

Universidad Autónoma Metropolitana

Unidad Cuajimalpa

Asignatura

SEMINARIO DE TECNOLOGÍAS DE LA
INFORMACIÓN II

Trabajo para entregar

Three.js

Alumno:

-FONSECA NEOS SALVADOR MICHEL

Grupo:

DK01T

Profesor

DR. ISAAC JUAN RUDOMIN GOLDBERG

Escena-1

1: esta segunda tarea se utilizó JavaScript para poder manipular todo lo que pongamos en el escenario. De igual manera en la primera escena se tenía que colocar cámara, luz y objetos. Para colocar la cámara se debe usar la sentencia `var camera`, en esta es donde pones el FOV para tener una mejor visión del escenario.

```
var scene = new THREE.Scene();  
var camera = new THREE.PerspectiveCamera(120, window.innerWidth /  
window.innerHeight, 0.1, 1000);
```

2: Los objetos que yo utilice fue un cubo, Para poder poner una figura se debe usar la sentencia `Var geometry` aquí es donde se va a poner el tipo de figura que queremos usar. Yo por ejemplo puse `Boxgeometry` que en este caso me va a cargar un cubo, ya que tenemos la figura ahora se debe colocar las dimensiones de esta misma para poder ajustarla a nuestro escenario.

Otras dos sentencias importantes son la de `cube.castshadow` la cual es de tipo boolean en la que al darle `true` nos va a poner sombra a nuestra figura solo cuando tengamos iluminación, y la de `cube.position` que ayuda a cambiar la posición de nuestra figura.

```
var geometry = new THREE.BoxGeometry(13, 13, 13, 4, 4, 4);  
var material = new THREE.MeshStandardMaterial({ color: 0xDAF7A6});  
var Cube = new THREE.Mesh(geometry, material);  
Cube.castShadow = true;  
Cube.position.set(10,1,3);  
scene.add(Cube);
```



Para finalizar con este primer escenario se colocó la luz con la sentencia `var light`, se puede poner varios tipos de iluminación como por ejemplo `spot light`, `ambient light` o la que yo utilice `point light`; de igual modo que con las figuras, la luz también la podemos cambiar de posición y cambiar la intensidad.



```
var light = new THREE.PointLight(0xffffff, 20, 200);  
    light.position.set(40, 16, 2);  
    light.castShadow = true;
```

Escena-2

En este Segundo escenario se agregó un modelo con extensión GLTF, de igual manera se tenía que cargar las texturas del modelo ya que si solo cargamos el modelo se vería nada más el molde de la figura.

En el three.js hay varias formas de cargar modelos, la que yo utilice es en la que debemos de poner el tipo de "LOADER" que vamos importar. Mi modelo es de extensión GLTF por lo que debemos de utilizar la sentencia: (La sintaxis varía entre diferentes cargadores).

```
import * as THREE from
'https://cdnjs.cloudflare.com/ajax/libs/three.js/r128/three.module.js';
    import { GLTFLoader } from
'https://threejsfundamentals.org/threejs/resources/threejs/r132/examples/jsm
/loaders/GLTFLoader.js';

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

2: para poder cargar el modelo se debe usar la sentencia `loader.load`, esta sentencia de `load` va a tomar dos argumentos el primero es el nombre del archivo del GLTF y el segundo la palabra función, dentro de esta función se va a crear un argumento, este argumento esencialmente va a ser el evento que va a pasar, en este caso el evento va a ser la información del GLTF.

```
loader.load("scene.gltf",function(gltf)
```

ahora solo para agregar el GLTF a nuestra escena se pone la sentencia:

```
scene.add(gltf.scene);
```



3: El último ejercicio era agregar sonido, esto se hace con la sentencia `audioLoader`. Nosotros podemos poner que la música se repita en bucle, poner el sonido he incluso se puede poner sonido a los objetos.

```
const audioLoader = new THREE.AudioLoader();  
    audioLoader.load( './Ice & Fire - King Canyon.mp3', function( buffer
```

