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

```
// Use BoxGeometry to create the cube
const geometry = new THREE.BoxGeometry(10, 10, 10);
geometry.center();
```

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

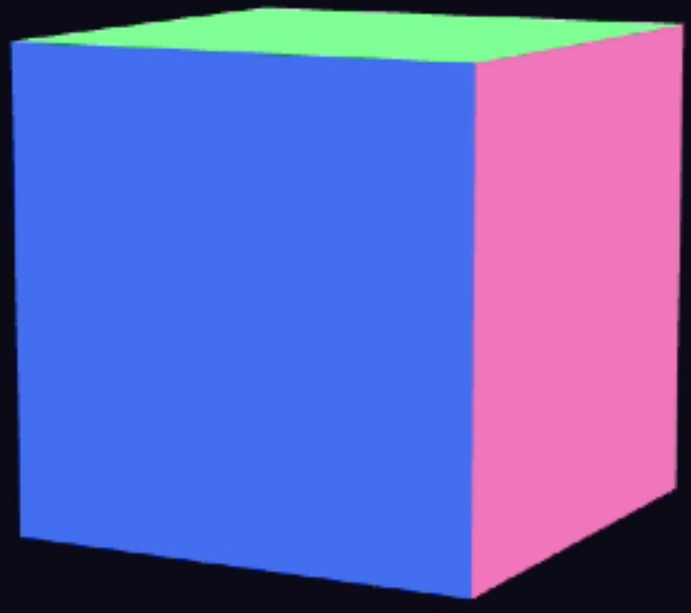
```
// Set the camera position
const camera = new THREE.PerspectiveCamera();
camera.position.set(10, 10, 60);
camera.lookAt(0, 0, 0);
```

```
// Setup rendering options
const renderer = new THREE.WebGLRenderer({ antialias: true });
const app = document.querySelector("#app");
app.appendChild(renderer.domElement);
renderer.setSize(window.innerWidth, window.innerHeight);
renderer.setPixelRatio(window.devicePixelRatio);
```

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

```
// Render the scene
const render = () => {
  renderer.render(scene, camera);
};

render();
```



Open Codesandbox

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

```
// Use BoxGeometry to create the cube
```

```
const geometry = new THREE.BoxGeometry(10, 10, 10);
```

```
geometry.center();
```

```
const cubeMesh = new THREE.Mesh(geometry, material);
```

```
// Create Mesh
```

```
scene.add(cubeMesh);
```

```
const material = new THREE.MeshNormalMaterial({ side: THREE.DoubleSide });
```

```
// Set the camera position
```

```
const camera = new THREE.PerspectiveCamera();
```

```
camera.lookAt(0, 0, 0);
```

```
camera.position.set(10, 10, 60);
```

```
const app = document.querySelector("#app");
```

```
const renderer = new THREE.WebGLRenderer({ antialias: true });
```

```
renderer.setPixelRatio(window.devicePixelRatio);
```

```
renderer.setSize(window.innerWidth, window.innerHeight);
```

```
// Setup rendering options
```

```
app.appendChild(renderer.domElement);
```

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

```
const scene = new THREE.Scene();
```

render();

```
const render = () => {
```

```
renderer.render(scene, camera);
```

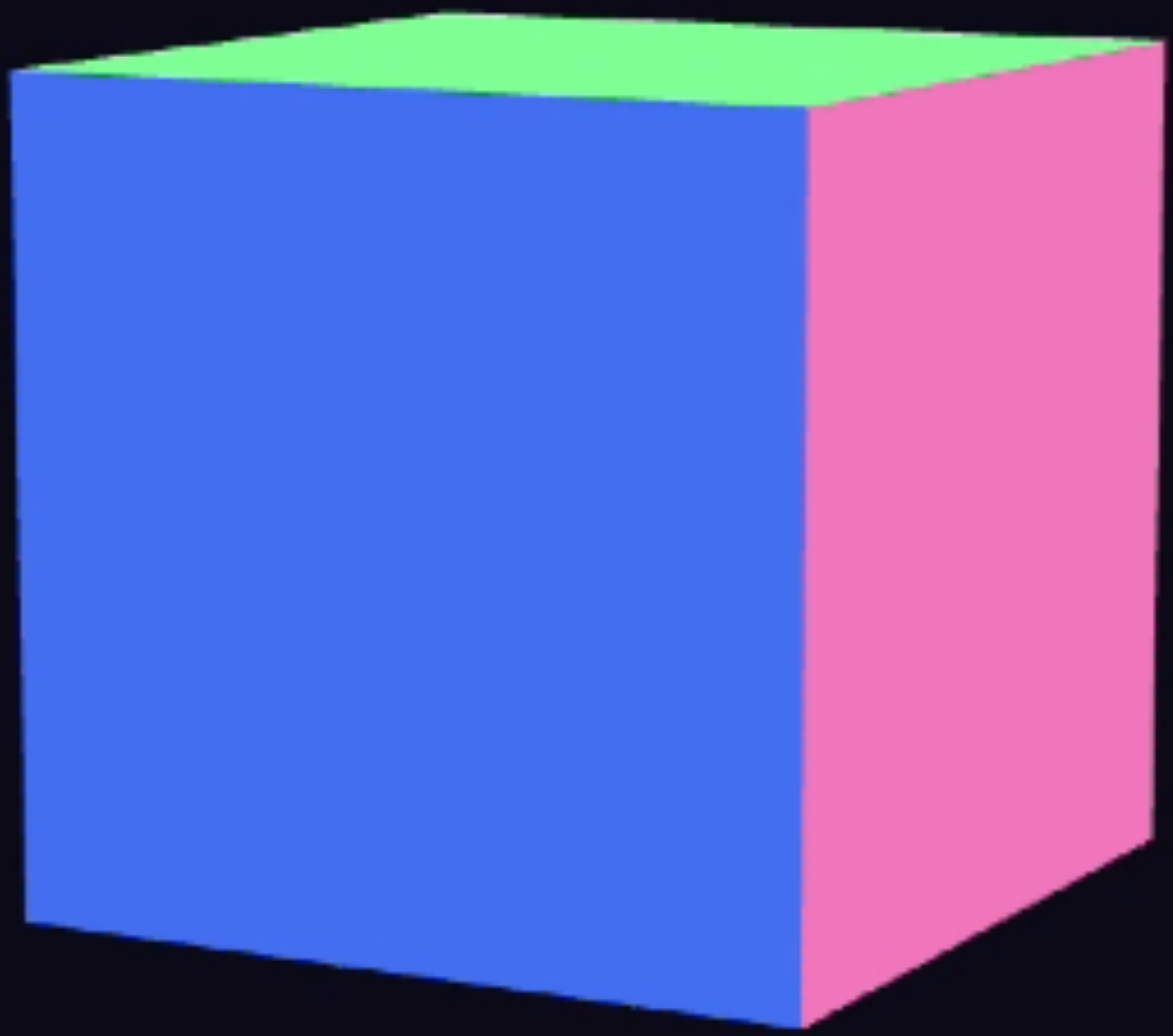
```
// Render the scene
```

```
// Rotate the cube along the y axis smoothly at 60 FPS
```

```
cubeMesh.rotation.y += 0.01;
```

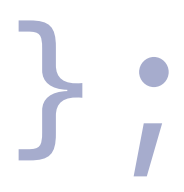
```
requestAnimationFrame(render);
```


Open Codesandbox



render();

```
const render = () => {
```



```
renderer.render(scene, camera);
```

```
// Render the scene
```