

# TP Three.js

LP DAWIN

2015-2016

## 1 Initialisation

Commencez par récupérer les fichiers `three.min.js` et `OrbitControls.js`. Vous aurez besoin d'un fichier html contenant uniquement les balises `head` et `body` et qui inclue les deux fichiers js. C'est `three.js` qui sera chargé de générer le canvas. Vous pouvez mettre votre code directement entre des balises `script`, ou dans un fichier js à part.

Pour commencer, il faut créer quelques objets :

1. Un `THREE.WebGLRenderer`
2. Une `THREE.Scene`
3. Une `THREE.PerspectiveCamera(fov, ratio, near, far)`

Ensuite, il faut ajouter à la page le canvas créé par le renderer avec `document.body.appendChild(renderer.domElement)`

Vous pouvez aussi ajouter des controles avec un `new THREE.OrbitControls(camera, renderer.domElement)`

Pour dessiner la scene, on utilise `renderer.render(scene, camera)`

## 2 Premier Mesh

Dans `three.js`, les objets 3D (ou Mesh) sont composés d'une `Geometry` et d'un `Material`. Pour faire un cube :

1. Créer la géométrie du cube avec `THREE.BoxGeometry`
2. Créer le matériau du cube avec `THREE.BasicMeshMaterial`
3. Combiner les deux en créant un Objet 3D avec `THREE.Mesh(geometry, material)`
4. Ajouter le mesh à la scene avec `scene.add(mesh)`

## 3 Systeme solaire

Pour faire un sphere texturée :

1. Utiliser la `SphereGeometry`
2. Charger la texture avec `THREE.ImageUtils.loadTexture(file)`
3. Appliquer la texture sur le matériau en changeant la valeur de `material.map`

Réaliser comme cela la Terre et le Soleil grâce aux textures disponibles sur <http://planetpixemporium.com/planets.html>

Ajouter les transformations nécessaires pour que la Terre tourne sur elle meme et autour du soleil. Vous pouvez ajouter des transformations sur chaque Mesh avec les champs position, rotation et scale. De plus, materialiser l'orbite de la Terre avec une RingGeometry.

## 4 Lumiere

Maintenant que vous avez des objets, vous pouvez ajouter de l'éclairage. Pour cela, ajouter une `THREE.PointLight(color)` à la scène, et la placer dans le soleil. Le `BasicMeshMaterial` n'est pas affecté par la lumière, il faut le remplacer par `MeshPhongMaterial`.