

EL JUEGO DE LA OCA

Actividad de desarrollo (UF2)



- Diseño de Interfaces Web.
- Delfín Rojas Espina.

Índice:

- ▶ Explicación de los archivos .html:
 - EL GRAN JUEGO DE LA OCA.html
 - selección_color.html
 - Tablero.html

Diapositivas.....3, 4, 5.
- ▶ Explicación del archivo de JAVASCRIPT “selección.js”:

Diapositivas.....6, 7, 8,
- ▶ Explicación del archivo “EstiloSeleccion.css”:

(Archivo que da estilo a los archivos html “EL GRAN JUEGO DE LA OCA.html” y “seleccion_color.html”).

Diapositivas.....9, 10, 11, 12.
- ▶ Explicación del archivo “EstiloTablero.css”:

(Archivo que da estilo al archivo “Tablero.html”).

Diapositivas.....13, 14, 15, 16.
- ▶ Explicación del archivo de JAVASCRIPT “LogicaJuego.js”:

Diapositivas..... 17, 18, 19, 20,

FileEditSelectionViewGoRun...<-->ElJuegoDeLaOcaDelfinRojas

EXPLORER...<-->Tablero.htmlseleccion_color.htmlEL GRAN JUEGO DE LA OCA.html xEstiloSeleccion.cssEstiloTablero.cssJS LogicaJuego.jsJS seleccion.js

OPEN EDITORS<-->Tablero.htmlseleccion_color.htmlx EL GRAN JUEGO DE LA OCA.htmlEstiloSeleccion.css CSSEstiloTablero.css CSSJS LogicaJuego.js JSseleccion.js JS

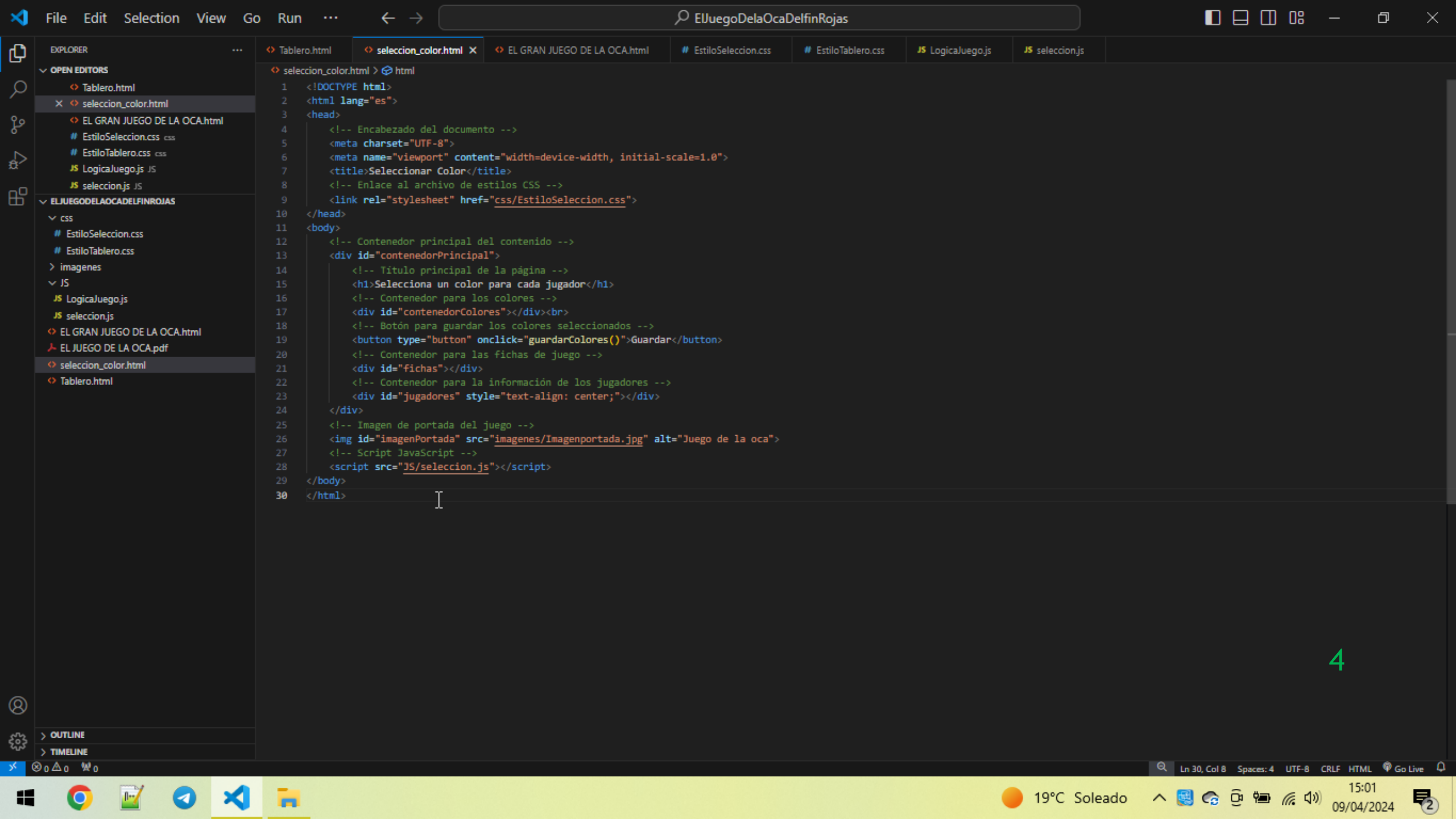
EL JUEGO DE LA OCA DEL FIN ROJAS<-->cssEstiloSeleccion.cssEstiloTablero.cssimagenesJSLogicaJuego.jsseleccion.jsEL GRAN JUEGO DE LA OCA.htmlEL JUEGO DE LA OCA.pdfseleccion_color.htmlTablero.html

OUTLINETIMELINE

EL GRAN JUEGO DE LA OCA.html > html<-->1<!DOCTYPE html>2<html lang="es">3<head><!-- Encabezado del documento: Aquí se incluyen los metadatos del documento y se enlaza el archivo CSS externo.-->4<meta charset="UTF-8">5<meta name="viewport" content="width=device-width, initial-scale=1.0">6<title>El juego de la Oca</title>7<link rel="stylesheet" href="css/EstiloSeleccion.css">8</head>9<body><!-- Contenido principal: Aquí se encuentra todo el contenido visible de la página. -->1011<div id="contenedor"> <!-- Aquí se incluye todo el contenido principal de la página,12como el título del juego, el formulario13y cualquier otro elemento relacionado con la interfaz de usuario. -->1415<h2>EL JUEGO DE LA OCA</h2><!-- Título principal del juego -->1617<h1>Selecciona el número de jugadores</h1><!-- Título para la selección del número de jugadores -->1819<form id="myForm"> <!-- Formulario que contiene la selección del número de jugadores20y un botón para aceptar la selección. -->2122<label for="numeroJugadores"></label>23<select id="numeroJugadores">24<option value="2">2 Jugadores</option>25<option value="3">3 Jugadores</option>26<option value="4">4 Jugadores</option>27</select>2829<button type="button" onclick="redireccionar()">Aceptar</button>30<!-- Elemento button que tiene la acción predeterminada de enviar datos de un formulario -->31</form>3233</div>343536<!-- Imagen de portada (img): Muestra la imagen de portada del juego. -->3738<script src="JS/seleccion.js"></script>39<!-- Enlaza el archivo JavaScript externo para manejar la funcionalidad de redireccionamiento. -->40</body>41</html>4243

3Ln 41, Col 8Spaces: 4UTF-8CRLFHTMLGo Live

19°C Soleado15:0109/04/2024



File

Edit

Selection

View

Go

Run

...

←

→

ElJuegoDeLaOcaDelfinRojas

□

□

□

□

—

□

×

EXPLORER

...

OPEN EDITORS

✕ Tablero.html

seleccion_color.html

EL GRAN JUEGO DE LA OCA.html

EstiloSeleccion.css

EstiloTablero.css

JS LogicaJuego.js

JS seleccion.js

ELJUEGODELAOCADELFINROJAS

css

EstiloSeleccion.css

EstiloTablero.css

imagenes

JS

JS LogicaJuego.js

JS seleccion.js

EL GRAN JUEGO DE LA OCA.html

EL JUEGO DE LA OCA.pdf

seleccion_color.html

Tablero.html

OUTLINE

TIMELINE

Tablero.html

html

1 <!DOCTYPE html>

2 <html lang="es">

3

4 <head>

5 <meta charset="UTF-8">

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

7 <title>Tablero</title>

8 <link rel="stylesheet" href="css/EstiloTablero.css">

9

10 <body>

11 <h1 style="text-align: center;">Lista de Jugadores</h1><!-- Título centrado -->

12 <div id="jugadores" style="text-align: center;"></div><!-- Contenedor de jugadores centrado -->

13 <div id="tablero" style="background-image: url('imagenes/En blanco Linea recta.JPG');"></div><!-- Contenedor del tablero -->

14 </button><!-- Botón para lanzar el dado -->

15 <button id="btn3">Reiniciar</button><!-- Botón para reiniciar -->

16 <table id="tablaJugadoresFinalizados"><!-- Definición de la tabla de jugadores finalizados -->

17 <thead><!-- Encabezado de la tabla -->

18 <tr><!-- Fila de encabezado -->

19 <th>Nombre del jugador</th><!-- Celda de encabezado para el nombre del jugador -->

20 <th>Posición</th><!-- Celda de encabezado para la posición del jugador -->

21 </tr>

22 </thead>

23 <tbody> <!-- Cuerpo de la tabla para los datos -->

24

25 </tbody> <!-- Aquí se llenará dinámicamente con los jugadores finalizados -->

26 </table>

27

28 <!-- Imagen de portada del juego -->

29

30

31 <!-- Importar jQuery -->

32 <script src="https://code.jquery.com/jquery-3.6.0.min.js"></script>

33

34 <!-- Importar el script de la lógica del juego -->

35 <script src="JS/LogicaJuego.js"></script>

36 </body>

37

38 </html>

Ln 38, Col 8

Spaces: 4

UTF-8

CRLF

HTML

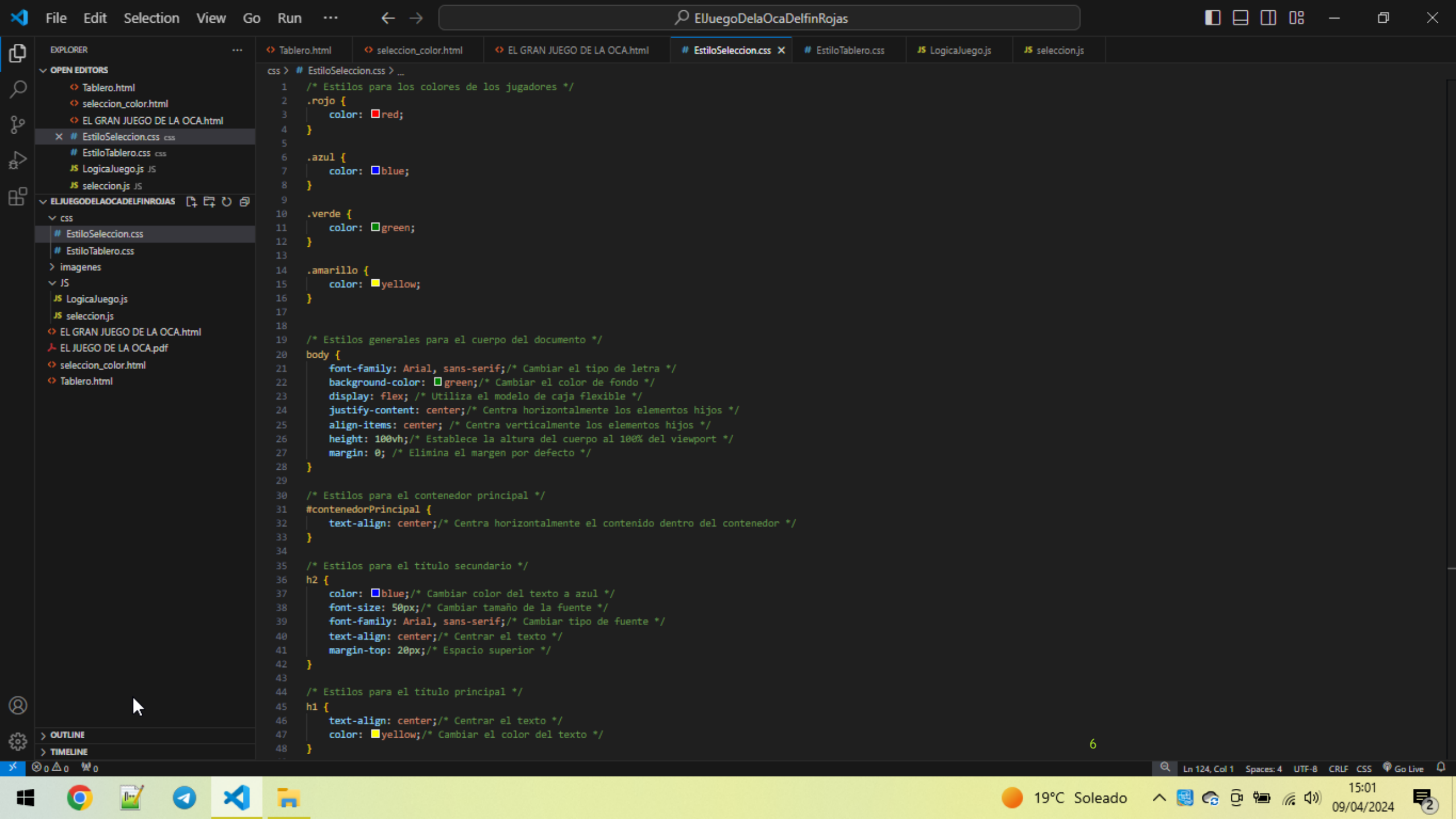
Go Live

19°C

Soleado

15:00

09/04/2024



```
File Edit Selection View Go Run ... < > ElJuegoDeLaOcaDelfinRojas
```

EXPLORER

OPEN EDITORS

Tablero.html

seleccion_color.html

EL GRAN JUEGO DE LA OCA.html

EstiloSeleccion.css

EstiloTablero.css

LogicaJuego.js

seleccion.js

ELJUEGODELAOCADELFINROJAS

css

EstiloSeleccion.css

EstiloTablero.css

imagenes

JS

LogicaJuego.js

seleccion.js

EL GRAN JUEGO DE LA OCA.html

EL JUEGO DE LA OCA.pdf

seleccion_color.html

Tablero.html

OUTLINE

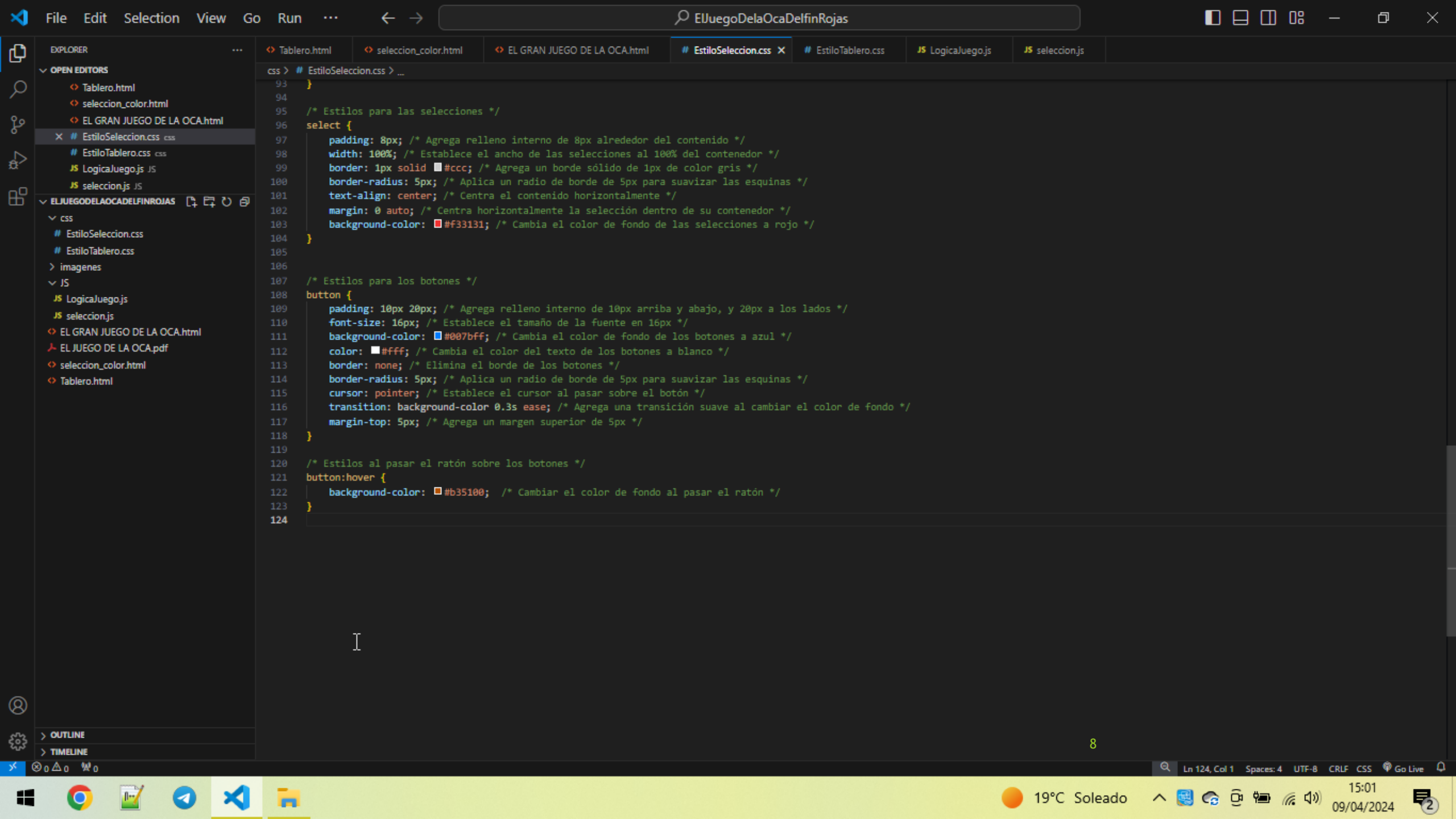
TIMELINE

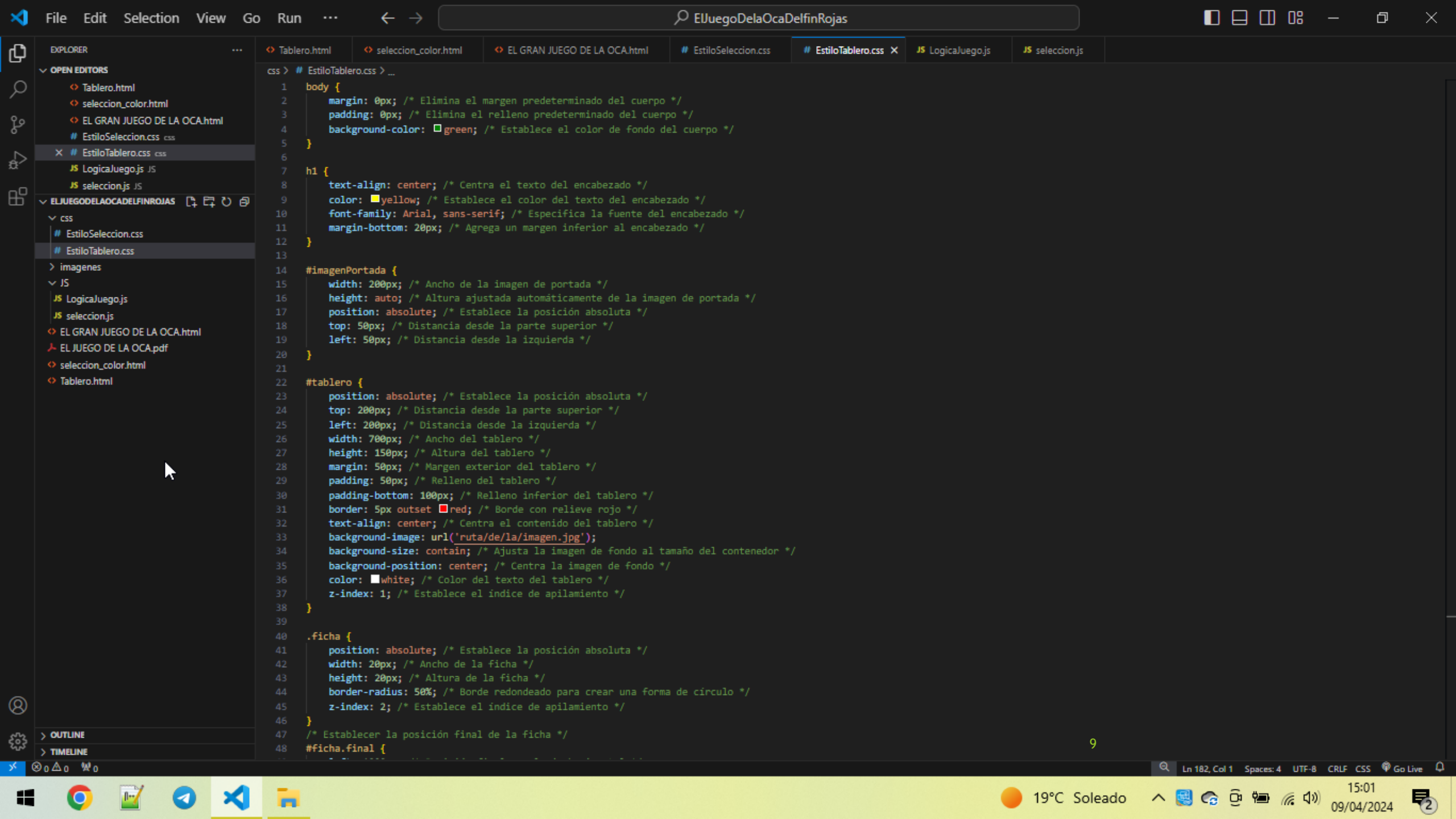
css > # EstiloSeleccion.css > ...

```
1  /* Estilos para los colores de los jugadores */
2  .rojo {
3      color: red;
4  }
5
6  .azul {
7      color: blue;
8  }
9
10 .verde {
11     color: green;
12 }
13
14 .amarillo {
15     color: yellow;
16 }
17
18
19 /* Estilos generales para el cuerpo del documento */
20 body {
21     font-family: Arial, sans-serif; /* Cambiar el tipo de letra */
22     background-color: green; /* Cambiar el color de fondo */
23     display: flex; /* Utiliza el modelo de caja flexible */
24     justify-content: center; /* Centra horizontalmente los elementos hijos */
25     align-items: center; /* Centra verticalmente los elementos hijos */
26     height: 100vh; /* Establece la altura del cuerpo al 100% del viewport */
27     margin: 0; /* Elimina el margen por defecto */
28 }
29
30 /* Estilos para el contenedor principal */
31 #contenedorPrincipal {
32     text-align: center; /* Centra horizontalmente el contenido dentro del contenedor */
33 }
34
35 /* Estilos para el titulo secundario */
36 h2 {
37     color: blue; /* Cambiar color del texto a azul */
38     font-size: 50px; /* Cambiar tamaño de la fuente */
39     font-family: Arial, sans-serif; /* Cambiar tipo de fuente */
40     text-align: center; /* Centrar el texto */
41     margin-top: 20px; /* Espacio superior */
42 }
43
44 /* Estilos para el titulo principal */
45 h1 {
46     text-align: center; /* Centrar el texto */
47     color: yellow; /* Cambiar el color del texto */
48 }
```

Ln 124, Col 1 Spaces: 4 UTF-8 CRLF CSS Go Live

19°C Soleado 15:01 09/04/2024





```
File Edit Selection View Go Run ... < > ElJuegoDeLaOcaDelfinRojas
```

```
EXPLORER
```

```
OPEN EDITORS
```

```
Tablero.html
```

```
seleccion_color.html
```

```
EL GRAN JUEGO DE LA OCA.html
```

```
# EstiloSeleccion.css
```

```
# EstiloTablero.css
```

```
JS LogicaJuego.js
```

```
JS seleccion.js
```

```
ELJUEGODELAOCADELFINROJAS
```

```
css
```

```
# EstiloSeleccion.css
```

```
# EstiloTablero.css
```

```
imagenes
```

```
JS
```

```
JS LogicaJuego.js
```

```
JS seleccion.js
```

```
EL GRAN JUEGO DE LA OCA.html
```

```
EL JUEGO DE LA OCA.pdf
```

```
seleccion_color.html
```

```
Tablero.html
```

```
OUTLINE
```

```
TIMELINE
```

```
css > # EstiloTablero.css > ...
```

```
1 body {
```

```
2   margin: 0px; /* Elimina el margen predeterminado del cuerpo */
```

```
3   padding: 0px; /* Elimina el relleno predeterminado del cuerpo */
```

```
4   background-color: green; /* Establece el color de fondo del cuerpo */
```

```
5 }
```

```
6
```

```
7 h1 {
```

```
8   text-align: center; /* Centra el texto del encabezado */
```

```
9   color: yellow; /* Establece el color del texto del encabezado */
```

```
10  font-family: Arial, sans-serif; /* Especifica la fuente del encabezado */
```

```
11  margin-bottom: 20px; /* Agrega un margen inferior al encabezado */
```

```
12 }
```

```
13
```

```
14 #imagenPortada {
```

```
15   width: 200px; /* Ancho de la imagen de portada */
```

```
16   height: auto; /* Altura ajustada automáticamente de la imagen de portada */
```

```
17   position: absolute; /* Establece la posición absoluta */
```

```
18   top: 50px; /* Distancia desde la parte superior */
```

```
19   left: 50px; /* Distancia desde la izquierda */
```

```
20 }
```

```
21
```

```
22 #tablero {
```

```
23   position: absolute; /* Establece la posición absoluta */
```

```
24   top: 200px; /* Distancia desde la parte superior */
```

```
25   left: 200px; /* Distancia desde la izquierda */
```

```
26   width: 700px; /* Ancho del tablero */
```

```
27   height: 150px; /* Altura del tablero */
```

```
28   margin: 50px; /* Margen exterior del tablero */
```

```
29   padding: 50px; /* Relleno del tablero */
```

```
30   padding-bottom: 100px; /* Relleno inferior del tablero */
```

```
31   border: 5px outset red; /* Borde con relieve rojo */
```

```
32   text-align: center; /* Centra el contenido del tablero */
```

```
33   background-image: url('ruta/de/la/imagen.jpg');
```

```
34   background-size: contain; /* Ajusta la imagen de fondo al tamaño del contenedor */
```

```
35   background-position: center; /* Centra la imagen de fondo */
```

```
36   color: white; /* Color del texto del tablero */
```

```
37   z-index: 1; /* Establece el índice de apilamiento */
```

```
38 }
```

```
39
```

```
40 .ficha {
```

```
41   position: absolute; /* Establece la posición absoluta */
```

```
42   width: 20px; /* Ancho de la ficha */
```

```
43   height: 20px; /* Altura de la ficha */
```

```
44   border-radius: 50%; /* Borde redondeado para crear una forma de círculo */
```

```
45   z-index: 2; /* Establece el índice de apilamiento */
```

```
46 }
```

```
47 /* Establecer la posición final de la ficha */
```

```
48 #ficha.final {
```


FileEditSelectionViewGoRun

ElJuegoDeLaOcaDelfinRojas

EXPLORER

OPEN EDITORS

Tablero.html

seleccion_color.html

EL GRAN JUEGO DE LA OCA.html

EstiloSeleccion.css

EstiloTablero.css

LogicaJuego.js

seleccion.js

ELJUEGODELAOCADELFINROJAS

css

EstiloSeleccion.css

EstiloTablero.css

imagenes

JS

LogicaJuego.js

seleccion.js

EL GRAN JUEGO DE LA OCA.html

EL JUEGO DE LA OCA.pdf

seleccion_color.html

Tablero.html

OUTLINE

TIMELINE

Tablero.html

seleccion_color.html

EL GRAN JUEGO DE LA OCA.html

EstiloSeleccion.css

EstiloTablero.css

LogicaJuego.js

seleccion.js

css > # EstiloTablero.css > ...

134 #btn3 {

136 top: 400px; /* Distancia desde la parte superior */

137 height: 60px; /* Altura del botón */

138 border: 2px solid #09dd62; /* Borde de 2px sólido con color verde */

139 font-size: 16px; /* Establece el tamaño de la fuente en 16px */

140 background-color: #007bff; /* Cambia el color de fondo de los botones a azul */

141 color: #fff; /* Cambia el color del texto de los botones a blanco */

142 border: none; /* Elimina el borde de los botones */

143 border-radius: 5px; /* Aplica un radio de borde de 5px para suavizar las esquinas */

144 cursor: pointer; /* Establece el cursor al pasar sobre el botón */

145 transition: background-color 0.3s ease; /* Agrega una transición suave al cambiar el color de fondo */

146 }

147 #btn3:hover {

148 background-color: #b35100; /* Cambia el color de fondo cuando el cursor está sobre el botón */

149 }

150 #tablaJugadoresFinalizados {

151 width: 380px; /*Ancho de la Tabla*/

152 height: 50px; /*Altura de la tabla*/

153 position: absolute; /* Posición absoluta */

154 top: 50px; /* Distancia desde la parte superior */

155 left: 980px; /* Distancia desde la izquierda */

156 color: red; /* Color del texto */

157 font-family: Arial, sans-serif; /* Fuente */

158 font-size: 20px; /* Tamaño de fuente */

159 }

160

161 #tablaJugadoresFinalizados th, #tablaJugadoresFinalizados td {

162

163 text-align: left; /* Alineación del texto a la izquierda */

164 }

165

166 #tablaJugadoresFinalizados th {

167 background-color: yellow; /* Fondo amarillo para las celdas de encabezado */

168 color: green;

169 }

170

171 #tablaJugadoresFinalizados tr:nth-child(even) {

172 background-color: #ddd; /* Fondo amarillo para filas pares */

173 color: red;; /* Texto amarillo para filas pares */

174 }

175 #tablaJugadoresFinalizados tr:nth-child(odd) {

176 background-color: lightblue; /* Fondo para filas impares */

177 color: green; /* Color de texto para filas impares */

178 }

179 #tablaJugadoresFinalizados tr:hover {

180 background-color: #ddd; /* Fondo gris claro cuando se pasa el ratón por encima */

181 }

182

Ln 182, Col 1 Spaces: 4 UTF-8 CRLF CSS Go Live

19°C Soleado 15:02 09/04/2024

FileEditSelectionViewGoRun

ElJuegoDeLaOcaDelfinRojas

EXPLORER

OPEN EDITORS

ELJUEGODELAOCADELFINROJAS

OUTLINE

TIMELINE

Tablero.html

seleccion_color.html

EL GRAN JUEGO DE LA OCA.html

EstiloSeleccion.css

EstiloTablero.css

JS LogicaJuego.js

JS seleccion.js

css

EstiloSeleccion.css

EstiloTablero.css

imagenes

JS

LogicaJuego.js

seleccion.js

EL GRAN JUEGO DE LA OCA.html

EL JUEGO DE LA OCA.pdf

seleccion_color.html

Tablero.html

JS > JS LogicaJuego.js > ...

```
1 // Obtener los jugadores del almacenamiento local
2 /*Este código JavaScript está obteniendo los datos de los jugadores almacenados en el almacenamiento local del navegador.*/
3 var jugadores = JSON.parse(localStorage.getItem('jugadores'));
4
5 // Función para mostrar los jugadores y sus colores
6 function mostrarJugadores() {
7     var jugadoresContenedor = $('#jugadores');/*Selecciona el contenedor de jugadores usando su ID y
8                                                /lo almacena en la variable jugadoresContenedor*/
9     jugadoresContenedor.html(''); // Limpiar el contenido anterior
10    /*la función se asegura de que el contenedor de jugadores esté vacío al principio, para luego actualizarlo con los
11    jugadores actuales durante la ejecución de la función. Esto garantiza que no se acumulen jugadores anteriores y
12    que solo se muestren los jugadores actuales en la página.*/
13
14
15    // Iterar sobre el array de jugadores
16    $.each(jugadores, function (index, jugador) {
17        // Crear un div para representar al jugador y establecer su texto con el nombre del jugador
18        var divJugador = $('<div class="jugador">')
19            .text('Jugador ' + (index + 1) + ': ' + jugador.nombre)
20            .append($('<p>').text('Pulsar dado').addClass('pulsarDado'));
21        // Establecer el color del texto del div según el color de la ficha del jugador
22        divJugador.css('color', jugador.color);
23
24        // Crear una ficha (un div) con la clase correspondiente al color del jugador
25        var ficha = $('<div class="ficha ' + jugador.color + '">');
26
27        // Agregar la ficha al contenedor del jugador
28        divJugador.append(ficha);
29
30        // Agregar el jugador al contenedor de jugadores
31        jugadoresContenedor.append(divJugador);
32
33        // Crear un botón para lanzar el dado y configurar su estilo
34        var dadoButton = $('<button class="dadoButton" style="background: transparent; border: none; padding: 0; margin: 0;">' +
35            '' +
36            '</button>');
37
38        divJugador.append(dadoButton); // Agregar el botón de dado al jugador
39
40    });
41 }
42
43
44 var ganadorEncontrado = false; // Un booleano que indica si se ha encontrado un ganador.
45 var totalJugadores = jugadores.length; // Almacena el número total de jugadores en el juego.
46 var jugadoresfinalizados = 0; // Un contador que lleva el registro de cuántos jugadores han completado el juego. Se inicializa en 0.
47
48 //Array para almacenar los jugadores finalizados
```

Ln 47, Col 1Spaces: 4UTF-8CRLFJavaScriptGo Live

19°C Soleado

15:0209/04/2024

File

Edit

Selection

View

Go

Run

...

←

→

ElJuegoDeLaOcaDelfinRojas

EXPLORER

OPEN EDITORS

Tablero.html

seleccion_color.html

EL GRAN JUEGO DE LA OCA.html

EstiloSeleccion.css

EstiloTablero.css

JS LogicaJuego.js

seleccion.js

ELJUEGODELAOCADELFINROJAS

css

EstiloSeleccion.css

EstiloTablero.css

imagenes

JS

JS LogicaJuego.js

seleccion.js

EL GRAN JUEGO DE LA OCA.html

EL JUEGO DE LA OCA.pdf

seleccion_color.html

Tablero.html

OUTLINE

TIMELINE

JS > JS LogicaJuego.js > ...

43

44 var ganadorEncontrado = false; // Un booleano que indica si se ha encontrado un ganador.

45 var totalJugadores = jugadores.length; // Almacena el número total de jugadores en el juego.

46 var jugadoresFinalizados = 0; // Un contador que lleva el registro de cuántos jugadores han completado el juego. Se inicializa en 0.

47

48 //Array para almacenar los jugadores finalizados

49 var jugadoresFinalizadosArray = [];/* Esta linea declara una variable llamada jugadoresFinalizadosArray y la inicializa como un

50 | array vacio []. Esta variable se utiliza para almacenar información sobre los jugadores que han finalizado el juego.*/

51 var primerJugadorGanador = true;//Esta variable se utiliza para controlar si ya ha habido un jugador ganador.

52

53

54 // Función para mover la ficha del jugador actual

55 function moverFicha(jugadorIndex, resultadoDado) {

56

57 var ficha = \$('.ficha').eq(jugadorIndex);// Seleccionar la ficha del jugador actual

58 var posicionActual = parseInt(ficha.css('left'), 10); // Obtener la posición actual de la ficha en el eje horizontal

59 var nuevaPosicion = posicionActual + (resultadoDado * 37);// Calcular la nueva posición de la ficha sumando el resultado del dado por 37

60

61 // Verificar si la ficha ya llegó a la posición final

62 if (ficha.hasClass('final')) {

63 | return; // Si ya llegó, salir de la función sin hacer nada

64 }

65

66 var animacion = ficha.animate({

67 | left: nuevaPosicion + 'px'

68 }, 1000, function () // Animar la ficha hacia la nueva posición durante 1000 milisegundos (1 segundo)

69 {

70

71 // Verificar si el jugador ha pasado los 1020 pixeles

72 if (nuevaPosicion >= 1020) {

73 | // Mostrar alerta solo para el primer jugador ganador

74 | if (primerJugadorGanador) {

75 | | alert('¡Ha ganado el jugador ' + (jugadorIndex + 1) + '!');

76 | | primerJugadorGanador = false; // Cambiar el estado para que no se muestre más la alerta

77 | }

78 | // Establecer la posición de la ficha en 1030px y añadir la clase 'final'

79 | ficha.css('left', '1030px');

80 | ficha.addClass('final');

81 |

82 |

83 | // Agregar el jugador al array de jugadores finalizados

84 | jugadoresFinalizadosArray.push({

85 | | nombre: jugadores[jugadorIndex].nombre,

86 | | posicion: nuevaPosicion

87 | });

88 |

89 | // Verificar si todos los jugadores han llegado a la meta

90 | if (jugadoresFinalizadosArray.length === totalJugadores) {

91 | | alert('¡Todos los jugadores han llegado a la meta! FIN DEL JUEGO!');

Ln 47, Col 1

Spaces: 4

UTF-8

CRLF

JavaScript

Go Live

19°C

Soleado

15:03

09/04/2024

FileEditSelectionViewGoRun...<-->

ElJuegoDeLaOcaDelfinRojas

EXPLORER

OPEN EDITORS

ELJUEGODELAOCADELFINROJAS

OUTLINE

TIMELINE

Tablero.html

seleccion_color.html

EL GRAN JUEGO DE LA OCA.html

EstiloSeleccion.css

EstiloTablero.css

JS LogicaJuego.js

JS seleccion.js

EL GRAN JUEGO DE LA OCA.html

EL JUEGO DE LA OCA.pdf

seleccion_color.html

Tablero.html

JS > JS LogicaJuego.js > ...

55function moverFicha(jugadorIndex, resultadoDado) {

66var animacion = ficha.animate({

68}, 1000, function () // Animar la ficha hacia la nueva posición durante 1000 milisegundos (1 segundo)

69

89// Verificar si todos los jugadores han llegado a la meta

90if (jugadoresFinalizadosArray.length === totalJugadores) {

91alert('¡Todos los jugadores han llegado a la meta! FIN DEL JUEGO!');

92} else {

93// Mostrar mensaje de la posición del jugador

94ficha.addClass('final');

95// Si otro jugador llega después del ganador, mostrar mensaje de su posición

96alert('Jugador ' + (jugadorIndex + 1) + ' ha quedado en la posición ' + jugadoresFinalizadosArray.length + '.');

97}

98

99// Llamar a la función para rellenar la tabla

100llenarTabla();

101/*La función moverFicha se encarga de mover la ficha de un jugador en el tablero de juego según el resultado del dado

102y de manejar las acciones asociadas cuando un jugador llega a la posición final.

103Aquí está el resumen de lo que hace la función:

104-Obtención de la ficha y su posición actual: La función selecciona la ficha del jugador actual utilizando su índice y

105obtiene su posición actual en el eje horizontal.

106-Cálculo de la nueva posición: Calcula la nueva posición de la ficha sumando el resultado del dado multiplicado

107por un valor específico.

108-Verificación de la posición final: Verifica si la ficha ha llegado a la posición final. Si lo ha hecho,

109no realiza ninguna acción adicional y sale de la función.

110-Animación de la ficha: Anima la ficha para moverla hacia la nueva posición durante un periodo de tiempo determinado.

111Manejo de acciones al llegar a la posición final: Cuando la ficha llega a la posición final, realiza las siguientes acciones:

112Muestra una alerta si el jugador es el primer ganador.

113Establece la posición de la ficha en la posición final.

114Agrega la clase final a la ficha para indicar que ha llegado a la posición final.

115Agrega al jugador al array de jugadores finalizados, almacenando su nombre y posición.

116Verifica si todos los jugadores han llegado a la meta y muestra una alerta si es así.

117Muestra un mensaje con la posición del jugador si todavía hay jugadores que no han llegado a la meta.

118Llama a la función llenarTabla para actualizar la tabla con la información de los jugadores finalizados.

119En resumen, la función moverFicha gestiona el movimiento de la ficha de un jugador en el tablero de juego y las acciones asociadas

120cuando un jugador alcanza la posición final del tablero.*/

121

122}

123});

124}

125

126

127// Función para rellenar la tabla con los jugadores finalizados

128function llenarTabla() {

129var tabla = \$('#tablaJugadoresFinalizados tbody');

130tabla.empty(); // Limpiar la tabla antes de llenarla de nuevo

131

132// Iterar sobre el array de jugadores finalizados y agregar cada jugador a la tabla

133jugadoresFinalizadosArray.forEach(function (jugador, index) {

Ln 47, Col 1

Spaces: 4

UTF-8

CRLF

() JavaScript

Go Live

19°C Soleado

15:03

09/04/2024

FileEditSelectionViewGoRun

ElJuegoDeLaOcaDelfinRojas

15:0309/04/2024

EXPLORER

OPEN EDITORS

Tablero.htmlseleccion_color.htmlEL GRAN JUEGO DE LA OCA.htmlEstiloSeleccion.csscssEstiloTablero.csscssJS LogicaJuego.js JSJS seleccion.js JS

ELJUEGODELAOCADELFINROJAS

css

EstiloSeleccion.cssEstiloTablero.css

imagenes

JS

JS LogicaJuego.jsJS seleccion.js

EL GRAN JUEGO DE LA OCA.htmlEL JUEGO DE LA OCA.pdfseleccion_color.htmlTablero.html

OUTLINE

TIMELINE

JS > JS LogicaJuego.js > ...

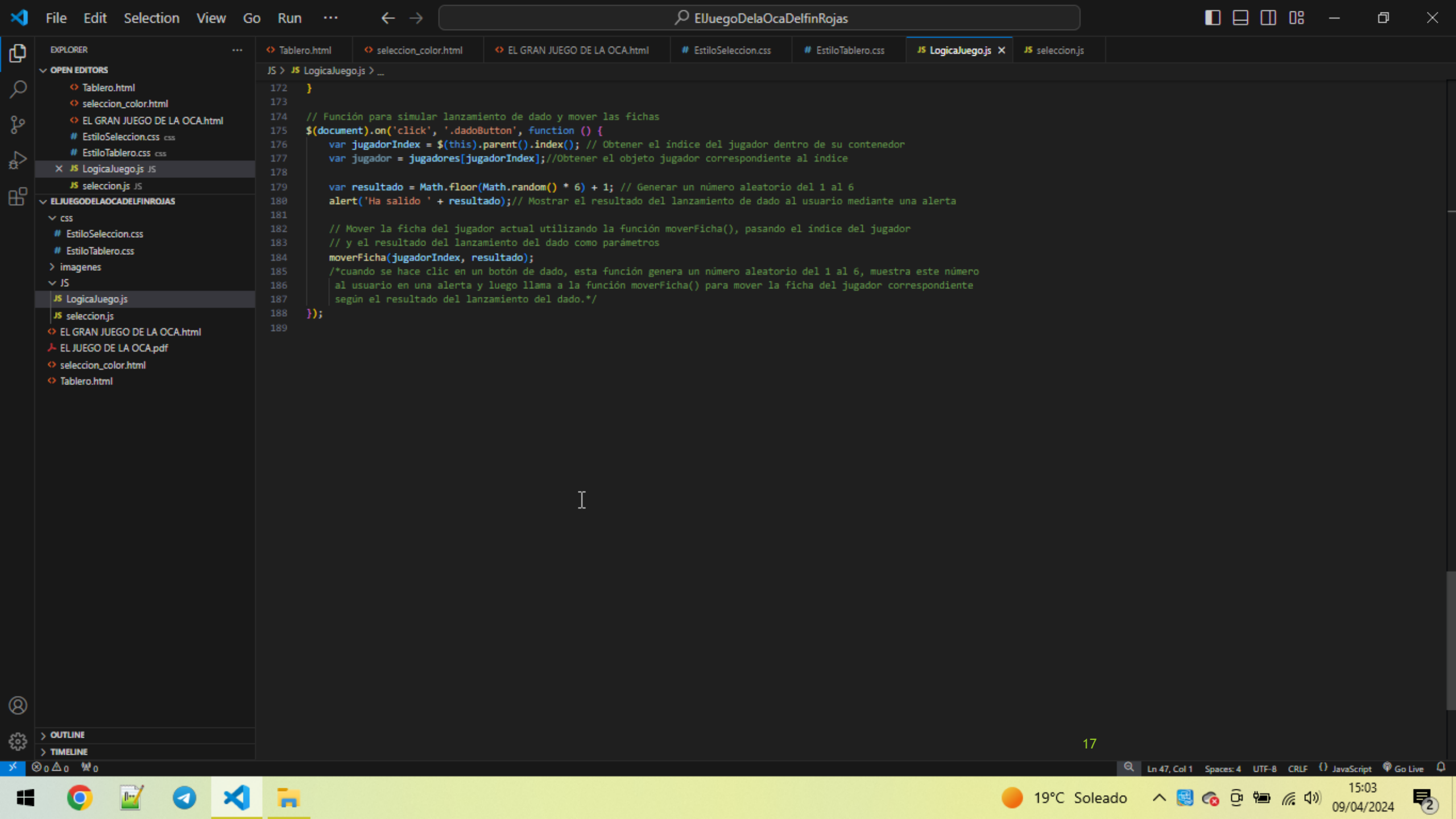
```
126
127 // Función para rellenar la tabla con los jugadores finalizados
128 function llenarTabla() {
129     var tabla = $('#tablaJugadoresFinalizados tbody');
130     tabla.empty(); // Limpiar la tabla antes de llenarla de nuevo
131
132     // Iterar sobre el array de jugadores finalizados y agregar cada jugador a la tabla
133     jugadoresFinalizadosArray.forEach(function (jugador, index) {
134         var fila = $('<tr>'); // Crear una nueva fila para cada jugador
135         fila.append($('<td>').text(jugador.nombre)); // Agregar el nombre del jugador como una celda de la fila
136         fila.append($('<td>').text(index + 1)); // Agregar el número de posición del jugador como una celda de la fila
137         tabla.append(fila); // Agregar la fila a la tabla
138     });
139     /*La función llenarTabla() toma la información de los jugadores que han finalizado el juego y la muestra en una tabla en el
140     documento HTML, mostrando el nombre del jugador y su posición en la lista de jugadores finalizados.*/
141 }
142
143
144
145 // Función para reiniciar las fichas cuando se hace clic en el botón "Reiniciar"
146 $(document).ready(function () {
147     mostrarJugadores(); // Mostrar los jugadores al cargar la página
148
149     // Establecer un controlador de eventos para el clic en el botón "Reiniciar"
150     $('#btn3').click(function () {
151         // Llamar a la función para reiniciar las fichas
152         reiniciarFichas();
153     });
154
155 });
156
157 function reiniciarFichas() {
158     var posicionInicial = 255; // Posición inicial deseada para las fichas
159     $(".ficha").stop().css({ left: posicionInicial }); // Reiniciar posición de todas las fichas
160     $(".ficha").animate({ left: posicionInicial }, 1000); // Animar todas las fichas hacia la posición inicial
161     $('#tablaJugadoresFinalizados tbody').empty(); // Vaciar la tabla de jugadores finalizados
162     var posicionInicial = 255; // Posición inicial deseada para las fichas
163     $(".ficha").stop().css({ left: posicionInicial }); // Reiniciar posición de todas las fichas
164     $(".ficha").animate({ left: posicionInicial }, 1000); // Animar todas las fichas hacia la posición inicial
165
166     // Eliminar la clase 'final' de todas las fichas para permitir que se muevan nuevamente
167     $(".ficha").removeClass('final');
168
169     /*Esta función asegura que cuando el documento HTML esté completamente cargado, se mostrarán los jugadores en la página
170     y se configurará un controlador de eventos para el botón "Reiniciar", de modo que cuando se haga clic en él,
171     se reiniciarán las fichas de los jugadores , se vaciará la tabla y los jugadores podrán volver a jugar */
172 }
173
```

Ln 47, Col 1Spaces: 4UTF-8CRLFJavaScriptGo Live

16

19°C Soleado

15:0309/04/2024



FileEditSelectionViewGoRun...<-->

ElJuegoDeLaOcaDelfinRojas

EXPLORER

OPEN EDITORS

Tablero.html

seleccion_color.html

EL GRAN JUEGO DE LA OCA.html

EstiloSeleccion.css

EstiloTablero.css

LogicaJuego.js

seleccion.js

ELJUEGODELAOCADELFINROJAS

css

EstiloSeleccion.css

EstiloTablero.css

imagenes

JS

LogicaJuego.js

seleccion.js

EL GRAN JUEGO DE LA OCA.html

EL JUEGO DE LA OCA.pdf

seleccion_color.html

Tablero.html

OUTLINE

TIMELINE

Tablero.html

seleccion_color.html

EL GRAN JUEGO DE LA OCA.html

EstiloSeleccion.css

EstiloTablero.css

LogicaJuego.js

seleccion.js

JS > JS seleccion.js > crearCamposColor

1 var opcionesSeleccionadas = []; // Array para almacenar las opciones seleccionadas

2

3 function redireccionar() {

4

5 // Obtengo el valor seleccionado del elemento con el id 'numeroJugadores'

6 var numeroJugadores = document.getElementById('numeroJugadores').value;

7

8 // Verifico si se ha seleccionado un número de jugadores

9 if (numeroJugadores !== "") {

10 // Si se ha seleccionado un número de jugadores, lo agrego al array 'opcionesSeleccionadas'

11 opcionesSeleccionadas.push(numeroJugadores);

12

13 // Redirecciono a otra página con el número de jugadores como parámetro en la URL

14 window.location.href = "seleccion_color.html?num_jugadores=" + encodeURIComponent(numeroJugadores);

15 } else {

16 // Si no se ha seleccionado un número de jugadores, mostrar un mensaje de alerta

17 alert("Por favor selecciona el número de jugadores antes de continuar.");

18 }

19

20 /*La función redireccionar() se activa cuando se hace clic en el botón de aceptar en la página. Su propósito es obtener

21 el número de jugadores seleccionado por el usuario y redirigir a otra página (seleccionar_color.html)

22 con ese número de jugadores como parámetro en la URL.*/

23 }

24 //

25

26

27 /* Array de colores permitidos :aquí se define un array llamado coloresPermitidos que contiene los nombres

28 de los colores permitidos para seleccionar. Esto proporciona una lista de opciones válidas para que los jugadores elijan

29 sus colores.*/

30 var coloresPermitidos = ['rojo', 'azul', 'verde', 'amarillo'];

31

32 /* Función para crear los campos de selección de color esta función acepta un parámetro numJugadores,

33 //que indica el número de campos de selección de color que se deben crear.*/

34 function crearCamposColor(numJugadores) {

35 // Obtener el contenedor donde se agregarán los campos de selección de color

36 var contenedor = document.getElementById('contenedorColores');

37 // Limpiar el contenido anterior del contenedor

38 contenedor.innerHTML = ''; // Limpiar el contenido anterior

39

40

41 /*Inicializar un arreglo para almacenar los colores seleccionados por cada jugador

42 Este array se utilizará para rastrear qué colores han sido seleccionados por cada jugador durante el juego.

43 Cada índice en el array corresponderá a un color específico en el array coloresPermitidos. */

44 var coloresSeleccionados = new Array(coloresPermitidos.length).fill(false);

45

46

47 //Se inicia un bucle for que iterará numJugadores veces para crear campos para cada jugador.

48 for (var i = 1; i <= numJugadores; i++) {

Ln 47, Col 97 Spaces: 4 UTF-8 CRLF JavaScript Go Live

19°C Soleado

15:03

09/04/2024

FileEditSelectionViewGoRun...<-->ElJuegoDeLaOcaDelfinRojas

EXPLORER

OPEN EDITORS

Tablero.html

seleccion_color.html

EL GRAN JUEGO DE LA OCA.html

EstiloSeleccion.css

EstiloTablero.css

LogicaJuego.js

seleccion.js

ELJUEGODELAOCADELFINROJAS

css

EstiloSeleccion.css

EstiloTablero.css

imagenes

JS

LogicaJuego.js

seleccion.js

EL GRAN JUEGO DE LA OCA.html

EL JUEGO DE LA OCA.pdf

seleccion_color.html

Tablero.html

OUTLINE

TIMELINE

Tablero.html

seleccion_color.html

EL GRAN JUEGO DE LA OCA.html

EstiloSeleccion.css

EstiloTablero.css

LogicaJuego.js

seleccion.js

JS > JS seleccion.js > crearCamposColor

34function crearCamposColor(numJugadores) {

41/*Inicializar un arreglo para almacenar los colores seleccionados por cada jugador

42Este array se utilizará para rastrear qué colores han sido seleccionados por cada jugador durante el juego.

43Cada índice en el array corresponderá a un color específico en el array coloresPermitidos. */

44var coloresSeleccionados = new Array(coloresPermitidos.length).fill(false);

45

46

47//Se inicia un bucle for que iterará numJugadores veces para crear campos para cada jugador.

48for (var i = 1; i <= numJugadores; i++) {

49// Creo un elemento label para mostrar el nombre del jugador

50var label = document.createElement('label');

51label.textContent = 'Jugador ' + i + ':';

52/*Este elemento label mostrará el nombre del jugador. El texto de este label se establece dinámicamente para mostrar

53"Jugador 1:", "Jugador 2:", etc.*/

54

55// Crear un input para que el usuario pueda ingresar el nombre del jugador

56var inputNombre = document.createElement('input');

57inputNombre.type = 'text';//Con este input el usuario pueda ingresar el nombre del jugador

58inputNombre.placeholder = 'Nombre del jugador ' + i;//Con este input e establece un marcador de posición dinámico

59inputNombre.id = 'nombreJugador' + i; //Con este input se asigna un id único a cada input para identificarlo más tarde

60

61// Crear un select para que el usuario pueda elegir el color del jugador

62var selectColor = document.createElement('select');

63selectColor.name = 'colorJugador' + i; // Establecer un nombre único para el select

64selectColor.id = 'colorJugador' + i;// Establecer un id único para el select

65

66// Agregar opciones de color al desplegable

67for (var j = 0; j < coloresPermitidos.length; j++) {

68var color = coloresPermitidos[j];

69var option = document.createElement('option');//Crear un elemento option

70option.value = color;// Establecer el valor del option como el nombre del color

71option.textContent = color;// Establecer el texto visible del option como el nombre del color

72option.classList.add(color); // Agregar clase CSS al option para darle estilo

73selectColor.appendChild(option);// Agregar el option al select

74}

75/*Se inicia un bucle for para recorrer el array coloresPermitidos, que contiene los nombres de los colores

76que el usuario puede elegir.

77En cada iteración del bucle, se crea un elemento <option> que representará una opción de color en el menú desplegable.

78Se establece el valor (value) del <option> como el nombre del color.

79Se establece el texto visible del <option> como el nombre del color.

80Se agrega una clase CSS al <option> para darle estilo.

81Finalmente, se agrega el <option> al <select>.*/*

82

83// Deshabilitar la opción del color seleccionado para evitar repeticiones

84selectColor.addEventListener('change', function (event) {

85// Obtener el índice del color seleccionado en el array coloresPermitidos

86var selectedIndex = coloresPermitidos.indexOf(event.target.value);

87// Marcar el color seleccionado como 'true' en el array coloresSeleccionados

Ln 83, Col 82

Spaces: 4

UTF-8

CRLF

JavaScript

Go Live

19°C Soleado

15:04

09/04/2024

FileEditSelectionViewGoRun...<-->

ElJuegoDeLaOcaDelfinRojas

EXPLORER

OPEN EDITORS

Tablero.html

seleccion_color.html

EL GRAN JUEGO DE LA OCA.html

EstiloSeleccion.css

EstiloTablero.css

LogicaJuego.js

seleccion.js

ELJUEGODELAOCADELFINROJAS

css

EstiloSeleccion.css

EstiloTablero.css

imagenes

JS

LogicaJuego.js

seleccion.js

EL GRAN JUEGO DE LA OCA.html

EL JUEGO DE LA OCA.pdf

seleccion_color.html

Tablero.html

OUTLINE

TIMELINE

Tablero.html

seleccion_color.html

EL GRAN JUEGO DE LA OCA.html

EstiloSeleccion.css

EstiloTablero.css

LogicaJuego.js

seleccion.js

JS > JS seleccion.js > guardarColores

34 function crearCamposColor(numJugadores) {

83 // Deshabilitar la opción del color seleccionado para evitar repeticiones

84 selectColor.addEventListener('change', function (event) {

85 // Obtener el índice del color seleccionado en el array coloresPermitidos

86 var selectedIndex = coloresPermitidos.indexOf(event.target.value);

87 // Marcar el color seleccionado como 'true' en el array coloresSeleccionados

88 coloresSeleccionados[selectedIndex] = true;

89

90

91 //Se inicia un bucle for que recorre todos los select de colores de los demás jugadores.

92 for (var k = 1; k <= numJugadores; k++) {

93 if (k !== i) {

94 var otherSelect = document.getElementById('colorJugador' + k);

95 otherSelect.querySelectorAll('option').forEach(function (option) {

96 if (option.value === event.target.value) {

97 option.disabled = true;

98 }

99 };

100 }

101 }

102 };

103

104 label.appendChild(inputNombre); // Añadir el input al label

105 label.appendChild(selectColor); // Añadir el select al label

106 contenedor.appendChild(label); // Agregar el label al contenedor de colores

107 contenedor.appendChild(document.createElement('br')); // Agregar un salto de línea al contenedor

108

109 }

110 }

111

112 // Obtener el número de jugadores de la URL

113 var parametrosURL = new URLSearchParams(window.location.search);

114 // Crear un nuevo objeto URLSearchParams para extraer los parámetros de la URL actual

115 var numJugadores = parseInt(parametrosURL.get('num_jugadores'));

116 // Obtener el valor del parámetro 'num_jugadores' de la URL y convertirlo a un número entero.

117

118

119 // Llamar a la función para crear los campos de color al cargar la página

120 window.onload = function () {

121 // Esta función se ejecutará cuando la página se haya cargado completamente

122 // La función crearCamposColor() se invoca con el número de jugadores obtenido de la URL

123 crearCamposColor(numJugadores);

124 };

125

126 // Función para guardar los nombres de jugadores y colores seleccionados

127 function guardarColores() {

128 var jugadores = []; // Crear un array para almacenar la información de cada jugador

129

Ln 128, Col 87

Spaces: 4

UTF-8

CRLF

JavaScript

Go Live

19°C

Soleado

15:04

09/04/2024

FileEditSelectionViewGoRun...<-->

ElJuegoDeLaOcaDelfinRojas

EXPLORER

OPEN EDITORS

EL JUEGO DE LA OCA DEL FIN ROJAS

OUTLINE

TIMELINE

Tablero.html

seleccion_color.html

EL GRAN JUEGO DE LA OCA.html

EstiloSeleccion.css

EstiloTablero.css

LogicaJuego.js

seleccion.js

JS > seleccion.js > guardarColores

```
124  });
125
126  // Función para guardar los nombres de jugadores y colores seleccionados
127  function guardarColores() {
128    var jugadores = []; // Crear un array para almacenar la información de cada jugador
129
130    // Obtener los nombres y colores seleccionados por cada jugador
131    for (var i = 1; i <= numJugadores; i++) {
132      // Obtener los elementos de input y select correspondientes al jugador actual
133      var nombreInput = document.getElementById('nombreJugador' + i);
134      var colorSelect = document.getElementById('colorJugador' + i);
135      // Obtener el nombre y color seleccionados por el jugador actual
136      var nombreJugador = nombreInput.value;
137      var colorJugador = colorSelect.value;
138      // Crear un objeto jugador con el nombre y color seleccionados
139      var jugador = {
140        nombre: nombreJugador,
141        color: colorJugador
142      };
143      // Agregar el objeto jugador al array de jugadores
144      jugadores.push(jugador);
145    }
146    function guardarColor() {
147      var color = document.getElementById('colorPicker').value;
148      // Aquí puedes realizar la acción para guardar el color seleccionado junto con la opción correspondiente en tu sistema
149      // Por ahora, simplemente mostraremos el color seleccionado
150      alert("Color seleccionado: " + color);
151      // Aquí podrías redirigir a otra página o hacer cualquier otra acción necesaria después de guardar el color
152    }
153    // Almacenar los datos en localStorage
154    localStorage.setItem('jugadores', JSON.stringify(jugadores));
155
156    // Redireccionar a la página de visualización de fichas
157    window.location.href = 'Tablero.html';
158    /*Esta función guardarColores() se encarga de recopilar la información (nombre y color) de cada jugador
159    y almacenarla en el almacenamiento local (localStorage).
160    Se itera a través de los jugadores utilizando un bucle for,
161    obteniendo los valores de los campos de entrada de nombre (nombreJugador) y los campos de selección de color (colorJugador)
162    para cada jugador.
163    Se crea un objeto jugador para cada jugador, que contiene el nombre y el color seleccionado.
164    Estos objetos jugador se agregan a un array llamado jugadores.
165    El array jugadores se convierte a una cadena JSON utilizando JSON.stringify() y se almacena en el almacenamiento local
166    con la clave 'jugadores'.
167    Finalmente, se redirecciona al usuario a la página de visualización de fichas (visualizar_fichas.html).*/
168
169
170
```

Ln 128, Col 87

Spaces: 4

UTF-8

CRLF

JavaScript

Go Live

19°C

Soleado

15:04

09/04/2024