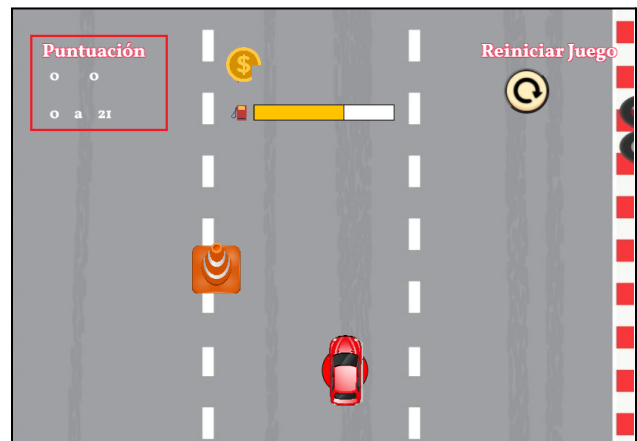


TABLA DE PUNTUACIÓN



¿Cuál fue nuestro objetivo para este módulo?

Aprendimos a crear una tabla de puntuación para mostrar la puntuación en la pantalla.

¿Qué logramos en clase el día de hoy?

- Agregamos una tabla de puntuación para mostrar la puntuación del jugador en la pantalla.

¿Qué conceptos y bloques de código vimos hoy?

- p5.DOM.elements.
- Tabla de puntuación.

¿Cómo hicimos las actividades?

1. Creamos elementos de texto para la tabla de puntuación en **constructor()** en **game.js**.

```
constructor() {  
  this.resetTitle = createElement("h2");  
  this.resetButton = createButton("");  
  
  this.leadeboardTitle = createElement("h2");  
  this.leader1 = createElement("h2");  
  this.leader2 = createElement("h2");  
}
```

2. Mostramos las propiedades de la tabla de puntuación usando **handleElement()**.

```
handleElements() {  
    form.hide();  
    form.titleImg.position(40, 50);  
    form.titleImg.class("gameTitleAfterEffect");  
  
    this.resetTitle.html("Reiniciar juego");  
    this.resetTitle.class("resetText");  
    this.resetTitle.position(width / 2 + 200, 40);  
  
    this.resetButton.class("resetButton");  
    this.resetButton.position(width / 2 + 230, 100);  
  
    this.leadeboardTitle.html("Tabla de puntuación");  
    this.leadeboardTitle.class("resetText");  
    this.leadeboardTitle.position(width / 3 - 60, 40);  
  
    this.leader1.class("leadersText");  
    this.leader1.position(width / 3 - 50, 80);  
  
    this.leader2.class("leadersText");  
    this.leader2.position(width / 3 - 50, 130);  
}
```

3. Agregamos los estilos para **resetText** y **LeadersText** dentro de **style.css**

```

JS Game.js M X JS sketch.js # style.css X <> index.html M
# style.css > ...
86
87 .resetText{
88     font-size: 25px;
89     color: white;
90     font-family: vollkorn;
91     text-align: center;
92     text-shadow: -1px 1px 0 #ff1744,
93                 1px 1px 0 #ff1744,
94                 1px -1px 0 #ff1744,
95                 -1px -1px 0 #ff1744;
96 }
97
98 .leadersText{
99     font-size: 20px;
100    color: white;
101    font-family: vollkorn;
102    text-align: center;
103 }

```

Creamos una función **showLeader()**.

- Establecimos leader1 y leader2 según la posición del jugador.

```

showLeaderboard() {
    var leader1, leader2;
    var players = Object.values(allPlayers);
    if (
        (players[0].rank === 0 && players[1].rank === 0) ||
        players[0].rank === 1
    ) {
        // &emsp; Esta etiqueta se utiliza para mostrar cuatro espacios
        leader1 =
            players[0].rank +
            "&emsp;" +
            players[0].name +
            "&emsp;" +
            players[0].score;

        leader2 =
            players[1].rank +
            "&emsp;" +
            players[1].name +
            "&emsp;" +
            players[1].score;
    }
}

```

- Utilizamos `.html()` para mostrar las variables `leader1` y `leader2`.

```
    this.leader1.html(leader1);  
    this.leader2.html(leader2);  
}
```

OUTPUT:



¿Qué sigue?

En la siguiente clase, daremos posiciones a los jugadores según su desempeño en el juego de carreras de autos.

AMPLÍEN SU CONOCIMIENTO:

1. Guarden el siguiente enlace para obtener más información sobre las consultas de la base de datos de Firebase:

<https://firebase.google.com/docs/database/web/read-and-write>