Project Dependencies

The project has only two dependencies.

a. Vitejs (Development Dependency)

Vitejs is a build tool that is used for productivity and for better development experience as it provides many features without any configurations.

Learn more at https://vitejs.dev/

b. Threejs

Threejs is an open-source JavaScript library that uses WebGL for creating 3D graphics on web.

Learn More at https://threejs.org/

Code

- ### Index.html

  In Index.html we linked our **Style** and **Script** files and added a canvas element with an id of **"bg"** so that we can reference it in script.

- ### camera.js

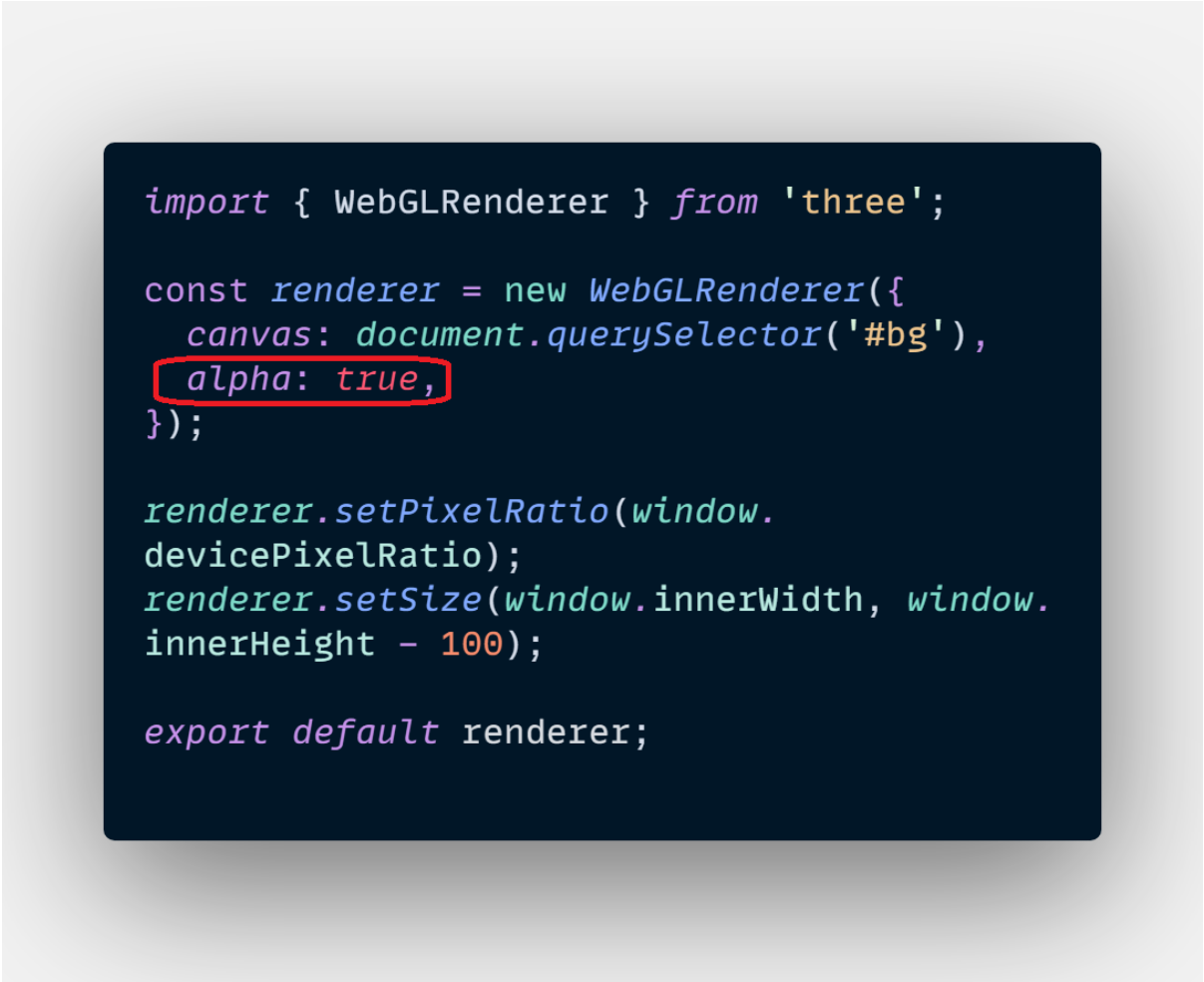
  All the configurations related to camera are stored in camera.js. In this project **PerspectiveCamera** is used which is the most basic Camera.
  Learn More at
  https://threejs.org/docs/index.html?q=perspe#api/en/cameras/PerspectiveCamera/

- ### renderer.js

  In order to render the scenes we need a renderer and all the configurations related to renderer is stored in **Renderer.js.** In this project the most common renderer **WebGLRenderer** is used**.** We've set alpha property to true to make the background visible the other configurations are self-explanatory.

  Learn More at

https://threejs.org/docs/index.html?q=WebGl#api/en/renderers/WebGLRenderer/



```
import { WebGLRenderer } from 'three';

const renderer = new WebGLRenderer({
  canvas: document.querySelector('#bg'),
  alpha: true,
});

renderer.setPixelRatio(window.
devicePixelRatio);
renderer.setSize(window.innerWidth, window.
innerHeight - 100);

export default renderer;
```

- main.js

In **main.js** we first setup the scene and import our **camera** and **renderer** from the respective modules.

Rendering the scene

For rendering the scene, we need what's called a render or animate loop. We implement this loop in **animate function** which causes the renderer to draw the scene every time the screen is refreshed.

Loading the model

In order to load the model we used GLTFLoader. We import GLTFLoader and initialize it. We use the its **load** method to our model. In the callback of this method we add the model into the scene.

Learn more at
https://threejs.org/docs/index.html?q=gltf#examples/en/loaders/GLTFLoader/

Adding the lights

Without lights the model seems miscoloured. We've used the directional lights. A light that gets emitted in a specific direction. This light will behave as though it is infinitely far away and the rays produced from it are all parallel.

Learn more at

https://threejs.org/docs/index.html?q=directional#api/en/lights/DirectionalLight/

Three-D Rotation

For this purpose we use **OrbitControls.** Learn more at

https://threejs.org/docs/index.html?q=orbitcontrols#examples/en/controls/OrbitControls/