

Práctica JavaScript – La reina en su color

Objetivo

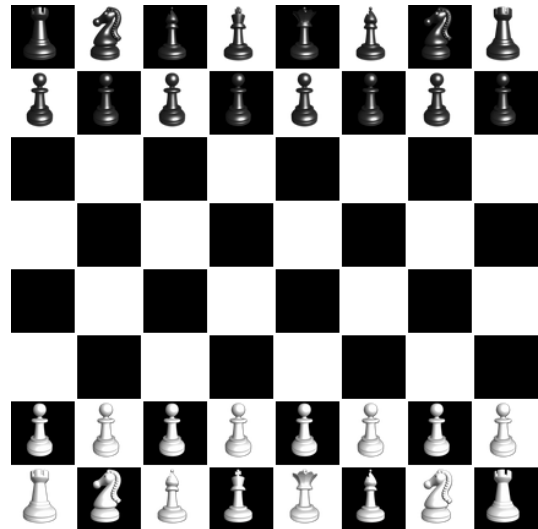
Aprender a modificar el DOM de una página HTML usando jQuery.

Instrucciones

Lee toda la práctica antes de empezar.

En esta práctica, se va a crear un tablero de ajedrez en una página HTML. Será un tablero convencional como el mostrado en la foto. El tablero permitirá mover las piezas entre las casillas con cierta validación.

Se usarán tablas, se aplicará estilo CSS y la interacción con el usuario será mediante jQuery (JavaScript).



Parte 1 – Crear el tablero

1. Crear el tablero:

- Todo comienza por crear una tabla de 8 filas y 8 celdas en cada una, dando el tablero completo de 64 casillas.
- Cada casilla debe tener su atributo *id* con la estructura “fila_columna”, por ejemplo: “1_1”, “1_2”, “2_3”. No debe haber repetidos.

2. Da estilo al tablero:

- Indica en las celdas de color negro la clase ‘celdaNegra’.
- Crea un bloque CSS en el que definas la clase ‘celdaNegra’ y le apliques un color de fondo negro (u otro color muy oscuro).
- Define con CSS estilo para todos los elementos TD (celdas). Indica que deben tener medir 70x70px, borde blanco de 1 pixel y tener fondo blanco.
- Define un color de fondo para el cuerpo de la página en gris (u otro color medio).

3. Coloca piezas

- Escoge cuatro imágenes de las piezas de ajedrez incluidas con el material (dos negras y dos blancas) e insértalas en sus celdas correspondientes en el tablero. Ajusta el tamaño usando CSS.

En este punto, nuestra tabla ya debe verse como un tablero de ajedrez y tener algunas piezas.

Parte 2 – Detectar el clic

4. Implementación del primer script:

- Crea un bloque de JavaScript en el encabezado y define una función “*celdaPresionada(id)*”
- La función debe imprimir en consola el atributo recibido *id*.

5. Asignación de acciones a las celdas

- Asigna a cada celda del tablero la propiedad *onClick*=“*celdaPresionada(id)*”. Esto llamará a la función *celdaPresionada* cada vez que le demos clic y enviará su atributo *id* como argumento.

- b. Haz pruebas dando clic en las diferentes celdas. Cada vez que lo hagas, en la consola del navegador Web debe aparecer el id de la celda presionada.

En este punto ya podemos interactuar con nuestro tablero y saber qué celda ha sido presionada.

Parte 3 – Mover las fichas

El procedimiento para mover una ficha es como sigue:

- a. Se da clic en la casilla de origen; lo que invoca a la función *celdaPresionada* y ésta guarda el id de la casilla presionada en una variable temporal llamada *origen*.
- b. Se da clic en la casilla de destino. Se vuelve a invocar la función pero como ya se tiene un origen, se copia el contenido del origen al destino y se borra el contenido del origen.

6. Inclusión de jQuery

- a. Introduce el *API jQuery* a tu página creando un elemento script en el encabezado y definiendo en el atributo *src* la ruta de la librería *jQuery* que tienes.

7. Complemento de la función para mover

- a. Complementa la función *celdaPresionada* para que tome el contenido de una celda y lo mueva a otro de la siguiente manera:
 - i. Declara una variable llamada *origen = null* **fuera** de la función *celdaPresionada*. Esta es nuestra variable global temporal.
 - ii. Dentro de *celdaPresionada*, verifica si la variable *origen* es *null*. Si es así, entonces asigna el valor del argumento *id* a la variable *origen*, *cambia el color de la celda a rojo* y termina.
 - iii. Si la variable *origen* no es *null*, intercambia el contenido:
 1. Guarda en una variable temporal el contenido HTML de la celda de origen usando el método *html()* de jQuery (sin argumentos).
 2. Asigna a la celda de destino el contenido que pusiste en la variable temporal usando el método *html(contenido)* de jQuery.
 3. Limpia el contenido de la celda de origen usando el mismo método.
 4. Regresa el color de la celda origen a blanco.

En este punto nuestro tablero ya funciona :D

Si te complicas mucho para diseñar la función, puedes verla al final de este documento. Pero haz tu mejor esfuerzo por hacerla sol@.

Parte 4 – Validar casillas

Es importante que no se pueda depositar una ficha en una casilla que contiene otra del mismo color. Si es de distinto color, no hay problema porque se la estaría comiendo. Para eso, nos auxiliaremos en el hecho de que el nombre del archivo de las fichas contiene su color.

8. Validamos si son de color distinto.

- a. Crea una función nueva llamada *sonDiferenteColor(contenidoA, contenidoB)*.
- b. En dicha función, utiliza la función *indexOf* para determinar si el argumento A contiene la cadena "white". El resultado de esta pregunta guárdalo en una variable *var aEsBlanca (booleana)*.

- c. Haz el procedimiento análogo para el argumento B.
- d. El valor de retorno de la función debe ser booleano, preguntando si *aEsBlanca* es distinto de *bEsBlanca*. Eso nos dirá si ambas fichas son de distinto color.

9. Incluimos la condición en nuestra función

- a. Cuando nuestra función *celdaPresionada* está a punto de mover una pieza debe considerar lo siguiente:
 - i. Si la celda de destino está vacía (texto en blanco), se puede mover sin problemas.
 - ii. Si la celda de destino contiene algo, debe llamar a la función *sonDiferenteColor* para asegurarse de que las piezas de distinto color.
 - 1. Si son distintas, la mueve sin problemas.
 - 2. Si son del mismo color, alerta a usuario indicando que el movimiento es inválido... y no mueve la ficha, claro.

Si te complicas mucho para diseñar la función, puedes verla al final de este documento. Pero tú puedes hacerla sol@.

Listo, nuestro tablero de ajedrez funciona muy bien... ¡Felicidades!

Ve más allá colocando todas las fichas del tablero.

Función sin validación

```
var origen = null;

function celdaPresionada(id){
  if (origen == null){
    origen = id;
    console.log("Origen " + id);
  }
  else{
    console.log("Moviendo de" + origen + " a " + id);

    var contenidoOrigen = $("#" + origen).html();

    $("#" + id).html(contenidoOrigen);
    $("#" + origen).html("");

    origen = null;
  }
}
```

Función con validación

```
var origen = null;

function sonDiferenteColor(contenidoA, contenidoB){
  var aEsBlanca = contenidoA.indexOf("white") > 0;
  var bEsBlanca = contenidoB.indexOf("white") > 0;

  return aEsBlanca != bEsBlanca;
}

function celdaPresionada(id){
  if (origen == null){
    origen = id;
    console.log("Origen " + id);
  }
  else{
    console.log("Moviendo de" + origen + " a " + id);

    var contenidoOrigen = $("#" + origen).html();
    var contenidoDestino = $("#" + id).html();

    if (contenidoDestino.trim() == ""){
      $("#" + id).html(contenidoOrigen);
      $("#" + origen).html("");
    }
    else if (sonDiferenteColor(contenidoOrigen, contenidoDestino)){
      $("#" + id).html(contenidoOrigen);
      $("#" + origen).html("");
    }
    else{
      alert("No te puedes comer a una ficha de tu mismo color.");
    }
    origen = null;
  }
}
```