HTML

Es un lenguaje de estructurado de etiquetas que permite tomar texto e imágenes y más es decir contenido y permite presentarlos de una forma jerarquizada.

El inventor es Tim Berners-Lee.

Las etiquetas delimitan el tipo de contenido y siempre hay una de apertura y otra de cierre.

<title>Mi primer página</title>

En el <head></head> va el contenido y recursos que tienen que ser cargados y procesados antes de cargar el <body></body> de la página.

<!DOCTYPE html> <!--Tipo de documento html5-->

<html lang="en"> <!--Es la etiqueta padre donde va todo el contenido-->

    <head> <!--Descendiente directo de html-->

        <meta charset="UTF-8">

        <meta name="viewport" content="width=device-width, initial-scale=1.0">

        <title>Mi primer página</title>

    </head>

    <body>

    </body>

</html>

Dentro de body va el contenido que se ve en el navegador.

**HEADINGS**

Los headings sirven para expresar que el contenido entre las etiquetas es un titular o subtitular

<body>

        <h1>Mayor importancia</h1>

        <h2>Segunda importancia</h2>

        <h3>Third</h3>

        <h4>Fourth</h4>

</body>

**PÁRRAFO**

<p>Aquí va lo del párrafo</p>

**LISTAS DESORDENADA Y ORDENADA**

<ul>

            <li>Hoy</li>

            <li>Mañana</li>

            <li>Silla</li>

</ul>

        <h5>Meses del año</h5>

<ol>

            <li>Enero</li>

            <li>Febrero</li>

            <ol>Marzo</ol>

</ol>

**HIPERENLACES**

Es una estructura creada para conectar diferentes partes de una página web con información y recursos, externos o internos. Se utiliza la etiqueta a con su atributo href

<a href="https://twitter.com/jc2bcs">Julio César</a>

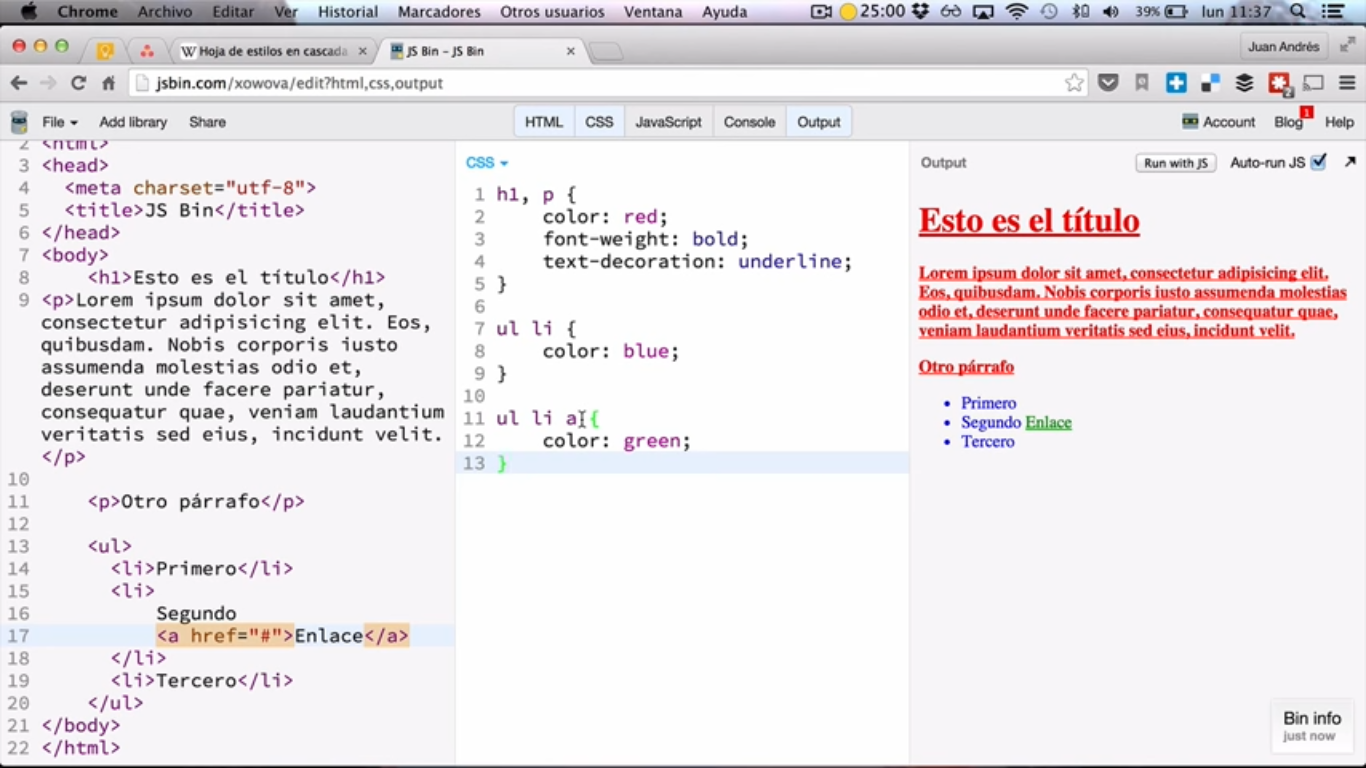
Para que se abra en otra pestaña se agrega el atributo target=”\_blank”:

<p>Puedes consultar mi twitter en: <a href="https://twitter.com/jc2bcs" target="\_blank">Julio César</a></p>

**CSS**

**¿Qué es CSS (cascading style sheet)?**

La idea que se encuentra del desarrollo de CSS es separa la estructura de un documento de su presentación.



Para darle formato a una etiqueta la sintaxis es la siguiente:

Nombredelaetiqueta, nombredeotraetiqueta {

Propiedades de diseño

color: red;

}

Cuando queremos ingresar a un elemento que esta dentro de otro etiqueta se deben separar por espacios y se lee de derecha a izquierda

**Pseudoclases**

Sirven para definir un estado, una relación o una posición en específico o especial de un elemento, para llegar a una posición especifica de un elemento:

* a:hover {

color:pink;

font-size:50px;

}

Estamos diciendo que cualquier enlace cuando se posicione el cursor en él se agrandará por 50px

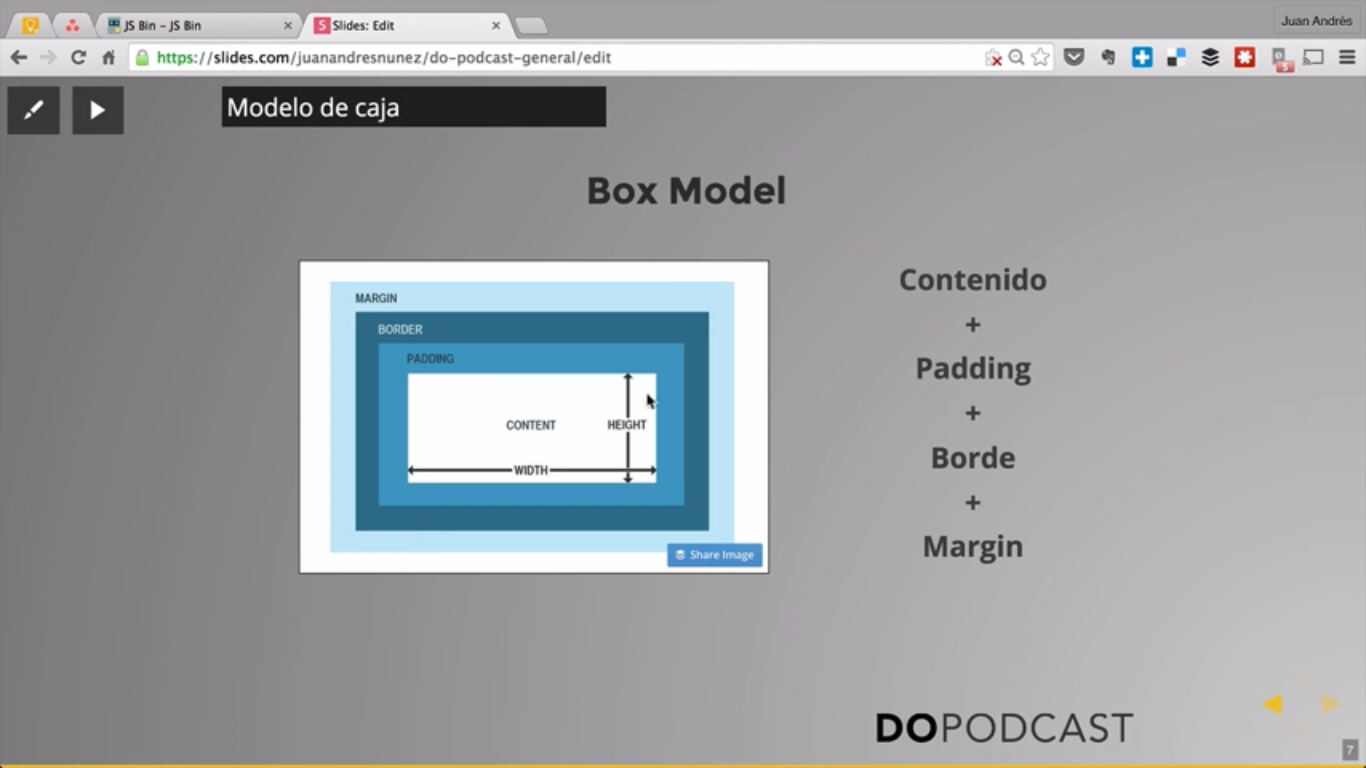
* p:first-of-type{

color:green;

}

Esto solo llegaría al primer párrafo

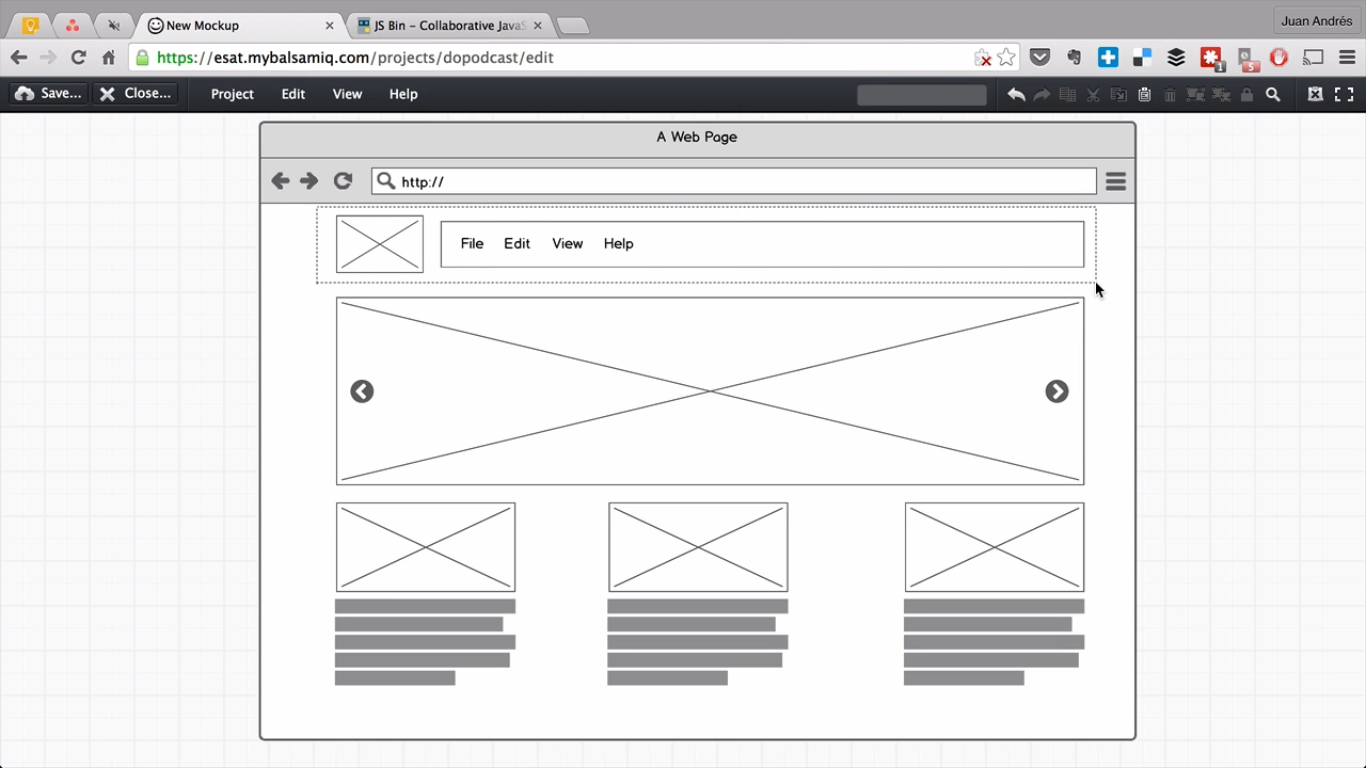
**BOX MODEL**



El navegador asigna unos por default pero nosotros podemos alterarlos a través de nuestros documentos de estilos css.

**ELEMENTOS DIV**

Viene de divisor y sirven para dividir la página en contenido relacionado:



Siguen siendo los bloques de estructura actualmente.

Tienen display block.

Puede haber un div que contiene todos los demás divs

**FORMULARIOS**

Se define con form

<form action=""></form>

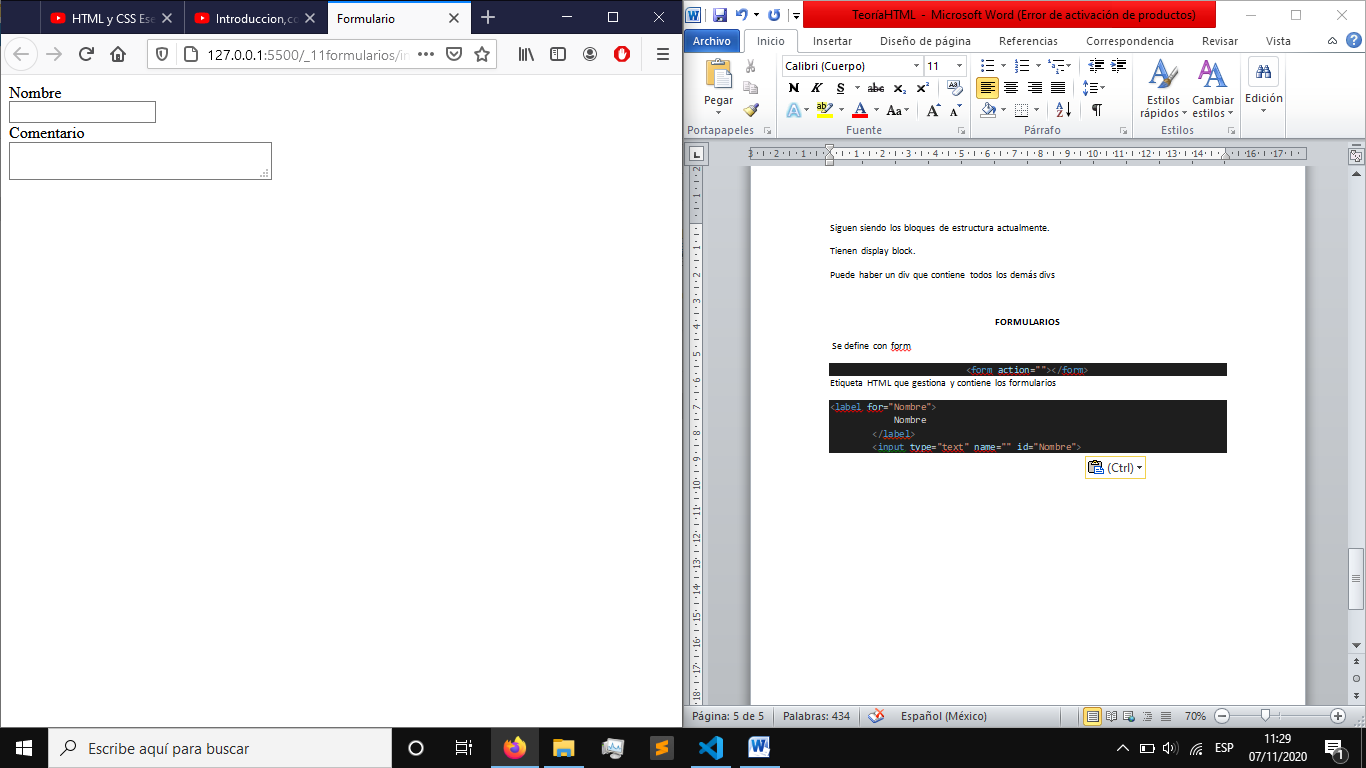
Etiqueta HTML que gestiona y contiene los formularios

<label for="Nombre">

            Nombre

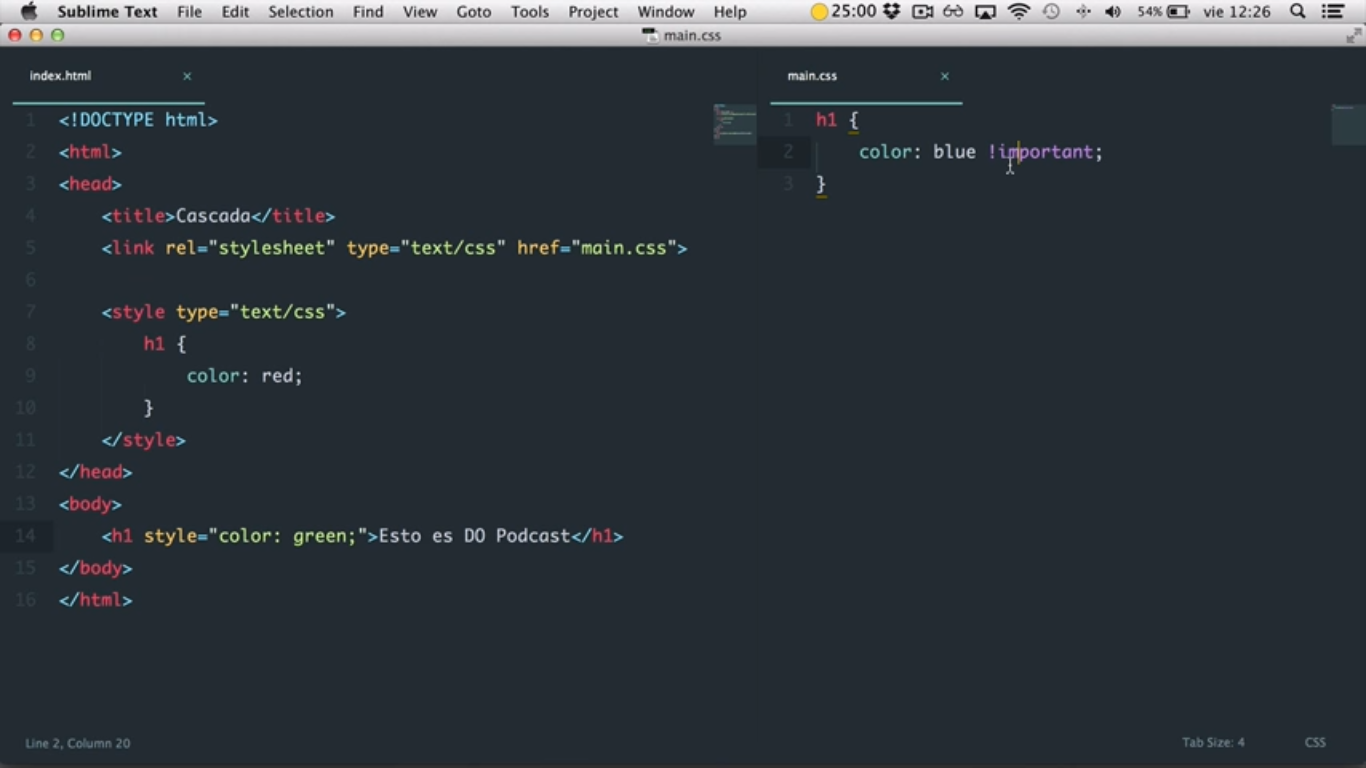
        </label>

        <input type="text" name="" id="Nombre">



**CASCADA**

Determina el orden de aplicación y de precedencia del css, es decir hay una jerarquía



El css se puede ejecutar desde un archivo externo, desde el documento html dentro del head dentro de las etiquetas style o se puede ejecutar también dentro de las propias etiquetas de los elementos, esto lleva un orden de ejecución de arriba hacia abajo, en caso de que quieramos que pese más la del documento css escribiremos **!important**

**ELEMENTOS FLOTANTES**

Los floats eliminan un elemento del flujo normal de un documento. El resto del contenido al mismo nivel dentro de ese elemento flotante fluirá alrededor de ese elemento flotante.

Los elementos flotantes se van apilando unos a otros hasta llegar a saturarse.

Para que el contendor padre <div></div> pueda cubrir esos elementos floatantes se puede usar el overflow=hidden; y listo :D

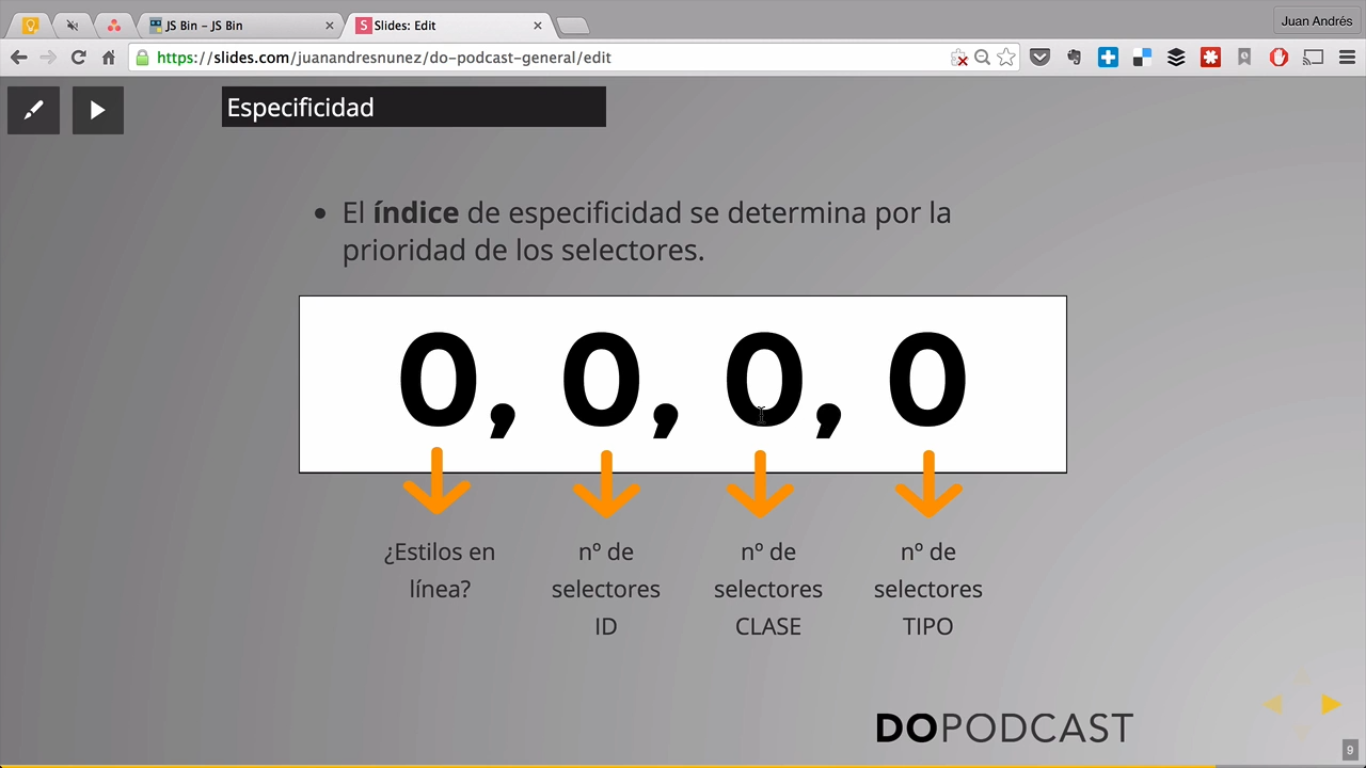
**HERENCIA**

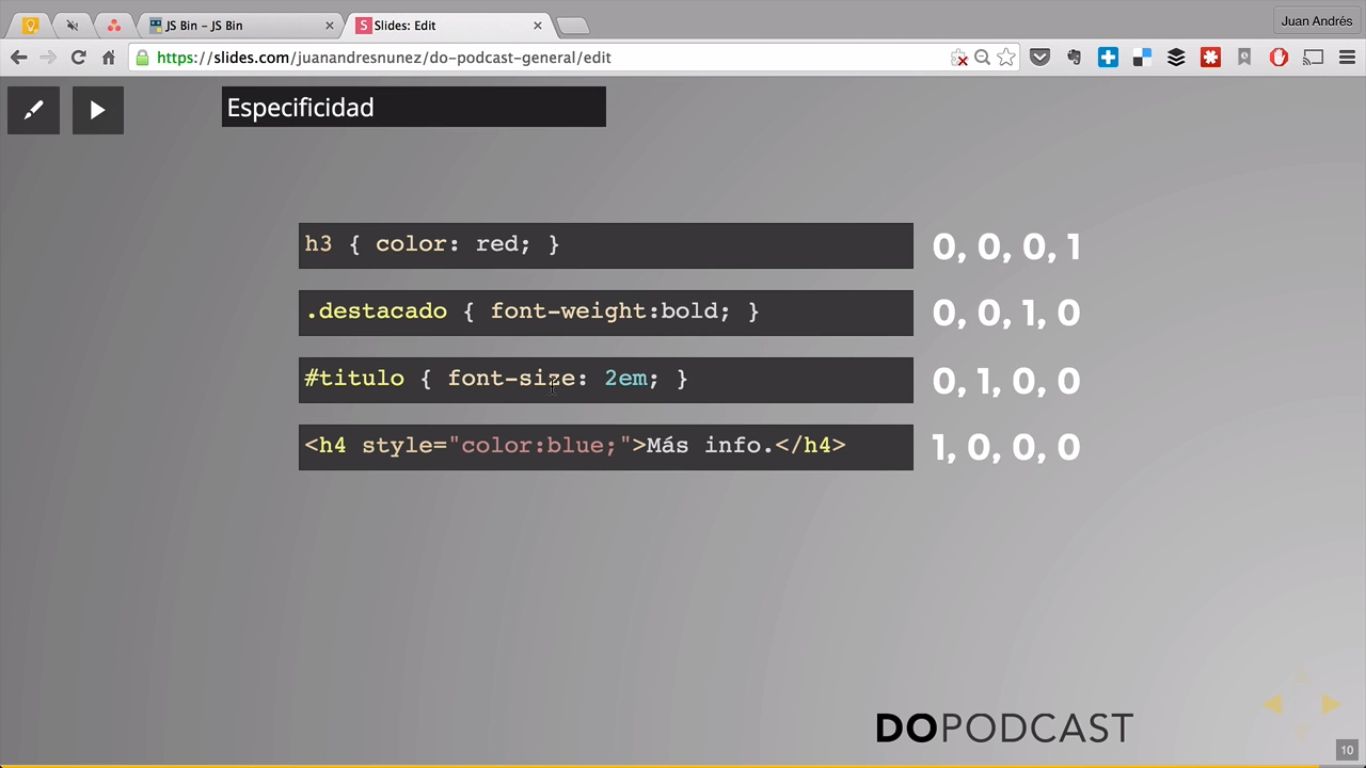
Una propiedad o fenómeno que ocurre por el cual los elementos hijos heredan las propiedades de sus elemento padre.

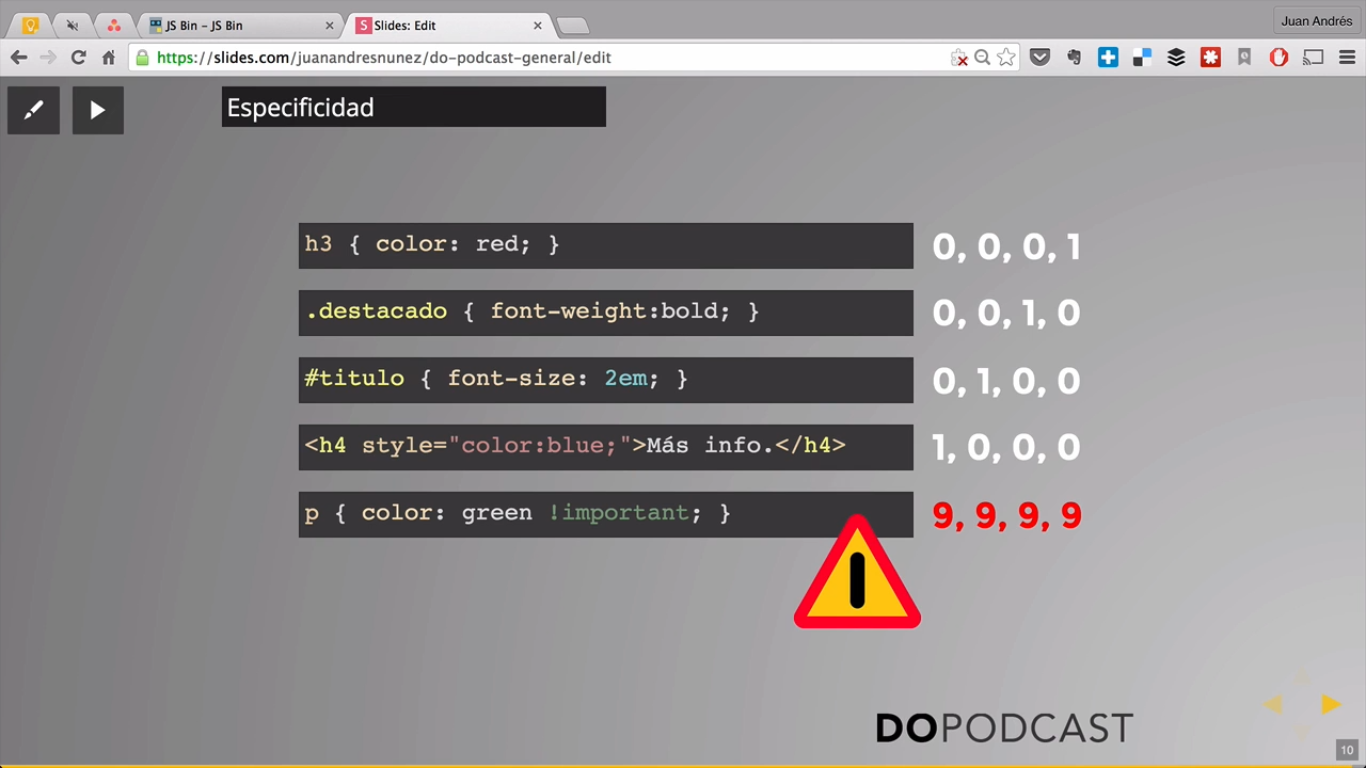
No todas las propiedades son heredables por defecto y algunas que no se pueden heredar.

**ESPECIFICIDAD**

Las IDs tienen mas peso que las clases en los selectores.







OVERFLOW

Desbordamiento:

Qué ocurre cuando un elemento sobrepasa en tamaño a su contenedor:

Con él se puede hacer **scroll**

**POSICIONAMIENTO**

En el css esta la propiedad position:

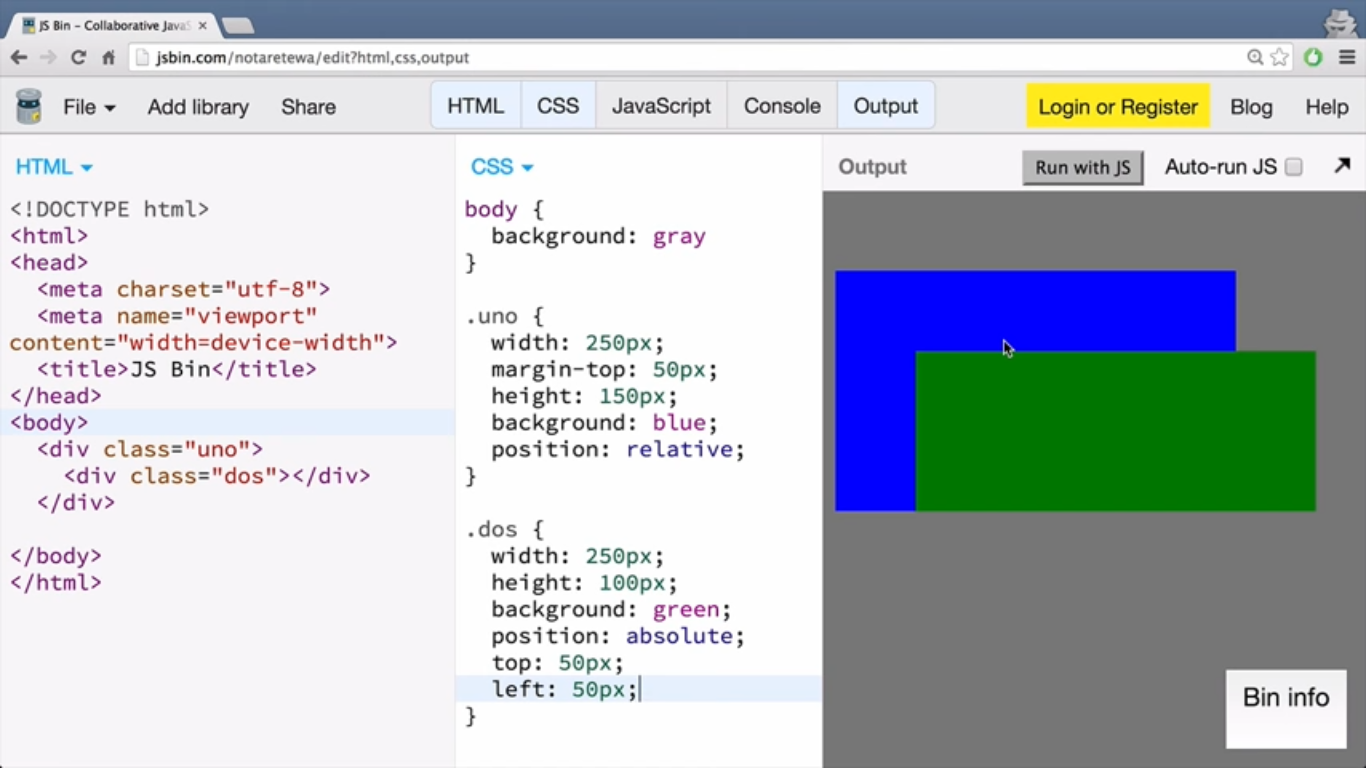
Position: static; #Es la que tiene por default

Position: relative;

Tenemos acceso a las porpiedades:

Top: 50px; lo podemos desplazar hacia abajo respecto a sí mismo

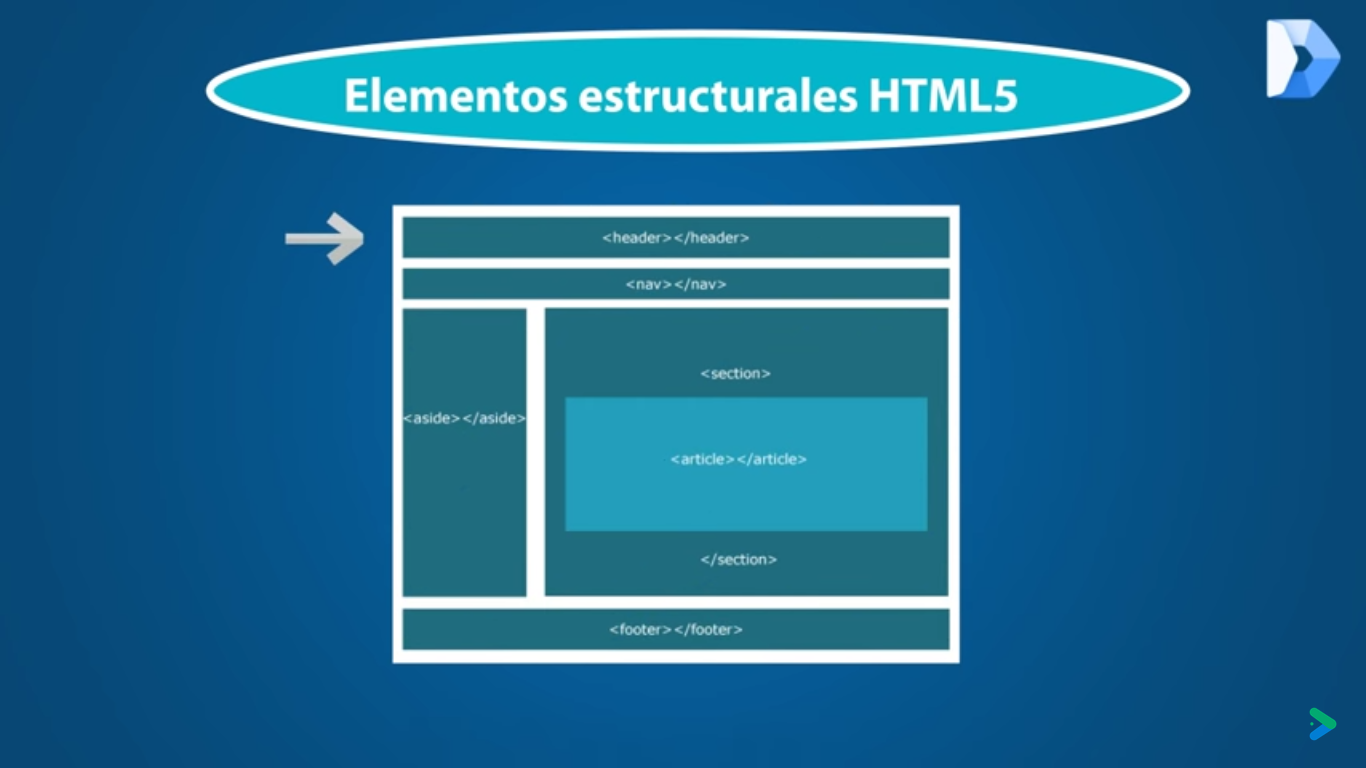
Position **absolute** es lo mismo que relative, respecto a otro elemento generalmente a su parent el body u otro parent posicionado



**Fixed fija el elemento a la pantalla.**

**INDEX Z**

Sólo se puede utilizar con elementos posicionados.



**FORMULARIOS:**

Se debe usar el elemento <form></form>

Tiene dos atributos importantes: el atributo **action** y el atributo **method**

Especificamos la url o la dirección en dónde se enviará la informaciónuna vez llenado el formulario.

En el atributo **method** definimos el método http que los navegadores usan para enviar la información del formulario, estos pueden ser get o post