

Desarrollo Web en Entorno Cliente



DOM

VERÓNICA BONIS MARTÍN
MARIA CARMEN CORREA HERAS
ÁNGEL SÁNCHEZ-SIERRA CRUZ
JOSÉ MARÍA TENREIRO EIRANOVA

<https://github.com/ITDDAW/02-DOM.git>

Actividad 2. DOM

Se quiere generar una página web completamente por JavaScript, modificando el árbol DOM. La página web solo tendrá un contenedor (DIV) y sobre dicho contenedor y por medio de JavaScript se añadirán elementos a dicho contenedor. La temática de la web se deja libre al alumno. No es necesario usar CSS.

Se añadirá al menos los siguientes elementos HTML a un formulario

- Campos de texto donde poner el DNI, el nombre y los apellidos, la dirección y el teléfono.
- Dos radio button, con 4 opciones a elegir
- Cinco checkbox diferentes
- 2 imágenes
- Un campo select
- Un text area

Valoración: 8 puntos sobre 10

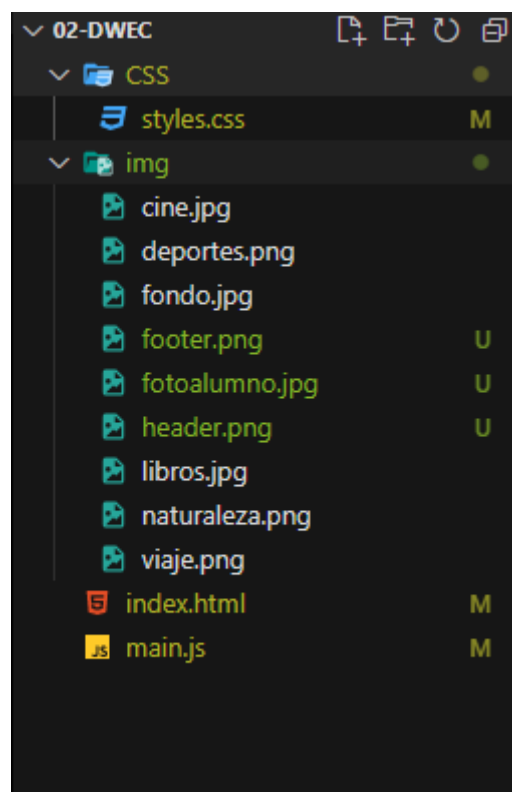
Requerimiento 2

Se valorará los estilos usados en la página web (CSS) y el uso de HTML5.

Valoración: 2 puntos sobre 10

Para la creación de nuestra página web hemos elegido un diseño de un formulario para un alumno de la escuela Edix.

La organización de los directorios en nuestro proyecto será el siguiente:



Actividad 2. DOM

Tenemos dos carpetas llamadas CSS e img en donde almacenaremos nuestra hoja de estilos y nuestras imágenes respectivamente.

Después tenemos los dos archivos importantes de la actividad, el index.html y el main.js.

Tal como se nos dice en la actividad nuestro archivo html solamente puede contener un bloque div en el que después podremos añadir dentro más elementos, pero solamente mediante Javascript. Por lo tanto, nuestro body en el index.html será de la siguiente forma:

```
<!-- onload se utiliza en el <body> para ejecutar un script una vez la página web se ha cargado completamente todo el contenido
Nosotros llamamos a la función cargar() que tenemos definida en el archivo main.js-->

<body onload="cargar()">

  <div class="contenedor" id="contenedor">


  </div>

</body>
</html>
```

Con onload llamamos la función cargar() que tenemos definida en nuestro archivo main.js. Cuando el body tiene todo cargado llama a esta función.

Nuestro archivo main.js es donde vamos a crear todos los elementos necesarios para dotar a nuestra página web de funcionalidad tal como la habíamos pensado que queríamos estructurarla.

Un buen diseño de elementos HTML mediante JavaScript depende totalmente de un buen orden y estructura del código por parte del desarrollador. Cualquier error puede llevar a que nuestra página no cargue y resultar muy difícil encontrar donde cometimos tal error. Por eso, implementar bloque a bloque por orden y no ir dando saltos en el diseño es muy importante para llegar a buen puerto.

Nuestra página web se estructura en tres grandes bloques dentro del bloque contenedor de HTML :

Actividad 2. DOM

```
... <body onload="cargar()" == $0
  <div class="contenedor" id="contenedor"> flex
    <div id="divheader">...</div>
    <div id="divformulario">...</div>
    <div id="divfooter">...</div>
  </div>
  <!-- Code injected by live-server -->
  <script type="text/javascript">...</script>
</body>
```

Estas serán los bloques padres de nuestro documento y cada uno como su nombre indica, va a contener los elementos básicos de este bloque.

La forma de crear los bloques div con Javascript se realiza de la siguiente forma:

```
let divheader=document.createElement("div");
let divformulario=document.createElement("div");
let divfooter=document.createElement("div");
```

En el bloque divheader, introduciremos el elemento header para establecer ahí la cabecera de nuestra página web mediante una imagen:

```
let header=document.createElement("header");
```

En el bloque divformulario, introduciremos nuestro formulario:

```
let formulario = document.createElement("form");
```

Y dentro de este todos los elementos creados correspondientes:

```
let fieldset= document.createElement("fieldset");
let legend= document.createElement("legend");
let labeldni=document.createElement("label");
let inputdni = document.createElement("input");
let fotoestudiante=document.createElement("img");
let radio1= document.createElement("input");
radio1.id="radio1";
radio1.type="radio";
let inputh1= document.createElement("input");
inputh1.id="inputh1";
inputh1.type="checkbox";
let select=document.createElement("select");
select.id="select";

//Creamos las diferentes opciones del select
let option1=document.createElement("option");
let textarea=document.createElement("textarea");
let boton =document.createElement("input");
boton.type="submit";
let footer=document.createElement("footer");
```

Actividad 2. DOM

Estos son solamente unos ejemplos de cómo se crean elementos de HTML con Javascript.

Después de crear elementos es también muy importante seguir un orden al agregarlos a sus bloques correspondientes:

Por ejemplo, hemos creado una etiqueta label y un input que a su vez se encuentran dentro de un bloque div llamado textos y este bloque a su vez se encuentra dentro de otro bloque div llamado cabecera, la forma de ir agregándolo en cada bloque correspondiente seria la siguiente:

```
textos.appendChild(labeldni);
textos.appendChild(inputdni);/
cabecera.appendChild(textos);
fieldset.appendChild(cabecera);
formulario.appendChild(fieldset);
divformulario.appendChild(formulario);
```

Cuando ya queremos añadir nuestro bloque divformulario a nuestro bloque div contenedor del HTML, es decir, divformulario ya queremos que sea hijo directo de div contenedor, entonces tenemos que recoger este div del HTML en una variable de la siguiente manera:

```
let contenedor=document.querySelector(".contenedor");
```

Y a continuación, añadir nuestro bloque divformulario a este div contenedor:

```
contenedor.appendChild(divformulario);
```

El código que se forma en Javascript para realizar bloques o elementos HTML es muy extenso, por eso en esta memoria solo hemos descrito algunos de los pasos para crear estos elementos o bloques.

Al final, la estructura de nuestro proyecto queda de la siguiente manera:

BLOQUE DIVHEADER

```
▼<div id="divheader">
  <header></header>
</div>
```

BLOQUE FORMULARIO

```

*** ▼<div id="divformulario"> == $0
  ▼<form id="formulario" action="#" method="post">
    ▼<fieldset>
      <legend>Ficha personal alumno</legend>
      ▼<div id="cabecera"> grid
        ▼<div id="textos"> flex
          <label>DNI ALUMNO</label>
          <input id="inputdni" type="text" placeholder="Introduce aqui tu DNI">
          <label>NOMBRE</label>
          <input id="inputnombre" placeholder="Introduce aqui tu Nombre">
          <label>APELLIDOS</label>
          <input id="inputapellidos" placeholder="Introduce aqui tus apellidos">
          <label>DIRECCIÓN</label>
          <input id="inputdireccion" placeholder="Introduce aqui tu direccion">
          <label>TELÉFONO</label>
          <input id="inputdireccion" type="number" placeholder="Introduce aqui tu telefono">
        </div>
        
      </div>
      ▼<div id="radios"> grid
        ▶<div id="gruporadio1">...</div>
        ▶<div id="gruporadio2">...</div>
      </div>
      ▼<div id="divbloqueustos"> grid
        ▶<div id="divhobbies">...</div>
        ▶<div id="divselect">...</div>
      </div>
      ▼<div id="textareadiv"> flex
        <label>Describe tus condiciones en 5 lineas</label>
        <textarea id="textarea" cols="100" rows="5"></textarea>
        <input type="submit" id="boton">
      </div>
    </fieldset>
  </form>
</div>

```

BLOQUE FOOTER

```
<div id="divfooter">  
  <footer></footer>  
</div>
```

Después con una hoja de estilos y como le hemos establecido a prácticamente todos los elementos sus id correspondientes, a la vez que hemos introducido cada conjunto de elementos en bloques div, es muy fácil darle estilo a la página web creada mediante javascript. De tal manera que quedaría como en la siguiente captura:

edix Digital
Workers

Sobre edixCursosFormación ProfesionalAction LearningBlog

Acceder

Te llamamos

Ficha personal alumno

DNI ALUMNO
Introduce aqui tu DNI

NOMBRE
Introduce aqui tu Nombre

APELLIDOS
Introduce aqui tus apellidos

DIRECCIÓN
Introduce aqui tu direccion

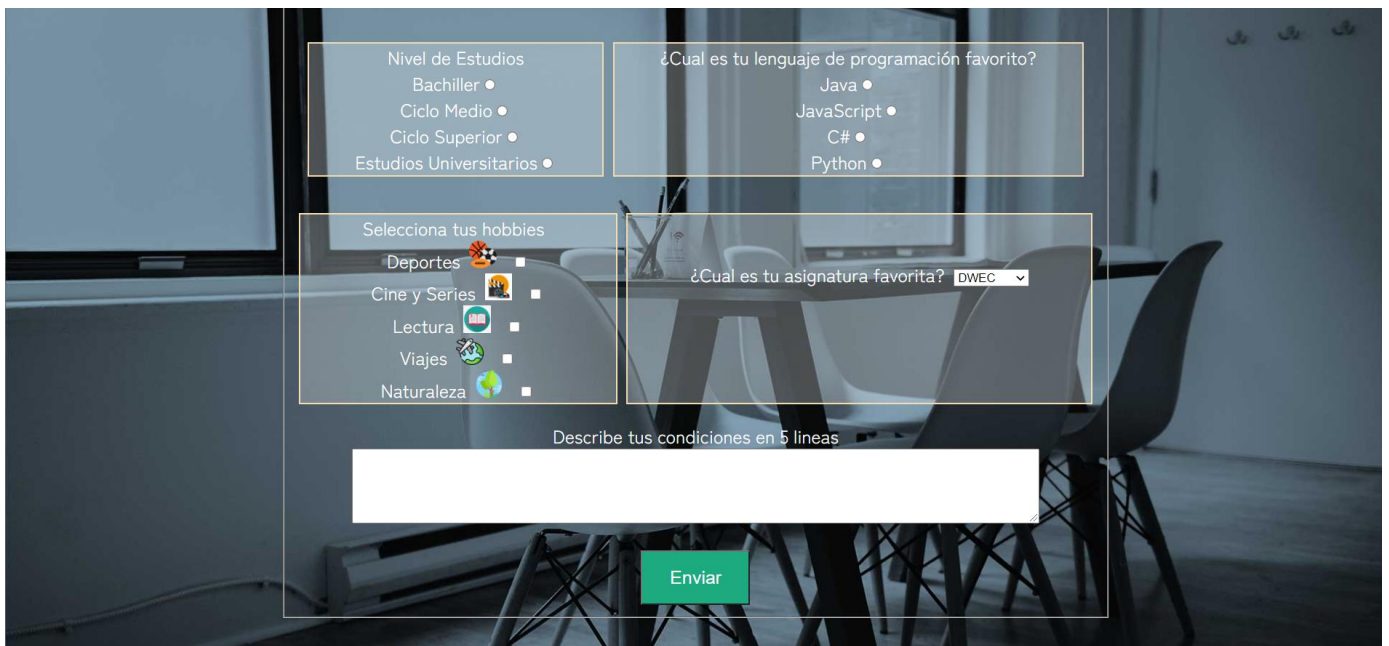
TELÉFONO
Introduce aqui tu telefono

Nivel de Estudios
Bachiller •
Ciclo Medio •
Ciclo Superior •
Estudios Universitarios •

¿Cual es tu lenguaje de programación favorito?
Java •
JavaScript •
C# •
Python •

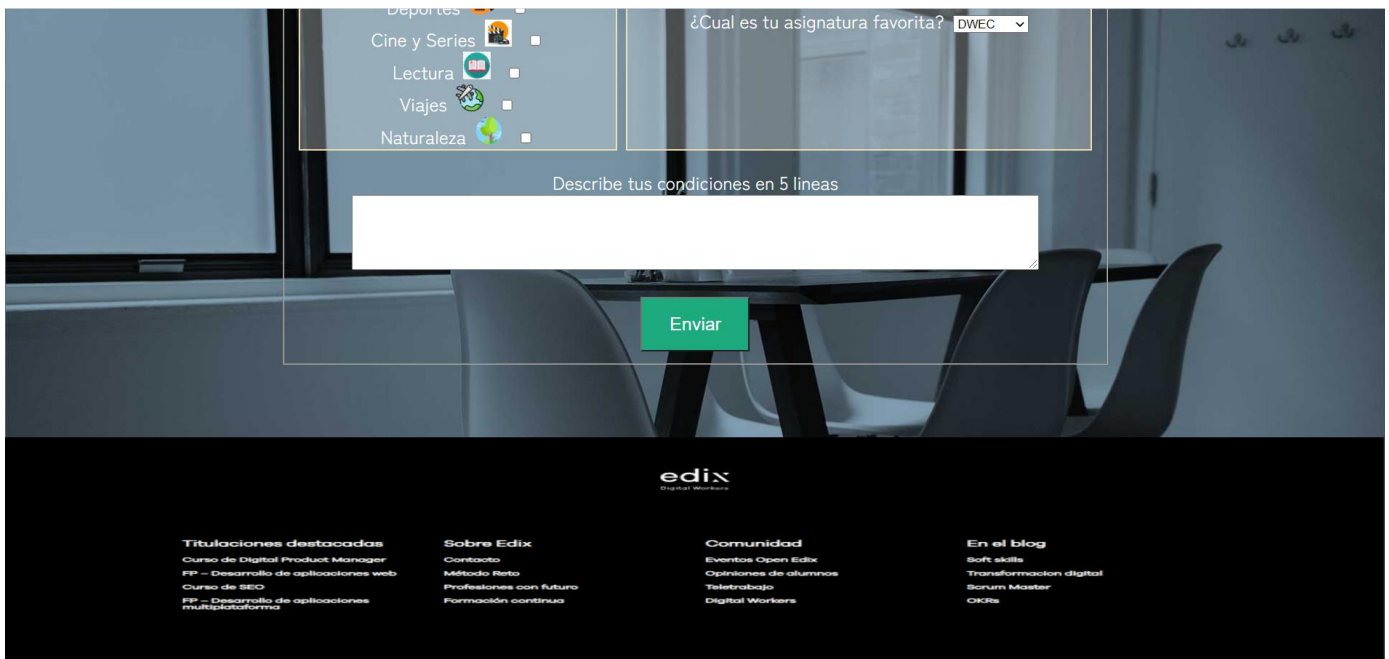
Selecciona tus hobbies
Deportes

Actividad 2. DOM



Formulario de registro de Edix. El formulario está superpuesto sobre una imagen de fondo que muestra una sala de estudio con una pizarra y una mesa con sillas. El formulario contiene los siguientes elementos:

- Nivel de Estudios:** Una lista de opciones con radio buttons: Bachiller, Ciclo Medio, Ciclo Superior, Estudios Universitarios.
- ¿Cual es tu lenguaje de programación favorito?:** Una lista de opciones con radio buttons: Java, JavaScript, C#, Python.
- Selecciona tus hobbies:** Una lista de opciones con checkboxes y iconos: Deportes, Cine y Series, Lectura, Viajes, Naturaleza.
- ¿Cual es tu asignatura favorita?:** Un menú desplegable con la opción DWEC seleccionada.
- Describe tus condiciones en 5 líneas:** Un campo de texto grande.
- Enviar:** Un botón verde.



Formulario de registro de Edix (detalle). Este bloque muestra una vista más detallada del formulario, destacando los elementos de selección de hobbies y la asignatura favorita. El formulario está superpuesto sobre una imagen de fondo que muestra una sala de estudio con una pizarra y una mesa con sillas. El formulario contiene los siguientes elementos:

- Selecciona tus hobbies:** Una lista de opciones con checkboxes y iconos: Deportes, Cine y Series, Lectura, Viajes, Naturaleza.
- ¿Cual es tu asignatura favorita?:** Un menú desplegable con la opción DWEC seleccionada.
- Describe tus condiciones en 5 líneas:** Un campo de texto grande.
- Enviar:** Un botón verde.

En la parte inferior del formulario, se encuentra el logo de Edix Digital Workers y una barra de navegación con los siguientes enlaces:

- Titulaciones destacadas:** Curso de Digital Product Manager, FP – Desarrollo de aplicaciones web, Curso de SEO, FP – Desarrollo de aplicaciones multiplataforma.
- Sobre Edix:** Contacto, Método Reto, Profesiones con futuro, Formación continua.
- Comunidad:** Eventos Open Edix, Opiniones de alumnos, Teletrabajo, Digital Workers.
- En el blog:** Soft skills, Transformación digital, Scrum Master, OKRs.

El proyecto se encuentra totalmente comentado siguiendo los pasos que hemos desarrollado.