index.html

```
                                <script>
        //CApturamos el elemento canvas y el creamos el plano 2d
                    let canvas = document.getElementById("canvas");
                          let ctx = canvas.getContext("2d");

     //Creamos el Objeto pelota con sus atributos y el objeto barra
                  let pelota = new Pelota(ctx, 25, "blue", 200, 0);
                        let barra = new Barra(ctx, 100, 15);
   //Creamos dos objetos Image para cambiar el color del objeto pelota
                           //y barra por imagenes
                           let imagenBarra = new Image();
                          let imagenPelota = new Image();
                        imagenPelota.onload = function () {
                             pelota.imagen = imagenPelota;
                                        };
                        imagenBarra.onload = function () {
                              barra.imagen = imagenBarra;
                                      iniciar();
                                        };
                          imagenBarra.src = 'barra.png';
                        imagenPelota.src = 'asteroide.png';

        //Esta función se llama a si misma y se ejecuta en bucle,
  //la función dibuja la pelota, la barra y comprueba la posición de la
            //barra y la pelota para verificar las colisiones
                               function iniciar(){
                                   pelota.draw();
                      barra.dibujarBarra(200, 450, 100, 10, 20);
                                pelota.mover(barra);
                             requestAnimationFrame(iniciar)


                                        }

  //Captura el teclado para comprobar si se pulsan las teclas para mover
                                  //la barra
                          function desplazarBarra(event) {
                            if(event.key === "ArrowRight"){
                               barra.moverBarra("Derecha");
                          }else if(event.key === "ArrowLeft"){
                               barra.moverBarra("Izquierda");
                    }else if(event.key === "p" || event.key === "P"){
```

```
                alert("PAUSA");
                    }

                }
//Captura la ventana y agrega un evento al pulsar cualquier tecla del
        //teclado la cual llama a la función desplazarBarra
            window.addEventListener('keydown', desplazarBarra);
                </script>
```

Pelota.js

```
                class Pelota{
        //Constructor para el objeto pelota con sus atributos
            constructor(contexto, radio, color, x, y){
                this.contexto = contexto;
                this.radio = radio;
                this.color = color;
                    this.x = x;
                  this.y = y + 20;
                this.xVelocidad = 5;
                this.yVelocidad = -5;

                this.imagen = new Image();
            this.imagen.src = 'asteroide.png';
                        }
  //Función que comprueba la posición de la pelota para manejar las
                        //colisiones
                        mover(objeto) {
          //Maneja las colisiones horizontales con la caja
       if (this.x + this.radio > this.contexto.canvas.width || this.x
                - this.radio < 0) {
              this.xVelocidad = -this.xVelocidad;
                        }

            // Maneja las colisiones con la barra
                        if (
                this.y + this.radio > objeto.y &&
          this.y - this.radio < objeto.y + objeto.alto &&
                this.x + this.radio > objeto.x &&
           this.x - this.radio < objeto.x + objeto.largo
                        ) {
              this.yVelocidad = -this.yVelocidad;
                        }
```

```
                // Maneja las colisión con la parte superior
                        if (this.y - this.radio < 0) {
                        this.yVelocidad = -this.yVelocidad;
                                        }

                // Maneja las colisión con la parte inferior
            if (this.y + this.radio > this.contexto.canvas.height) {
                      this.x = this.contexto.canvas.width / 2;
                              this.y = 0 + this.radio;
                                 alert("YOU LOSE");
  //Si toca la parte inferior vuelve a dibujar la pelota como en el
                        //inicio a modo de reset
                                    this.draw();
                                        }

                           this.x += this.xVelocidad;
                           this.y += this.yVelocidad;
                                     }


 //FUnción que dibuja la pelota y le agrega una imagen como fondo
                                  draw(){
                    this.contexto.clearRect(0, 0, 500, 500);
                           this.contexto.beginPath();
       this.contexto.arc(this.x, this.y, this.radio, 0, 2 * Math.PI);

                        // Dibujar la imagen como fondo
          this.contexto.drawImage(this.imagen, this.x - this.radio,
        this.y - this.radio, this.radio * 2, this.radio * 2);

          this.contexto.fillStyle = "transparent";  // Establecer el
                  color de relleno como transparente
                             this.contexto.fill();
                          this.contexto.closePath();

                                    }

                                  }
```

```
                        class Barra{
            //COnstructor con los atributos de la barra
                  constructor(contexto, largo, alto){
                        this.contexto = contexto;
                           this.largo = largo;
                            this.alto = alto;
                              this.x = 200;
                              this.y = 450;
                           this.velocidad = 50;

                        this.imagen = new Image();
                     this.imagen.src = 'barra.png';
                                 }

//FUnción para redondear los bordes de la barra, con la imagen de fondo
      //esta función no se aprecia pero con un color de fondo si
              redondearBordes(x, y, width, height, radius) {
                        this.contexto.beginPath();
                    this.contexto.moveTo(x + radius, y);
          this.contexto.arcTo(x + width, y, x + width, y + height,
                              radius);
         this.contexto.arcTo(x + width, y + height, x, y + height,
                              radius);
             this.contexto.arcTo(x, y + height, x, y, radius);
              this.contexto.arcTo(x, y, x + width, y, radius);
                        this.contexto.closePath();
                                 }
                  //Función para dibujar la barra
                           dibujarBarra() {
        // Llamar a la función para dibujar el rectángulo con bordes
                              redondeados
       this.redondearBordes(this.x, 450, this.largo, this.alto, 5); //
      Puedes ajustar el valor del radio según tus preferencias
        this.contexto.drawImage(this.imagen, this.x, 450, this.largo,
                        this.alto);

                                 }
                                /*
                           dibujarBarra(){
          this.contexto.fillRect(this.x, 450, this.largo, this.alto);
                               }*/
```

```
  //Si la tecla capturada es la flecha hacia la derecha se envia como
//parámetro la palabra “Derecha” de lo contrario “Izquierda” siempre y
    //cuando las teclas pulsadas sean flecha derecha o izquierda.
                        moverBarra(direccion){
                    if(direccion == "Derecha" && this.x <
              this.contexto.canvas.width - this.largo){
                                this.x += this.velocidad;
              }else if(direccion == "Izquierda" && this.x > 0){
                                this.x -= this.velocidad;
                                          }
                                        }
                                      }
```