

# ThreeJS

---

ThreeJS est une librairie javascript pour créer des applications 3D dans un navigateur internet.

- Créer un fichier HTML qui contient le code suivant :

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8">
    <title>My first three.js app</title>
    <style>
      body { margin: 0; }
    </style>
  </head>
  <body>
    <script src="three.js"></script>
    <script>
      // Our Javascript will go here.
    </script>
  </body>
</html>
```

- Récupérer la librairie <https://threejs.org/build/three.js>
- Ajouter le fichier three.js dans le même dossier que le fichier HTML créé précédemment
- Pour afficher quelque chose dans three.js, on a besoin de 3 éléments : Une scène, une camera et un renderer. On va donc rajouter

```
const scene = new THREE.Scene();
const camera = new THREE.PerspectiveCamera( 75, window.innerWidth /
window.innerHeight, 0.1, 1000 );

const renderer = new THREE.WebGLRenderer();
renderer.setSize( window.innerWidth, window.innerHeight );
document.body.appendChild( renderer.domElement );
```

Ce code va créer un objet scene, puis créer une caméra de type "perspective". On va ensuite créer un renderer qu'on va intégrer dans le document HTML.

- On va ensuite rajouter le code suivant pour créer un cube dans notre application web :

```
const geometry = new THREE.BoxGeometry( 1, 1, 1 );
const material = new THREE.MeshBasicMaterial( { color: 0x00ff00 } );
const cube = new THREE.Mesh( geometry, material );
scene.add( cube );
```

```
camera.position.z = 5;
```

- On rajoute une fonction d'animation pour bouger le cube :

```
function animate() {  
    requestAnimationFrame( animate );  
  
    cube.rotation.x += 0.01;  
    cube.rotation.y += 0.01;  
    renderer.render( scene, camera );  
}  
animate();
```