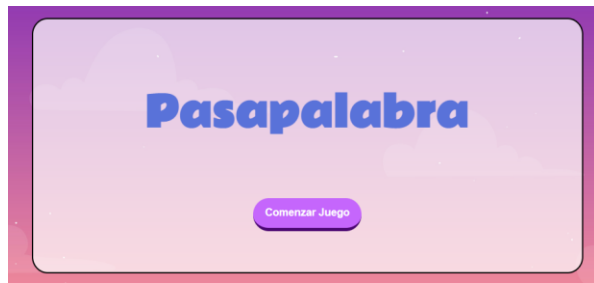


Pasapalabra

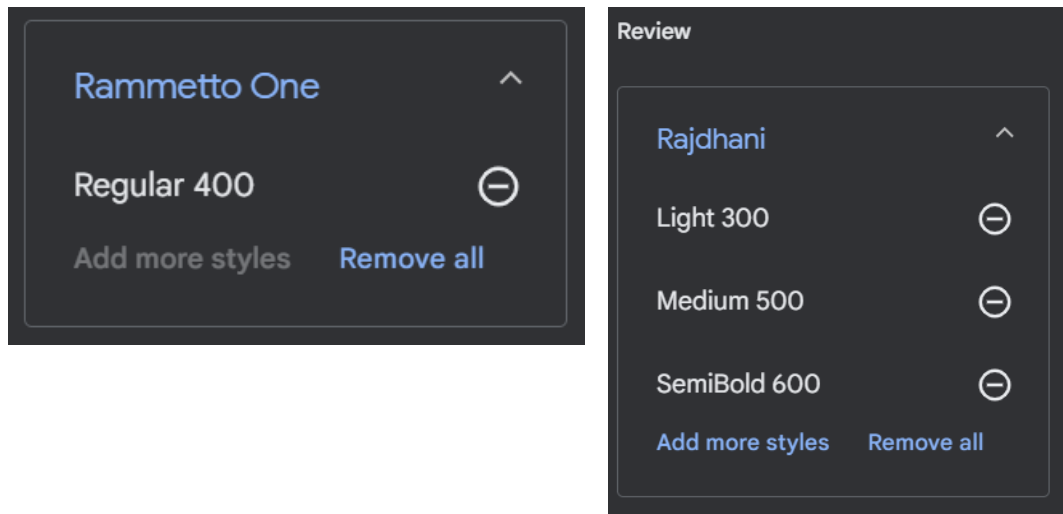


Realizaremos nuestro primer juego con JavaScript, para ello prepararemos nuestro html con el siguiente código:



```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <link rel="stylesheet" href="css/styles.css">
  <title>Pasapalabra</title>
</head>
<body>
  <section id="pantalla-inicial">
    <h1>Pasapalabra</h1>
    <button id="comenzar">Comenzar Juego</button>
  </section>
  <script src="js/script.js"></script>
</body>
</html>
```

Ahora realizaremos nuestro estilo css, lo primero será importar la fuente con la cual trabajaremos:



Y la colocaremos en nuestro archivo css:

```
1 /* Importamos las fuentes con las cuales trabajaremos */
2 @import url('https://fonts.googleapis.com/css2?family=Rajdhani:wght@300;500;600&family=Rammetto+One&display=swap');
```

A continuación, daremos estilo a nuestra web:

```
*{
  box-sizing: border-box;
}

body{
  font-family: 'Rajdhani';
  background: url(..img/fondo.jpg);
  background-attachment: fixed;
  background-size: cover;
}

#pantalla-inicial{
  display: block;
  max-width: 700px;
  margin: 50px auto;
  text-align: center;
  background: rgba(255,255,255,.7);
  padding: 30px;
  border: 2px solid #000;
  border-radius: 20px;
}

#pantalla-inicial h1{
  font-family: 'Rammetto One';
  font-size: 50px;
  color: #5972d9;
}
```

```

button{
  background: #c566fc;
  border: none;
  color: #fff;
  font-weight: bold;
  padding: 10px 15px;
  cursor: pointer;
  margin: 0 20px;
  border-radius: 30px;
  border-bottom: 4px solid #4b0872;
}

button:hover{
  border-bottom: 4px solid transparent;
  background-color: #7705b9;
}

```

Nos quedara de la siguiente manera nuestra pantalla inicial:



Ahora iniciaremos la sección del juego:

```

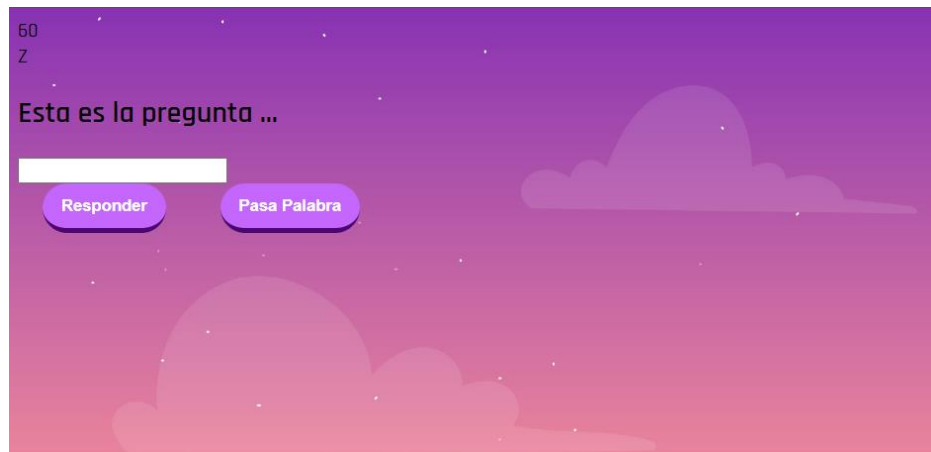
<!-- Seccion del juego -->
<section id="pantalla-juego">
  <!-- Aqui se generaran de forma dinamica las letras -->
  <div class="container">
    <span id="tiempo">60</span>
  </div>
  <div class="contenedor-pregunta">
    <span id="letra-pregunta">Z</span>
    <h2 id="pregunta">Esta es la pregunta ... </h2>
    <input type="text" id="respuesta">
    <div class="botones">
      <button id="responder">Responder</button>
      <button id="pasar">Pasa Palabra</button>
    </div>
  </div>
</section>
</body>
</html>

```

Para verificar como va nuestra sección, iremos a nuestro css y en #pantalla-inicial cambiaremos la propiedad de **display: block** por **none**.

```
#pantalla-inicial{
  display: none;
  max-width: 700px;
  margin: 50px auto;
  text-align: center;
  background: rgba(255,255,255,.7);
  padding: 30px;
  border: 2px solid #000;
  border-radius: 20px;
}
```

Y nos quedara de la siguiente manera en nuestro navegador:



Ahora aplicaremos el css para nuestra sección del juego:

```
/* Pantalla inicial del juego */

#pantalla-juego{
  display: block;
}

/* Esta clase se agregara dinamicamente */

.pregunta-actual{
  background-color: #000;
  color: #fff;
}

/* Esta clase se agregara dinamicamente cuando haya respondido correctamente una pregunta */

.bien-respondida{
  background-color: #3edbbb;
  color: #fff;
}

/* Esta clase se agregara dinamicamente cuando haya respondido incorrectamente una pregunta */

.mal-respondida{
  background-color: #fb797b;
  color: #fff;
}
```

```

.contenedor-pregunta{
  background-color: rgba(255,255,255,.7);
  text-align: center;
  max-width: 700px;
  margin: auto;
  border: 2px solid #000;
  border-radius: 20px;
  padding: 0 20px;
}

.contenedor-pregunta #pregunta{
  margin: 0;
  font-weight: normal;
}

.contenedor-pregunta input{
  display: inline-block;
  margin: 10px 0;
  font-size: 25px;
  padding: 10px;
  text-align: center;
  width: 60%;
  border-radius: 30px;
  border: none;
  outline: none;
  border: 2px solid #ccc;
}

```

```

.contenedor-pregunta #letra-pregunta{
  border: 2px solid #000;
  display: inline-block;
  color: #000;
  font-weight: bold;
  padding: 5px 10px;
  font-size: 30px;
  position: relative;
  bottom: 25px;
  background: #fff;
}

.container{
  position: relative;
  width: 255px;
  height: 255px;
  margin: 50px auto;
  border: 2px solid #000;
  border-radius: 50%;
  background: rgba(255, 255, 255, .7);
  display: flex;
  justify-content: center;
  align-items: center;
}

.container #tiempo{
  font-size: 50px;
  font-weight: bold;
}

```

```
/* La siguiente clase es para los círculos de cada letra */
```

```
.circle{  
  position: absolute;  
  width: 50px;  
  height: 50px;  
  border: 1px solid #000;  
  background: #fff;  
  color: #5972d9;  
  border-radius: 50%;  
  display: flex;  
  justify-content: center;  
  align-items: center;  
  font-size: 20px;  
  font-weight: bold;  
  text-transform: uppercase;  
}
```

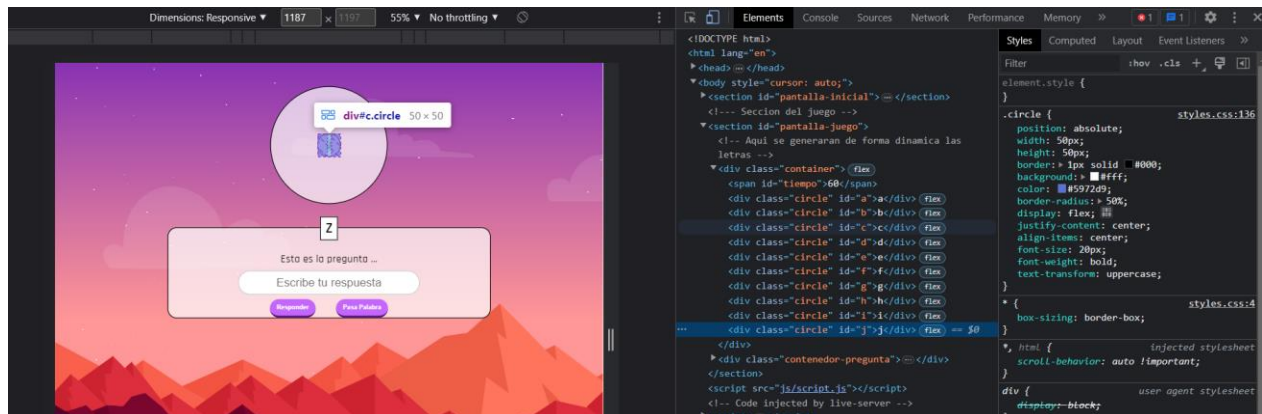
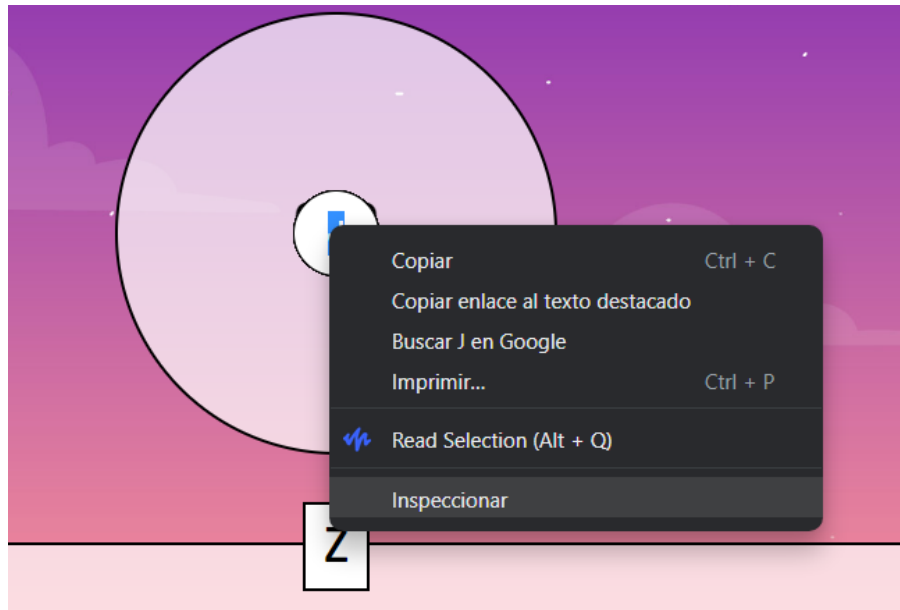
Ahora realizaremos nuestro js:

```
//Total de preguntas del juego  
  
const TOTAL_PREGUNTAS = 10;  
  
//Creamos las letras de la A a la J  
  
const container = document.querySelector('.container');  
for (let i = 1; i ≤ TOTAL_PREGUNTAS; i++){  
  const circle = document.createElement('div');  
  circle.classList.add('circle');  
  circle.textContent = String.fromCharCode(i + 96);  
  circle.id = String.fromCharCode(i + 96);  
  container.appendChild(circle);  
}
```

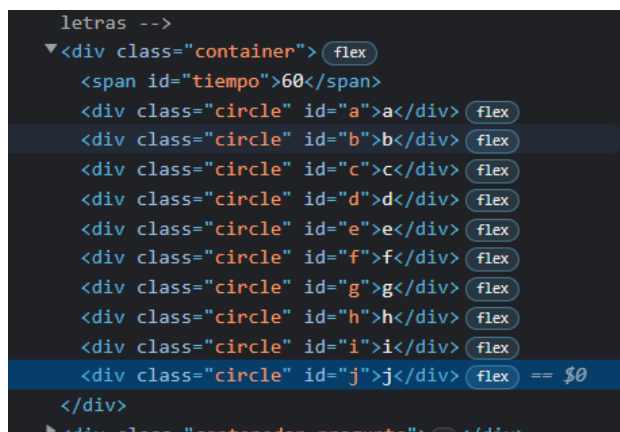
Nos debe quedar de la siguiente manera nuestra vista web:



Para verificar que se han generado seleccionamos la letra J del círculo y daremos clic derecho inspeccionar y revisamos que se estén construyendo correctamente los divs de nuestras letras:



Nos debe generar los divs de las letras automáticamente:



Ahora agregaremos las siguientes líneas de código dentro para agregar los círculos:

```
const container = document.querySelector(".container");
for (let i = 1; i ≤ TOTAL_PREGUNTAS; i++) {
  const circle = document.createElement("div");
  circle.classList.add("circle");
  circle.textContent = String.fromCharCode(i + 96);
  circle.id = String.fromCharCode(i + 96).toUpperCase();
  container.appendChild(circle);

  const angle = ((i - 1) / TOTAL_PREGUNTAS) * Math.PI * 2 - (Math.PI / 2);
  const x = Math.round(95 + 120 * Math.cos(angle));
  const y = Math.round(95 + 120 * Math.sin(angle));
  circle.style.left = `${x}px`;
  circle.style.top = `${y}px`;
}
```

Asignamos un "id" al círculo, que es la misma letra del contenido de texto del círculo, luego calculamos la posición de cada círculo en un círculo completo, utilizando una fórmula trigonométrica para obtener la posición X e Y del círculo en relación con el centro del contenedor. Finalmente, se establecen las posiciones X e Y del círculo utilizando estilos CSS en línea.

Para que nos quede de esta manera:



Ahora modificamos nuestro css en la clase #pantalla-inicial: none por Block

```
#pantalla-inicial{
  display: block;
  max-width: 700px;
  margin: 50px auto;
  text-align: center;
  background: rgba(255, 255, 255, .7);
  padding: 30px;
  border: 2px solid #000;
  border-radius: 20px;
}
```

Y en la clase #pantalla-juego:block cambiamos por none

```
#pantalla-juego{
  display: none;
}
```

Ahora en nuestro js agregaremos un evento para que al dar clic sobre el botón comenzar nos muestre el inicio del juego:

```
let comenzar = document.getElementById('comenzar');
comenzar.addEventListener('click', () => {
  document.getElementById('pantalla-inicial').style.display = 'none';
  document.getElementById('pantalla-juego').style.display = 'block';
});
```

De esta manera al presionar clic nos llevará a la pantalla inicial del juego, cuando refrescamos la pagina volverá al inicio y así sucesivamente.

Ahora cargaremos las variables para llevar nuestro tiempo del juego estas las colocaremos al inicio :

```
//Total de preguntas del juego
const TOTAL_PREGUNTAS = 10;

//Variables para controlar el tiempo
const timer = document.getElementById('tiempo');
//Tiempo del juego en segundos
const TIEMPO_DEL_JUEGO = 60;
//Variable que indica el tiempo restante
let tiempoRestante = TIEMPO_DEL_JUEGO;
//Variable que maneja el contador de tiempo
let countdown;

//Creamos las letras de la A a la J
```

Agregamos la función al momento de dar clic en comenzar que se ejecute el contador:

```
let comenzar = document.getElementById('comenzar');
comenzar.addEventListener('click', () => {
  document.getElementById('pantalla-inicial').style.display = 'none';
  document.getElementById('pantalla-juego').style.display = 'block';

  //Iniciamos el contador de tiempo
  iniciarContador();
});
```

Y creamos la función iniciarContador():

```
//Funcion para iniciar el contador de tiempo
function iniciarContador() {
  countdown = setInterval(()=>{
    //restamos un segundo al tiempo restante
    tiempoRestante--;
    //Actualizamos el tiempo en el DOM
    timer.innerText = tiempoRestante;

    //Si el tiempo llega a 0, detiene el contador
    if(tiempoRestante === 0){
      clearInterval(countdown);
    }
  }, 1000);
}
```

De esta manera el contador iniciara desde 60 a contar.



Ahora crearemos la base de datos de las preguntas que saldrán en nuestro juego:

```
//Total de preguntas del juego
const TOTAL_PREGUNTAS = 10;

//Base de datos para las preguntas del juego
const bd_juego = [
  {
    id: 'A',
    pregunta: "¿Empresa reconocida que se dedica a los servidores?",
    respuesta: "amazon"
  },
  {
    id: 'B',
    pregunta: "¿Termino en ingles que hace referencia a una copia de seguridad?",
    respuesta: "backup"
  },
  {
    id: 'C',
    pregunta: "¿Nombre de la memoria que almacena temporalmente los datos de la computadora?",
    respuesta: "cache"
  },
  {
    id: 'D',
    pregunta: "¿Archivo que controla los perifericos que se conectan a la computadora?",
    respuesta: "driver"
  },
  {
    id: 'E',
    pregunta: "¿Mezclar los datos para protegerlos como medida de seguridad, es decir convertir texto normal a texto cifrado?",
    respuesta: "encriptar"
  },
  {
    id: 'F',
    pregunta: "¿Famosa red social creada por Mark Zuckerberg?",
    respuesta: "facebook"
  },

```

```
  {
    id: 'G',
    pregunta: "¿Lenguaje de programacion creado por Google?",
    respuesta: "go"
  },
  {
    id: 'H',
    pregunta: "¿Lenguaje utilizado para estructurar las paginas web?",
    respuesta: "html"
  },
  {
    id: 'I',
    pregunta: "¿Aspecto que presentan los programas tras su ejecucion mediante el cual ejercemos la comunicacion con estos?",
    respuesta: "interfaz"
  },
  {
    id: 'J',
    pregunta: "¿Lenguaje de programacion con el cual se diseño el sistema operativo android?",
    respuesta: "java"
  }
]
```

Agregamos el llamado de la función cargarPreguntas dentro de comenzar:

```
let comenzar = document.getElementById('comenzar');
comenzar.addEventListener('click', () => {
  document.getElementById('pantalla-inicial').style.display = 'none';
  document.getElementById('pantalla-juego').style.display = 'block';

  //Iniciamos el contador de tiempo
  iniciarContador();
  //Cargamos las Preguntas
  cargarPreguntas();
});
```

Creamos la variable para controlar la pregunta actual:

```
//Total de preguntas del juego

const TOTAL_PREGUNTAS = 10;

//Variable que controla la pregunta actual.
//Comienza en -1 porque la primera pregunta es la 0
let preguntaActual = -1;

//Base de datos para las preguntas del juego
const bd_juego = [
```

Y creamos la función para las preguntas:

```
    }, 1000);
}

//Funcion para cargar las preguntas
function cargarPreguntas() {
    preguntaActual++;
    //controlo si he llegado al final de las preguntas, para iniciar de nuevo.
    if (preguntaActual ≥ TOTAL_PREGUNTAS) {
        preguntaActual = 0;
    }
}
```

Haremos un arreglo para poder mantener los estados de las preguntas, si la ha contestado o no, 0 indica que no la ha contestado y 1 que sí; de esta manera cargaremos las 10:

```
//Comienza en -1 porque la primera pregunta es la 0
let preguntaActual = -1;

//Arreglo con los estados de las preguntas
let estadoPreguntas = [0, 0, 0, 0, 0, 0, 0, 0, 0, 0];

//Base de datos para las preguntas del juego
const bd_juego = [
```

Ahora debemos controlar que dentro del arreglo existen preguntas por contestar con estado 0

```
//Funcion para cargar las preguntas
function cargarPreguntas() {
    preguntaActual++;
    //controlo si he llegado al final de las preguntas, para iniciar de nuevo.
    if (preguntaActual ≥ TOTAL_PREGUNTAS) {
        preguntaActual = 0;
    }
    //controlo si la pregunta ya fue respondida
    if (estadoPreguntas.indexOf(0) ≥ 0) {
        while (estadoPreguntas[preguntaActual] === 1) {
            preguntaActual++;
            if (preguntaActual ≥ TOTAL_PREGUNTAS) {
                preguntaActual = 0;
            }
        }
    }
}
```

Ahora buscamos las preguntas y hacemos el cargue respectivo en la vista web

```
//Funcion para cargar las preguntas
function cargarPreguntas() {
  preguntaActual++;
  //controlo si he llegado al final de las preguntas, para iniciar de nuevo.
  if (preguntaActual ≥ TOTAL_PREGUNTAS) {
    preguntaActual = 0;
  }
  //controlo si la pregunta ya fue respondida
  if (estadoPreguntas.indexOf(0) ≥ 0){
    //Ahora buscamos dentro cual es la que esta sin responder es decir que esta en 0
    while (estadoPreguntas[preguntaActual] === 1) {
      preguntaActual++;
      if (preguntaActual ≥ TOTAL_PREGUNTAS) {
        preguntaActual = 0;
      }
    }
    //Ahora si buscamos la pregunta en la base de datos
    document.getElementById('letra-pregunta').textContent = bd_juego[preguntaActual].id;
    document.getElementById('pregunta').textContent = bd_juego[preguntaActual].pregunta;
    let letra = bd_juego[preguntaActual].id;
    document.getElementById(letra).classList.add('pregunta-actual');
  }else{
    //Si ya no hay preguntas sin responder, terminamos el juego
    clearInterval(countdown);
    //mostrarPantallaFinal();
  }
}
```



Ahora detectamos cada vez que haya un cambio en el input para ver cuando se presiona enter y controlamos si lo que ingreso es correcto o no.

```
//detectamos el cambio del input
let respuesta = document.getElementById("respuesta");
respuesta.addEventListener("keyup", function (event) {
    //Detecto si se presiona la tecla enter
    if (event.key === "Enter") {
        if (respuesta.value === "") {
            alert("Debe ingresar una respuesta");
            return;
        }
        //Obtengo la respuesta ingresada

        let respuestaIngresada = respuesta.value.toLowerCase();
        controlarRespuesta(respuestaIngresada);
    }
});
```

Nos saldrá el mensaje del alert:



Creo la función para controlarRespuesta:

```
function controlarRespuesta(respuestaIngresada){  
  //Controlo la respuesta correcta  
  if (respuestaIngresada == bd_juego[preguntaActual].respuesta){  
    alert("Respuesta Correcta");  
  }  
}
```

Para comprobarlo escribimos en el input y revisamos



Ahora agregamos cantidadAcertadas y comentamos el alert:

```
function controlarRespuesta(respuestaIngresada){  
  //Controlo la respuesta correcta  
  if (respuestaIngresada == bd_juego[preguntaActual].respuesta){  
    //alert("Respuesta Correcta");  
    cantidadAcertadas++;  
  }  
}
```

Cargamos la variable en el inicio para llevar la cantidad:

```
1 //Total de preguntas del juego  
2  
3 const TOTAL_PREGUNTAS = 10;  
4  
5 //Variable que me lleva la cantidad de preguntas respondidas correctamente  
6 let cantidadAcertadas = 0;  
7  
8  
9 //Variable que controla la pregunta actual.
```

Actualizo el estado de la pregunta actual a 1, para indicar que ya está respondida:

```
function controlarRespuesta(respuestaIngresada){
  //Controlo la respuesta correcta
  if (respuestaIngresada == bd_juego[preguntaActual].respuesta){
    //alert("Respuesta Correcta");
    cantidadAcertadas++;

    //cambio el estado de la pregunta a 1
    estadoPreguntas[preguntaActual] = 1;

    let letra = bd_juego[preguntaActual].id;
    document.getElementById(letra).classList.remove("pregunta-actual");
    document.getElementById(letra).classList.add("bien-respondida");
  }
}
```

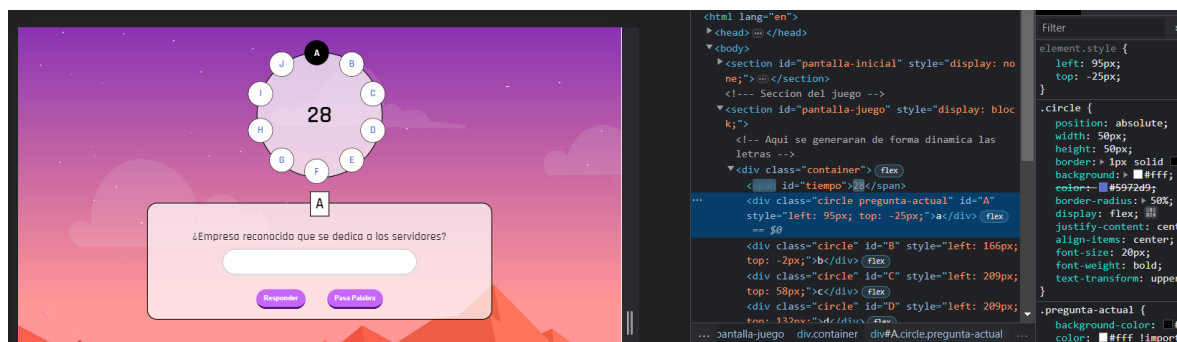
Agregamos en nuestro css !important a las clases, de esta manera al iniciar estará en color negro y cambiara según la respuesta correcta o incorrecta:

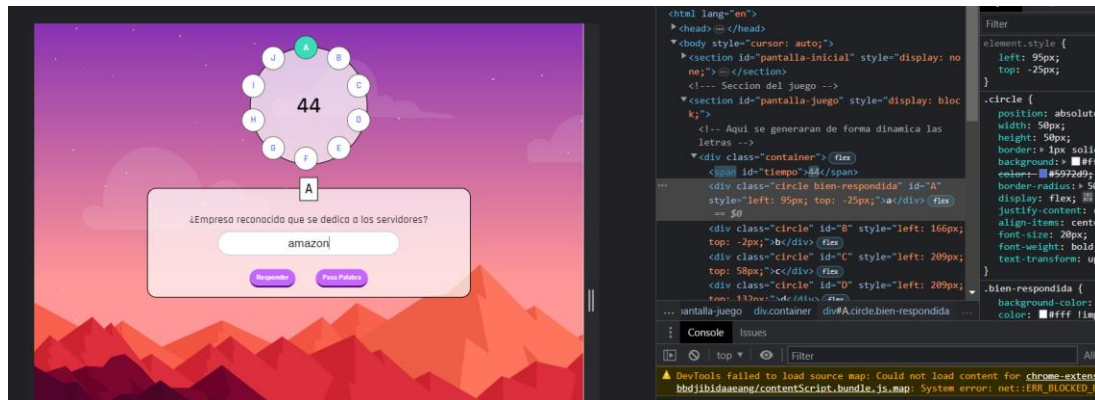
```
.pregunta-actual{
  background-color: #000 !important;
  color: #fff !important;
}

/* Esta clase se agregara dinamicamente cuando haya respondido correctamente una pregunta */
.bien-respondida{
  background-color: #3edbbb !important;
  color: #fff !important;
}

/* Esta clase se agregara dinamicamente cuando haya respondido incorrectamente una pregunta */
.mal-respondida{
  background-color: #fb797b !important;
  color: #fff !important;
}
```

Verificamos que al momento de ejecutar el código la clase circle cuando se conteste correctamente cambiará su estado a bien respondida y se pondrá de color verde.





Ahora aplicamos de la misma manera con un else para referirnos si contesta mal que se cambien las propiedades:

```
function controlarRespuesta(respuestaIngresada) {
  //Controlo la respuesta correcta
  if (respuestaIngresada == bd_juego[preguntaActual].respuesta) {
    //alert("Respuesta Correcta");
    cantidadAcertadas++;

    //cambio el estado de la pregunta a 1
    estadoPreguntas[preguntaActual] = 1;

    let letra = bd_juego[preguntaActual].id;
    document.getElementById(letra).classList.remove("pregunta-actual");
    document.getElementById(letra).classList.add("bien-respondida");
  } else {
    estadoPreguntas[preguntaActual] = 1;
    let letra = bd_juego[preguntaActual].id;
    document.getElementById(letra).classList.remove("pregunta-actual");
    document.getElementById(letra).classList.add("mal-respondida");
  }
}
```

Al hacer la verificación notaremos que al escribir la palabra incorrecta y presionar enter se nos colocara de color rojo:



Por último, limpiamos el input y cargamos la función para la próxima pregunta:

```
function controlarRespuesta(respuestaIngresada) {  
  //Controlo la respuesta correcta  
  if (respuestaIngresada == bd_juego[preguntaActual].respuesta) {  
    //alert("Respuesta Correcta");  
    cantidadAcertadas++;  
  
    //cambio el estado de la pregunta a 1  
    estadoPreguntas[preguntaActual] = 1;  
  
    let letra = bd_juego[preguntaActual].id;  
    document.getElementById(letra).classList.remove("pregunta-actual");  
    document.getElementById(letra).classList.add("bien-respondida");  
  } else {  
    estadoPreguntas[preguntaActual] = 1;  
    let letra = bd_juego[preguntaActual].id;  
    document.getElementById(letra).classList.remove("pregunta-actual");  
    document.getElementById(letra).classList.add("mal-respondida");  
  }  
  //limpio el input  
  respuesta.value = "";  
  //carga la siguiente pregunta  
  cargarPreguntas();  
}
```

Al hacer la verificación debemos notar que al escribir la respuesta se pondrá en verde o rojo el círculo y continuará a la siguiente pregunta:



Ahora agregamos el código para activar el botón de pasa palabra y que nos permita saltar la pregunta:

```
//Boton para pasar de pregunta sin contestar
let pasar = document.getElementById("pasar");
pasar.addEventListener("click", function () {
    let letra = bd_juego[preguntaActual].id;
    document.getElementById(letra).classList.remove("pregunta-actual");
    cargarPreguntas();
});
```

De esta manera al darle clic sobre el botón pasar palabra saltaremos la pregunta; seguimos con el botón para responder la pregunta al darle clic:

```
//Boton para responder la pregunta
let responder = document.getElementById("responder");
responder.addEventListener("click", function () {
    if (respuesta.value === "") {
        alert("Debe ingresar una respuesta");
        respuesta.focus();
        respuesta.value = "";
        return;
    }
    let respuestaIngresada = respuesta.value.toLowerCase();
    controlarRespuesta(respuestaIngresada);
    respuesta.focus();
});
```

Ahora realizamos en el html la pantalla final:

```
9         </div>
10      </section>
11      <!-- Seccion pantalla final -->
12      <section id="pantalla-final">
13          <h1>Resumen del Juego</h1>
14          <h3>Acertadas</h3>
15          <span id="acertadas">10</span>
16          <h3 id="puntaje">40% de acierto</h3>
17          <button id="reiniciar">Jugar de Nuevo</button>
18      </section>
19
20      <script src="js/script.js"></script>
21  </body>
22  </html>
```

Ahora agregamos nuestro css, vamos a modificar la clase #pantalla-inicial en la propiedad display y colocaremos por el momento en none para poder ver la pantalla inicial:

```
#pantalla-inicial{
  display: none;
  max-width: 700px;
  margin: 50px auto;
  text-align: center;
  background: rgba(255, 255, 255, .7);
```

Y procedemos a colocar nuestra clase #pantalla-final

```
/* Clase para la pantalla final */
#pantalla-final{
  display: block;
  max-width: 700px;
  margin: auto;
  text-align: center;
  background: rgba(255, 255, 255, .7);
  padding: 30px;
  border: 2px solid #000;
  border-radius: 20px;
```

Y ya se empieza a mostrar de esta manera:



Ahora agregamos la clase para nuestro span:

```
#pantalla-final span{
  background: #000;
  width: 100px;
  height: 100px;
  margin: auto;
  color: #fff;
  display: flex;
  justify-content: center;
  align-items: center;
  font-size: 60px;
  font-weight: bold;
  border-radius: 50%;
  margin-bottom: 30px;
}
```

Quedando de la siguiente manera:



Modificamos de nuevo la clase #pantalla-final en display por none:

```
#pantalla-final{
  display: none;
  max-width: 700px;
  margin: auto;
  text-align: center;
}
```

Y ahora modificamos #pantalla-inicial

```
#pantalla-inicial{
  display: block;
  max-width: 700px;
  margin: 50px auto;
}
```

Agregamos nuestro js:

```
//Mostrar Pantalla Final
function mostrarPantallaFinal() {
    document.getElementById("acertadas").textContent = cantidadAcertadas;
    document.getElementById("puntaje").textContent = (cantidadAcertadas * 100) / TOTAL_PREGUNTAS + "% de aciertos";
    document.getElementById("pantalla-juego").style.display = "none";
    document.getElementById("pantalla-final").style.display = "block";
}
```

Y habilitamos mostrarPantallaFinal en cargarPreguntas:

```
document.getElementById(tetra).classList.add("pregunta-actual");
} else {
    //Si ya no hay preguntas sin responder, terminamos el juego
    clearInterval(countdown);
    mostrarPantallaFinal();
}
```

También en iniciarContador:

```
function iniciarContador() {
    countdown = setInterval(() => {
        //restamos un segundo al tiempo restante
        tiempoRestante--;
        //Actualizamos el tiempo en el DOM
        timer.innerText = tiempoRestante;

        //Si el tiempo llega a 0, detiene el contador
        if (tiempoRestante === 0) {
            clearInterval(countdown);
            mostrarPantallaFinal();
        }
    }, 1000);
}
```

De esta manera ya tendremos nuestro juego casi terminado, ahora programamos el botón para jugar de nuevo:

```
//Boton para reiniciar el juego
let reiniciar = document.getElementById("reiniciar");
reiniciar.addEventListener("click", function () {
    //Reiniciamos las variables
    cantidadAcertadas = 0;
    preguntaActual = -1;
    tiempoRestante = TIEMPO_DEL_JUEGO;
    timer.innerText = tiempoRestante;
    estadoPreguntas = [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0];

    //Quitamos las clases de los círculos
    let circulos = document.getElementsByClassName("circle");
    for (i=0; i<circulos.length; i++) {
        circulos[i].classList.remove("pregunta-actual");
        circulos[i].classList.remove("bien-respondida");
        circulos[i].classList.remove("mal-respondida");
    }

    //Mostramos la pantalla inicial
    document.getElementById("pantalla-final").style.display = "none";
    document.getElementById("pantalla-juego").style.display = "block";
    respuesta.value = "";
    respuesta.focus();
    iniciarContador();
    cargarPreguntas();
});
```

Así hemos finalizado nuestro juego y ya podemos hacer las respectivas pruebas para verificarlo.