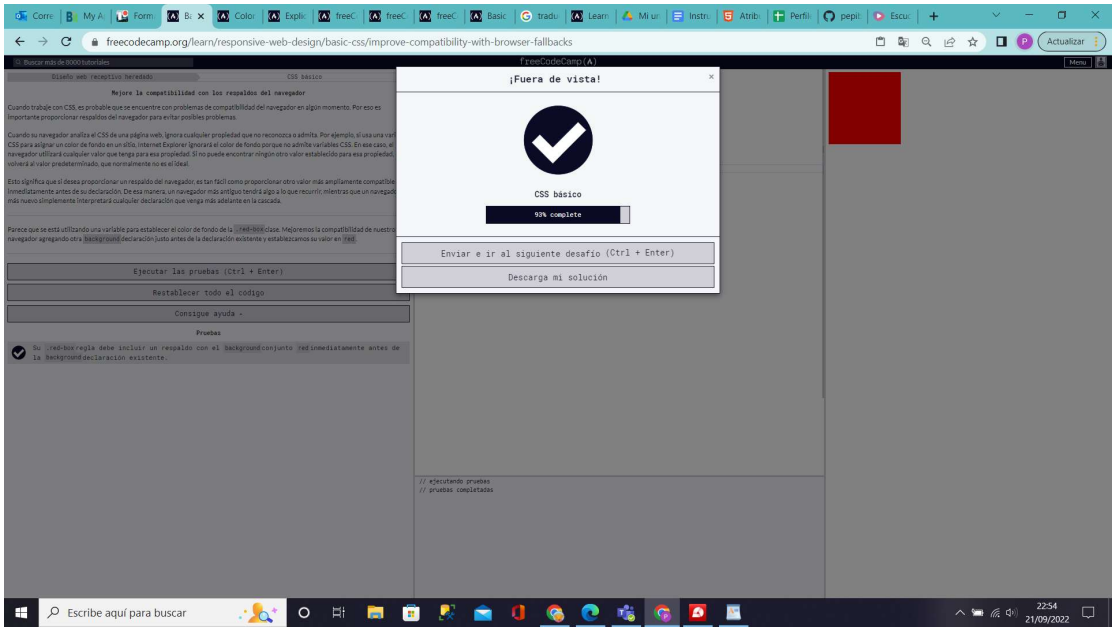
Cuando trabaje con CSS, es probable que se encuentre con problemas de compatibilidad del navegador en algún momento. Por eso es importante proporcionar respaldos del navegador para evitar posibles problemas.

Cuando su navegador analiza el CSS de una página web, ignora cualquier propiedad que no reconozca o admita. Por ejemplo, si usa una variable CSS para asignar un color de fondo en un sitio, Internet Explorer ignorará el color de fondo porque no admite variables CSS. En ese caso, el navegador utilizará cualquier valor que tenga para esa propiedad. Si no puede encontrar ningún otro valor establecido para esa propiedad, volverá al valor predeterminado, que normalmente no es el ideal.

Esto significa que **si desea proporcionar un respaldo del navegador**, es tan fácil como **proporcionar otro valor más** ampliamente compatible **inmediatamente antes de su declaración**. De esa manera, un navegador más antiguo tendrá algo a lo que recurrir, mientras que un navegador más nuevo simplemente interpretará cualquier declaración que venga más adelante en la cascada.

Parece que se está utilizando una variable para establecer el color de fondo de la **.red-box** clase. Mejoremos la compatibilidad de nuestro navegador agregando otra **background** declaración justo antes de la declaración existente y establezcamos su valor en **red**.






```
<style>
 :root {
   --red-color: red;
 }
 .red-box {
   background: red;
   background: var(--red-color);
   height: 200px;
   width:200px;
 }
</style>
<div class="red-box"></div>
```

Heredar variables CSS

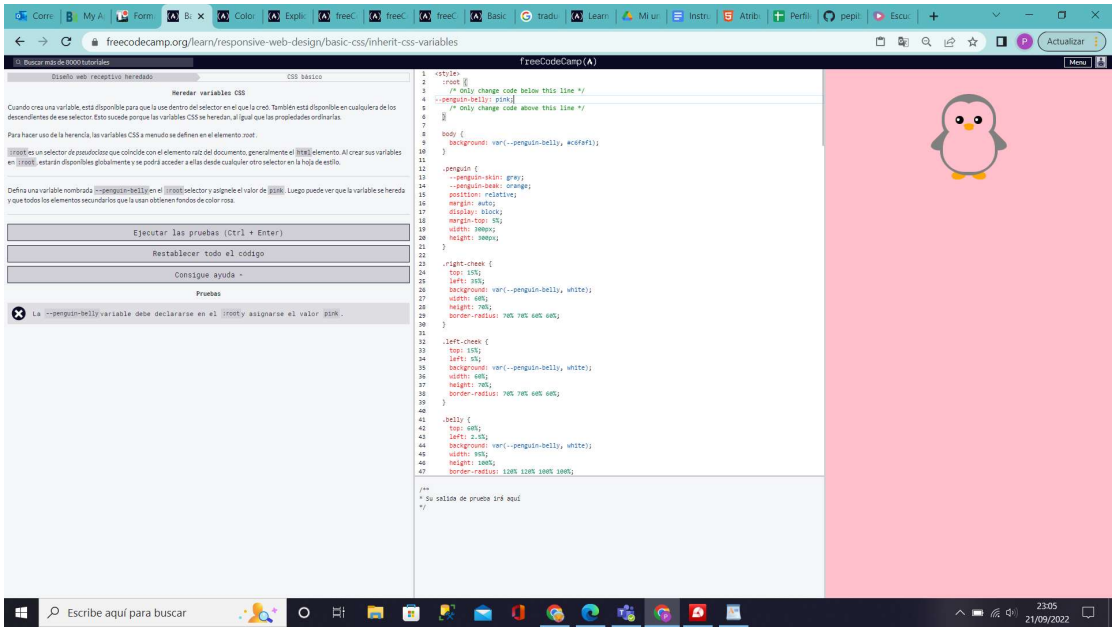
Cuando crea una variable, está disponible para que la use dentro del selector en el que la creó. También está disponible en cualquiera de los descendientes de ese selector. Esto sucede porque **las variables CSS se heredan**, al **igual que las propiedades ordinarias**.

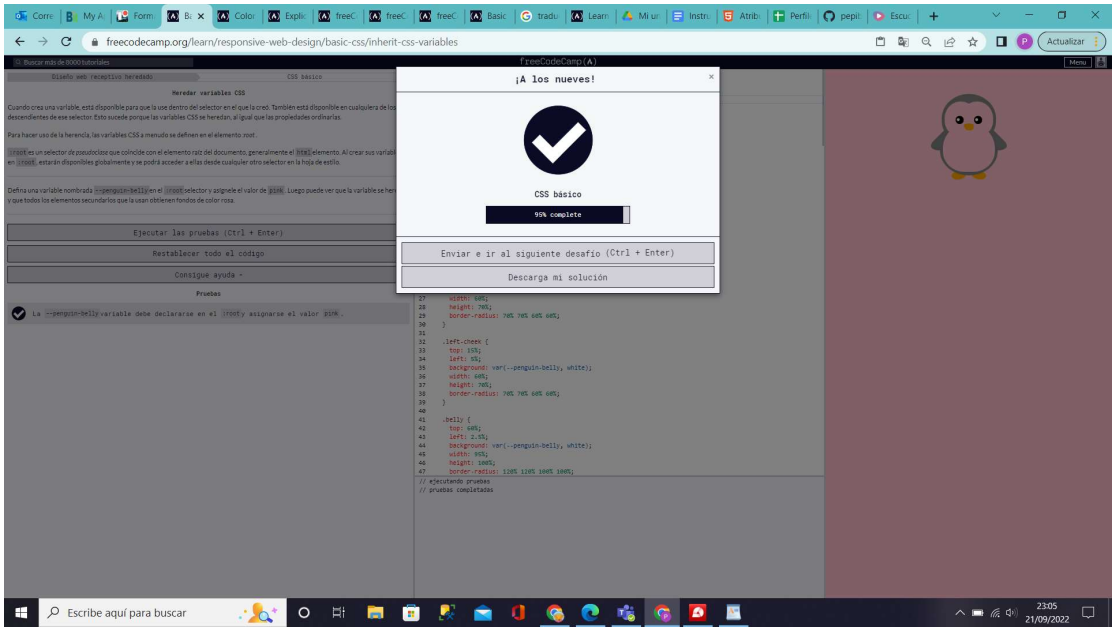
Para hacer uso de la herencia, las variables CSS a menudo se definen en el

elemento :root .

:root es un selector de pseudoclase que coincide con el elemento raíz del documento, generalmente el **html** elemento. Al crear sus variables en **:root**, estarán disponibles globalmente y se podrá acceder a ellas desde cualquier otro selector en la hoja de estilo.

Defina una variable nombrada **--penguin-belly** en el **:root** selector y asígnele el valor de **pink**. Luego puede ver que la variable se hereda y que todos los elementos secundarios que la usan obtienen fondos de color rosa.






```
<style>
 :root {
   /* Only change code below this line */
--penguin-belly: pink;
   /* Only change code above this line */
 }
 body {
   background: var(--penguin-belly, #c6faf1);
 }
 .penguin {
   --penguin-skin: gray;
   --penguin-beak: orange;
   position: relative;
   margin: auto;
   display: block;
   margin-top: 5%;
   width: 300px;
```

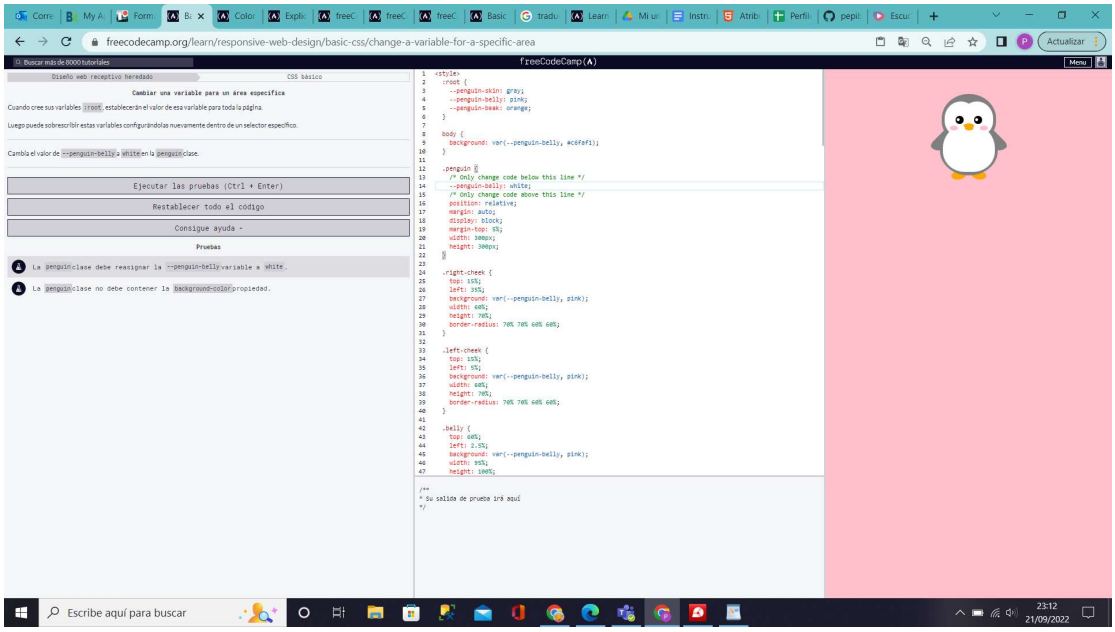
```
  height: 300px;
}
```

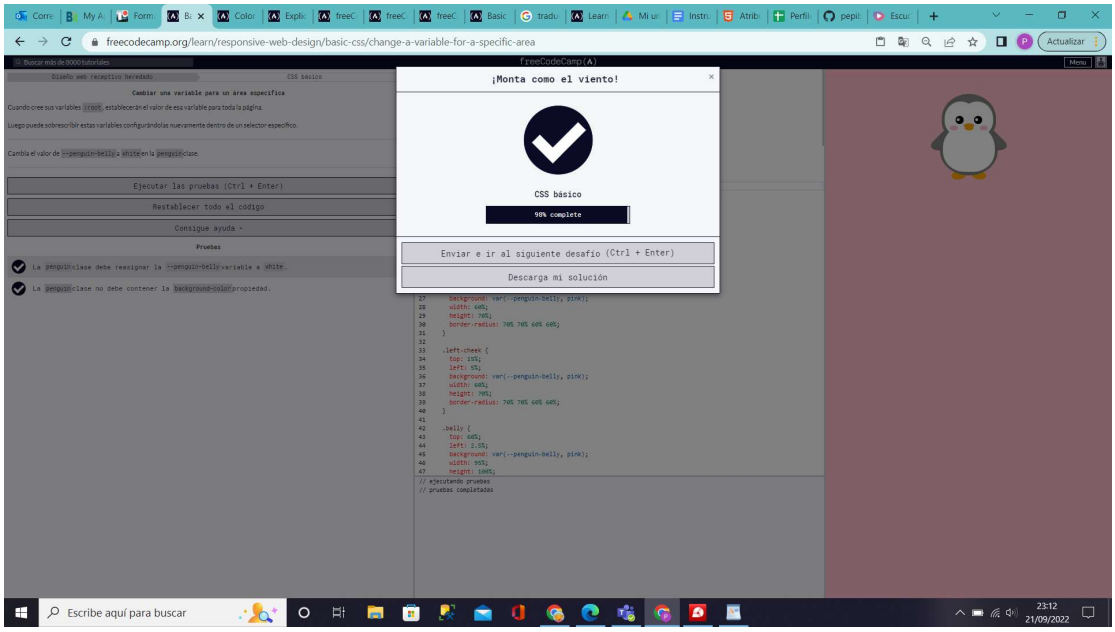
Cambiar una variable para un área específica

Cuando cree sus variables **:root**, establecerán **el valor de esa variable para toda la página.**

Luego **puede sobrescribir estas variables configurándolas** nuevamente **dentro de un selector específico.**

Cambia el valor de **--penguin-belly** a **white** en la penguin clase.






```
<style>
 :root {
   --penguin-skin: gray;
   --penguin-belly: pink;
   --penguin-beak: orange;
 }
 body {
   background: var(--penguin-belly, #c6faf1);
 }
 .penguin {
   /* Only change code below this line */
   --penguin-belly: white;
   /* Only change code above this line */
   position: relative;
   margin: auto;
   display: block;
   margin-top: 5%;
```

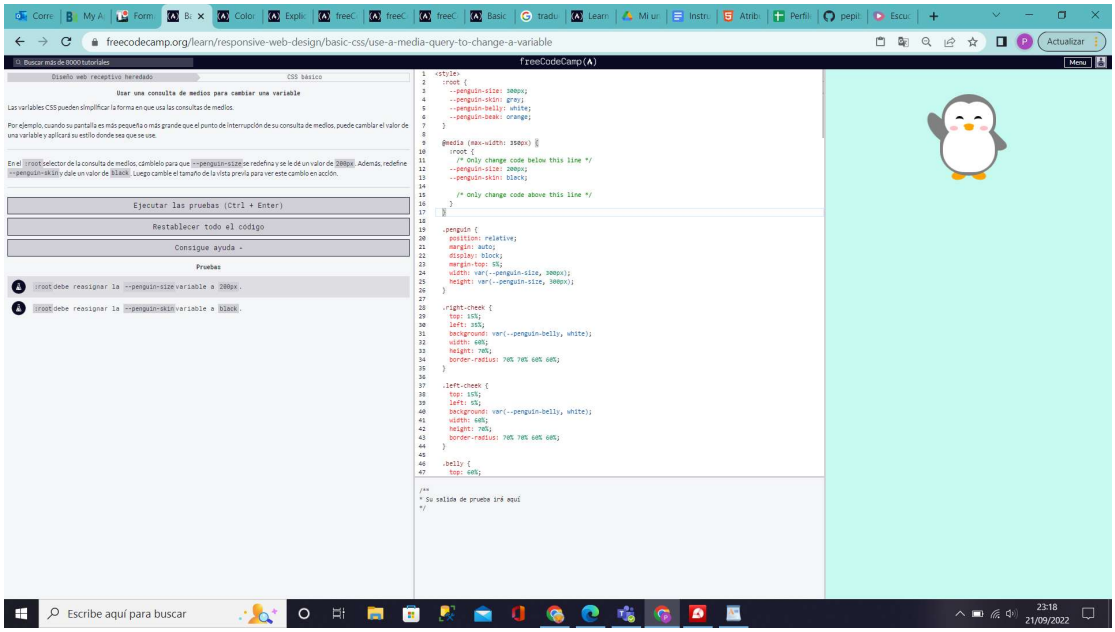
```
  width: 300px;

  height: 300px;

}
```

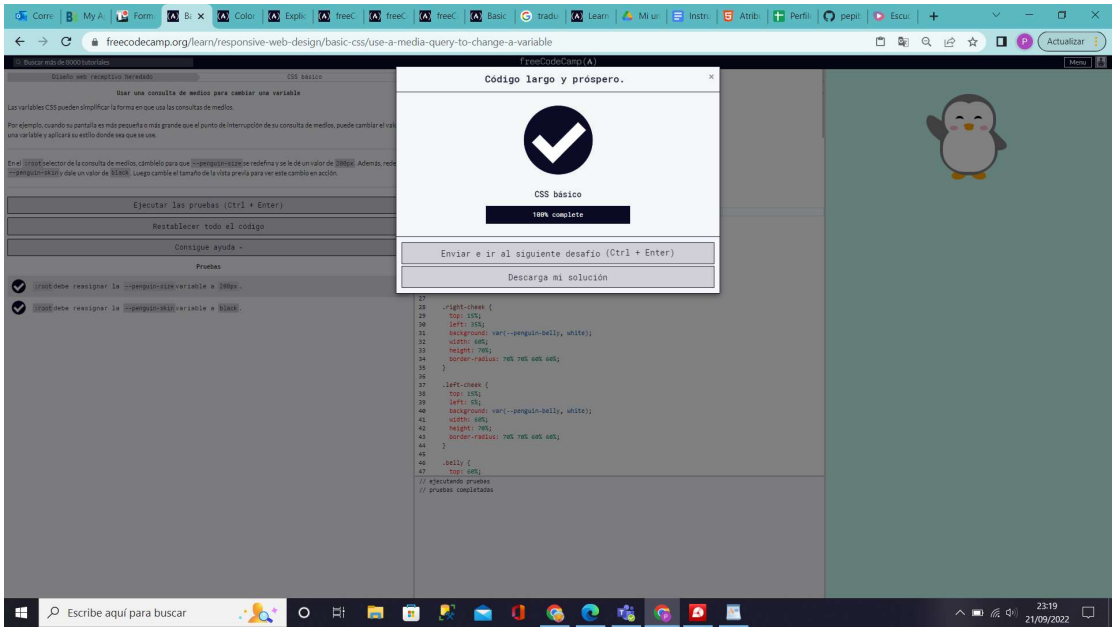
Usar una consulta de medios para cambiar una variable

Las variables CSS pueden simplificar la forma en que usa las consultas de medios.

Por ejemplo, **cuando su pantalla es más pequeña o más grande** que el punto de interrupción de su consulta de medios, puede cambiar el valor de una variable y aplicará su estilo donde sea que se use.

En el **:root** selector de la consulta de medios, cámbielo para que **--penguin-size** se redefina y se le dé un valor de **200px**. Además, redefine **--penguin-skin** y dale un valor de **black**. Luego cambie el tamaño de la vista previa para ver este cambio en acción.






```
<style>
 :root {
   --penguin-size: 300px;
   --penguin-skin: gray;
   --penguin-belly: white;
   --penguin-beak: orange;
 }
 @media (max-width: 350px) {
   :root {
     /* Only change code below this line */
   --penguin-size: 200px;
   --penguin-skin: black;
     /* Only change code above this line */
   }
 }
 .penguin {
   position: relative;
```

```
  margin: auto;
  display: block;
  margin-top: 5%;
  width: var(--penguin-size, 300px);
  height: var(--penguin-size, 300px);
}
```