

Devolución del trabajo práctico N°3

Alumnos: Bontorno Jeremías, Ignacio Rodriguez.

Calificación: Aprobado.

Revisiones:

Punto 1:

A continuación marco una corrección en su resolución:

```
div {  
    background-image: linear-gradient(45deg, #ff9900, #ff5e62);  
    height: 100px;  
    line-height: 100px; 🙌 /* Atención */  
}
```

Aunque presenta el resultado esperado, el uso de line-height en este caso no es correcto. Si necesito generar espacio alrededor de un elemento, no modifico propiedades de su contenido, utilizo propiedades como margin o padding para expandir en función de las necesidades.

Punto 4:

La idea de este punto sería tener una barra lateral con links funcionales, por lo que sería bueno cambiar todos los “p” por “a”. En la línea 14 hacen referencia a una propiedad “text-decoration: none” que sólo es aplicable a links “a”, pero está aplicada a un “li”.

Sugiero los siguientes cambios:

```
div { 🙌 /* 1 */  
    width: 30vw;  
    height: 100vh;  
    background-color: aquamarine;  
}  
div ul {  
    padding-inline-start: 0px;  
}  
div li { 🙌 /* 2 */  
    list-style: none;  
    padding: 8px;  
    border: 1px solid grey;  
    /* Lo de abajo puede ir para li o para a en este caso */  
    font-family: "Playfair Display", serif;
```

```
font-weight: 700;  
font-size: 22px;  
}
```

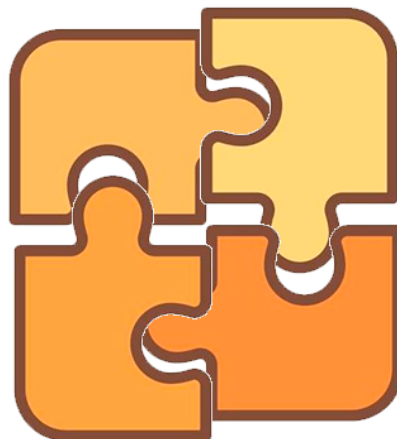
1. Delego la responsabilidad del ancho y alto al contenedor padre de todos esos links, haciendo que los hijos se manejen respecto a lo que pase con el padre.
2. Vean la forma en la que selecciono. En este caso particular, yo sé que en mi documento HTML hay un único div que tiene li en su interior, además sé que los li van siempre dentro de una lista, por lo que obviamos la parte de seleccionar mediante “div ul li”. En esto último decimos, “para el li que está dentro del ul que está dentro del div”, lo cual por lo que dijimos, es redundante.

Punto 5:

En el punto A, la consigna es ambigua y se da a interpretación. La solución está bien implementada. Lo que esperaba del ejercicio era que colocaran `position absolute` en la caja 2. Esto provocaría que dicha caja se salga del flujo de contenido, permitiendo que la caja 3 tome su lugar. Nuevamente, la solución es correcta debido a la ambigüedad del ejercicio.

Punto 6:

Este punto sirve para demostrar que, si bien es importante pensar en cualquier tamaño de pantalla (y por lo tanto usar porcentajes, `vh` o `vw`), a veces hay que verlo desde un punto de vista relativo. El problema de usar `vh` y `vw` acá es el siguiente. En mi pantalla estoy viendo esto:



Cual es el problema que causó esto? Como los tamaños de las pantallas son diferentes, `vh` y `vw` son diferentes también. La solución está en pensar las cosas relativamente. Vamos a colocar la pieza superior izquierda, nuestra primera pieza. Como nosotros sabemos que todas las piezas miden lo mismo por más que abra la página en una pantalla enorme o una muy chica, puedo trabajar relativamente a la pieza ya colocada. Si mis piezas miden 100px de ancho y alto, yo sé que la pieza superior derecha debe ir a 80px a la derecha de mi primera pieza (para que la pieza en png encaje con la primera). La pieza inferior izquierda debe ir 80px hacia abajo de la primera pieza y así.

La forma de hacer esto funcionar es mediante la interacción padre-hijo con position. Si a los hijos les pongo position absolute, al padre le pongo position relative para sostenerlos y permitirme jugar con las piezas a mi antojo. Con esto estoy diciendo que es necesario crear un contenedor con todas las imágenes dentro.