



Néstor stiven Aguilar garzón - 6000423

Preguntas chatgtp



crea el array de un icosaedro para THREE JS



Para crear un array de un icosaedro en THREE.js, podemos seguir un enfoque similar al del dodecaedro. Primero, necesitamos definir los vértices del icosaedro en un arreglo. Luego, podemos usar estos vértices para definir las caras del icosaedro.

Aquí está el código para crear el arreglo de un icosaedro:

less

Copy code

```
// Definir los vértices del icosaedro
const t = (1 + Math.sqrt(5)) / 2;
const icosavertices = [ [-1, t, 0],
  [1, t, 0],
  [-1, -t, 0],
  [1, -t, 0],
  [0, -1, t],
  [0, 1, t],
  [0, -1, -t],
  [0, 1, -t],
  [t, 0, -1],
  [t, 0, 1],
  [-t, 0, -1],
  [-t, 0, 1]
];

// Definir las caras del icosaedro usando los vértices
const icosafaces = [ [0, 11, 5],
  [0, 5, 1],
  [0, 1, 7],
  [0, 7, 10],
  [0, 10, 11],
  [1, 5, 9],
  [5, 11, 4],
  [11, 10, 2],
  [10, 7, 6],
  [7, 1, 8],
  [3, 9, 4],
  [3, 4, 2],
  [3, 2, 6],
  [3, 6, 8],
  [3, 8, 9],
  [4, 9, 5],
  [2, 4, 11],
  [6, 2, 10],
  [8, 6, 7],
  [9, 8, 1]
];
```

```

nima Copy code

<!DOCTYPE html>
<html>
  <head>
    <title>Dodecaedro</title>
    <script src="https://cdn.jsdelivr.net/npm/three.js@r128/three.min.js"></script>
    <style>
      body {
        margin: 0;
      }
      canvas {
        display: block;
      }
    </style>
  </head>
  <body>
    <script>
      // Crear la escena
      const scene = new THREE.Scene();

      // Crear la cámara
      const camera = new THREE.PerspectiveCamera(75, window.innerWidth / window.innerHeight, 0.1, 1000);
      camera.position.set(0, 0, 20);

      // Crear el renderizador
      const renderer = new THREE.WebGLRenderer();
      renderer.setSize(window.innerWidth, window.innerHeight);
      document.body.appendChild(renderer.domElement);

      // Crear los materiales
      const material = new THREE.MeshBasicMaterial({ color: 0xffffff, wireframe: true });

      // Crear la geometría
      const geometry = new THREE.DodecahedronGeometry(5, 0);

      // Crear la malla y añadirla a la escena
      const mesh = new THREE.Mesh(geometry, material);
      scene.add(mesh);

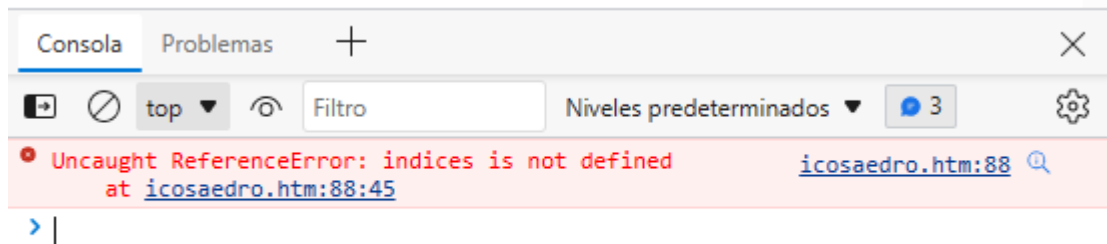
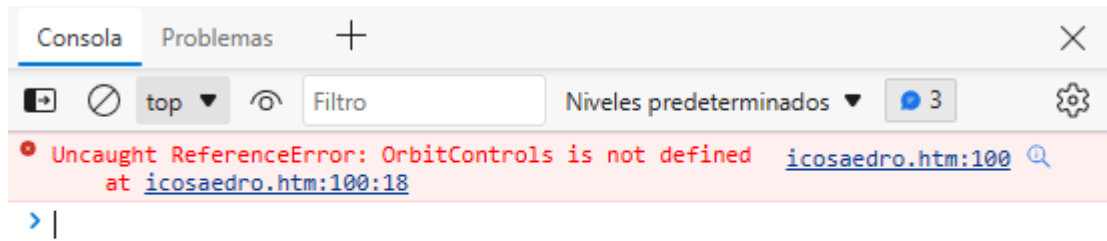
      // Animar la escena
      function animate() {
        requestAnimationFrame(animate);
        mesh.rotation.x += 0.01;
        mesh.rotation.y += 0.01;
        renderer.render(scene, camera);
      }

      animate();
    </script>
  </body>
</html>

```

ERRORES

1. el navegador web no ha podido encontrar una referencia a ese objeto o función. Esto podría ser porque el archivo JavaScript que define "OrbitControls" no se ha cargado correctamente, o porque hay un error en el propio archivo JavaScript que impide que se defina "OrbitControls".



- 2.

Es posible que hayas escrito mal el nombre de la variable o que la hayas eliminado accidentalmente en alguna parte del código.