20-4-2023

PAC DESARROLLO M09. DESARROLLO DE INTERFACES

J. Víctor Rodríguez Barbeira

Contenido

Estructura

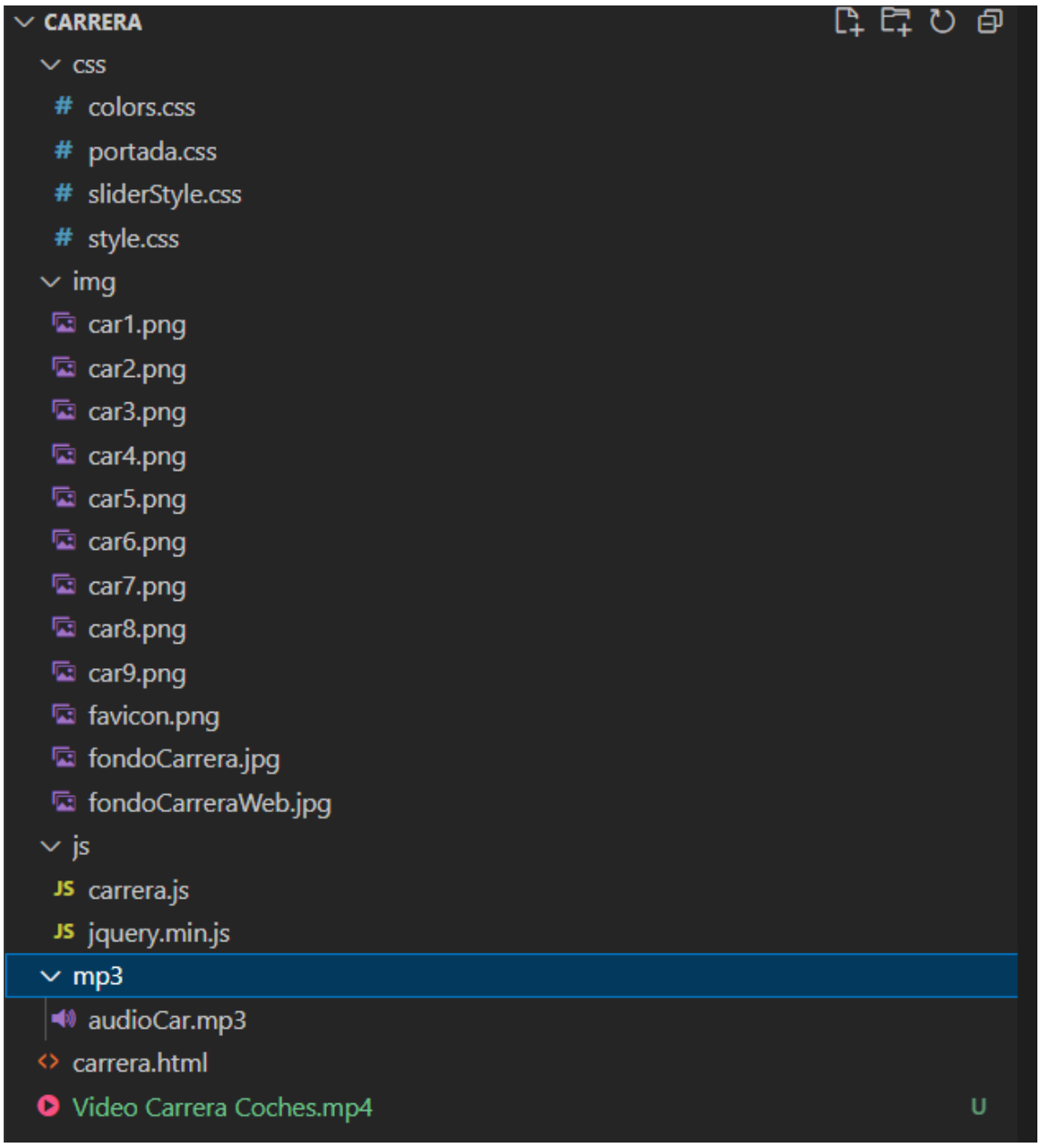
Hemos separado los archivos correspondientes a cada lenguaje en carpetas diferentes para una mejor organización.

Archivo jquery.min.js: Para garantizar el funcionamiento del juego en un futuro, aportamos la librería de jquery en local, que no sea que alguna variación de la misma en un futuro provoque un funcionamiento distinto del juego.

$(document).ready(function(): También se puede escribir de la siguiente forma ->

$(funcion(){ Es un método abreviado que hace lo mismo (fuente. Kiko Palomares)

Su utilidad radica en que no se ejecutará ningún código javascript hasta que se cargue todo el DOM.



Analizando fuentes externas he observado que podemos separar en varios archivos css para estructurar mejor nuestro programa

style.css M

```
css > # style.css > *
1   @import "colors.css";
2   @import "portada.css";
3
4
5   *{
6       margin: 0;
7       padding: 0;
8       box-sizing: border-box;
9       color: var(--color-secondary);
10  }
11
12  .container{
13      position: absoluta;
14      background-image: linear-gradient(var(--color-secondary),var(--color-primary));
15      width: 100%;
16  }
```

Además podemos definir variables con nuestros valores y así referenciarlos durante todo el programa y si necesitamos cambiarlos, hacerlo solo una vez en un único sitio.

```
css > # colors.css > :root
1   :root{
2       --color-primary: grey;
3       --color-secondary: black;
4
5       --color-goal: yellow;
6       --color-pista:rgb(62, 59, 59)
7   }
```

1. Crear un desplegable para seleccionar el número de participantes.

```html
<div class="participantes">
    <label for="participantes">Seleccione número de participantes</label>
    <select name="participantes" id="participantes">
        <option value="1">1</option>
        <option value="2">2</option>
        <option value="3">3</option>
        <option value="4">4</option>
        <option value="5">5</option>
        <option value="6">6</option>
        <option value="7">7</option>
        <option value="8">8</option>
        <option value="9">9</option>
    </select>
</div> <!-- participantes -->
```

2. Crear **un array** con tantas posiciones como participantes se hayan seleccionado. Se adjuntará una imagen de un coche en cada posición.

```javascript
function crearArray(numberElements){
    //Declaramos el array
    carList=[];

    //Inicializamos el array
    for (let i=0; i<numberElements;i++){
        carList[i]=`img/car${i+1}.png`;
    }
    return carList;
}
//Coloca los coches en la línea de salida
function muestraParticipantes(carList){
    //asigna a pista el  div id=pista
    const pista=document.getElementById("pista");
    //inserta el código HTML de img y sus coches
    pista.innerHTML="";
    for (let i=0;i<carList.length;i++){
        pista.innerHTML+=`<img class="car" id="${i}" src=${carList[i]} alt="">`;
    }
    pista.innerHTML+="</div>"

    // Variable global que tiene el array de los coches
    cars =  document.getElementsByClassName("car");
}
```

3. Todos los coches deben situarse uno debajo del otro con un **"margin-left" de 0px**.

```css
*{
    margin: 0;
    padding: 0;
    box-sizing: border-box;
    color: var(--color-secondary);
}
```

```css
.car{
    position: relative;
    width: 100px;
    display: flex;
    flex-direction: column;
}
```

4. Indicar un valor fijando una línea de meta.

```css
#pista{
    display: block;
    height: 440px;
    width: 1000px;
    background: var(--color-pista);
    border-right: 10px solid var(--color-goal);

}
```

5. Se trata de ir incrementando el margen desplazando los coches hacia la derecha (con el método **animate**) hasta rebasar la línea de meta (valor indicado).

```
function generaTiemposCorredores(){
    //Establece aleatoriamente los tiempos de llegada  para cada participante
    //Los anima a partir del método animate
    for (let i=0;i<cars.length;i++){    //cars[].length variable Global devuelve el número de participantes
        //2 variables locales recogen el id del elemento html y el tiempo de desplazamiento
        let ident=`#${i}`;
        let time=(Math.random(1)+1)*1000;
        $(ident).animate({"left":"1000px"},time);

        //guardamos en un array Global los tiempos para cada participante
        times[i]=time;
    }
}
```

6. Se crearán **dos botones**, "Iniciar" y "Reiniciar". Al cargar la página, solamente debe mostrarse el botón inicio, reiniciar debe estar oculto. Sin embargo, cada vez que pulsamos uno de ellos, desparece y se muestra el otro.

```
// B O T Ó N   I N I C I A R
$("#iniciar").click(function(){
    //Actualiza los botones que se han de mostrar
    $("#iniciar").hide();
    $("#reiniciar").show();
    $("#parar").show();
```

7. Al pulsar el botón inicio, los coches empezarán a avanzar. En cambio, si se pulsa el botón "Reiniciar" los coches tendrán que volver al puesto inicial.

```
// B O T Ó N   P A R A R
$( "#parar" ).click(function() {
    //Detiene las animaciones jQuery animate
    $( ".car" ).stop();

    //Actualiza los estados de los botones
    $("#iniciar").show();
    $("#reiniciar").show();
    $("#parar").hide();
});
```

```
// B O T Ó N   R E I N I C I A R
$( "#reiniciar" ).click(function() {
    //Animación animate para retroceder los coches
    $( ".car" ).animate({ left: "0px" }, 3000 );

    //Actualiza el estado de los botones
    $("#iniciar").show();
    $("#reiniciar").hide();
    $("#parar").hide();

});
```

8. El avance de cada coche se realiza añadiendo de forma aleatoria valores entre 1 y 10.

Hemos cambiado los valores para adaptar el desplazamiento de los vehículos a unos valores más visuales.

El método Math.random(1) nos generará valores aleatorios no enteros entre 0 y 1 sin incluir éste último.

```
function generaTiemposCorredores(){
    //Establece aleatoriamente los tiempos de llegada  para cada participante
    //Los anima a partir del método animate
    for (let i=0;i<cars.length;i++){    //cars[].length variable Global devuelve el número de participante
        //2 variables locales recogen el id del elemento html y el tiempo de desplazamiento
        let ident=`#${i}`;
        let time=(Math.random(1)+1)*1000;
        $(ident).animate({"left":"1000px"},time);

        //guardamos en un array Global los tiempos para cada participante
        times[i]=time;
    }
}
```

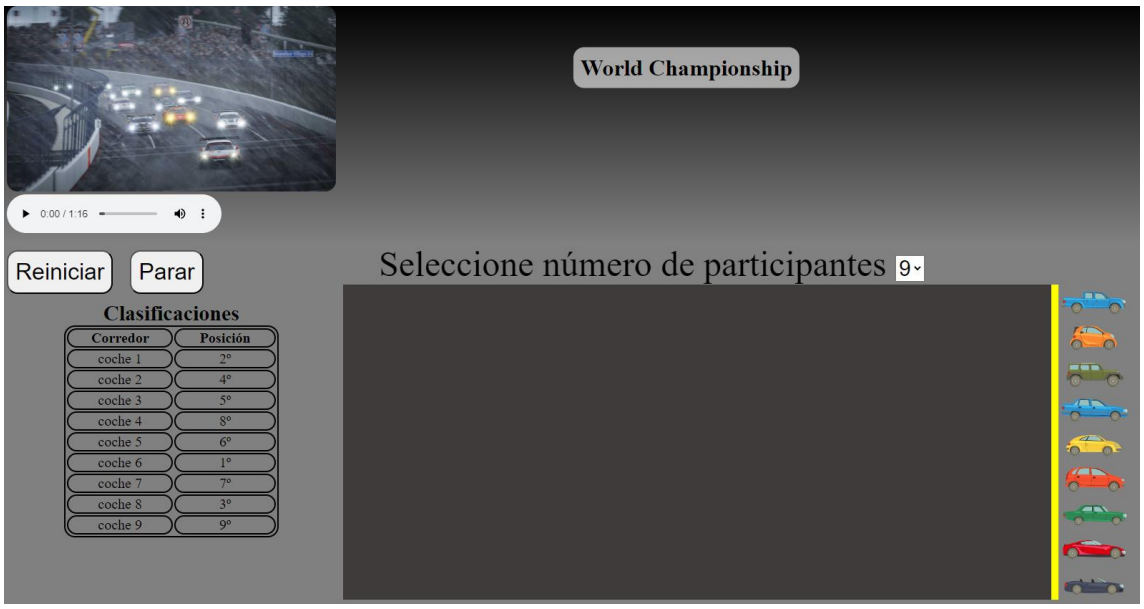
9. El primer coche que llegue al valor fijado como línea de meta, será el ganador.

Para establecer la posición de cada participante contamos cuantas veces hay un tiempo mejor que el propio, si es 0 es el primero, si hay 1 es el segundo, etc.

```
let lose=0;
for (let i=0;i<cars.length;i++){
    lose=0;
    for (let j=0;j<cars.length;j++){
        if (times[j]<times[i])  lose++;
    }
    posicion[i]=lose;
```

10. Mostrar una tabla con las posiciones de cada participante.

11. Se debe diseñar el entorno del aplicativo web, cuanto más creativo mejor.



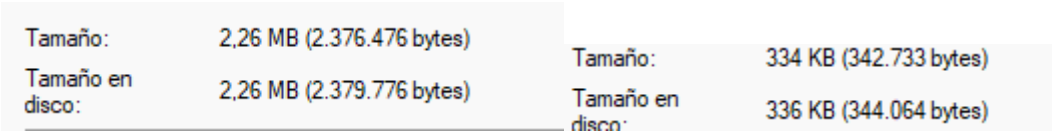
Modificar el tamaño de las imágenes

[1]Recomendaciones para realizar la optimización. La resolución que generalmente utilizamos en un ordenador suele ser de 96 ppp (resolución 1.600 x 1.200), por lo que no sería recomendable usar mayores resoluciones, ya que aumentaría mucho el tamaño del archivo.

Conviene guardar las imágenes originales en **formato BMP**, **TIFF** o **JPEG** sin realizar compresión y realizar copias en otros formatos con los que podamos tratarlas para publicarlas en la web.

Mediante Gimp hemos podido reducir el tamaño de la imagen de la portada de 2,26 MB a 336KB

https://www.gimp.org/downloads/



[1] Fuente. Documentación Ilerna

Licencias

Como imagen para el juego hemos utilizado la siguiente. Su licencia es creative commons y se nos indica que debemos mencionar al autor.

"Project CARS 2 / Rainy Race For The Finish" by Stefans02 is licensed under CC BY 2.0.

Project CARS 2 / Rainy Race For The Finish
by Stefans02

Get this image

How to use

Visit the image's website to download and use it. Make sure to credit the creator by showing the attribution information where you are sharing your work.

License

This image was marked with a CC BY 2.0 license:

Credit the creator.

Credit the creator

Rich Text | HTML | Plain text

"Project CARS 2 / Rainy Race For The Finish" by Stefans02 is licensed under CC BY 2.0.