**Roll Dice Documentation**

**Instructions**

Simply press the “Roll Dice” button and a random number between 1 – 6 will be generated and displayed on the dice.

**Building “main.html”**

I began by creating a basic HTML file in which our 3D object can be viewed called “main.html”. It contains all the basic component of a blank html file. “Viewport” under the head section allows the webpage to be scalable, especially for mobile viewing. The canvas will take up the entire window of the webpage using the “canvas” style. The “three.js” script is linked in the body of the html file. It allows three.js to run without requiring a local copy. The body also calls a local file called “main.js”, where the 3D object will be built. I also created a button labeled “Roll Dice” for simple user input.

**Building “main.js”**

1. I started by creating our function “init()” and function “update()”. Then added “scene”, “camera”, “renderer” and “cube” elements using the global “let”.
   1. Under the function “init()”
      1. Added scene graph “THREE.Scene”.
      2. Added perspective camera with
         1. Field of view of 100
         2. Aspect ratio which automatically adjust based on window size
         3. and Near and Far frustum set at 0.1 and 100 respectively.
      3. Added WebGL renderer and set size to whole screen and added to “canvas” element on “main.html”.
      4. Add cube object
      5. Got textures for each side of the cube from flaticon.com and mapped them to each side of the cube
      6. Used “cube = new THREE.Mesh(geometry, materials)” to put it all together
      7. Added the cube to the scene along with ambient and directional lighting
      8. Positioned the camera to z = 5
   2. Under the function “update()”
      1. Set button to visible
      2. Used “cube.rotation” to rotate the cube slowly along all axis’s
      3. Render the scene and camera angle by use “renderer(scene, camera)”
      4. Animates function “update()” using “requestAnimationFrame”
2. Created function “getRandomNumber()” and “roll()”
   1. Under “getRandomNumber()”
      1. Random number between 1 and 6 generated
      2. “roll()” function is called
   2. Under “roll()”
      1. Update function animation is stopped using “cancelAnimationFrame”
      2. Animates function “roll()” using “requestAnimationFrame”
      3. Set button to hidden so user can’t unintentionally press button
      4. Counter counts the number of frames/cycles of the animation
         1. If counter is below 50, it rotates the dice rapidly in all directions to simulate dice roll
         2. If counter is between 50 and 150 display side of dice which corresponds with random number (1-6) using “cube.rotation.set”
         3. If counter is above 150, end roll function animation and call update()
      5. Render the scene and camera angle by use “renderer(scene, camera)”
3. Created function “onWindowResize()” and called using an event listener
   1. “onWindowResize()” is called when window size changes. Changes canvas size and keeps dice centered as window size changes
4. Call “init()” and “update()” on launch
5. Uses event listener to determine if button is pressed. If pressed, calls “getRandomNumber” function

**Attribution:**

Dice icons created by Freepik – Flaticon.com

<https://www.flaticon.com/free-icons/dice>

This attribution is also displayed on the “main.html” file. Simply scroll down on the “main.html” file.