Assignment 3: Checkers

To do this assignment I have my code structured like the following steps:

Setup camera, trackball and scene:

First of all I have defined the camera variables, the trackball controls and created the three.js scene like the following:

```
//Camera variables
        var VIEW_ANGLE = 65, //65 FOV is most 'natural' FOV
          ASPECT = WIDTH / HEIGHT,
          NEAR = 0.1,          //these elements are needed for cameras to
          FAR = 10000;         //partition space correctly
        var camera =
          new THREE.PerspectiveCamera(
                VIEW_ANGLE,
                ASPECT,
                NEAR,
                FAR);
        camera.position.z = 400;
        var controls = new THREE.TrackballControls( camera );
        controls.target.set( 0, 0, 0 )

        var scene = new THREE.Scene();
        scene.add(camera);
```

Checkers board:

Then, I have created the checkers board object as three.js cube geometry and added to it a checkers board image texture.

Positions:

I have created all the initial positions for the black pieces at the upper levels of the board and save them into an array called positionblack[]. In the same way, I have created all the initial positions for the white pieces and save them into an array called positionwhites[].

Moreover, I have saved all the white squares position in the board into an array called allwhiteboardpositions[] and all the black squares position into an array called allblackboardpositions[].

Create the pieces

To create the pieces I have call the function loader and load the JSON data file with the pieces. I have created a piece for every white piece initial position and a piece for every black piece initial position. I have saved those pieces into arrays called whitepieces[] and blackpieces[]

As an example, this is the way I have created the black pieces:

```
for(var j=0;j<positionblack.length;j++){

        //Black color to black pieces

        var material=new THREE.MeshBasicMaterial( { color: 0x666666 });

        //create the piece geometry

        blackpiecesmesh = new THREE.Mesh( geometry, material);

        //scale, position and rotate to place the piece over the board
        blackpiecesmesh.scale.set( 60, 60, 60 );

        blackpiecesmesh.rotation.y += Math.PI/2;
        blackpiecesmesh.rotation.z += Math.PI/2;

        //Position, taken  from the positionwhites array

        blackpiecesmesh.position= positionblack[j];

        //console.log(blackpiecesmesh.position);

        //save the piece into the blackpieces array
        blackpieces.push(blackpiecesmesh);


        //add it to the scene
        scene.add(blackpiecesmesh);

}//for j
```

Randomize button

To randomize the pieces in the board once the randomize button is clicked I have created a shuffle function to randomize the arrays of allwhiteboardpositions[] and allblackboardpositions[]. This shuffle function follows the Fisher-Yates algorithm that I have searched.

Once the arrays are shuffled, I have assigned one of the randomized allwhiteboardpositions to each of the black pieces and one of the randomized allblackboardpositions to each of the white pieces.

To move them smoothly from its original position I have used tween.js library and creating a tween object for every piece that is how I have use it:

```
$.each (blackpieces, function(key, piece){
        //smoothy move the blackpieces from the original initial position
        var tween = new TWEEN.Tween( { x: piece.position.x, y: piece.position.y } )
                //to the target (allwhitepositions[key])
                .to( { x: allwhiteboardpositions[key].x,y:allwhiteboardpositions[key].y
}, 500 )
                .onUpdate( function () {
                        piece.position.x=this.x;
                        piece.position.y=this.y;

        } ).start();

})//each
```

Finally, I have to do TWEEN.update in the renderLoop position to have the tweens updated.

Change color button

To change the color of the board once the button change color is clicked I have changed the texture of the board mesh loading a new image as its texture.

CSS

To make my submission looks better, I have included some CSS that can be found in the css /style.css file.