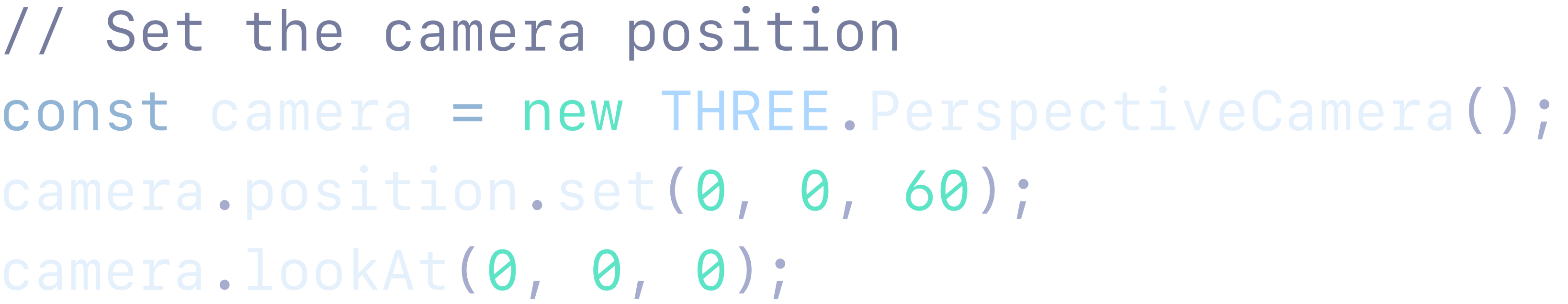
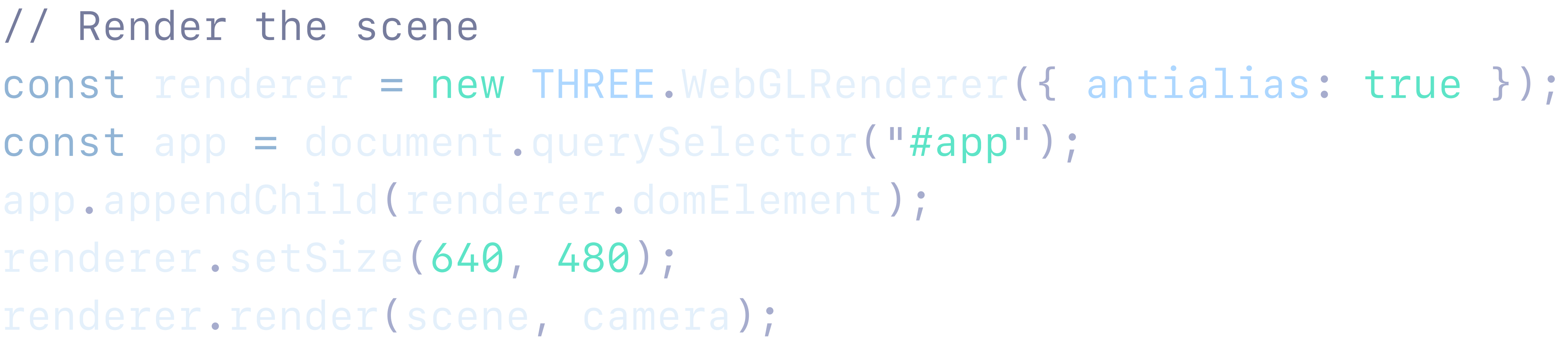
```
import * as THREE from "three";
```

```
// Create a triangle shape, draw lines between (x, y) coordinates
const triangle = new THREE.Shape();
triangle.lineTo(0, 10);
triangle.lineTo(10, -10);
triangle.lineTo(0, -10);
const geometry = new THREE.ShapeGeometry(triangle);
geometry.center();
```

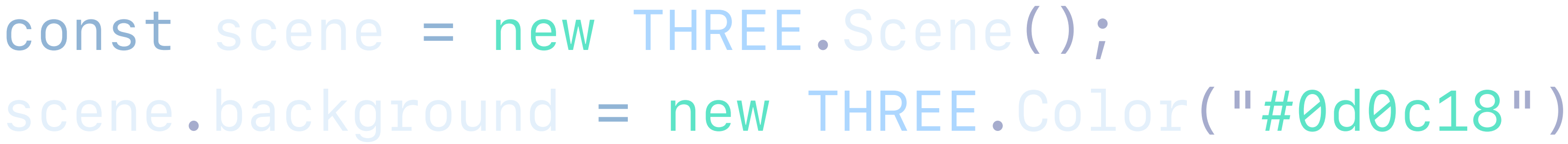
```
// Create Mesh
const material = new THREE.MeshNormalMaterial({ side: THREE.DoubleSide });
const mesh = new THREE.Mesh(geometry, material);
scene.add(mesh);
```



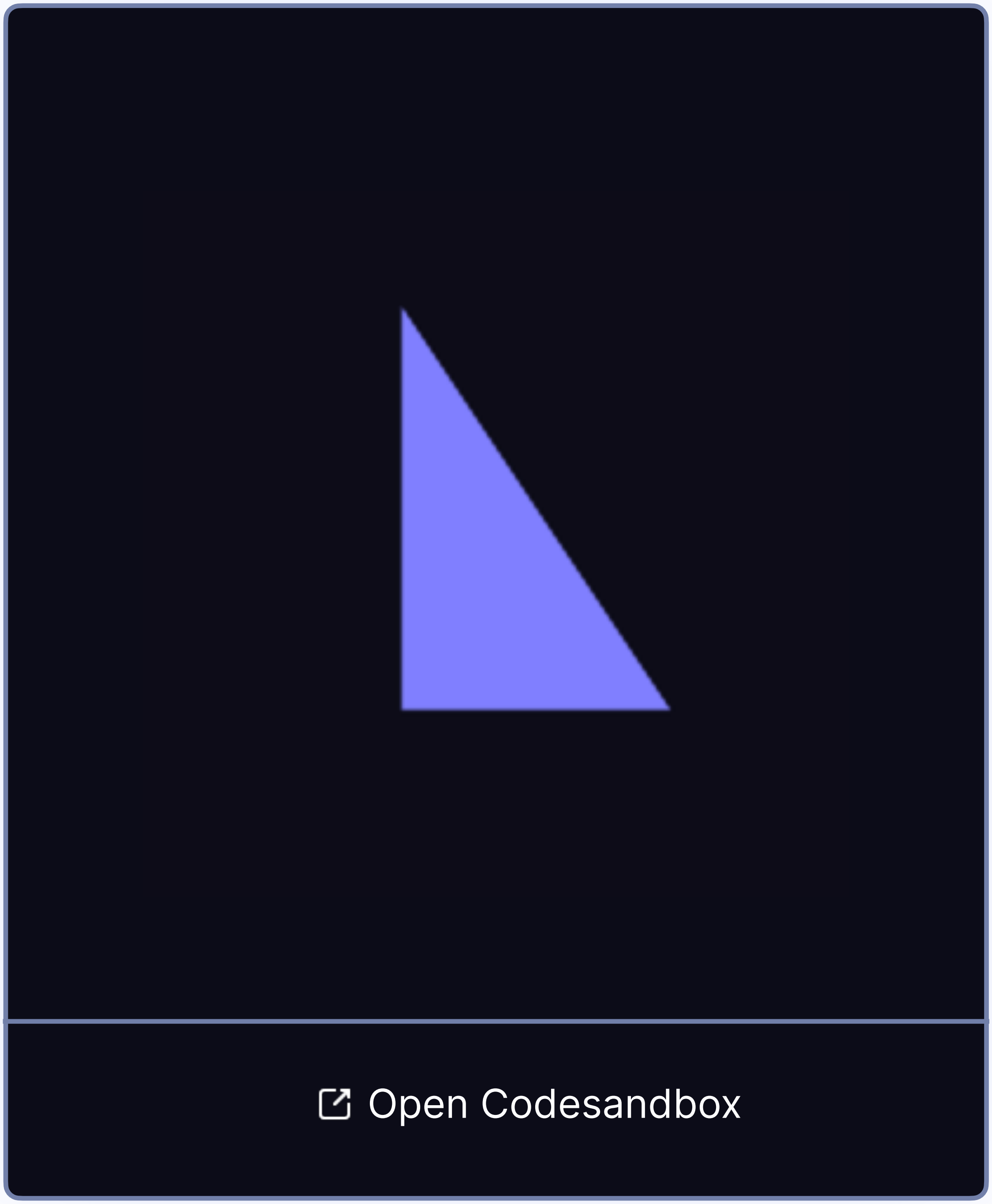
```
// Set the camera position
const camera = new THREE.PerspectiveCamera();
camera.position.set(0, 0, 60);
camera.lookAt(0, 0, 0);
```



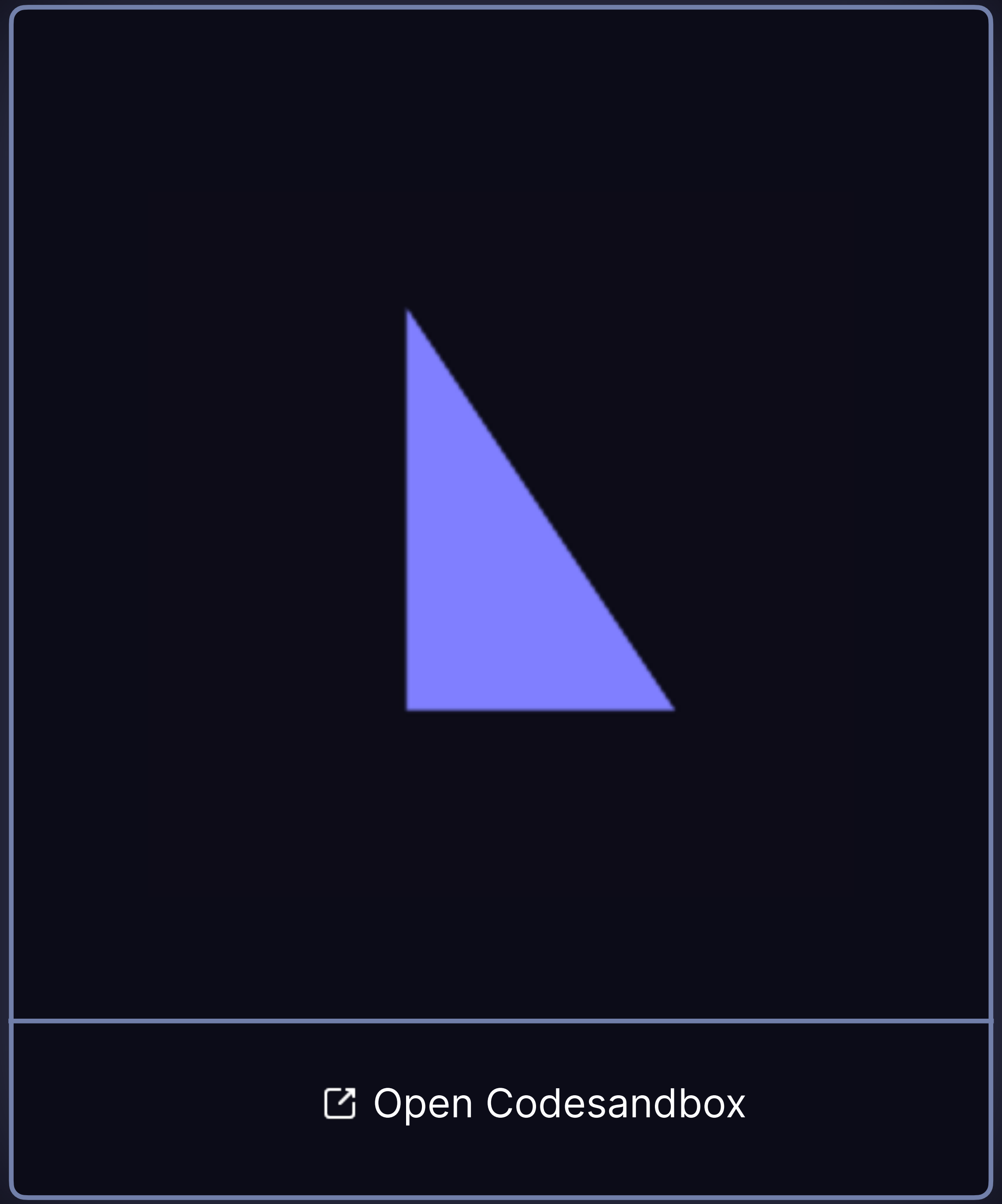
```
// Render the scene
const renderer = new THREE.WebGLRenderer({ antialias: true });
const app = document.querySelector("#app");
app.appendChild(renderer.domElement);
renderer.setSize(640, 480);
renderer.render(scene, camera);
```

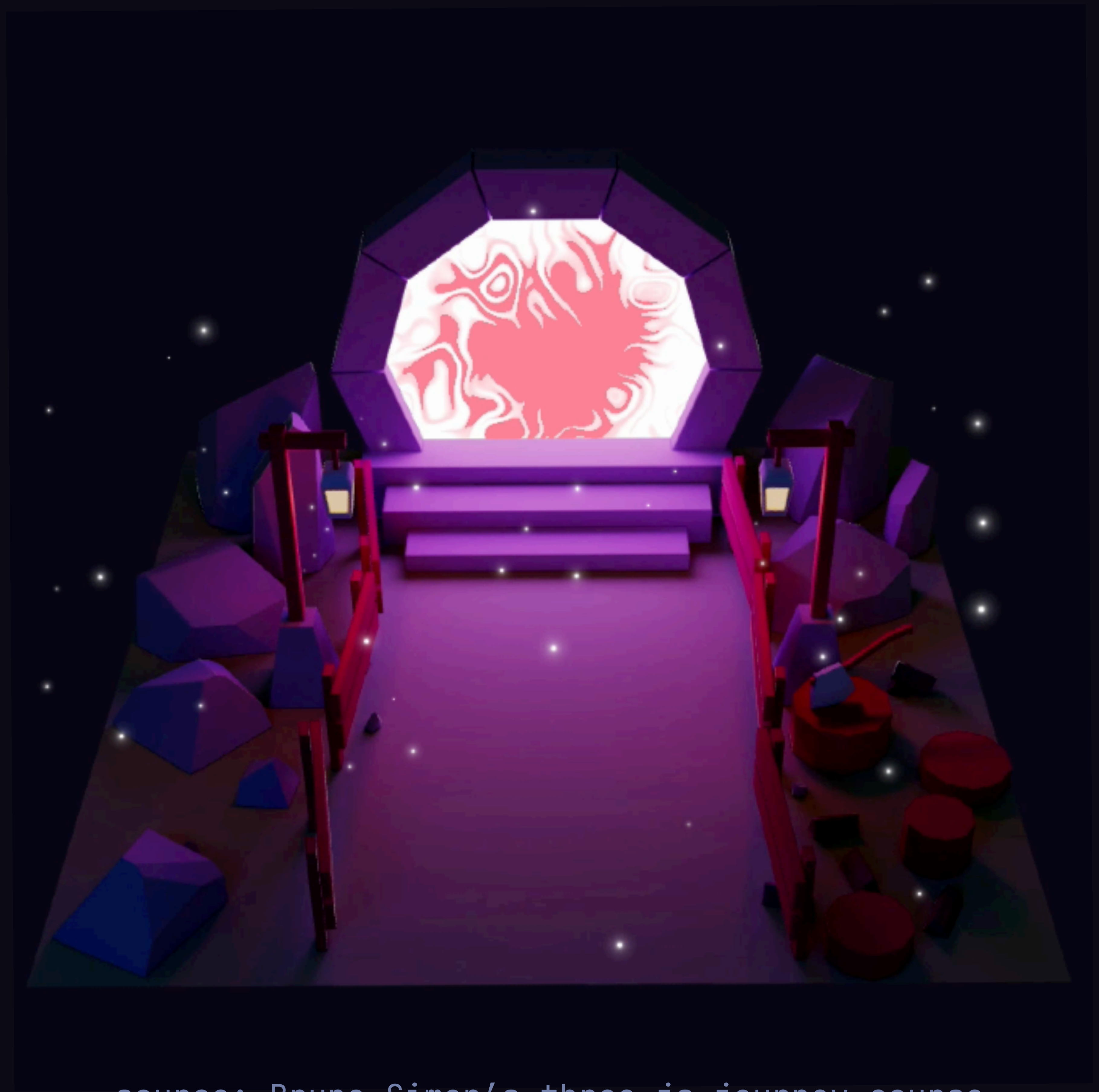


```
const scene = new THREE.Scene();
scene.background = new THREE.Color("#0d0c18")
```


Open Codesandbox

```
import * as THREE from "three";

const scene = new THREE.Scene();
scene.background = new THREE.Color("#0d0c18")

// Create a triangle shape, draw lines between (x, y) coordinates
const triangle = new THREE.Shape();
triangle.lineTo(0, 10);
triangle.lineTo(10, -10);
triangle.lineTo(0, -10);
const geometry = new THREE.ShapeGeometry(triangle);
geometry.center();

// Create Mesh
const material = new THREE.MeshNormalMaterial({ side: THREE.DoubleSide });
const mesh = new THREE.Mesh(geometry, material);
scene.add(mesh);

// Set the camera position
const camera = new THREE.PerspectiveCamera();
camera.position.set(0, 0, 60);
camera.lookAt(0, 0, 0);

// Render the scene
const renderer = new THREE.WebGLRenderer({ antialias: true });
const app = document.querySelector("#app");
app.appendChild(renderer.domElement);
renderer.setSize(640, 480);
renderer.render(scene, camera);
```





source: Bruno Simon’s three.js-journey course