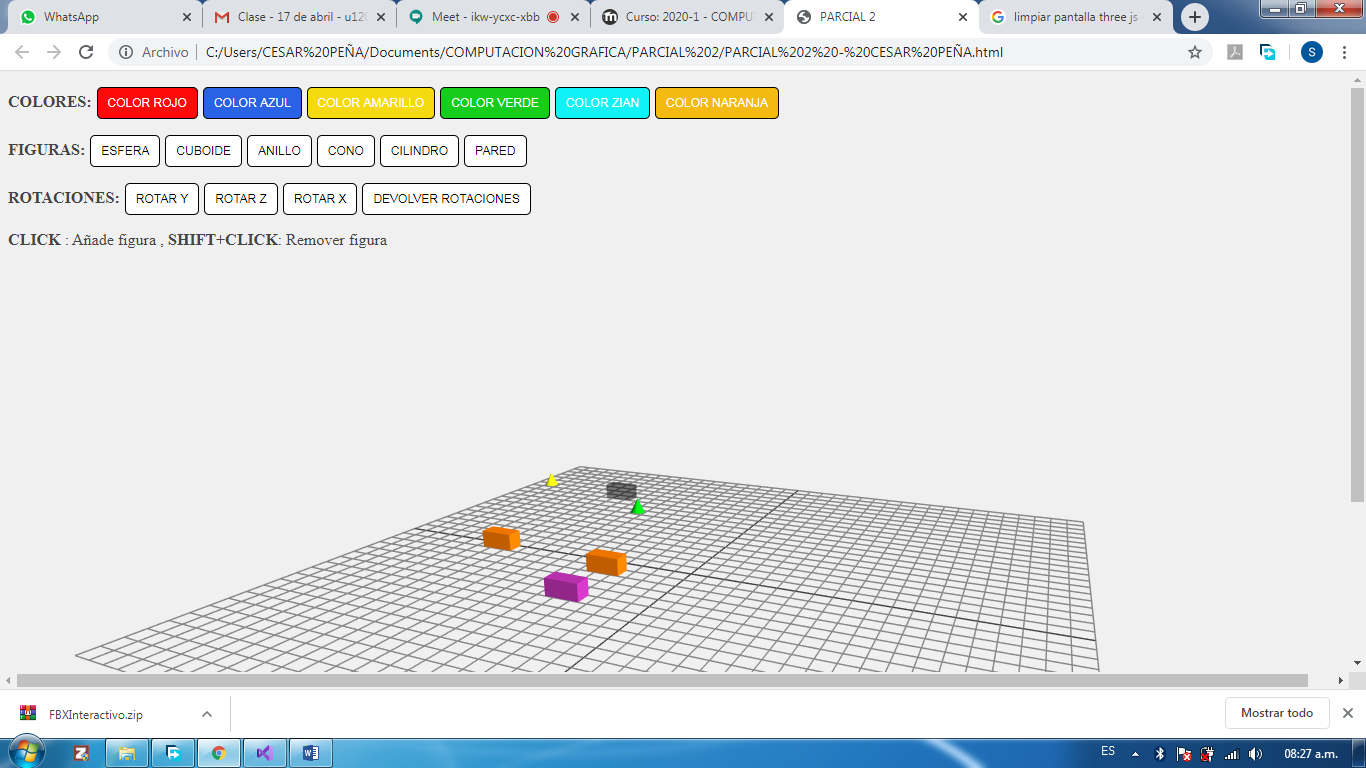
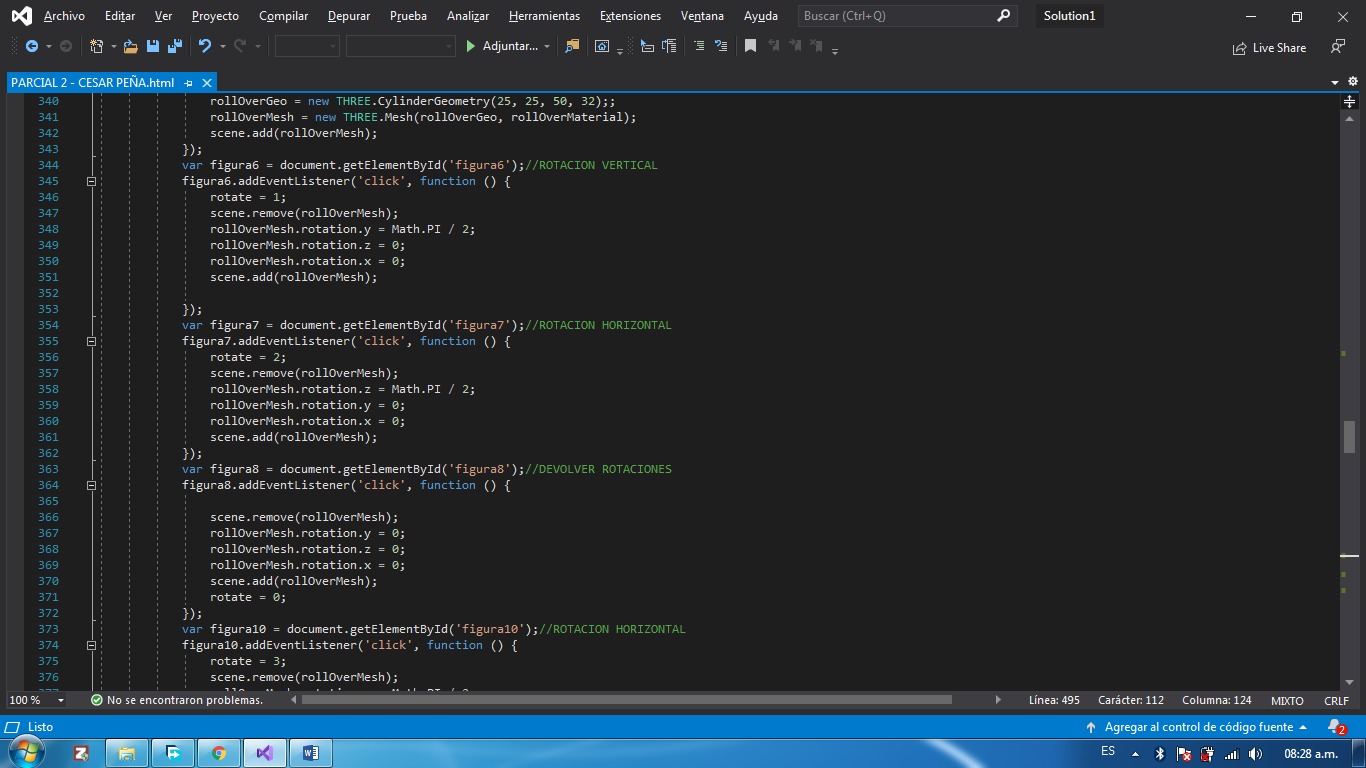
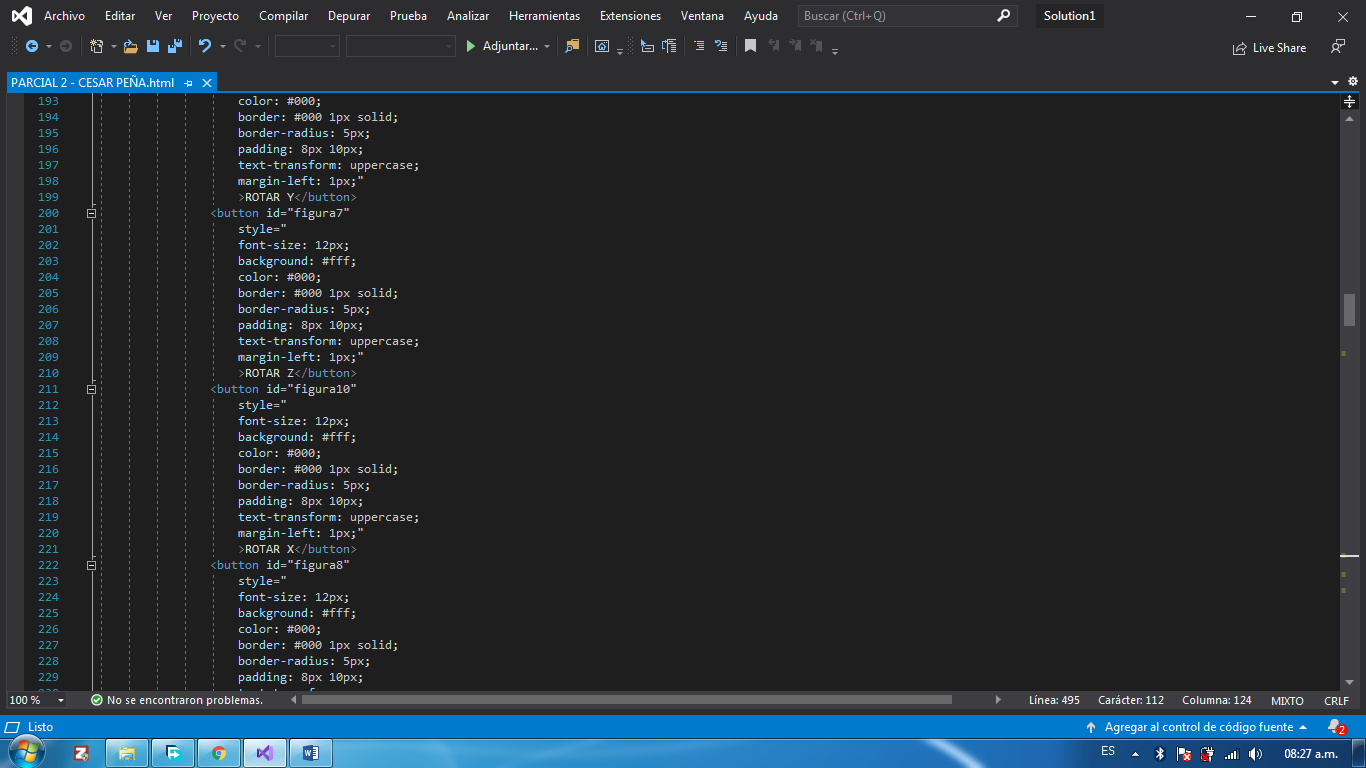
INFORME PARCIAL 2:

CESAR PEÑA – 1202190

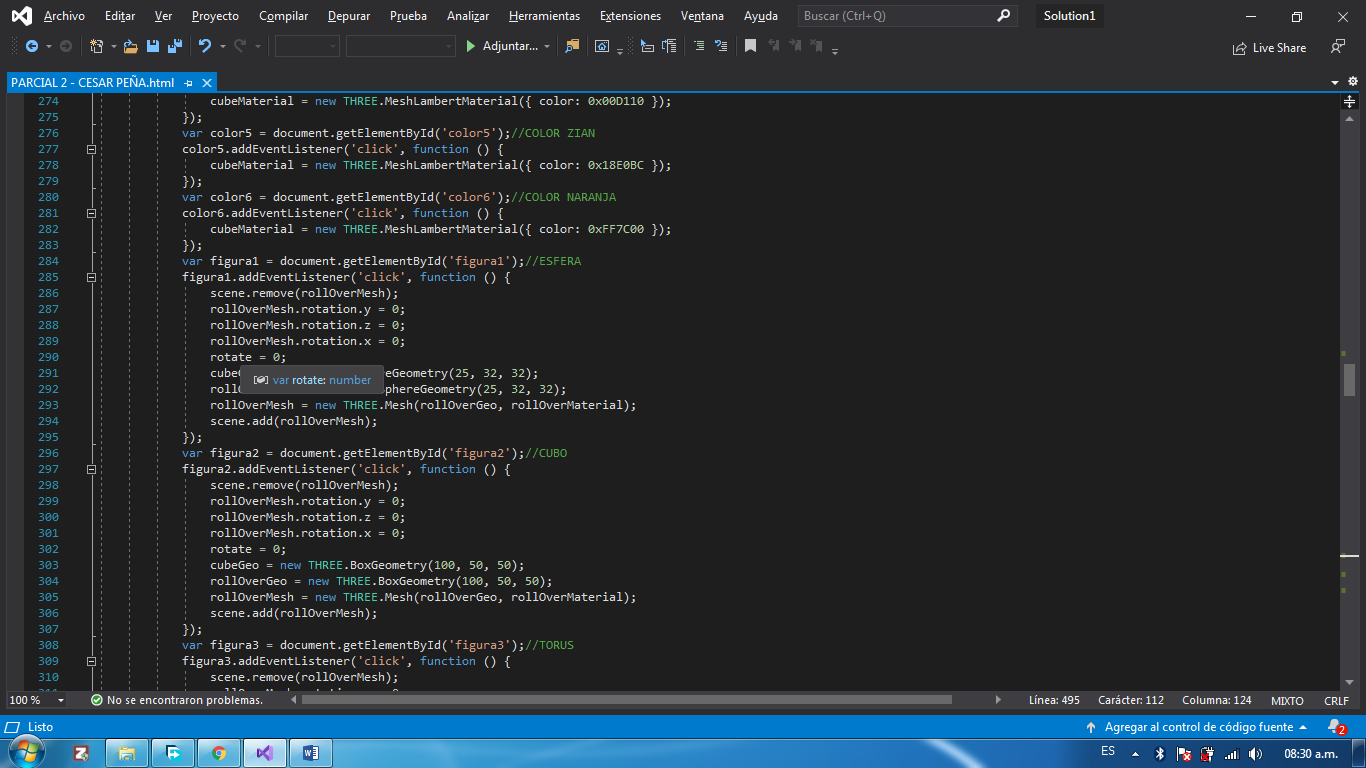
Para la elaboración del programa se usó el programa de la documentación de la página de Three.js el programa voxel pointer se utilizaron botones para la declaración de qué tipo de figura se va a utilizar, colores y rotaciones como se ve en pantalla:



Utilizando el siguiente código:

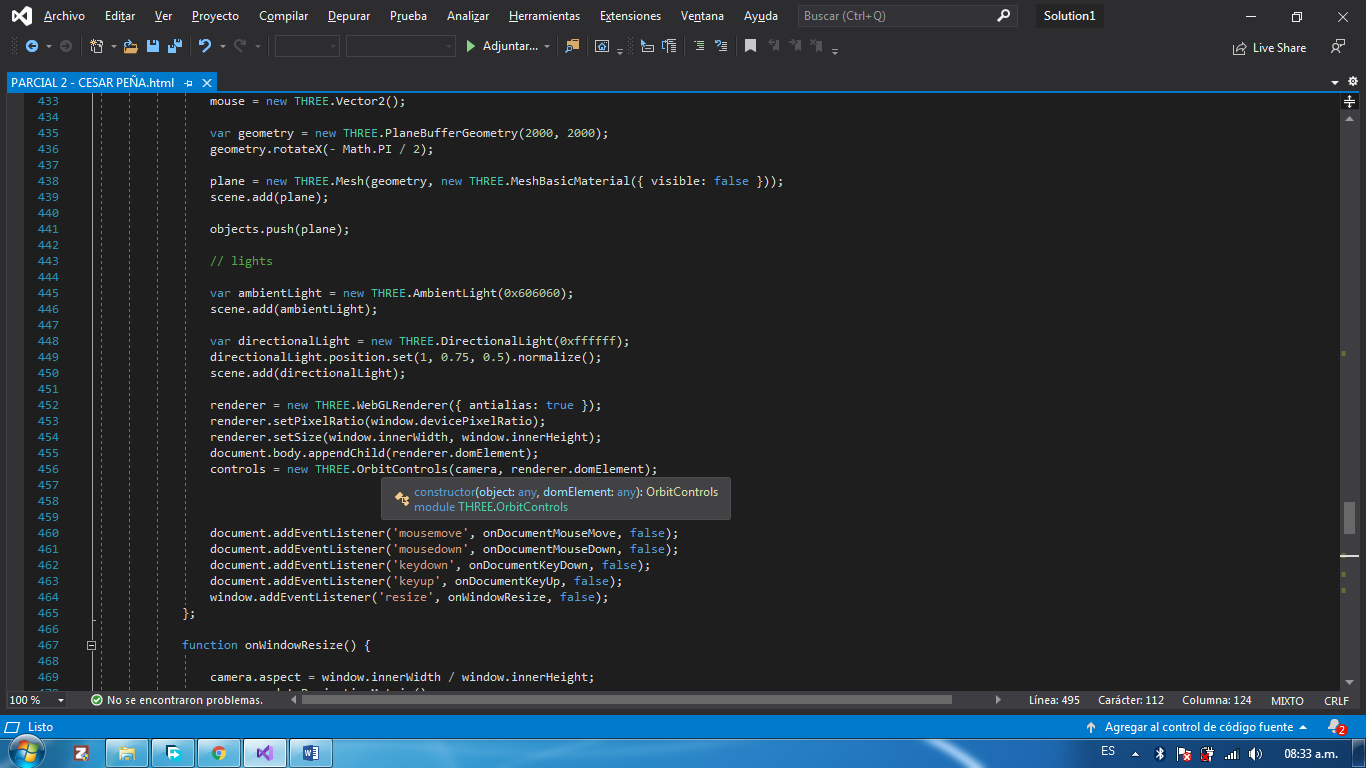
f

Después de esto se agregaron las diferentes geometrías inicializando:

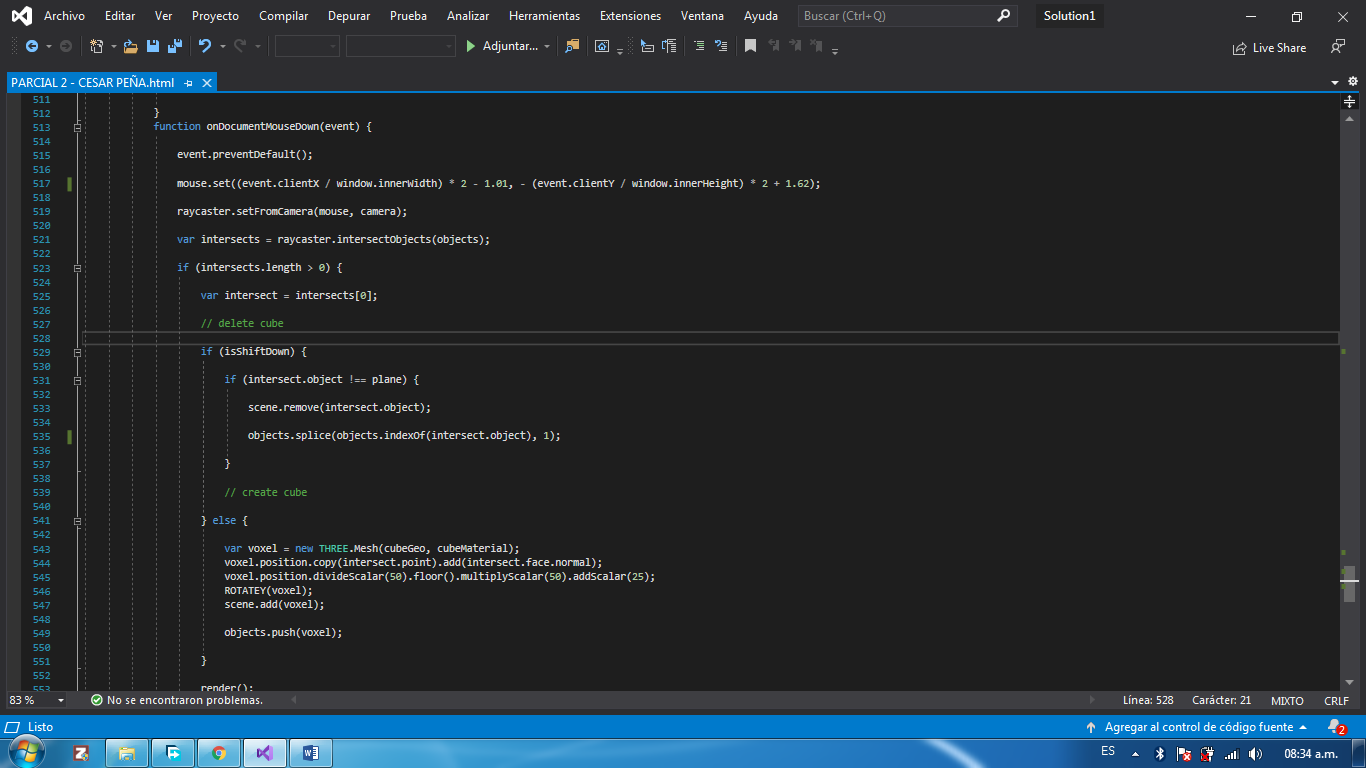


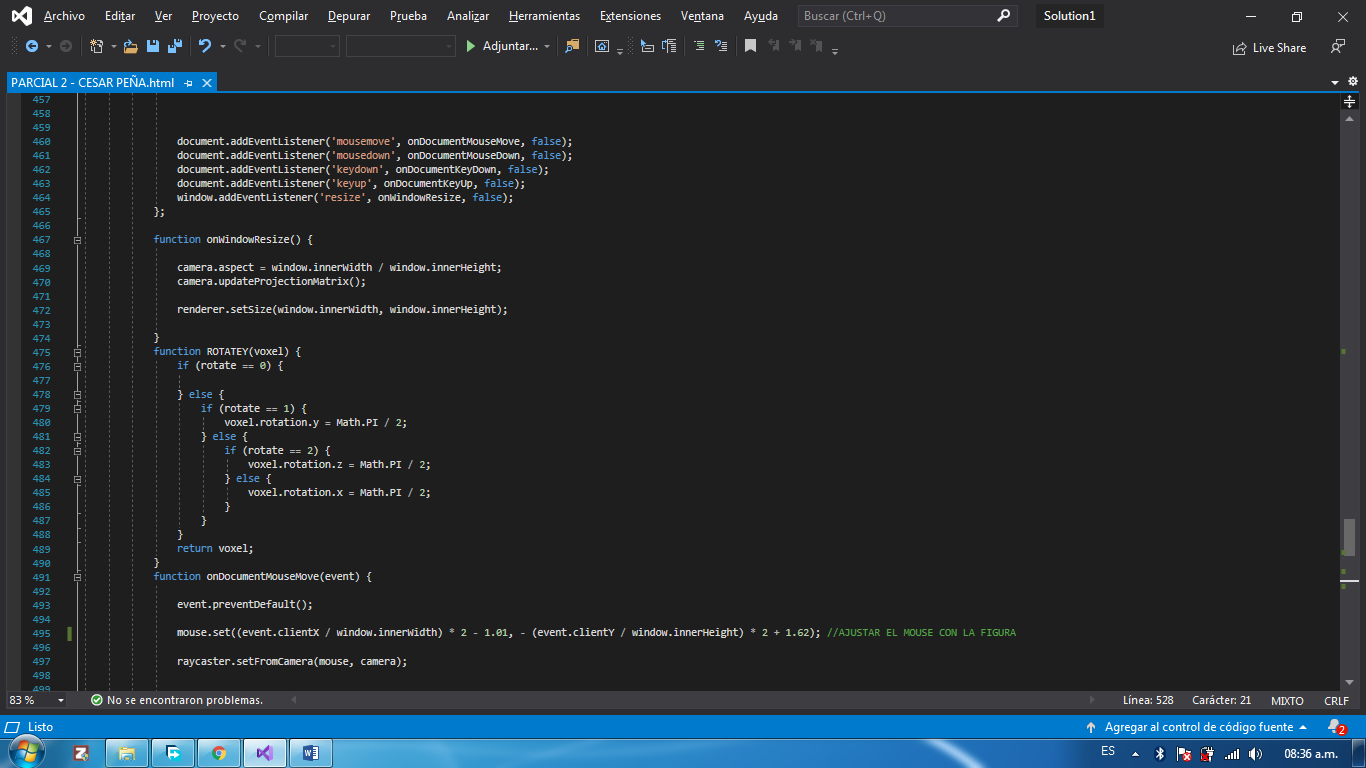
Ya con estas figuras se pueden lograr agregar botones y reiniciar los valores de CubeGeo y RollOverMesh para poder cambiar los valores de figura que se va a agregar.

El programa cuenta con los OrbitControls agregándolo en el momento que inicializa el render como se ve en la imagen:



Las figuras se rotaron a través de una llamada a una función dentro de la función mouseDown :

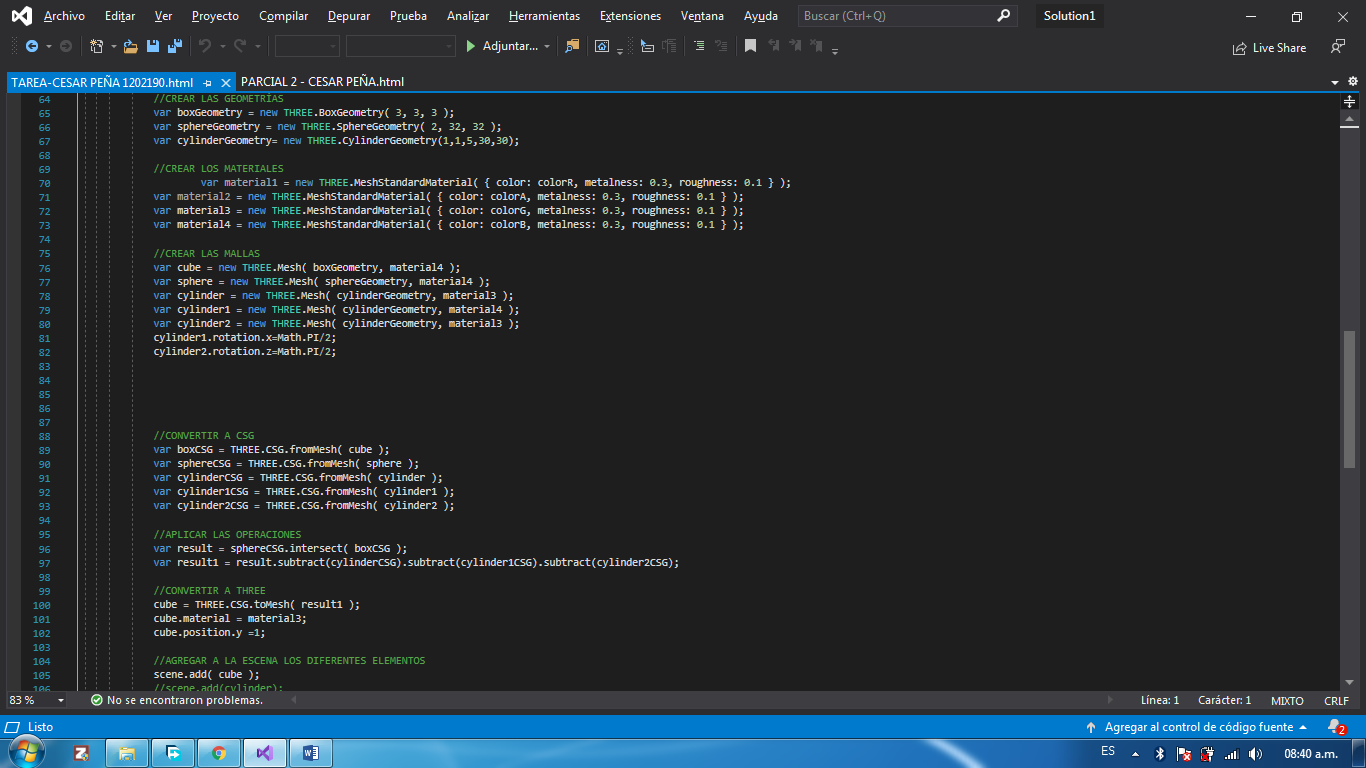




De igual manera ciertas figuras creadas a través de extrusiones o figuras creadas por mi autoría y las agrega de igual manera al programa:

Unos ejemplos de estas son:

 var shape = new THREE.Shape();  
shape.moveTo(0,0);  
shape.lineTo( 0, 2.5 );  
shape.lineTo( 0.5, 2.5 );  
shape.lineTo( 0.5, 3 );  
shape.lineTo( 1.5, 3 );  
shape.lineTo( 1.5, 2.5 );  
shape.lineTo( 2, 2.5);  
shape.lineTo( 2, 3 );  
shape.lineTo( 3, 3 );  
  shape.lineTo( 3, 2.5 );  
shape.lineTo( 3.5, 2.5 );  
shape.lineTo( 3.5, 0 );  
  
  
             
           var material = new THREE.LineBasicMaterial( { color : 0xff0000 } );  
           var resolution = 50;  
           var points = shape.getPoints( resolution );  
           var geometry = new THREE.BufferGeometry().setFromPoints( points );  
           // Create the final object to add to the scene  
           var curveObject = new THREE.Line( geometry, material );  
             
//EXTRUDE  
var extrudeSettings = {  
steps: 2,  
amount: 3.5,  
depth:1,  
bevelEnabled: false,  
};  
  
var geometryExt = new THREE.ExtrudeGeometry( shape, extrudeSettings );  
var materialExt= new THREE.MeshStandardMaterial( {  
color: 0xAA3333,  
metalness: 0,  
roughness: 0.1,  
opacity: 1,  
transparent: false  
} );  
  
var mesh = new THREE.Mesh( geometryExt, materialExt ) ;  
scene.add( mesh );



Ya como tal ese tipo de figuras son las que intenté meter en el programa, pero no las pude insertar debido a que el voxel no me las dejaba pintar por la geometría de malla que tiene por defectos