M4.252 - HTML y CSS aula 1

PEC2

Estudiante: Ivan Lenin Schartun Brito

Primera parte

Pregunta 1. Sobre la utilización de tablas responded a las siguientes preguntas:

* 1. **¿Qué atributos utilizaremos para fusionar celdas verticalmente y horizontalmente? Poned un ejemplo de código que muestre su uso.**

Respuesta. –

Para este caso se debería usar el atributo rowspan y colspan, el primero especifica el número de filas que debe abarcar una celda mientras que le segundo el número de columnas. El valor de estos atributos indica el número de celdas que desea fusionar respectivamente.

|  |  |  |
| --- | --- | --- |
| **HTML** | **CSS** | **Live** |
| <table>  <tr>  <th>Enero</th>  <th>Febrero</th>  <th>Marzo</th>  <th>Abril</th>  </tr>  <tr>  <td>123</td>  <td>134</td>  <td colspan="2">156</td>  </tr>  <tr>  <td>213</td>  <td rowspan="2">256</td>  <td>273</td>  <td>274</td>  </tr>  <tr>  <td>321</td>  <td>372</td>  <td>373</td>  </tr>  </table> | table \* {  border: 1px solid black;  } |  |

* 1. **Explicad el uso de cada uno de los elementos siguientes thead , tbody y tfoot .**

Respuesta. –

* *thead*: se utiliza para agrupar el contenido del encabezado en una tabla HTML.

|  |  |  |
| --- | --- | --- |
| **HTML** | **CSS** | **Live** |
| <table>  <thead>  <tr>  <th>Enero</th>  <th>Febrero</th>  <th>Marzo</th>  <th>Abril</th>  </tr>  </thead>  <tr>  <td>123</td>  <td>134</td>  <td colspan="2">156</td>  </tr>  <tr>  <td>213</td>  <td rowspan="2">256</td>  <td>273</td>  <td>274</td>  </tr>  <tr>  <td>321</td>  <td>372</td>  <td>373</td>  </tr>  </table> | table \* {  border: 1px solid black;  }  thead {  background-color: pink;  } |  |

* *tbody:* se usa para agrupar el contenido del cuerpo en una tabla dentro del documento.

|  |  |  |
| --- | --- | --- |
| **HTML** | **CSS** | **Live** |
| <table>  <thead>  <tr>  <th>Enero</th>  <th>Febrero</th>  <th>Marzo</th>  <th>Abril</th>  </tr>  </thead>  <tbody>  <tr>  <td>123</td>  <td>134</td>  <td colspan="2">156</td>  </tr>  <tr>  <td>213</td>  <td rowspan="2">256</td>  <td>273</td>  <td>274</td>  </tr>  <tr>  <td>321</td>  <td>372</td>  <td>373</td>  </tr>  </tbody>  </table> | table \* {  border: 1px solid black;  }  tbody {  background-color: pink;  } |  |

* *tfoot:* se usa para agrupar el contenido de pie de página dentro de una tabla de un documento.

|  |  |  |
| --- | --- | --- |
| **HTML** | **CSS** | **Live** |
| <table>  <thead>  <tr>  <th>Enero</th>  <th>Febrero</th>  <th>Marzo</th>  <th>Abril</th>  </tr>  </thead>  <tr>  <td>123</td>  <td>134</td>  <td colspan="2">156</td>  </tr>  <tr>  <td>213</td>  <td rowspan="2">256</td>  <td>273</td>  <td>274</td>  </tr>  <tr>  <td>321</td>  <td>372</td>  <td>373</td>  </tr>  <tfoot>  <tr>  <td colspan="4">  Pie de Pagina  </td>  </tr>  </tfoot>  </table> | table \* {  border: 1px solid black;  }  tfoot {  background-color: pink;  } |  |

* 1. **¿Para qué resulta útil el uso del atributo scope?** **¿Qué valores puede tener este atributo? ¿En qué elemento de tabla podemos utilizarlo?**

Respuesta. –

Este atributo permite identificar las celdas con las que se relaciona el encabezado (definido en el elemento ‎‎ ‎‎<th>‎)‎, es decir especifica si un <th> es un encabezado para una columna, fila o grupo de columnas o filas.‎

Los valores que se le pueden asignar a este atributo son:

* row‎: Especifica que la celda es un encabezado para una columna‎
* col‎: Especifica que la celda es un encabezado para una fila‎
* rowgroup‎: Especifica que la celda es un encabezado para un grupo de columnas‎
* colgroup‎: Especifica que la celda es un encabezado para un grupo de filas‎

Se debe utilizar en el elemento <th>

|  |  |  |
| --- | --- | --- |
| **HTML** | **CSS** | **Live** |
| <table>  <tr>  <th></th>  <th scope="col">Cantidad</th>  <th scope="col">Proyección</th>  </tr>  <tr>  <th scope="row">Enero</th>  <td>50</td>  <td>$100</td>  </tr>  <tr>  <th scope="row">Febrero</th>  <td>35</td>  <td>$80</td>  </tr>  </table> | table \* {  border: 1px solid black;  } |  |

**Pregunta 2**. Sobre la utilización de formularios responded a las siguientes preguntas:

**2.1 Explicad qué características tienen los siguientes tipos de input y poned ejemplos que indiquen en qué casos resulta útil su utilización:**

* + **radio**
  + **checkbox**
  + **password**
  + **number**

Respuesta. –

* *radio:* Es uno de los valores que se pueden asignar al atributo *type*, este tipo de valor permite que dentro de un grupo de opciones no más de una opción pueda ser seleccionada al mismo tiempo. Puede servir por ejemplo en un formulario donde se indague el tipo sanguíneo de una persona.

|  |  |  |
| --- | --- | --- |
| **HTML** | **CSS** | **Live** |
| <input type="radio" id="A\_positivo" name="tipoSangre" value="A\_positivo">  <label for="A\_positivo">(A+)</label><br>  <input type="radio" id="A\_negativo" name="tipoSangre" value="A\_negativo">  <label for="A\_negativo">(A-)</label><br>  <input type="radio" id="B\_positivo" name="tipoSangre" value="B\_positivo">  <label for="B\_positivo">(B+)</label><br> | NA |  |

* *checkbox:* Es uno de los valores que se pueden asignar al atributo *type*, este tipo de valor permite que dentro de un grupo de opciones el usuario pueda seleccionar no más de una opción al mismo tiempo tildando cada ítem deseado. Puede servir por ejemplo en un formulario donde se indague a que alimentos es alérgico una persona.

|  |  |  |
| --- | --- | --- |
| **HTML** | **CSS** | **Live** |
| <input type="checkbox" id="alergia1" name="alergia1" value="Nueces">  <label for="alergia1">Nueces</label><br>  <input type="checkbox" id="alergia2" name="alergia2" value="Maiz">  <label for="alergia2">Maiz</label><br>  <input type="checkbox" id="alergia3" name="alergia3" value="Chocolate">  <label for="alergia3">Chocolate - Lo lamento :-( -</label><br>label><br> | NA |  |

* *password:* Es uno de los valores que se pueden asignar al atributo *type*, este tipo de valor indica que la información introducida en el campo de entrada es una contraseña por lo que será de cierta forma segura. Los datos ingresados se reemplazarán por asterisco ("\*") o un punto ("•"), aunque esto depende del OS y UA. El caso más común seria para realizar un *login* en una cuenta segura.

|  |  |  |
| --- | --- | --- |
| **HTML** | **CSS** | **Live** |
| <label for="pwd">Ingrese contraseña:</label>  <input type="password" id="pwd" name="pwd"> | NA |  |

* *number:* Es uno de los valores que se pueden asignar al atributo *type*, este tipo de valor indica que la información introducida en el campo de entrada es un número. El navegador realizar una validación en el *front* que rechazaría la entrada si esta no fuese numérica. Su utilidad podría ser el ingreso de la edad de un usuario en un formulario de registro.

|  |  |  |
| --- | --- | --- |
| **HTML** | **CSS** | **Live** |
| <label for="quantity">Ingrese su edad:</label>  <input type="number" id="quantity" name="quantity"> | NA |  |

* 1. **Explicad para qué sirve y cómo debe utilizarse la etiqueta label dentro de un formulario.**

Respuesta. –

Es un elemento de HMTL que se emplean para dar una identificación (título) a los campos definidos en un formulario. En otras palabras, el elemento asocia algún texto a un campo de control en el *form*. Las mismas se deben usar para indicar el propósito del campo al cual está asociada.

Existen dos formas generales de etiquetado, la forma *explícita* y la *implícita*

En la forma *explícita* se debe establecer una asociación entre el tag y el elemento <input>, mediante los atributos *id*, y el *for* contenidos en cada tag respectivamente*.* El mismo valor para cada uno.

<label for="meal">Que desea comer? <label>

<br>

<input type="text" name="food" id="meal">

‎ En la forma *implícita*, el elemento se utiliza como contenedor tanto para el control de formulario como para el texto de la etiqueta, de modo que los dos están asociados implícitamente.

<label>

Nombre:

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

</label>

La asociación entre <label> y <input> permite que la primera sirva como un área en la que se puede hacer clic para seleccionar o activar algún control de forma. También garantiza que las tecnologías de asistencia puedan hacer referencia a la etiqueta correcta.

**Pregunta 3**. Sobre el modelo de caja (CSS box model) responded a las siguientes preguntas:

**3.1 ¿Qué elementos componen la estructura de una caja?**

Respuesta. –

El modelo de caja está ideado como una figura rectangular invisible que es apilada dentro de un documento HTML. Cada elemento que conforma el documento es participe de este modelo.

Dichas cajas posees a su vez propiedades que rigen su disposición en el flujo del formato visual. Las propiedades que manejar su comportamiento son: padding, margin, with y heigth, que según el tipo de elemento puedo o no variar (block, inline).

* *Padding*: Establece la cantidad de espacio entre el borde y el contenido del elemento
* *Marging*: Establece los márgenes de un elemento especificando con respectos a sus vecinos.
* *Width*: Establece el valor de anchura que posee el elemento
* *Height*: Establece el valor de altura que posee el elemento

Es importante destacar que las propiedades anteriores definirán el tamaño que un elemento tiene por defecto cuando se considera el modelo estándar de CSS.

**3.2 ¿Qué diferencia hay entre display:** block, display: inline y display: inline-block**?**

Respuesta. –

* Block: Este valor correspondiente a la propiedad display nos habla de la forma en cómo se dispondrá la visualización del elemento en el navegador. Para este caso en particular se habla de que el elemento se considerara como un bloque que comienza en una nueva línea, y toma todo el ancho‎ de la misma.
* Inline: Muestra un elemento como un ítem en línea, es decir un elemento que no comienza en una nueva línea y solo ocupa el ancho necesario de su contenido. Para este tipo de elementos las propiedades de altura y ancho no tienen efecto.
* Inline-Block: Podemos considerarla como un híbrido, es decir el elemento al cual se le aplique este valor se comportará como un elemento en línea, al cual se puede aplicar valores de alto y ancho. No ocupara todo el nivel de línea.

**3.3 ¿Para qué se utiliza la propiedad CSS box-sizing? ¿Qué valores puede tener esta propiedad?**

Respuesta. –

El box-sizing es una propiedad CSS que nos permite cambiar el modelo de caja por defecto de los navegadores a uno alternativo, con esto podemos indicar que un elemento solo considere como tamaño real el valor de su altura y anchura, no tomando en consideración para este cálculo las propiedades de padding y margin. Entre los valores que dicha propiedad puede aceptar se tienen:

* content-box, es el valor por defecto que maneja esta propiedad y el navegador. En el esquema al valor de *width* se le debe añadir el valor de *padding* y *border* para obtener el valor de anchura total del elemento, de la misma forma se aplicará esto a *height*.
  + width (anchura) + padding (relleno) + border (borde) = anchura real de la caja
  + height (altura) + padding (relleno) + border (borde) = altura real de la caja
* border-box, a diferencia del caso anterior con este valor los valores dispuestos en *width* y *height* corresponderán al valor total que abarca el elemento, es decir no se considera para el cálculo de las dimensiones al *padding* y *border*
  + width (anchura) = anchura real de la caja
  + height (altura) = altura real de la caja

Existen otros dos valores poco usados:

* *padding-box*‎, los valores de ancho y alto se aplican al contenido del elemento y su relleno (padding), es decir el relleno se empuja dentro de la caja del elemento y no afecta el valor del tamaño total. El borde se agrega al exterior de la caja y es considerado para el cálculo del tamaño total. Actualmente, solo es compatible con Firefox
* *inherit*‎, hereda el tamaño del elemento primario.‎

Segunda parte

Explicación de las entidades HTML y CSS utilizadas

1. **Sobre las entidades HTML: debe explicarse el uso de las etiquetas elegidas.**

Respuesta. –

*Entidades HMTL*

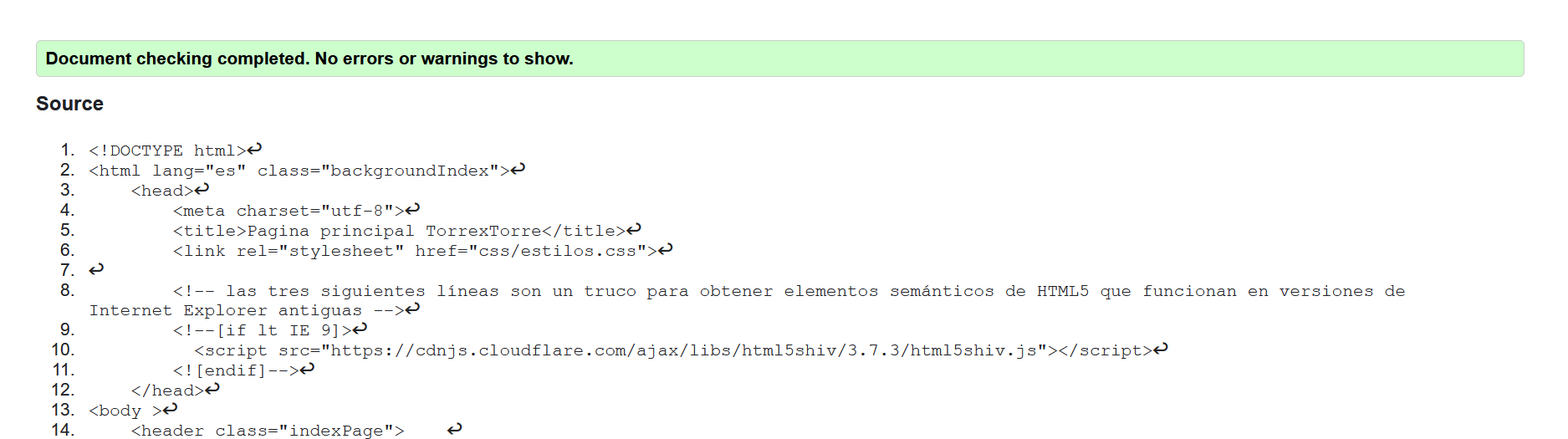
|  |  |
| --- | --- |
| **Etiqueta** | **Explicación** |
| <!DOCTYPE html> | Permite definir el tipo de documento y se agregó a cada página HTML |
| <html lang="es"> | Define la raíz del documento HTML. En este se usa el atributo “lang” para definir en qué tipo de idioma (español) está escrita la página española |
| <head> | Contiene la meta data y la información característica del documento. Enlaces a otros recursos, títulos de la pagina |
| <meta charset="utf-8"> | En el tag se definen los meta datos del documento. En este caso especificamos la codificación del HTML (utf-8) |
| <title> | Define el título del documento. En este ejercicio se usaron tres títulos diferentes. |
| <link> | Se usó para establecer una relación entre el HTML y otro documento externo, en este caso la de estilo. Se usan dos atributos “rel” que indica el tipo de documento, y “href” que da la dirección donde se encuentra la página de estilos. |
| *<!--  -- >* | Permite ingresar un comentario, en este ejercicio para comentar un truco de CSS en conjunto con un Script que ayuda a que las etiquetas semánticas funcionen en versiones antiguas de IE. |
| <body> | Con esta etiqueta delimítanos lo que considero cuerpo del documento |
| <header> | Definimos la cabecera del documento. Al header de index.html se le adiciona una clase (indexPage) particular distinto a los otros documentos de la página web |
| <section> | Permite encapsular un contenido genérico dentro del documento. Se uso en el caso de inscripcion.html para la forma con un id *sectionInscription* |
| <h1> | Con esta etiqueta definí los encabezados de cada artículo, section, y el utilizado en el encabezado principal de la compañía en index.html |
| <img> | Con este tag se pudo definir una imagen dentro del documento. En el indicamos la fuente de la imagen que queremos incluir con el atributo “src”. Además, tiene un atributo “alt” que proporciona una descripción de la imagen para los lectores de página, y casos en la que no sea posible mostrarla. Cada imagen en el proyecto cuenta con ambos atributos, |
| <a> | La etiqueta permite decir los hipervínculos en las páginas, tanto internos como externos.  Estos enlaces contaron con el atributo “href” y “target=\_blank”. Los enlaces internos no tienen target. |
| <span> | Es un contendor en línea. Permite agrupar texto o elemento dentro de los párrafos u otro texto. En este proyecto se usó para aplicar estilo especifico a un determinado texto. |
| <nav> | Es una etiqueta de tipo semántica que permite encapsular lo que se considera menú de navegación. En el caso de las 4 páginas esta sección es básicamente la misma en el header. |
| <ul> | Etiqueta que ayuda a definir una lista no ordenada. Se usaron para crear el menú de navegación, y otros elementos en la página cursos.html. |
| <li> | Etiqueta que define los ítems de las listas. Se usaron para crear el menú de navegación, y otros elementos en la página el-ajedrez.html. Es el caso de los <li> en el menú de navegación se incluyó el atributo class="current" que ayudaría a dar estilo e indicar la página en la que se encuentra actualmente el usuario. Esta clase va variando de posición en los <li> según la página en la que se encuentre el usuario |
| <main> | Especifica el contenido principal del documento |
| <p> | Define un párrafo |
| <footer> | Se usó para especificar el pie de página. Se adiciono el atributo “class="indexPage” en index.html para tener un manejo distinto a de las demás páginas en cuanto a CSS |
| <div> | Es una etiqueta no semántica que me permitió encapsular algunas áreas dentro de footer, cursos.html y inscripcion.html. Ayudan a separar por bloques para un mejor entendimiento de la estructura solicitada y aplicación de css, no se usó estructura semántica ya que lo que se necesitaba era solo aplicar CSS.  Los atributos de clase *right, ulleft* ayuda a lograr lo requerido en el mockups. |
| <article> | Es una etiqueta del tipo semántico que encapsula lo que considere como un artículo. Para este ejercicio los contenidos los considere similares a los que se podrían conseguir en un periódico. Usado en el-ajedrez.html y cursos.html ( este ultimo con una clase *cursos* para dar un estilo a las listas de esa página) |
| <strong> | Permite dar un marcado especial con énfasis. Se usó en palabras dentro de párrafo en el-ajedrez.html, e inscripciones.html |
| <h2> | Define encabezados de segundo nivel en el HTML, en este caso el para los que seguían al tipo h1 del apartado que correspondía. |
| </h3> | Define encabezados de tercer nivel en el HTML, en este caso el para los que seguían al tipo h2 del apartado que correspondía. |
| <figure> | Especifica un contenido independiente que hace referencia al texto principal. En el caso del ejercicio, fue usado para encapsular imágenes y tablas ya que según 3w puede contener este contenido de flujo. El-ajedrez.html, en el caso de la tabla se agrega el atributo “table”para asignar css |
| <figcaption> | Especifica la leyenda de cada información contenida en la etiqueta <figure> |
| <abbr> | Es una etiqueta que permitió dar información sobre abreviaciones dentro de los párrafos. En ella se usó el atributo “title” para ingresar lo que significaba la abreviación. |
| <em> | Es una etiqueta que permite dar un énfasis en el documento. Usado en el=ajedrez.html |
| <table> | Este tag permite organizar los datos en filas y columnas de la tabla dispuesta en el-ajedrez.html |
| <caption> | Permite establecer un titulo para la tabla en el-ajedrez.html |
| <thead> | El tag permite encapsular el contenido considerado como cabecera de la table |
| <tr> | Permite definir una fila las filas de la tabla |
| <th> | Con este tag se definieron las celdas consideradas como encabezado en la tabla. A este tag se le adiciona el atributo scope “row”o “col” dependiendo si era de una columna o fila |
| <tbody> | El tag permite encapsular el contenido considerado como cuerpo de la table |
| <time> | Define un tiempo específico de una forma que es entendible para la máquina. Se usó el atributo “datetime” con las fechas que se mostraban en dentro del contenido. |
| <dl> | Etiqueta que permite la creación de una lista de descripción. Usada en la página cursos.html |
| <dt> | Etiqueta que ayuda a definir un término en una lista de descripción. Usada en la página cursos.html |
| <dd> | Etiqueta que se usa para describir un término en una lista de descripción, está dentro del <dt>. Usada en la página cursos.html |
| <form> | Este elemento fue usado para la creación de un formulario dentro de la página inscripción.html. Se adiciono un atributo acción con valor *post* para indicar que queremos enviar los datos mediante el servidor y no la url |
| <fieldset> | Este tag se usó para agrupar elementos que guardan relación entre si dentro de la forma. En la página de inscripción.html se agruparon tres grupos: Selección del curso, Datos personales y de contacto y Otras informaciones, este ultimo con un atributo de clase *textarea* para poder aplicar css especifico |
| <legend> | Este tag se usó para para definir un título para cada elemento <fieldset>. |
| <label> | Usado para definir una etiqueta descriptiva para cada elemento de inserción de información dentro del formulario. Cada una de estas etiquetas tiene un atributo *for* cuyo valor debe ser igual al atributo *id* dentro de cada entrada para unirlos. |
| <select> | El elemento se utiliza para crear una lista desplegable en el formulario. Para este ejercicio se adicionan los atributos *name, id, required*. El primero es necesario para hacer referencia a los datos del formulario después de enviar el formulario, el segundo para estableces la asociación con el tag *label,* y el ultimo para indicar que este campo es necesario y no se puede obviar. |
| <option> | Este tag, define las opciones disponibles en la lista desplegable que presenta *selec*t. Posee un atributo *value* que indica que valor fue tomado en la selección para luego ser enviado en la forma |
| <input> | Especifica un campo de entrada donde el usuario puede ingresar datos. Según el atributo que se le asigne puede variar el tipo de entrada manejada, para el ejercicio se usaron *type : text, email, tel, date, checkbox.* El primero permita la entrada de tipo texto, el segundo de correo, teléfono, fecha, y chequeo de casilla. Para el caso de las entradas tipo texto se adiciono un atributo extra *pattern* para darle como valor una expresión regular *[a-zA-Z\u00C0-\u00FF\-]{2,128}* que ayuda a que solo ciertas combinaciones sean posibles ( según lo requerido en PEC2) |
| <textarea> | EL tag permite definir una de entrada de texto de varias líneas. Además de los atributos *id , name* , ya antes explicados, se le incluye el de *rows*, y *cols* que permiten especificar el tamaño del área de texto. |
| <button> | El elemento define un botón en el que se puede hacer clic. Posee un atributo *type* con valor *submit.* Que indica que tiene funcionalidad de envío de datos del formulario |

*Entidades CSS*

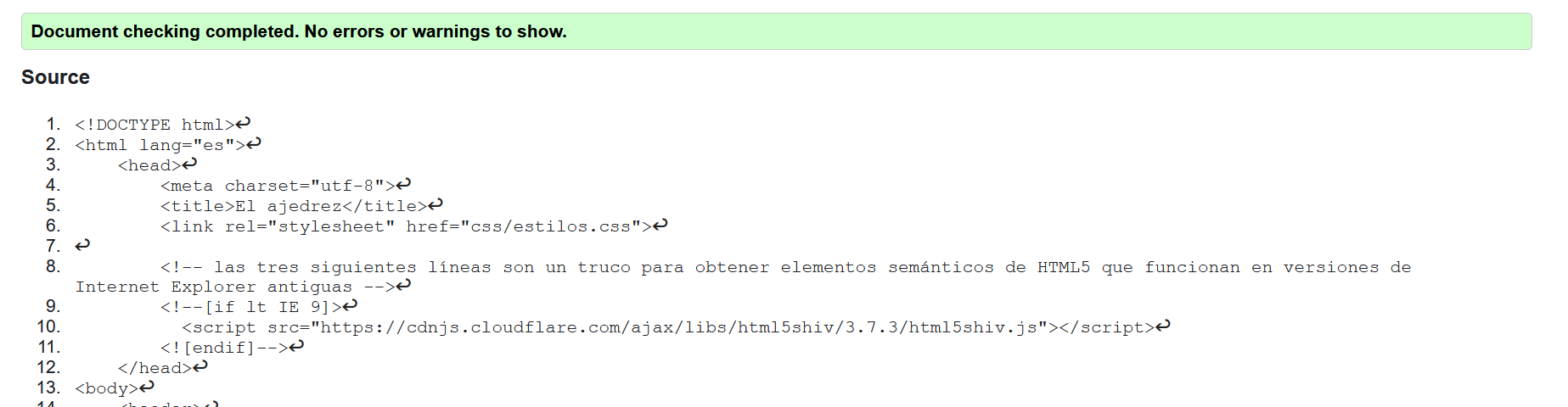
|  |  |  |
| --- | --- | --- |
|  | **Etiqueta** | **Explicación** |
|  | @import url("https://fonts.googleapis.com/css2?family=Noto+Sans:ital,wght@0,400;0,700;1,400;1,700&display=swap"); | Importamos el tipo de letra a usar |
| \* {    margin: 0;    padding: 0;    box-sizing: border-box;    font-size: 10px;    font-family: "Noto Sans", sans-serif;    color: rgb(23, 23, 23);  } | Regla que permite un reset general y establece un tamaño de letra estándar y color texto |
| En esta sección se dará formato general a páginas: ajedrez,cursos, Inscripción, e index | header div,  footer div,  article,  header.indexPage nav,  div.indexPage,  #sectionInscription {    width: 60%;    margin-left: auto;    margin-right: auto;  } | Regla para alinear contenido, cabecera y pie de pagina |
| footer div {    padding-top: 1rem;    padding-bottom: 3.5rem;  } | Regla para mover contenido texto footer |
| header:not(.indexPage),  footer:not(.indexPage) {    background-color: rgb(0, 0, 0);  } | Regla para colocar fondo negro cabecera y pie de página, salvo index.html |
| a {    text-decoration: none;  } | Regla para quitar línea baja por defecto de enlaces |
| a:hover {    color: rgb(153, 17, 17);  } | Regla para dar estilo a todos los enlaces cuando hover |
| article a {    border-bottom: 0.1rem dotted;  } | Regla para dar estilo a los enlaces del texto, línea inferior |
| a.exceptionHover,  nav \*,  footer \*,  div.indexPage h1,  div.indexPage p {    color: rgb(255, 255, 255);  } | Regla para evitar cambio de color en hover para encabezado torrextorre. Dar color a los enlaces en menú Nav, y contenido footer, y texto de p y h1 de index.html |
| nav ul li,  footer ul li {    list-style-type: none;    display: inline-block;  } | Regla para remover vinetas de UL en navegación |
| nav ul li::before,  footer ul li::before {    content: "·";    width: 5rem;    padding-right: 0.5rem;    padding-left: 0.25rem;  } | Creamos puntos falsos entre li de Nav y Foother |
| nav ul li:first-child::before,  footer ul li:first-child::before {    content: "";    padding-right: 0rem;    padding-left: 0rem;  } | Quitamos el primer punto falso del li de Nav y Foother |
| nav a,  nav li {    font-size: 1.4rem;  } | Regla que dar estilo a los enlaces en menú Nav |
| header div > a,  span {    font-size: 3.5rem;    font-weight: 700;  } | Regla para dar estilo de fuente en nombre de header |
| span {    color: rgb(153, 17, 17);  } |
| div > img {    width: 3%;    margin-right: 1.1rem;  } | Regla para dar estilo a logo de navegación, salvo index.hmtl |
| header div {    padding-top: 2rem;    padding-bottom: 1.3rem;  } | Regla para dar espacio en contenido de header |
| .current {  font-weight: 700;  } | Regla para resaltar en que página se encuentra usuario |
| article \*,  #sectionInscription \* {    font-size: 1.2rem;  } | Regla para establecer tamaño de fuente en texto |
| article,  #sectionInscription {    margin-top: 5rem;    margin-bottom: 5rem;  } | Regla para dar formato a contenedor de text, margen superior e inferior |
| article h1,  #sectionInscription h1 {    font-size: 3rem;    text-align: center;    margin-bottom: 1.5rem;  } | Reglas para dar formato a títulos dentro de texto |
| article h2,  h2 time {    font-size: 2rem;    margin-top: 1.2rem;    margin-bottom: 1.2rem;  }  article h3 {    font-size: 1.5rem;    margin-top: 1.2rem;    margin-bottom: 1.2rem;  } |
| p {    margin-top: 1.2rem;    line-height: 2rem;  } | Regla para espacio entre párrafos |
| footer p {    margin-top: 0rem;  } | Regla para reset de márgenes párrafos en footer |
| figure {    width: 65%;    margin-left: auto;    margin-right: auto;    margin-top: 2rem;    margin-bottom: 1.5rem;    text-align: center;  } | Reglas para centrar el contenido de figure, y dimensionar el contenido |
| figure img {    width: 80%;  } |  |
| figcaption {    font-size: 1rem;  } |  |
| En esta sección se dará formato a la tabla | figure.table {    width: 100%;  } | Regla para otorgar ancho completo al figure de la tabla |
| figure.table figcaption {    margin-top: 1.2rem;    text-align: left;  } | Regla para alinear figcaption de tabla |
| table {    border-collapse: collapse;    border: 1px solid rgb(23, 23, 23);    margin-left: auto;    margin-right: auto;    margin-top: 1.2rem;    width: 75%;  } | Regla para modificar border de tabla y alinearla con respecto al figure |
| thead {    border: 1px solid rgb(23, 23, 23);  } | Regla para dar estilo a bordes de cabecera |
| caption {    border: 1px solid rgb(23, 23, 23);    border-bottom-style: hidden;    font-size: 1.8rem;  } | Regla para dar estilo a caption de tabla, bordes y tamaño de fuente |
| thead tr th {    padding: 0.5rem 2rem 0.5rem 0.5rem;    text-align: left;    font-size: 1rem;  } | Regla para alinear contenido de cabecera y dar estilo a su fuente |
| tbody th,  td {    text-align: left;    padding-left: 0.5rem;    padding-top: 1rem;    padding-bottom: 1rem;  } | Regla para alinear contenido de cuerpo de tabla |
| tbody tr:nth-child(odd) {    background-color: rgba(0, 0, 0, 0.1);  } | Regla para crear líneas de cebra de la tabla |
| th[scope="row"] {    font-size: 1.5rem;  } | Regla para aumentar tamaño de fuente a cabeceras de cada row |
| thead tr th:nth-child(1),  thead tr th:nth-child(2) {    width: 20%;  } | Reglas para proporcionar anchos de celda según mok |
| thead tr th:nth-child(3) {    width: 17%;  } |
| td:nth-child(4) {    padding-right: 2.5rem;  } |
| En esta sección se dará formato a las listas de cursos | .ulLeft,  .right {    width: 70%;  } | Con estas reglas creamos dos contenedores para cada clase de división derecha e izquierda, de ul / dl |
| .ulLeft {    margin-left: 30%;  } | Con esta Reglas movemos a la izquierda la división con la clase asignada |
| article.cursos ul {    padding-left: 2.5rem;    line-height: 2rem;    list-style-type: square;  } | Con estas reglas damos formato de espacio a los contenidos dentro e ul y dl |
| article.cursos dl {    padding-left: 2rem;  } |
| dt {    font-size: 1.5rem;    font-weight: bold;    border-bottom: 1px solid rgb(0, 0, 0);    margin-top: 1rem;  } | Con la regla se da estilo a cada dt dentro de su dl |
| dd {    line-height: 2rem;  } | Regla que otorga espacio de línea entre dd |
| En esta sección se dará formato a index.html | .indexPage h1,  .indexPage span {    font-size: 11rem;    display: inline-block;  } | Regla establecer tamaño de fuente del encabezado y posicionamiento |
| .backgroundIndex {    background-image: url(../img/home-hero.jpg);    background-repeat: no-repeat;    background-attachment: fixed;    background-size: cover;  } | Regla para imagen de fonde index.html |
| div.indexPage img {    width: 9%;    margin-left: 0.7rem;    margin-bottom: 0.5rem;  } | Reglas para dar estilo a la imagen del encabezado, tamaño y posición |
| div.indexPage h1 + p {    font-size: 3rem;    font-weight: bold;  } | Regla para dar estilo a unico parrafo de encabezado |
| header.indexPage nav {    margin-top: 2rem;  } | Regla para dar espacion superior a menu de navegacion |
| div.indexPage {    margin-top: 15rem;    margin-bottom: 15rem;  } | Regla para dar espacio superior e inferior a sección que contiene el encabezado |
| En esta sección se dará formato al formulario | textarea {    overflow: auto;    resize: both;    max-width: 65rem;    width: 100%;  } | Regla para definir el ancho por defecto del textarea y su comportamiento en caso de overflow |
| legend {    font-weight: bold;  } | Regla para definir el estilo de la leyenda de tabla |
| fieldset {    padding-top: 3rem;    padding-right: 5rem;    padding-left: 5rem;    padding-bottom: 2rem;    border: 0.1rem solid rgb(23, 23, 23);    margin-top: 1.5rem;  } | Regla dar estilo a cada sección correspondiente a un fliedset |
| label,  textarea {    display: block;    margin-bottom: 1.5rem;  } | Reglas para permitir que las labels se disponga sobre sus entradas |
| select,  input {    margin: 0 0 1em 0;    padding: 10px;    width: 35%;  } | Reglas dar estilo a los campos donde el usuario seleccionara o ingresara información |
| div.datosPersonales {    display: inline-block;    width: 32rem;    margin-left: auto;    margin-right: auto;  } | Estilo para los cuadros de ingreso de información |
| div.datosPersonales input {    width: 100%;    margin: 0 0 1em 0;    padding: 10px;  } |
| div.infoExtra {    margin-top: 4rem;    width: 90%;    margin-left: auto;    margin-right: auto;  } |
| input[type="checkbox"] {    display: inline-block;    width: 0.9rem;    height: 0.9rem;  */\* Importante*  *It works well in Google Chrome, Microsoft Edge. But in Mozilla Firefox the clickable checkbox area is as defined, but it shows a default size checkbox. , in that case we could use transform property, but not cover in this PEC\*/*  } | Estilo para el checkbox del formulario |
| button {    background-color: rgb(153, 17, 17);    border: none;    color: white;    padding: 1.5rem 3.2rem;    text-align: center;    font-size: 1.6rem;    margin: 0.4rem 0.2rem;    border-radius: 5%;  } | Estilo para botón de formulario |
| #sectionInscription > p {    display: inline-block;    background-color: rgba(0, 0, 0, 0.1);  } | Estilo para remarcar información de texto inicial de formulario |
| input:invalid:required {    background-color: rgba(0, 0, 0, 0.1);  } | Estilo para indicar que un campo es invalido |
| .textarea {    max-width: 75.6rem;    min-width: 75.6rem;  } | Regla para evitar que el field set que contiene textarea sea modificado |

*Validaciones HMTL y CSS*

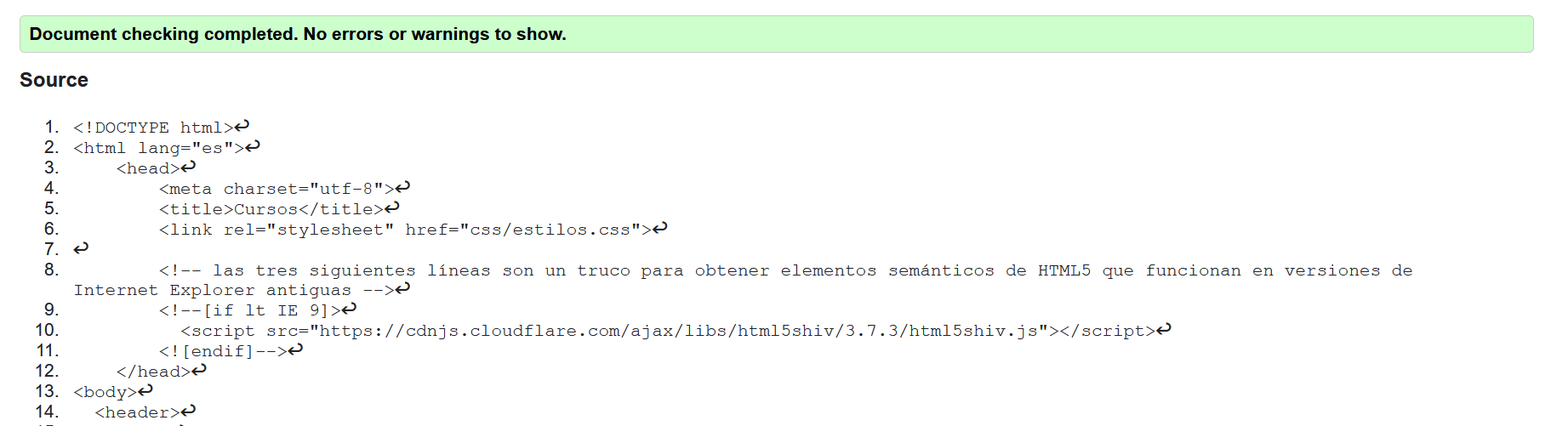
1.- index.hmtl



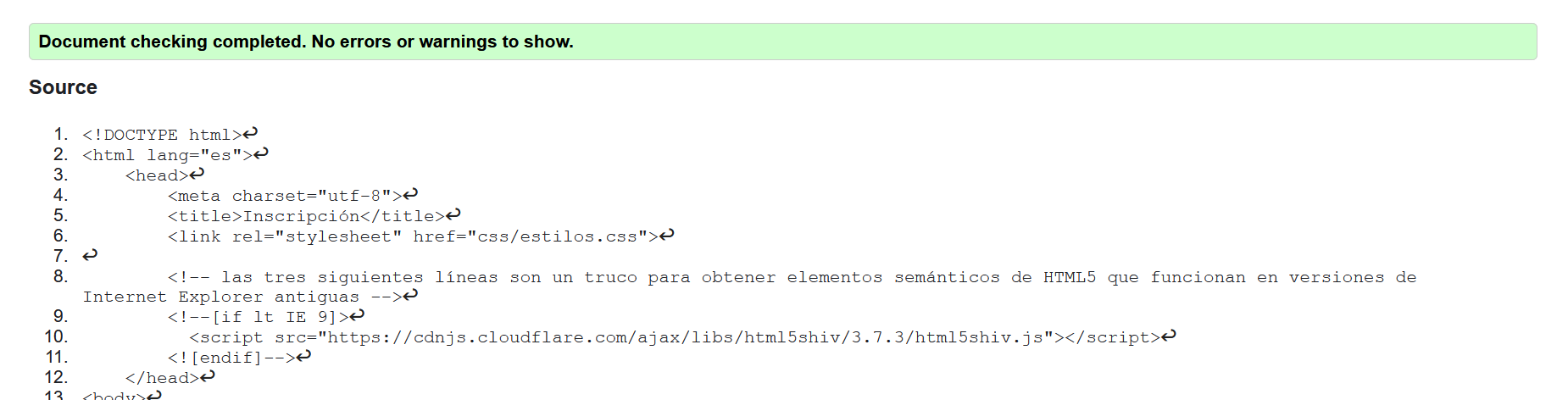
2.- el-ajedrez.html



3.- cursos.html



4.- inscripción.html



5.- estilos.css

