

Geometric Objects and Transformations

6TH WEEK, 2022



Modeling a Cube

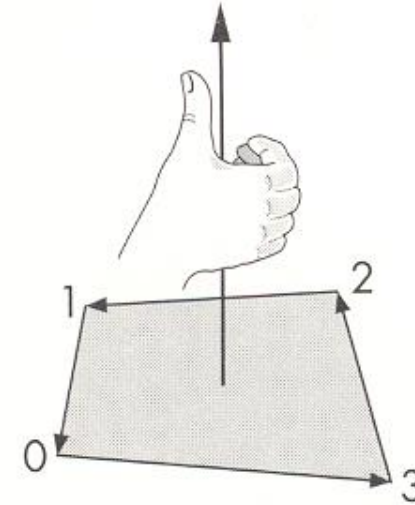
- Surface-based model
 - Outward-pointing face
 - Right-hand rule: counterclockwise order
- Data structure
 - Geometry: location of vertices

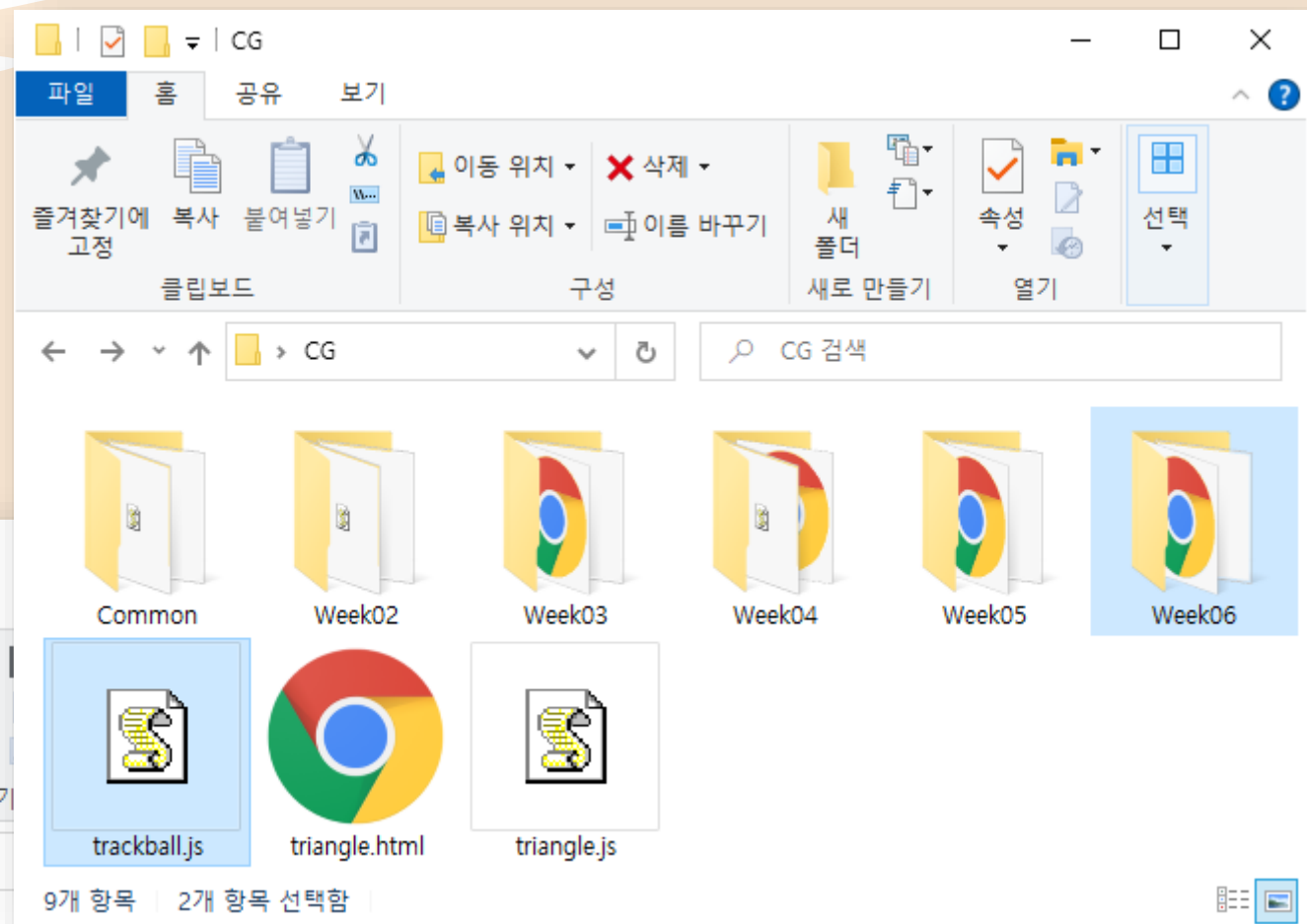
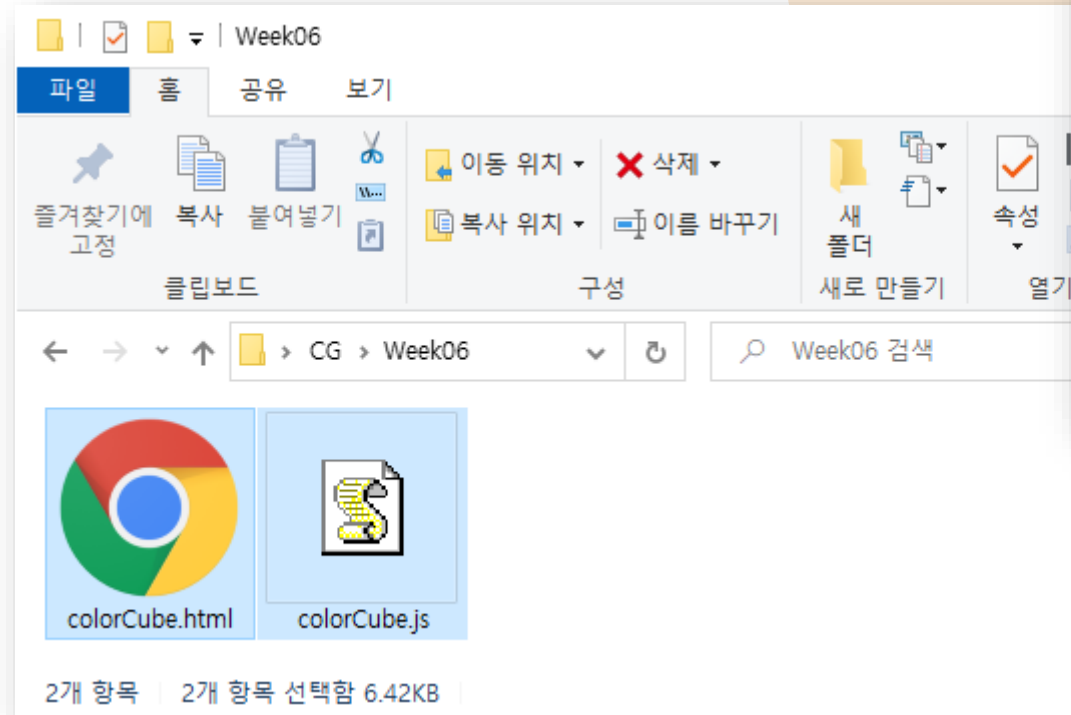
```
float vertices[8][3]= { {-1.0, -1.0, -1.0}, {1.0, -1.0, -1.0},  
                        {1.0, 1.0, -1.0}, {-1.0, 1.0, -1.0}, {-1.0, -1.0, 1.0},  
                        {1.0, -1.0, 1.0}, {1.0, 1.0, 1.0}, {-1.0, 1.0, 1.0}};
```

- Topology: connectivity

```
var indices = [ 0, 3, 2, 0, 2, 1 ];  
for ( var i = 0; i < indices.length; ++i )  
    points.push( vertices[indices[i]] );
```

```
gl.drawArrays( gl.TRIANGLES, 0, NumVertices );
```





File Edit Selection View Go Run Terminal Help

colorCube.html - Visual Studio Code

colorCube.html x JS colorCube.js

C: > Users > Sun-Jeong Kim > Desktop > CG > Week06 > <> colorCube.html > ...

```
1 <!DOCTYPE html>
2 <html>
3   <head>
4     <title>학번 이름 - Color Cube</title>
5     <script id="vertex-shader" type="x-shader/x-vertex">
6       attribute vec4 vPosition;
7       attribute vec4 vColor;
8       uniform vec3 theta;
9       varying vec4 fColor;
10
11     void main() {
12       // Compute the sines and cosines of theta for each of
13       // the three axes in one computation
14       vec3 angles = radians(theta);
15       vec3 c = cos(angles);
16       vec3 s = sin(angles);
17
18       // Remember: these matrices are column-major
19       mat4 rx = mat4( 1.0, 0.0, 0.0, 0.0,
20                     0.0, c.x, s.x, 0.0,
21                     0.0, -s.x, c.x, 0.0,
22                     0.0, 0.0, 0.0, 1.0 );
23
24       mat4 ry = mat4( c.y, 0.0, -s.y, 0.0,
25                     0.0, 1.0, 0.0, 0.0,
26                     s.y, 0.0, c.y, 0.0,
27                     0.0, 0.0, 0.0, 1.0 );
28
29       mat4 rz = mat4( c.z, s.z, 0.0, 0.0,
30                     -s.z, c.z, 0.0, 0.0,
31                     0.0, 0.0, 1.0, 0.0,
32                     0.0, 0.0, 0.0, 1.0 );
33
34       gl_Position = rz * ry * rx * vPosition;
35       fColor = vColor;
```


Ln 1, Col 1 Spaces: 4 UTF-8 CRLF HTML

File Edit Selection View Go Run Terminal Help colorCube.html - Visual Studio Code

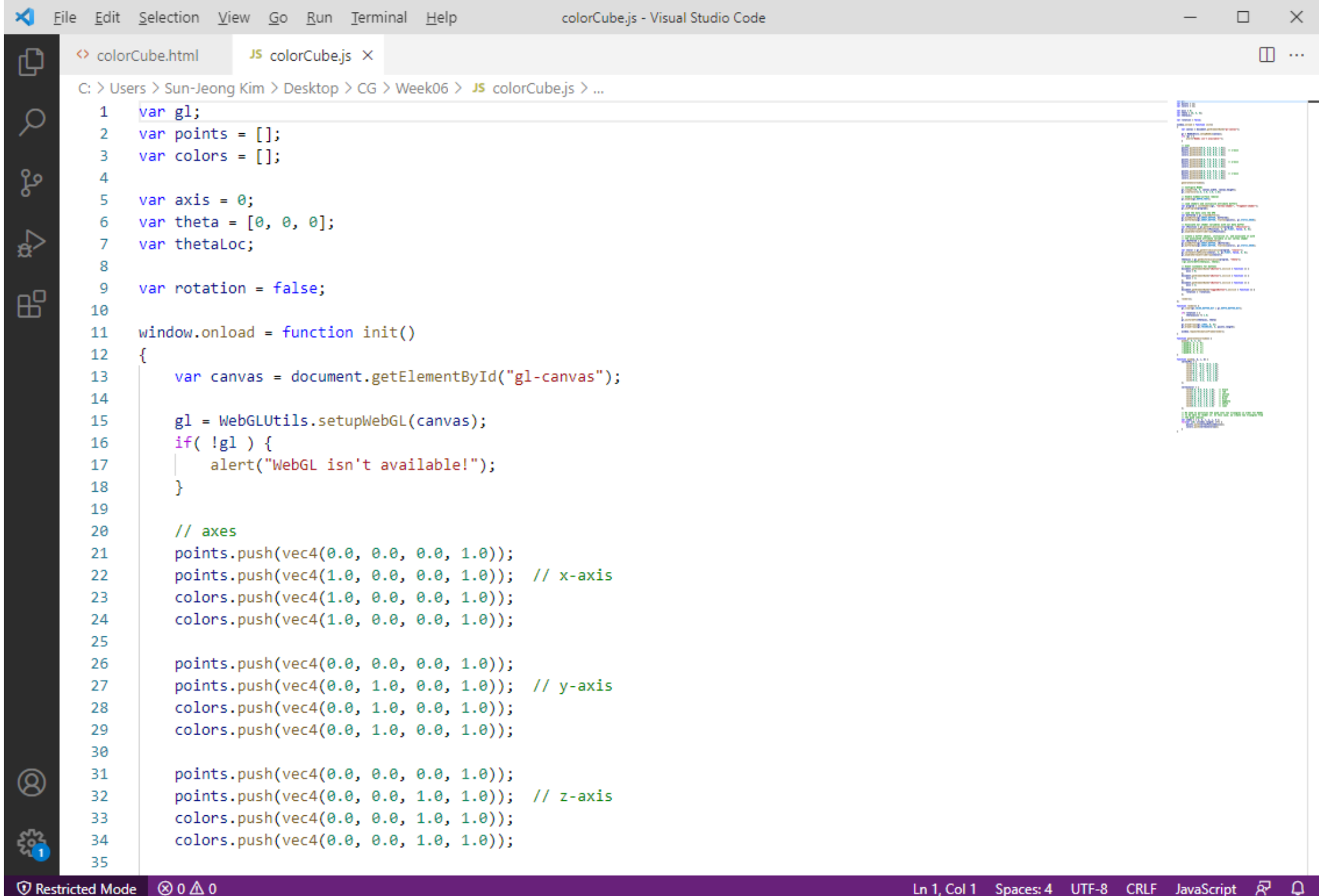
colorCube.html x JS colorCube.js

C: > Users > Sun-Jeong Kim > Desktop > CG > Week06 > <> colorCube.html > ...

```
33
34     gl_Position = rz * ry * rx * vPosition;
35     fColor = vColor;
36 }
37 </script>
38
39 <script id="fragment-shader" type="x-shader/x-fragment">
40 precision mediump float;
41 varying vec4 fColor;
42
43 void main() {
44     gl_FragColor = fColor;
45 }
46 </script>
47
48 <script type="text/javascript" src="../../Common/webgl-utils.js"></script>
49 <script type="text/javascript" src="../../Common/initShaders.js"></script>
50 <script type="text/javascript" src="../../Common/MV.js"></script>
51 <script type="text/javascript" src="colorCube.js"></script>
52 </head>
53 <body>
54     <div>
55         <button id="xButton">Rotate X</button>
56         <button id="yButton">Rotate Y</button>
57         <button id="zButton">Rotate Z</button>
58         <button id="toggleButton">Toggle Rotate</button>
59     </div>
60     <canvas id="gl-canvas" width="512" height="512">
61         Oops... your browser doesn't support the HTML5 canvas element!
62     </canvas>
63 </body>
64 </html>
```



Ln 1, Col 1 Spaces: 4 UTF-8 CRLF HTML



FileEditSelectionViewGoRunTerminalHelpcolorCube.js - Visual Studio Code

colorCube.htmlJS colorCube.js

C: > Users > Sun-Jeong Kim > Desktop > CG > Week06 > JS colorCube.js > ...

36generateColorCube();

37

38// Configure WebGL

39gl.viewport(0, 0, canvas.width, canvas.height);

40gl.clearColor(1.0, 1.0, 1.0, 1.0);

41

42// Load shaders and initialize attribute buffers

43var program = initShaders(gl, "vertex-shader", "fragment-shader");

44gl.useProgram(program);

45

46// Load the data into the GPU

47var bufferId = gl.createBuffer();

48gl.bindBuffer(gl.ARRAY_BUFFER, bufferId);

49gl.bufferData(gl.ARRAY_BUFFER, flatten(points), gl.STATIC_DRAW);

50

51// Associate our shader variables with our data buffer

52var vPosition = gl.getAttribLocation(program, "vPosition");

53gl.vertexAttribPointer(vPosition, 4, gl.FLOAT, false, 0, 0);

54gl.enableVertexAttribArray(vPosition);

55

56// Create a buffer object, initialize it, and associate it with

57// the associated attribute variable in our vertex shader

58var cBufferId = gl.createBuffer();

59gl.bindBuffer(gl.ARRAY_BUFFER, cBufferId);

60gl.bufferData(gl.ARRAY_BUFFER, flatten(colors), gl.STATIC_DRAW);

61

62var vColor = gl.getAttribLocation(program, "vColor");

63gl.vertexAttribPointer(vColor, 4, gl.FLOAT, false, 0, 0);

64gl.enableVertexAttribArray(vColor);

65

66thetaLoc = gl.getUniformLocation(program, "theta");

67//gl.uniform3fv(thetaLoc, theta);

68

69// Event listeners for buttons

70document.getElementById("xButton").onclick = function () {

7

Ln 1, Col 1Spaces: 4UTF-8CRLFJavaScript

7

File Edit Selection View Go Run Terminal HelpcolorCube.js - Visual Studio Code

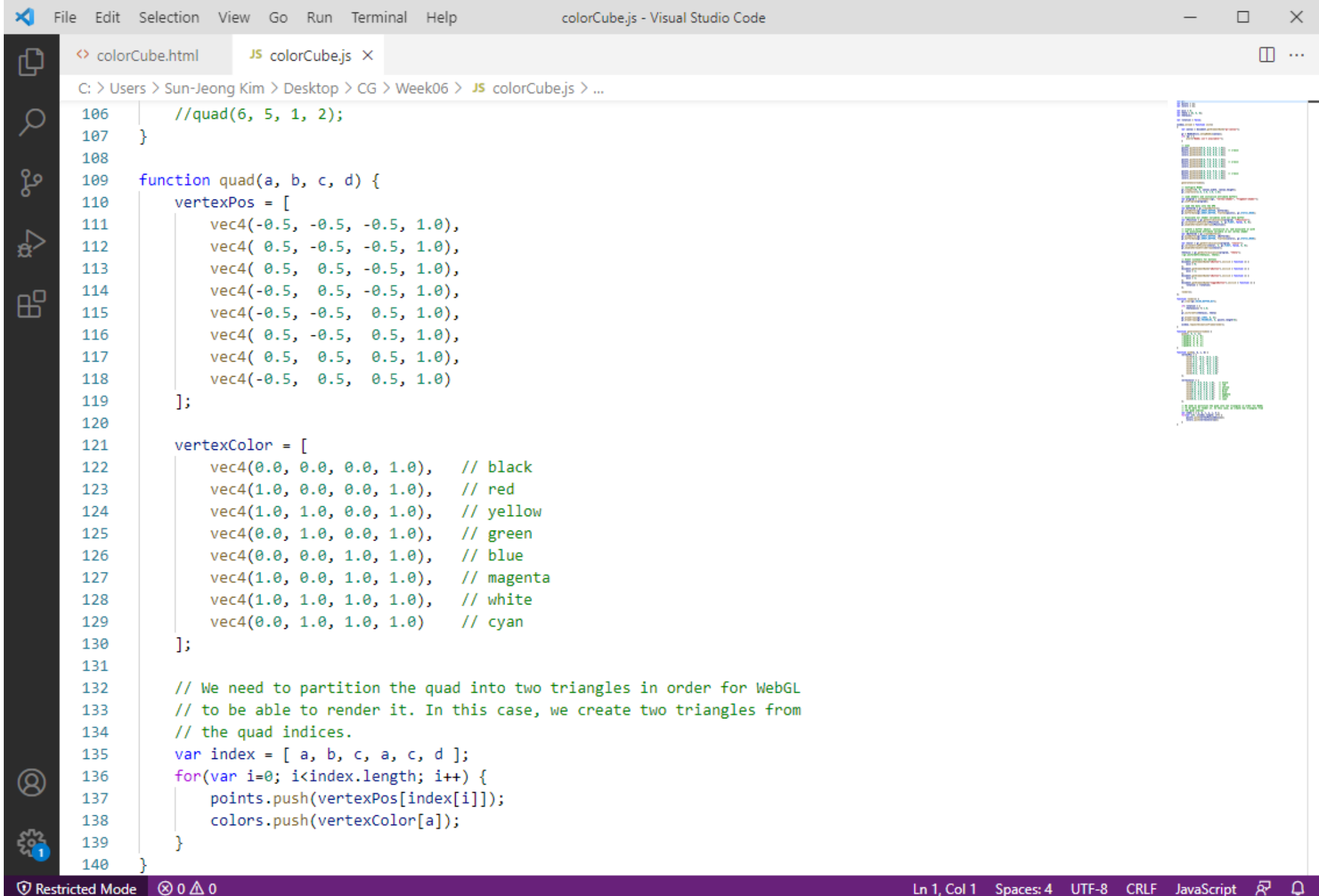
colorCube.htmlJS colorCube.js X

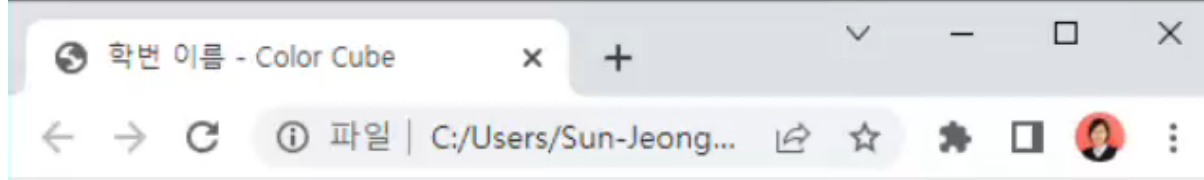
C: > Users > Sun-Jeong Kim > Desktop > CG > Week06 > JS colorCube.js > ...

```
71     axis = 0;
72 };
73 document.getElementById("yButton").onclick = function () {
74     axis = 1;
75 };
76 document.getElementById("zButton").onclick = function () {
77     axis = 2;
78 };
79 document.getElementById("toggleButton").onclick = function () {
80     rotation = !rotation;
81 };
82
83 render();
84 };
85
86 function render() {
87     gl.clear(gl.COLOR_BUFFER_BIT);
88
89     if( rotation ) {
90         theta[axis] += 2.0;
91     }
92     gl.uniform3fv(thetaLoc, theta)
93
94     gl.drawArrays(gl.LINES, 0, 6);
95     gl.drawArrays(gl.TRIANGLES, 6, points.length-6);
96
97     window.requestAnimationFrame(render);
98 }
99
100 function generateColorCube() {
101     quad(1, 0, 3, 2);
102     //quad(2, 3, 7, 6);
103     //quad(3, 0, 4, 7);
104     //quad(4, 5, 6, 7);
105     //quad(5, 4, 0, 1);
```

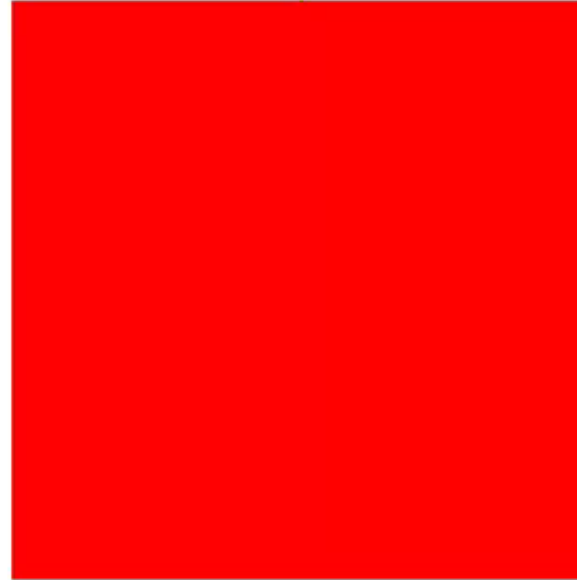
8

Ln 1, Col 1 Spaces: 4 UTF-8 CRLF JavaScript



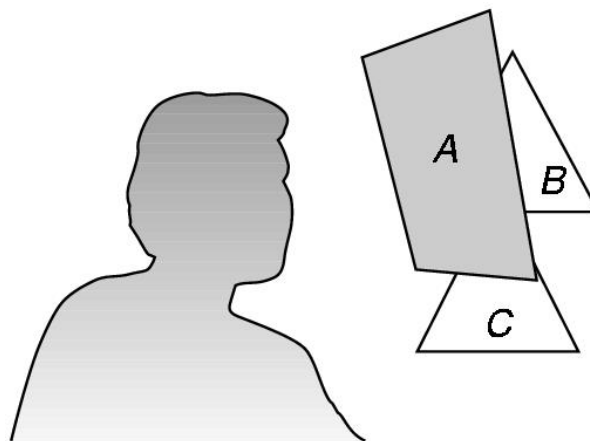


Rotate X Rotate Y Rotate Z Toggle Rotate



Hidden-Surface Removal

- To see only those surfaces in front of other surfaces
- Visible-surface algorithms or hidden-surface-removal algorithm
 - Algorithms for ordering objects
 - OpenGL uses the z-buffer algorithm that saves depth information



Hidden-Surface Problem

Using the Z-buffer Algorithm

- The algorithm uses an extra buffer, the z-buffer, to store depth information as geometry travels down the pipeline
- Depth buffer is required to available in WebGL
- It must be
 - Enabled
 - `gl.enable(gl.DEPTH_TEST);`
 - Cleared in for each render
 - `gl.clear(gl.COLOR_BUFFER_BIT | gl.DEPTH_BUFFER_BIT);`

File Edit Selection View Go Run Terminal HelpcolorCube.js - Visual Studio Code

colorCube.htmlJS colorCube.js X

C: > Users > Sun-Jeong Kim > Desktop > CG > Week06 > JS colorCube.js > init

```
36 generateColorCube();
37
38 // Configure WebGL
39 gl.viewport(0, 0, canvas.width, canvas.height);
40 gl.clearColor(1.0, 1.0, 1.0, 1.0);
41
42 // Enable hidden-surface removal
43 gl.enable(gl.DEPTH_TEST);
44
45 // Load shaders and initialize attribute buffers
46 var program = initShaders(gl, "vertex-shader", "fragment-shader");
47 gl.useProgram(program);
48
49 // Load the data into the GPU
50 var bufferId = gl.createBuffer();
51 gl.bindBuffer(gl.ARRAY_BUFFER, bufferId);
52 gl.bufferData(gl.ARRAY_BUFFER, flatten(points), gl.STATIC_DRAW);
53
54 // Associate our shader variables with our data buffer
55 var vPosition = gl.getAttribLocation(program, "vPosition");
56 gl.vertexAttribPointer(vPosition, 4, gl.FLOAT, false, 0, 0);
57 gl.enableVertexAttribArray(vPosition);
58
59 // Create a buffer object, initialize it, and associate it with
60 // the associated attribute variable in our vertex shader
61 var cBufferId = gl.createBuffer();
62 gl.bindBuffer(gl.ARRAY_BUFFER, cBufferId);
63 gl.bufferData(gl.ARRAY_BUFFER, flatten(colors), gl.STATIC_DRAW);
64
65 var vColor = gl.getAttribLocation(program, "vColor");
66 gl.vertexAttribPointer(vColor, 4, gl.FLOAT, false, 0, 0);
67 gl.enableVertexAttribArray(vColor);
68
69 thetaLoc = gl.getUniformLocation(program, "theta");
70 //gl.uniform3fv(thetaLoc, theta);
```

```
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
```

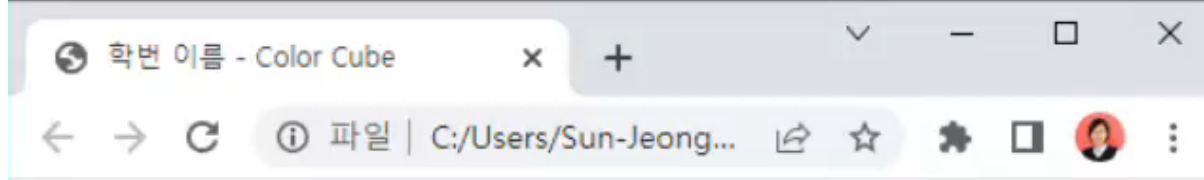
Ln 43, Col 30 Spaces: 4 UTF-8 CRLF JavaScript

colorCube.html JS colorCube.js X

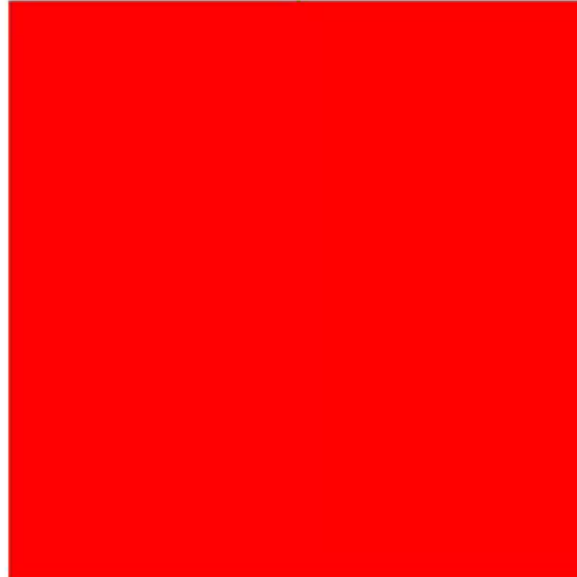
C: > Users > Sun-Jeong Kim > Desktop > CG > Week06 > JS colorCube.js > render

```
71
72 // Event listeners for buttons
73 document.getElementById("xButton").onclick = function () {
74     axis = 0;
75 };
76 document.getElementById("yButton").onclick = function () {
77     axis = 1;
78 };
79 document.getElementById("zButton").onclick = function () {
80     axis = 2;
81 };
82 document.getElementById("toggleButton").onclick = function () {
83     rotation = !rotation;
84 };
85
86 render();
87 };
88
89 function render() {
90     gl.clear(gl.COLOR_BUFFER_BIT | gl.DEPTH_BUFFER_BIT);
91
92     if( rotation ) {
93         theta[axis] += 2.0;
94     }
95     gl.uniform3fv(thetaLoc, theta)
96
97     gl.drawArrays(gl.LINES, 0, 6);
98     gl.drawArrays(gl.TRIANGLES, 6, points.length-6);
99
100     window.requestAnimationFrame(render);
101 }
102
103 function generateColorCube() {
104     quad(1, 0, 3, 2);
105     //quad(2, 3, 7, 6);
```





Rotate X Rotate Y Rotate Z Toggle Rotate



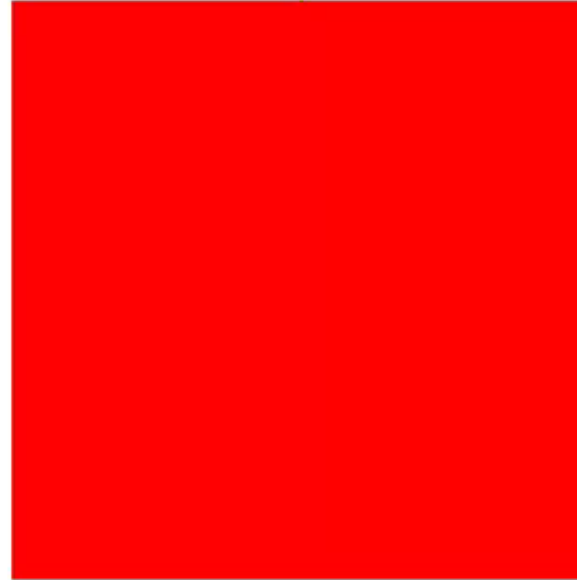

```
File Edit Selection View Go Run Terminal Help
colorCube.js - Visual Studio Code

colorCube.html JS colorCube.js X

C: > Users > Sun-Jeong Kim > Desktop > CG > Week06 > JS colorCube.js > generateColorCube

102
103 function generateColorCube() {
104     quad(1, 0, 3, 2);
105     quad(2, 3, 7, 6);
106     quad(3, 0, 4, 7);
107     //quad(4, 5, 6, 7);
108     quad(5, 4, 0, 1);
109     quad(6, 5, 1, 2);
110 }
111
112 function quad(a, b, c, d) {
113     vertexPos = [
114         vec4(-0.5, -0.5, -0.5, 1.0),
115         vec4( 0.5, -0.5, -0.5, 1.0),
116         vec4( 0.5,  0.5, -0.5, 1.0),
117         vec4(-0.5,  0.5, -0.5, 1.0),
118         vec4(-0.5, -0.5,  0.5, 1.0),
119         vec4( 0.5, -0.5,  0.5, 1.0),
120         vec4( 0.5,  0.5,  0.5, 1.0),
121         vec4(-0.5,  0.5,  0.5, 1.0)
122     ];
123
124     vertexColor = [
125         vec4(0.0, 0.0, 0.0, 1.0), // black
126         vec4(1.0, 0.0, 0.0, 1.0), // red
127         vec4(1.0, 1.0, 0.0, 1.0), // yellow
128         vec4(0.0, 1.0, 0.0, 1.0), // green
129         vec4(0.0, 0.0, 1.0, 1.0), // blue
130         vec4(1.0, 0.0, 1.0, 1.0), // magenta
131         vec4(1.0, 1.0, 1.0, 1.0), // white
132         vec4(0.0, 1.0, 1.0, 1.0) // cyan
133     ];
134
135     // We need to partition the quad into two triangles in order for WebGL
136     // to be able to render it. In this case, we create two triangles from
```

Rotate X Rotate Y Rotate Z Toggle Rotate



File Edit Selection View Go Run Terminal HelpcolorCube.js - Visual Studio Code

colorCube.htmlJS colorCube.js X

C: > Users > Sun-Jeong Kim > Desktop > CG > Week06 > JS colorCube.js > init

```
1  var gl;
2  var points = [];
3  var colors = [];
4
5  var axis = 0;
6  var theta = [0, 0, 0];
7  var thetaLoc;
8
9  var rotation = false;
10
11 window.onload = function init()
12 {
13     var canvas = document.getElementById("gl-canvas");
14
15     gl = WebGLUtils.setupWebGL(canvas);
16     if( !gl ) {
17         alert("WebGL isn't available!");
18     }
19     /*
20     // axes
21     points.push(vec4(0.0, 0.0, 0.0, 1.0));
22     points.push(vec4(1.0, 0.0, 0.0, 1.0)); // x-axis
23     colors.push(vec4(1.0, 0.0, 0.0, 1.0));
24     colors.push(vec4(1.0, 0.0, 0.0, 1.0));
25
26     points.push(vec4(0.0, 0.0, 0.0, 1.0));
27     points.push(vec4(0.0, 1.0, 0.0, 1.0)); // y-axis
28     colors.push(vec4(0.0, 1.0, 0.0, 1.0));
29     colors.push(vec4(0.0, 1.0, 0.0, 1.0));
30
31     points.push(vec4(0.0, 0.0, 0.0, 1.0));
32     points.push(vec4(0.0, 0.0, 1.0, 1.0)); // z-axis
33     colors.push(vec4(0.0, 0.0, 1.0, 1.0));
34     colors.push(vec4(0.0, 0.0, 1.0, 1.0));
35     */
```

```
1  gl = WebGLUtils.setupWebGL(canvas);
2  if( !gl ) {
3      alert("WebGL isn't available!");
4  }
5
6  // axes
7  points.push(vec4(0.0, 0.0, 0.0, 1.0));
8  points.push(vec4(1.0, 0.0, 0.0, 1.0)); // x-axis
9  colors.push(vec4(1.0, 0.0, 0.0, 1.0));
10 colors.push(vec4(1.0, 0.0, 0.0, 1.0));
11
12 points.push(vec4(0.0, 0.0, 0.0, 1.0));
13 points.push(vec4(0.0, 1.0, 0.0, 1.0)); // y-axis
14 colors.push(vec4(0.0, 1.0, 0.0, 1.0));
15 colors.push(vec4(0.0, 1.0, 0.0, 1.0));
16
17 points.push(vec4(0.0, 0.0, 0.0, 1.0));
18 points.push(vec4(0.0, 0.0, 1.0, 1.0)); // z-axis
19 colors.push(vec4(0.0, 0.0, 1.0, 1.0));
20 colors.push(vec4(0.0, 0.0, 1.0, 1.0));
```

Ln 35, Col 3 Spaces: 4 UTF-8 CRLF JavaScript

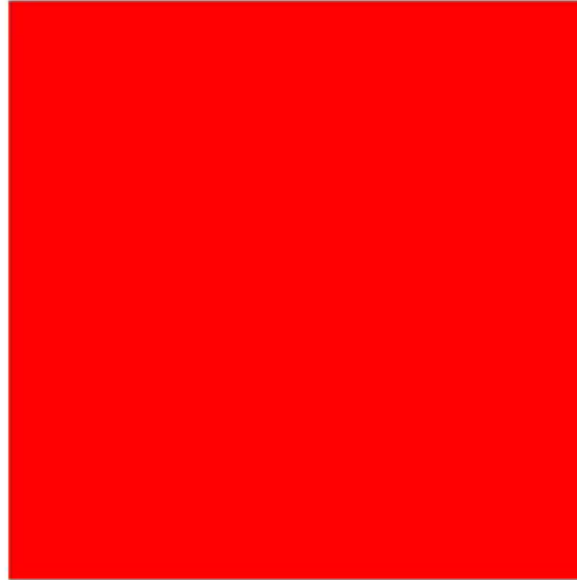
colorCube.html JS colorCube.js X

C: > Users > Sun-Jeong Kim > Desktop > CG > Week06 > JS colorCube.js > generateColorCube

```
76     document.getElementById("yButton").onclick = function () {
77         axis = 1;
78     };
79     document.getElementById("zButton").onclick = function () {
80         axis = 2;
81     };
82     document.getElementById("toggleButton").onclick = function () {
83         rotation = !rotation;
84     };
85
86     render();
87 };
88
89 function render() {
90     gl.clear(gl.COLOR_BUFFER_BIT | gl.DEPTH_BUFFER_BIT);
91
92     if( rotation ) {
93         theta[axis] += 2.0;
94     }
95     gl.uniform3fv(thetaLoc, theta)
96
97     //gl.drawArrays(gl.LINES, 0, 6);
98     gl.drawArrays(gl.TRIANGLES, 0, points.length);
99
100     window.requestAnimationFrame(render);
101 }
102
103 function generateColorCube() {
104     quad(1, 0, 3, 2);
105     quad(2, 3, 7, 6);
106     quad(3, 0, 4, 7);
107     quad(4, 5, 6, 7);
108     quad(5, 4, 0, 1);
109     quad(6, 5, 1, 2);
110 }
```

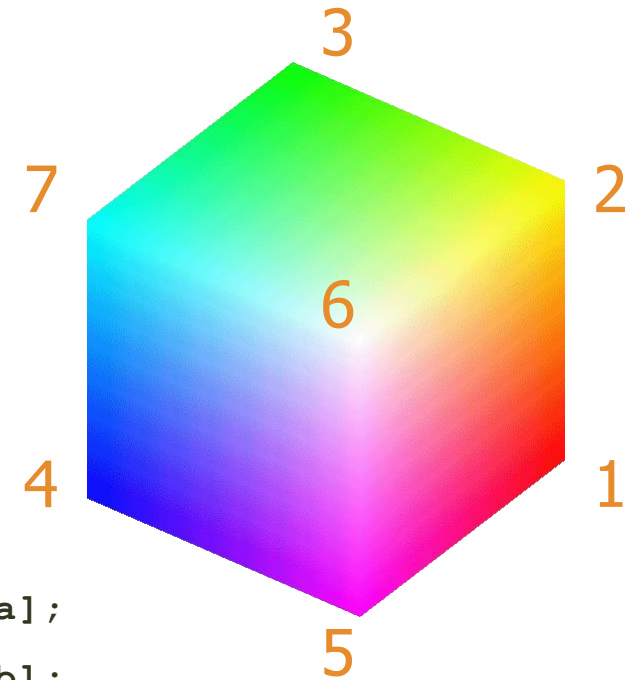


Rotate X Rotate Y Rotate Z Toggle Rotate



Color Cube (1)

```
float vertex_pos[8][3]= { {-1.0, -1.0, -1.0}, {1.0, -1.0, -1.0},  
                          {1.0, 1.0, -1.0}, {-1.0, 1.0, -1.0}, {-1.0, -1.0, 1.0},  
                          {1.0, -1.0, 1.0}, {1.0, 1.0, 1.0}, {-1.0, 1.0, 1.0}};  
  
float vertex_color[8][3] = { {0.0, 0.0, 0.0}, {1.0, 0.0, 0.0},  
                             {1.0, 1.0, 0.0}, {0.0, 1.0, 0.0}, {0.0, 0.0, 1.0},  
                             {1.0, 0.0, 1.0}, {1.0, 1.0, 1.0}, {0.0, 1.0, 1.0}};  
  
int Index = 0;  
void quad( int a, int b, int c, int d )  
{  
    colors[Index] = vertex_color[a];    points[Index] = vertex_pos[a];  
    Index++;  
    colors[Index] = vertex_color[b];    points[Index] = vertex_pos[b];  
    Index++;  
    colors[Index] = vertex_color[c];    points[Index] = vertex_pos[c];  
    Index++;  
  
    colors[Index] = vertex_color[a];    points[Index] = vertex_pos[a];  
    Index++;  
    colors[Index] = vertex_color[c];    points[Index] = vertex_pos[c];  
    Index++;  
    colors[Index] = vertex_color[b];    points[Index] = vertex_pos[b];  
    Index++;  
}
```



Color Cube (2)

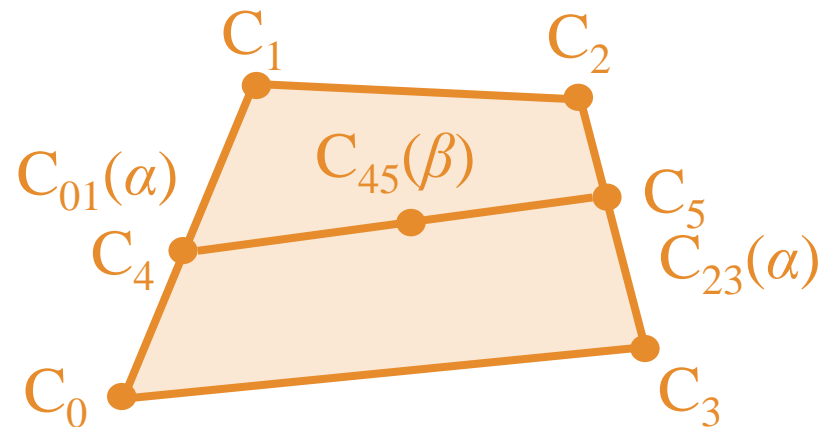
```
void generateColorCube()  
{  
    quad( 1, 0, 3, 2 );  
    quad( 2, 3, 7, 6 );  
    quad( 3, 0, 4, 7 );  
    quad( 4, 5, 6, 7 );  
    quad( 5, 4, 0, 1 );  
    quad( 6, 5, 1, 2 );  
}
```

- Bilinear interpolation

$$C_{01}(\alpha) = (1 - \alpha)C_0 + \alpha C_1$$

$$C_{23}(\alpha) = (1 - \alpha)C_2 + \alpha C_3$$

$$C_{45}(\beta) = (1 - \beta)C_4 + \beta C_5$$

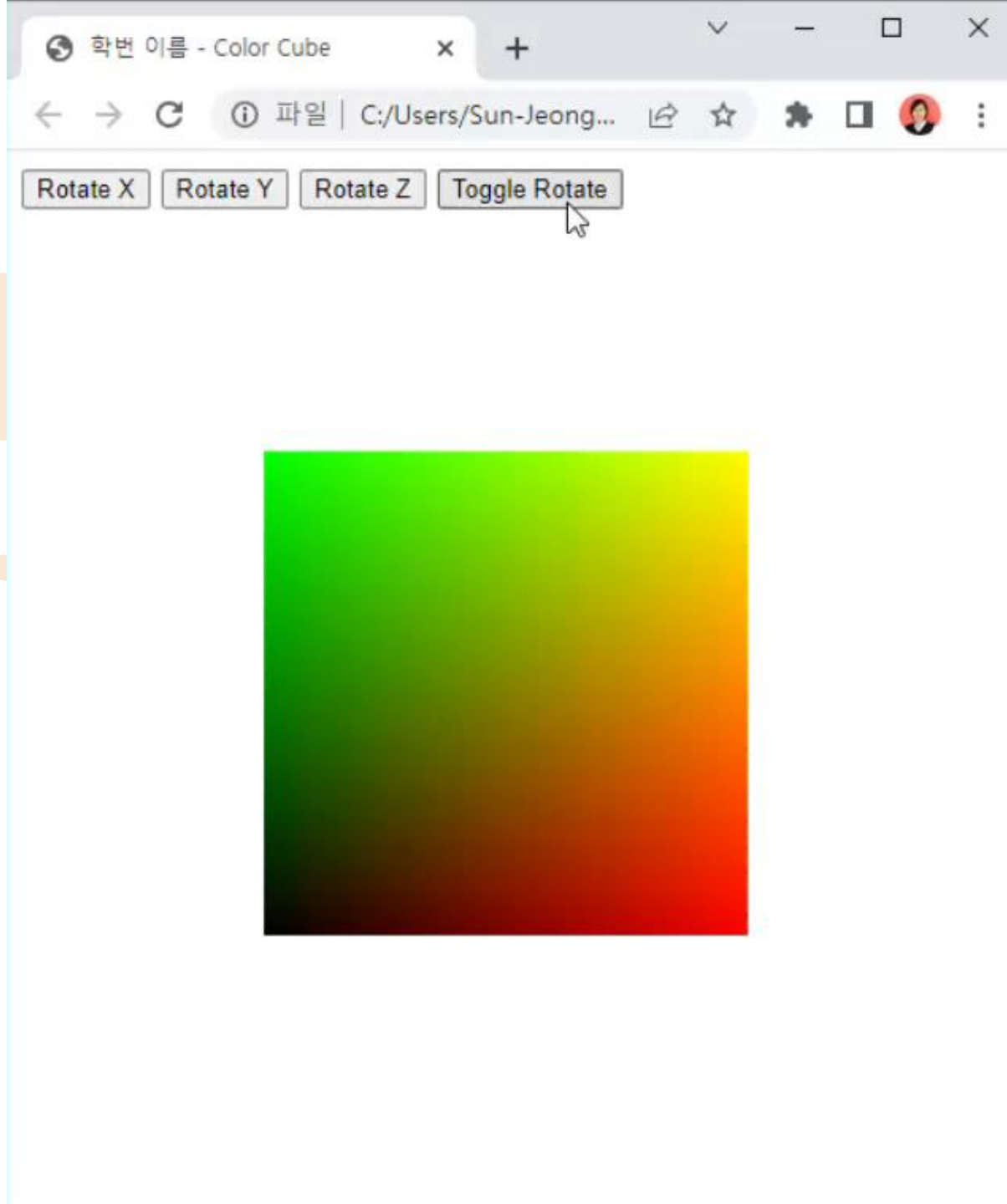



```

colorCube.html  JS colorCube.js X
C: > Users > Sun-Jeong Kim > Desktop > CG > Week06 > JS colorCube.js > quad

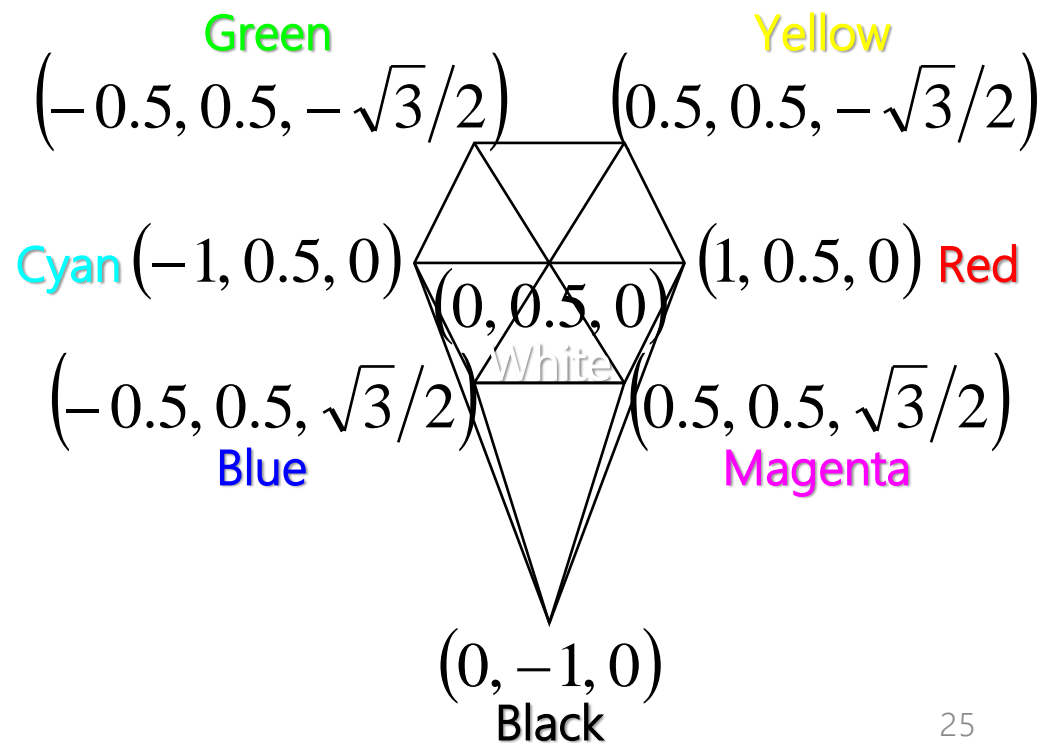
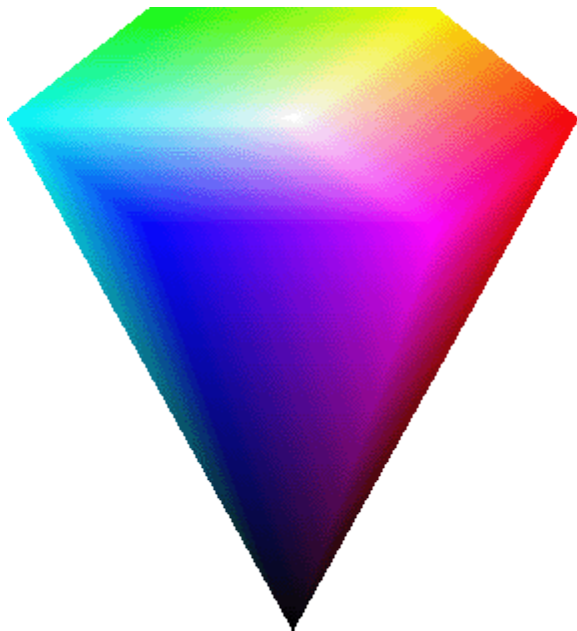
111
112 function quad(a, b, c, d) {
113     vertexPos = [
114         vec4(-0.5, -0.5, -0.5, 1.0),
115         vec4( 0.5, -0.5, -0.5, 1.0),
116         vec4( 0.5,  0.5, -0.5, 1.0),
117         vec4(-0.5,  0.5, -0.5, 1.0),
118         vec4(-0.5, -0.5,  0.5, 1.0),
119         vec4( 0.5, -0.5,  0.5, 1.0),
120         vec4( 0.5,  0.5,  0.5, 1.0),
121         vec4(-0.5,  0.5,  0.5, 1.0)
122     ];
123
124     vertexColor = [
125         vec4(0.0, 0.0, 0.0, 1.0), // black
126         vec4(1.0, 0.0, 0.0, 1.0), // red
127         vec4(1.0, 1.0, 0.0, 1.0), // yellow
128         vec4(0.0, 1.0, 0.0, 1.0), // green
129         vec4(0.0, 0.0, 1.0, 1.0), // blue
130         vec4(1.0, 0.0, 1.0, 1.0), // magenta
131         vec4(1.0, 1.0, 1.0, 1.0), // white
132         vec4(0.0, 1.0, 1.0, 1.0)  // cyan
133     ];
134
135     // We need to partition the quad into two triangles in order for WebGL
136     // to be able to render it. In this case, we create two triangles from
137     // the quad indices.
138     var index = [ a, b, c, a, c, d ];
139     for(var i=0; i<index.length; i++) {
140         points.push(vertexPos[index[i]]);
141         //colors.push(vertexColor[a]);
142         colors.push(vertexColor[index[i]]);
143     }
144 }
145

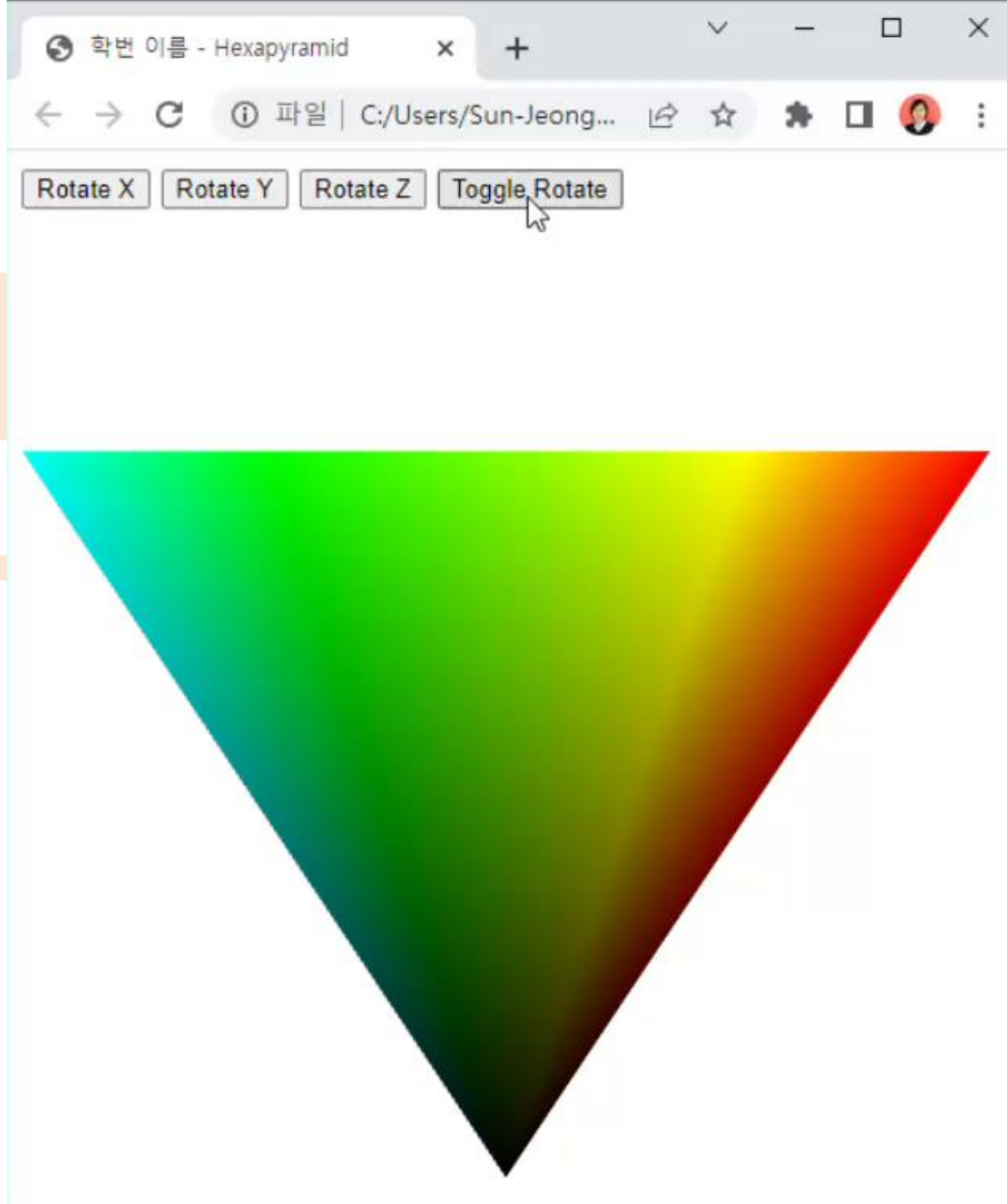
```



연습 문제 (1)

- Hexagonal Pyramid를 그리시오.
 - colorCube.html, js → hexaPyramid.html, js로 복사
 - Vertex 8개
 - Triangle 12개





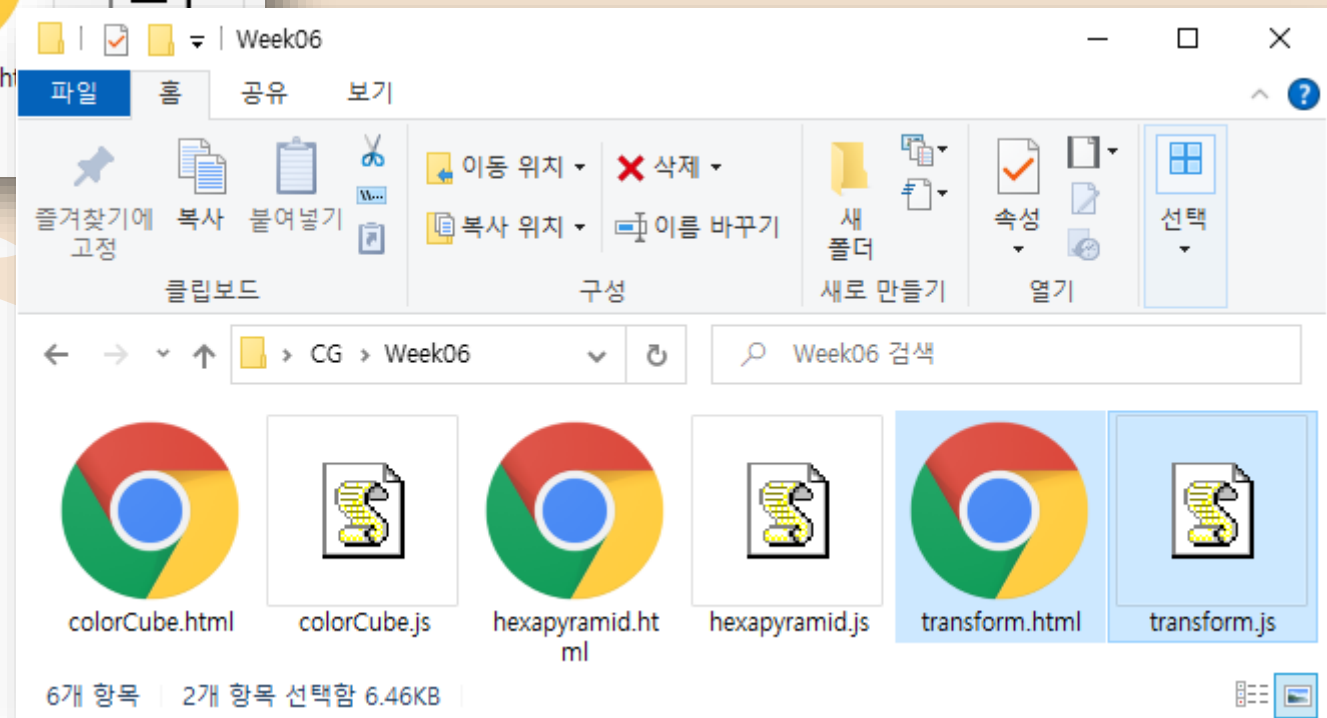
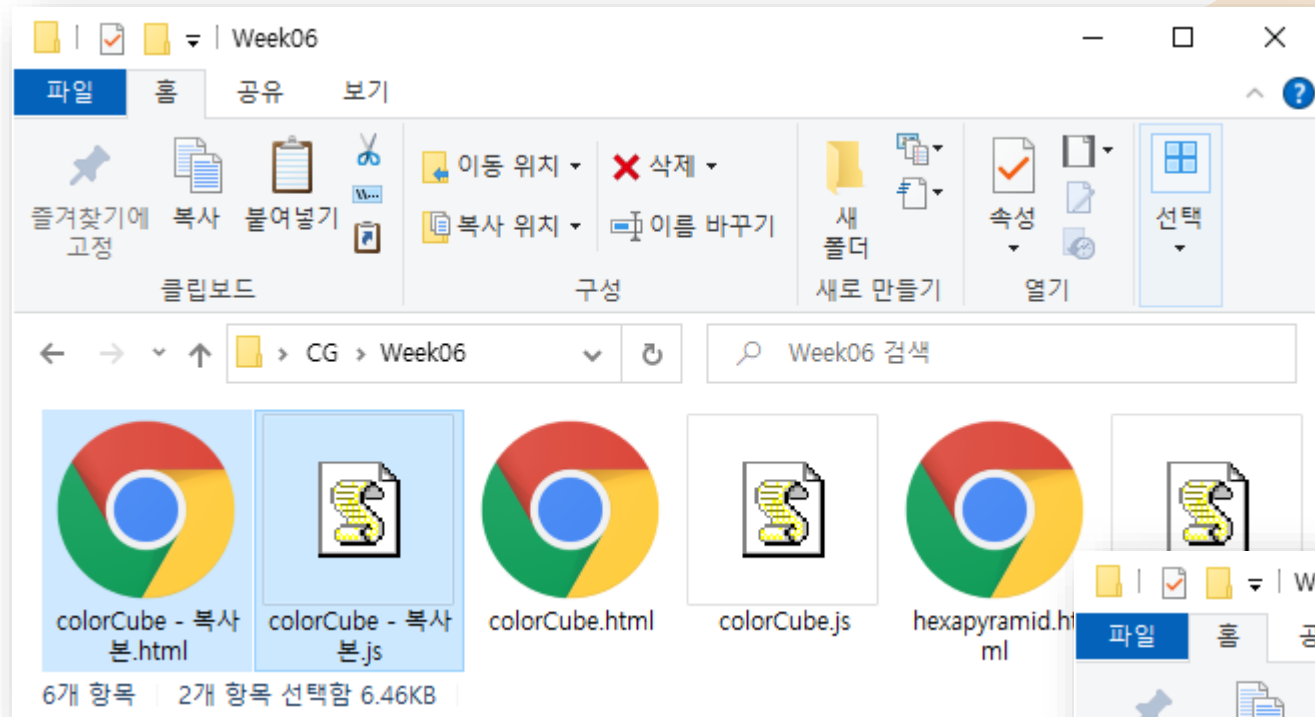
Line-Preserving Transformations

- Two classes of transformations of importance to CG that preserves lines
 - Affine and projective transformations
- Homogeneous coordinates
 - Vectors: $(x, y, z, \underline{0})$, points: $(x, y, z, w) = (x/w, y/w, z/w, \underline{1})$
 - 4x4 matrices: modeling, viewing, and projection

- OpenGL pipeline



- Model-view: to position objects relative to camera
- Projection: to specify clipping volume and map vertices to a normalized coordinate system



File Edit Selection View Go Run Terminal Helptransform.html - Visual Studio Code

colorCube.htmlJS colorCube.jshexapyramid.htmlJS hexapyramid.jstransform.html XJS transform.js

C: > Users > Sun-Jeong Kim > Desktop > CG > Week06 > <> transform.html > html > head > title

```
1 <!DOCTYPE html>
2 <html>
3 <head>
4 <title>학번 이름 - Transformations</title>
5 <script id="vertex-shader" type="x-shader/x-vertex">
6 attribute vec4 vPosition;
7 attribute vec4 vColor;
8 uniform vec3 theta;
9 varying vec4 fColor;
10
11 void main() {
12     // Compute the sines and cosines of theta for each of
13     // the three axes in one computation
14     vec3 angles = radians(theta);
15     vec3 c = cos(angles);
16     vec3 s = sin(angles);
17
18     // Remember: these matrices are column-major
19     mat4 rx = mat4( 1.0, 0.0, 0.0, 0.0,
20                   0.0, c.x, s.x, 0.0,
21                   0.0, -s.x, c.x, 0.0,
22                   0.0, 0.0, 0.0, 1.0 );
23
24     mat4 ry = mat4( c.y, 0.0, -s.y, 0.0,
25                   0.0, 1.0, 0.0, 0.0,
26                   s.y, 0.0, c.y, 0.0,
27                   0.0, 0.0, 0.0, 1.0 );
28
29     mat4 rz = mat4( c.z, s.z, 0.0, 0.0,
30                   -s.z, c.z, 0.0, 0.0,
31                   0.0, 0.0, 1.0, 0.0,
32                   0.0, 0.0, 0.0, 1.0 );
33
34     gl_Position = rz * ry * rx * vPosition;
35     fColor = vColor;
```

Ln 4, Col 39Spaces: 4UTF-8CRLFHTML

Restricted Mode0 0 0

transform.html - Visual Studio Code

File Edit Selection View Go Run Terminal Help

colorCube.html JS colorCube.js hexapyramid.html JS hexapyramid.js transform.html X JS transform.js

C: > Users > Sun-Jeong Kim > Desktop > CG > Week06 > <> transform.html > html > head > script

```
36     }
37     </script>
38
39     <script id="fragment-shader" type="x-shader/x-fragment">
40     precision mediump float;
41     varying vec4 fColor;
42
43     void main() {
44     |     gl_FragColor = fColor;
45     }
46     </script>
47
48     <script type="text/javascript" src="../../Common/webgl-utils.js"></script>
49     <script type="text/javascript" src="../../Common/initShaders.js"></script>
50     <script type="text/javascript" src="../../Common/MV.js"></script>
51     <script type="text/javascript" src="transform.js"></script>
52     </head>
53     <body>
54     <div>
55     |     <button id="xButton">Rotate X</button>
56     |     <button id="yButton">Rotate Y</button>
57     |     <button id="zButton">Rotate Z</button>
58     |     <button id="toggleButton">Toggle Rotate</button>
59     </div>
60     <canvas id="gl-canvas" width="512" height="512">
61     |     Oops... your browser doesn't support the HTML5 canvas element!
62     </canvas>
63     </body>
64 </html>
```

Ln 51, Col 54 Spaces: 4 UTF-8 CRLF HTML

File Edit Selection View Go Run Terminal Helptransform.js - Visual Studio Code

colorCube.htmlJS colorCube.jshexapyramid.htmlJS hexapyramid.jstransform.htmlJS transform.js X

C: > Users > Sun-Jeong Kim > Desktop > CG > Week06 > JS transform.js > init

```
1  var gl;
2  var points = [];
3  var colors = [];
4
5  var axis = 0;
6  var theta = [0, 0, 0];
7  var thetaLoc;
8
9  var rotation = false;
10
11 window.onload = function init()
12 {
13     var canvas = document.getElementById("gl-canvas");
14
15     gl = WebGLUtils.setupWebGL(canvas);
16     if( !gl ) {
17         alert("WebGL isn't available!");
18     }
19
20     generateColorCube();
21     generateHexaPyramid();
22
23     // Configure WebGL
24     gl.viewport(0, 0, canvas.width, canvas.height);
25     gl.clearColor(0.9, 0.9, 0.9, 1.0);
26
27     // Enable hidden-surface removal
28     gl.enable(gl.DEPTH_TEST);
29
30     // Load shaders and initialize attribute buffers
31     var program = initShaders(gl, "vertex-shader", "fragment-shader");
32     gl.useProgram(program);
33
34     // Load the data into the GPU
35     var bufferId = gl.createBuffer();
```

31

Restricted Mode 0 0 Ln 25, Col 39 Spaces: 4 UTF-8 CRLF JavaScript

File Edit Selection View Go Run Terminal Helptransform.js - Visual Studio Code

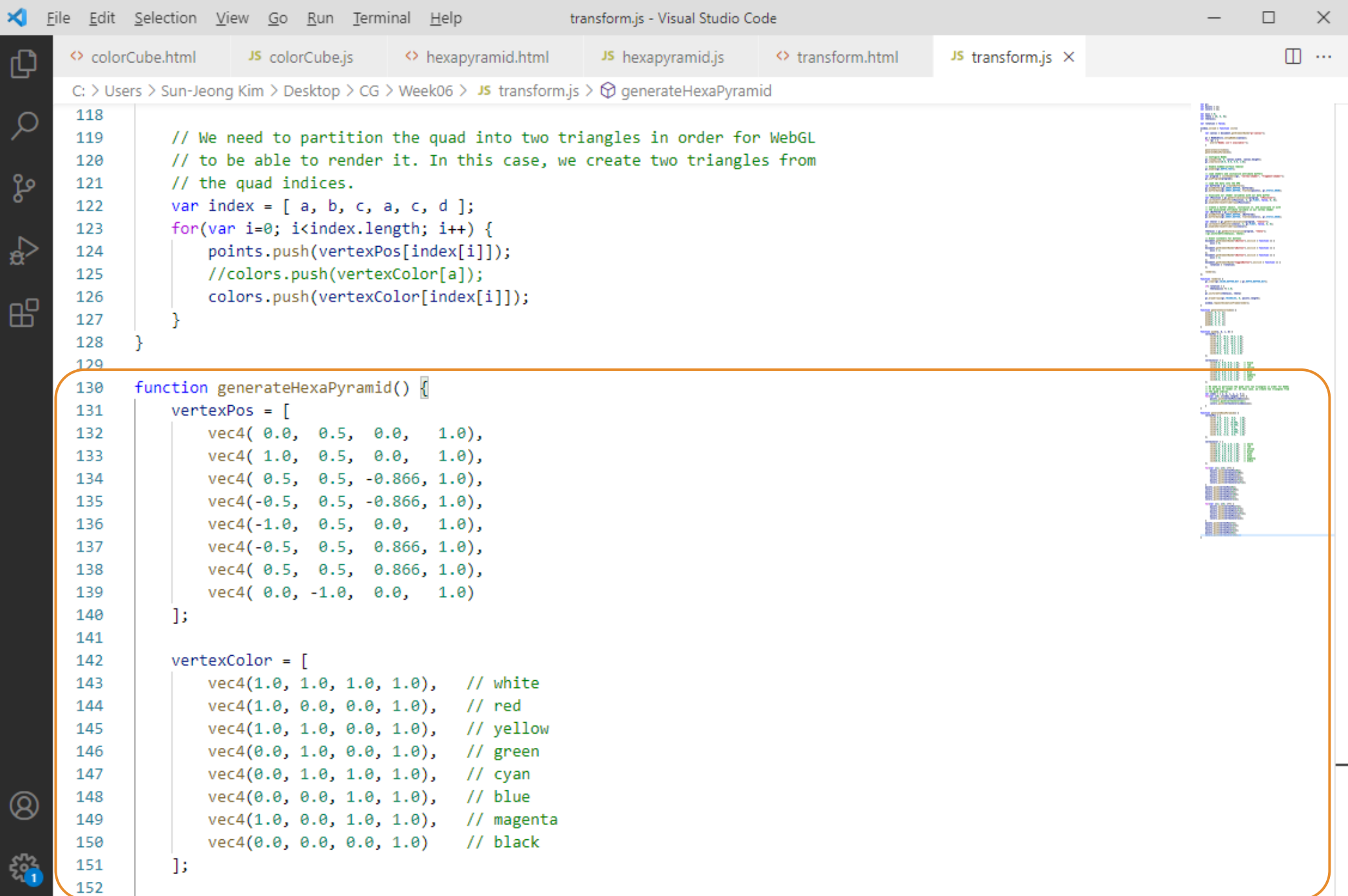
colorCube.htmlJS colorCube.jshexapyramid.htmlJS hexapyramid.jstransform.htmlJS transform.js X

C: > Users > Sun-Jeong Kim > Desktop > CG > Week06 > JS transform.js > render

```
73
74 function render() {
75     gl.clear(gl.COLOR_BUFFER_BIT | gl.DEPTH_BUFFER_BIT);
76
77     if( rotation ) {
78         theta[axis] += 2.0;
79     }
80     gl.uniform3fv(thetaLoc, theta)
81
82     gl.drawArrays(gl.TRIANGLES, 0, points.length);
83
84     window.requestAnimationFrame(render);
85 }
86
87 function generateColorCube() {
88     quad(1, 0, 3, 2);
89     quad(2, 3, 7, 6);
90     quad(3, 0, 4, 7);
91     quad(4, 5, 6, 7);
92     quad(5, 4, 0, 1);
93     quad(6, 5, 1, 2);
94 }
95
96 function quad(a, b, c, d) {
97     vertexPos = [
98         vec4(-0.5, -0.5, -0.5, 1.0),
99         vec4( 0.5, -0.5, -0.5, 1.0),
100        vec4( 0.5,  0.5, -0.5, 1.0),
101        vec4(-0.5,  0.5, -0.5, 1.0),
102        vec4(-0.5, -0.5,  0.5, 1.0),
103        vec4( 0.5, -0.5,  0.5, 1.0),
104        vec4( 0.5,  0.5,  0.5, 1.0),
105        vec4(-0.5,  0.5,  0.5, 1.0)
106     ];
107 }
```

32

Restricted Mode 0 0 Ln 81, Col 1 Spaces: 4 UTF-8 CRLF JavaScript



File Edit Selection View Go Run Terminal Helptransform.js - Visual Studio Code

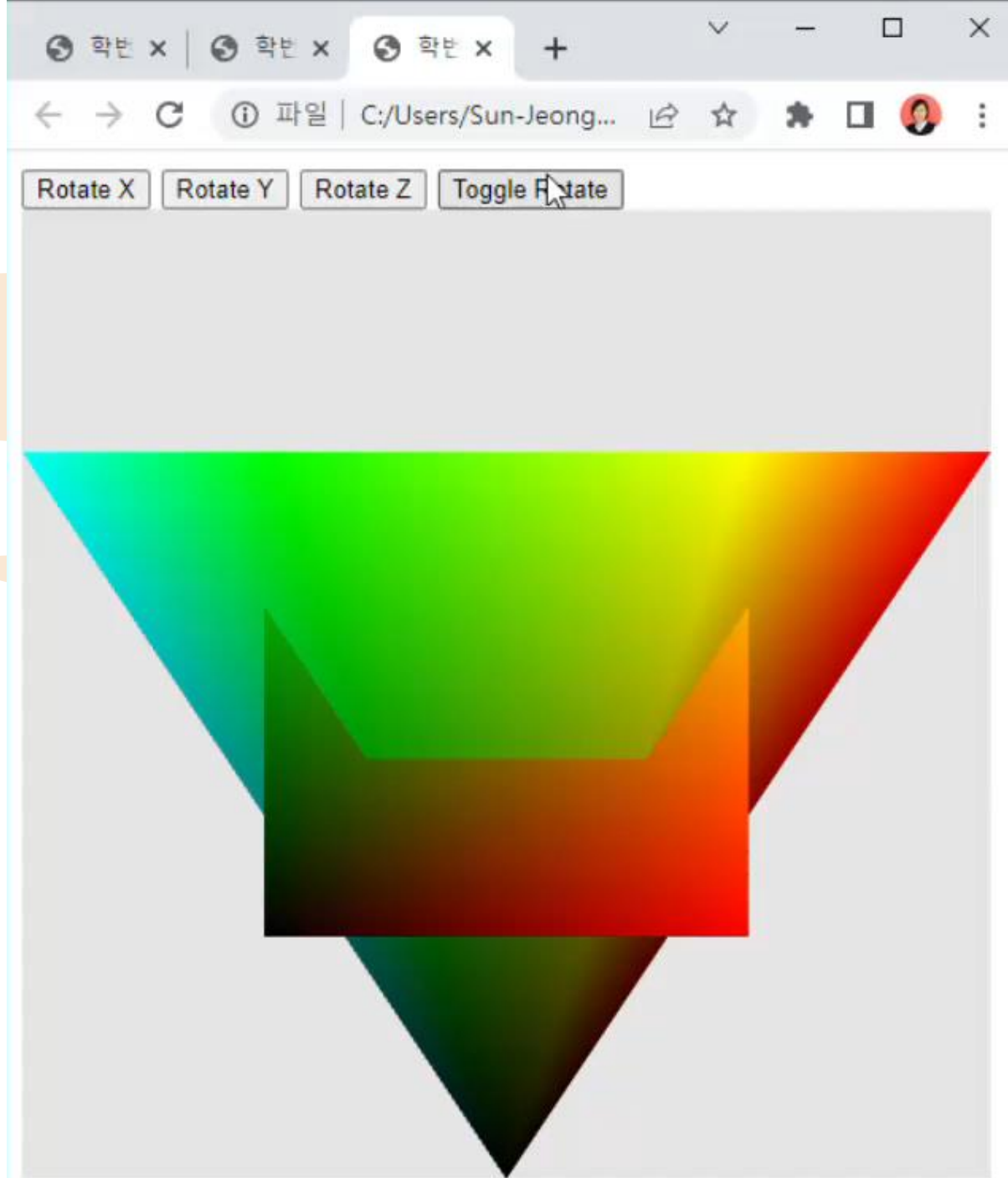
colorCube.htmlJS colorCube.jshexapyramid.htmlJS hexapyramid.jstransform.htmlJS transform.js X

C: > Users > Sun-Jeong Kim > Desktop > CG > Week06 > JS transform.js > generateHexaPyramid

```
152
153     for(var i=1; i<6; i++) {
154         points.push(vertexPos[0]);
155         colors.push(vertexColor[0]);
156         points.push(vertexPos[i]);
157         colors.push(vertexColor[i]);
158         points.push(vertexPos[i+1]);
159         colors.push(vertexColor[i+1]);
160     }
161     points.push(vertexPos[0]);
162     colors.push(vertexColor[0]);
163     points.push(vertexPos[6]);
164     colors.push(vertexColor[6]);
165     points.push(vertexPos[1]);
166     colors.push(vertexColor[1]);
167
168     for(var i=1; i<6; i++) {
169         points.push(vertexPos[7]);
170         colors.push(vertexColor[7]);
171         points.push(vertexPos[i+1]);
172         colors.push(vertexColor[i+1]);
173         points.push(vertexPos[i]);
174         colors.push(vertexColor[i]);
175     }
176     points.push(vertexPos[7]);
177     colors.push(vertexColor[7]);
178     points.push(vertexPos[1]);
179     colors.push(vertexColor[1]);
180     points.push(vertexPos[6]);
181     colors.push(vertexColor[6]);
182
183
```

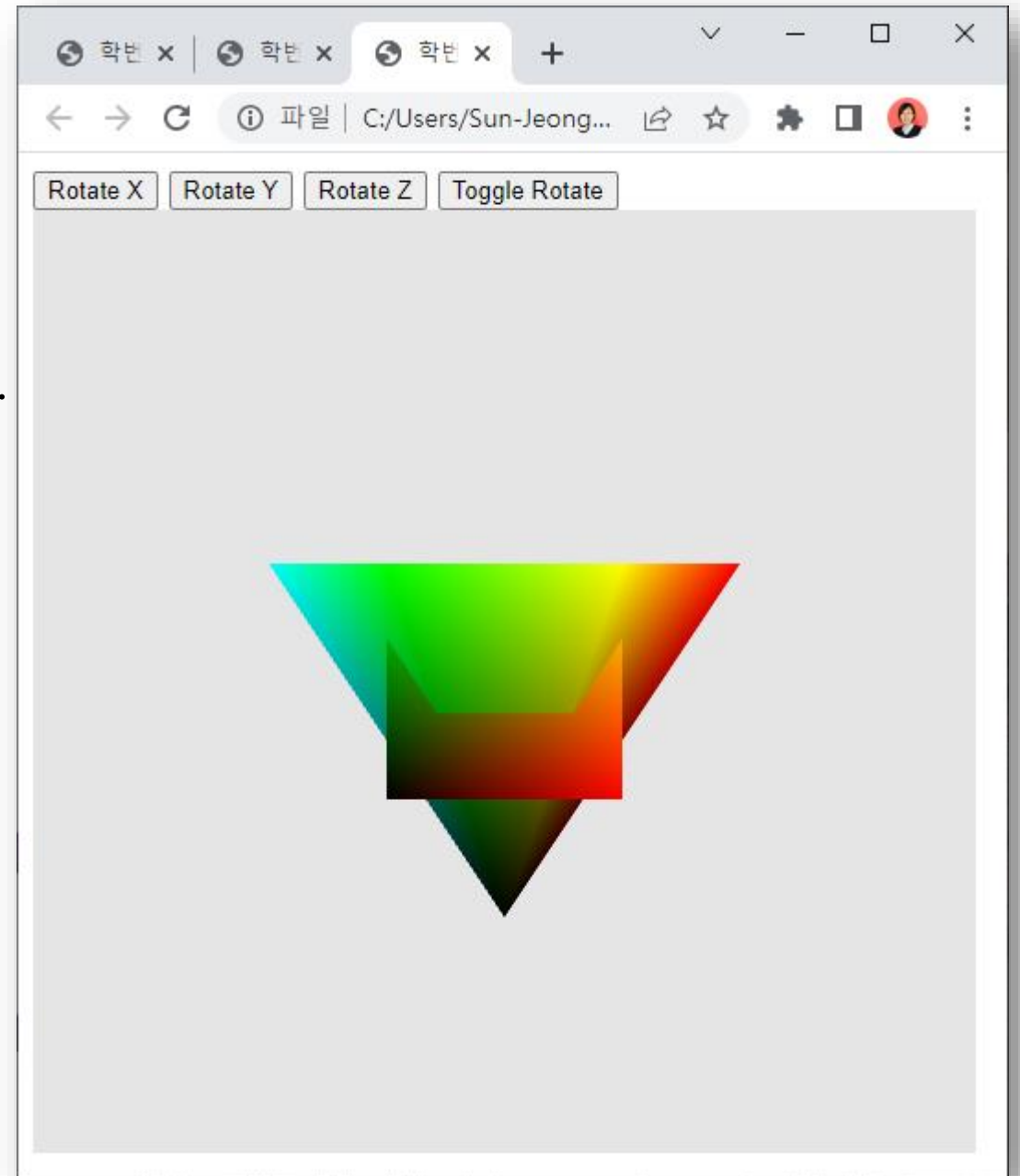
```
152
153
154
155
156
157
158
159
160
161
162
163
164
165
166
167
168
169
170
171
172
173
174
175
176
177
178
179
180
181
182
183
```

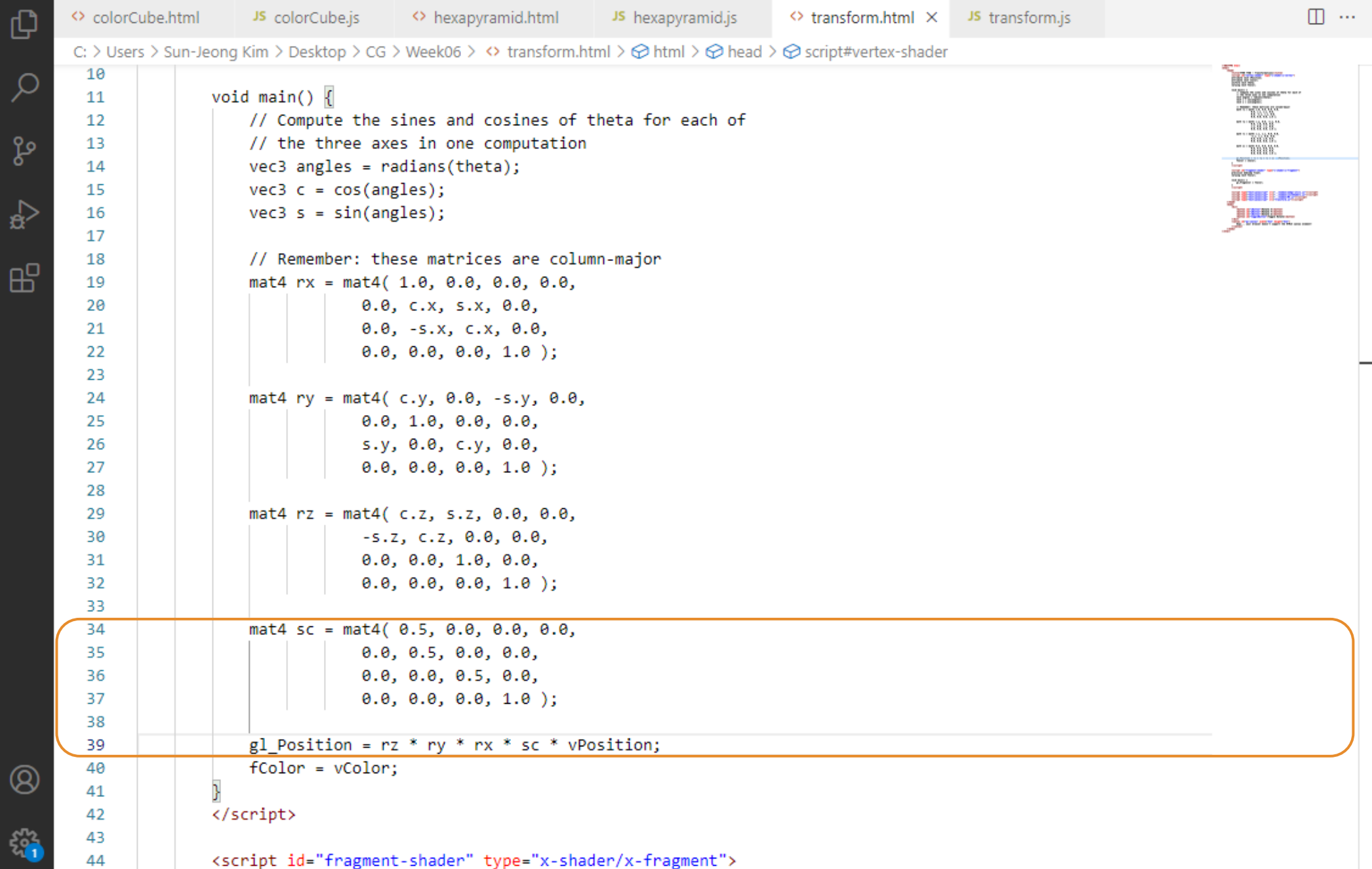
Ln 181, Col 33 Spaces: 4 UTF-8 CRLF JavaScript



연습 문제 (2)

- 오브젝트들의 크기를 반으로 줄이시오.

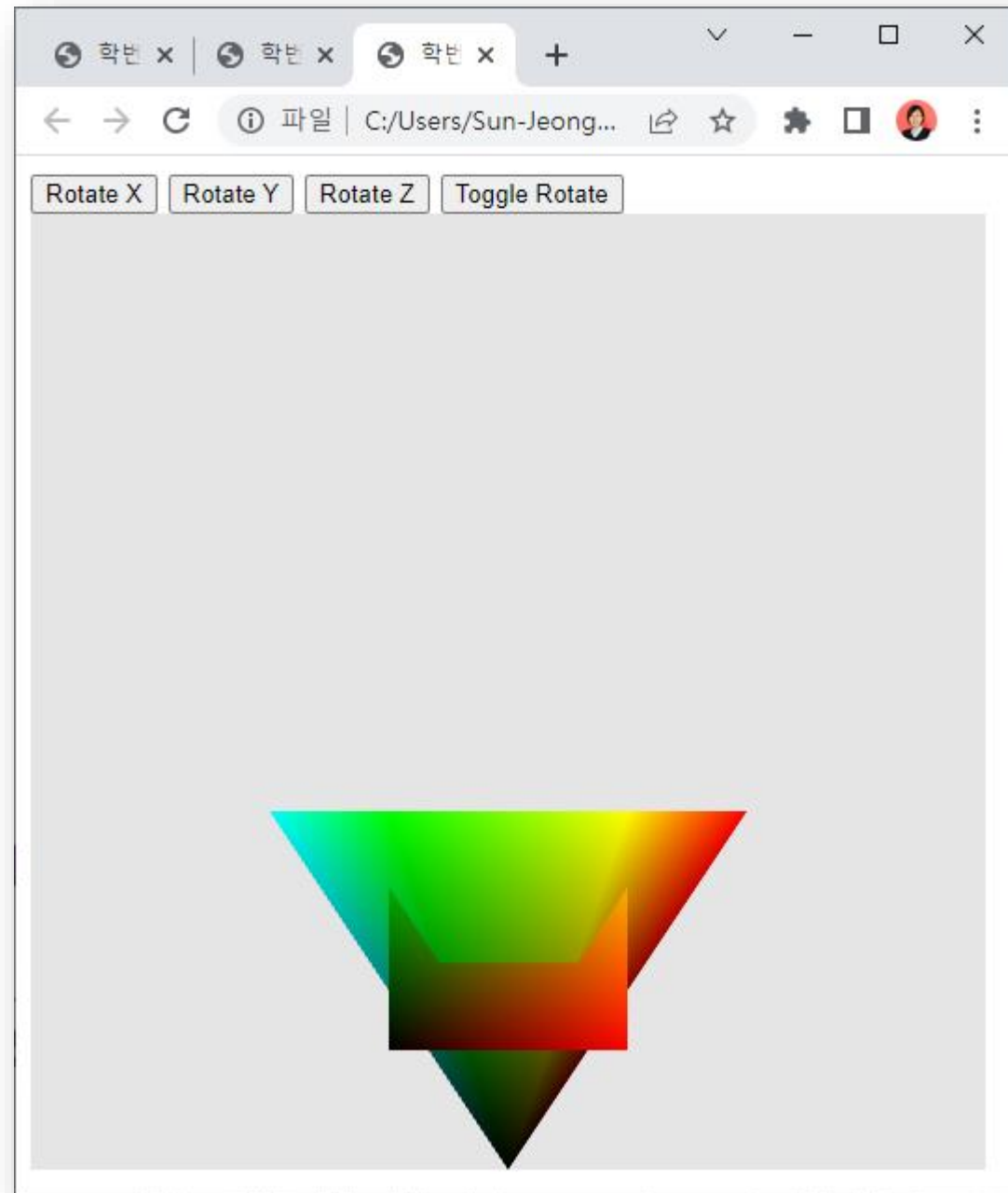




```
10
11 void main() {
12     // Compute the sines and cosines of theta for each of
13     // the three axes in one computation
14     vec3 angles = radians(theta);
15     vec3 c = cos(angles);
16     vec3 s = sin(angles);
17
18     // Remember: these matrices are column-major
19     mat4 rx = mat4( 1.0, 0.0, 0.0, 0.0,
20                   0.0, c.x, s.x, 0.0,
21                   0.0, -s.x, c.x, 0.0,
22                   0.0, 0.0, 0.0, 1.0 );
23
24     mat4 ry = mat4( c.y, 0.0, -s.y, 0.0,
25                   0.0, 1.0, 0.0, 0.0,
26                   s.y, 0.0, c.y, 0.0,
27                   0.0, 0.0, 0.0, 1.0 );
28
29     mat4 rz = mat4( c.z, s.z, 0.0, 0.0,
30                   -s.z, c.z, 0.0, 0.0,
31                   0.0, 0.0, 1.0, 0.0,
32                   0.0, 0.0, 0.0, 1.0 );
33
34     mat4 sc = mat4( 0.5, 0.0, 0.0, 0.0,
35                   0.0, 0.5, 0.0, 0.0,
36                   0.0, 0.0, 0.5, 0.0,
37                   0.0, 0.0, 0.0, 1.0 );
38
39     gl_Position = rz * ry * rx * sc * vPosition;
40     fColor = vColor;
41 }
42 </script>
43
44 <script id="fragment-shader" type="x-shader/x-fragment">
```

연습 문제 (3)

- 오브젝트들을 y 축 방향으로 -0.5 만큼 이동 시키시오.



C: > Users > Sun-Jeong Kim > Desktop > CG > Week06 > transform.html > html > head > script#vertex-shader

```

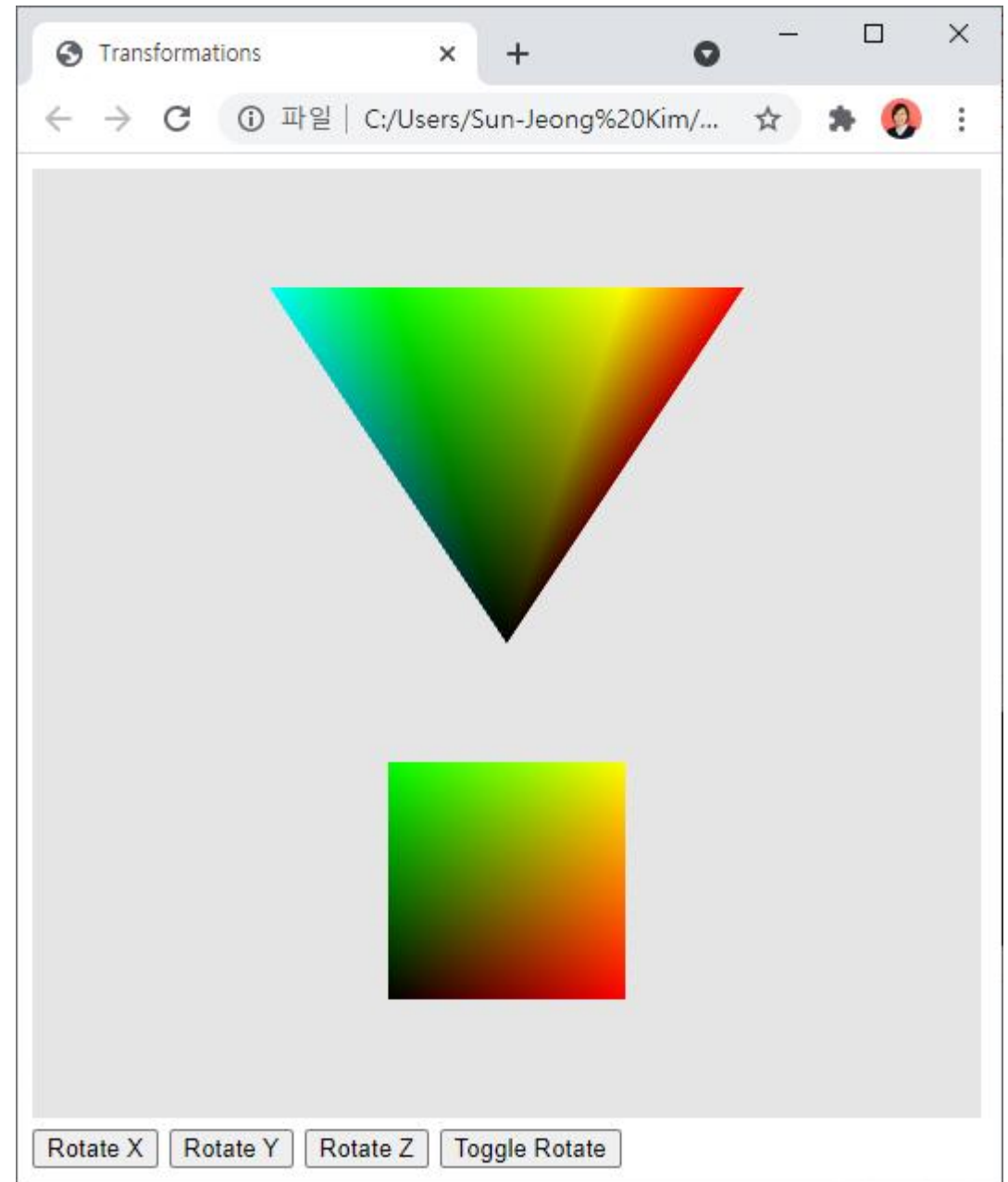
11 void main() {
12     // Compute the sines and cosines of theta for each of
13     // the three axes in one computation
14     vec3 angles = radians(theta);
15     vec3 c = cos(angles);
16     vec3 s = sin(angles);
17
18     // Remember: these matrices are column-major
19     mat4 rx = mat4( 1.0, 0.0, 0.0, 0.0,
20                   0.0, c.x, s.x, 0.0,
21                   0.0, -s.x, c.x, 0.0,
22                   0.0, 0.0, 0.0, 1.0 );
23
24     mat4 ry = mat4( c.y, 0.0, -s.y, 0.0,
25                   0.0, 1.0, 0.0, 0.0,
26                   s.y, 0.0, c.y, 0.0,
27                   0.0, 0.0, 0.0, 1.0 );
28
29     mat4 rz = mat4( c.z, s.z, 0.0, 0.0,
30                   -s.z, c.z, 0.0, 0.0,
31                   0.0, 0.0, 1.0, 0.0,
32                   0.0, 0.0, 0.0, 1.0 );
33
34     mat4 sc = mat4( 0.5, 0.0, 0.0, 0.0,
35                   0.0, 0.5, 0.0, 0.0,
36                   0.0, 0.0, 0.5, 0.0,
37                   0.0, 0.0, 0.0, 1.0 );
38
39     mat4 tr = mat4( 1.0, 0.0, 0.0, 0.0,
40                   0.0, 1.0, 0.0, 0.0,
41                   0.0, 0.0, 1.0, 0.0,
42                   0.0, -0.5, 0.0, 1.0 );
43
44     gl_Position = tr * rz * ry * rx * sc * vPosition;
45     fColor = vColor;

```



연습 문제 (4)

- Hexagonal Pyramid를 y축 방향으로 0.5만큼 이동 시키시오.



File Edit Selection View Go Run Terminal Help

transform.html - Visual Studio Code

colorCube.html JS colorCube.js hexapyramid.html JS hexapyramid.js transform.html X JS transform.js

C: > Users > Sun-Jeong Kim > Desktop > CG > Week06 > <> transform.html > html > head > script#vertex-shader

```
1 <!DOCTYPE html>
2 <html>
3   <head>
4     <title>학번 이름 - Transformations</title>
5     <script id="vertex-shader" type="x-shader/x-vertex">
6       attribute vec4 vPosition;
7       attribute vec4 vColor;
8       uniform vec3 theta;
9       uniform vec3 displ;
10      varying vec4 fColor;
11
12      void main() {
13        // Compute the sines and cosines of theta for each of
14        // the three axes in one computation
15        vec3 angles = radians(theta);
16        vec3 c = cos(angles);
17        vec3 s = sin(angles);
18
19        // Remember: these matrices are column-major
20        mat4 rx = mat4( 1.0, 0.0, 0.0, 0.0,
21                      0.0, c.x, s.x, 0.0,
22                      0.0, -s.x, c.x, 0.0,
23                      0.0, 0.0, 0.0, 1.0 );
24
25        mat4 ry = mat4( c.y, 0.0, -s.y, 0.0,
26                      0.0, 1.0, 0.0, 0.0,
27                      s.y, 0.0, c.y, 0.0,
28                      0.0, 0.0, 0.0, 1.0 );
29
30        mat4 rz = mat4( c.z, s.z, 0.0, 0.0,
31                      -s.z, c.z, 0.0, 0.0,
32                      0.0, 0.0, 1.0, 0.0,
33                      0.0, 0.0, 0.0, 1.0 );
34
35        mat4 sc = mat4( 0.5, 0.0, 0.0, 0.0,
```

Ln 9, Col 27 Spaces: 4 UTF-8 CRLF HTML

Restricted Mode 0 0 0

C: > Users > Sun-Jeong Kim > Desktop > CG > Week06 > <> transform.html > html > head > script#vertex-shader

```

12 void main() {
13     // Compute the sines and cosines of theta for each of
14     // the three axes in one computation
15     vec3 angles = radians(theta);
16     vec3 c = cos(angles);
17     vec3 s = sin(angles);
18
19     // Remember: these matrices are column-major
20     mat4 rx = mat4( 1.0, 0.0, 0.0, 0.0,
21                   0.0, c.x, s.x, 0.0,
22                   0.0, -s.x, c.x, 0.0,
23                   0.0, 0.0, 0.0, 1.0 );
24
25     mat4 ry = mat4( c.y, 0.0, -s.y, 0.0,
26                   0.0, 1.0, 0.0, 0.0,
27                   s.y, 0.0, c.y, 0.0,
28                   0.0, 0.0, 0.0, 1.0 );
29
30     mat4 rz = mat4( c.z, s.z, 0.0, 0.0,
31                   -s.z, c.z, 0.0, 0.0,
32                   0.0, 0.0, 1.0, 0.0,
33                   0.0, 0.0, 0.0, 1.0 );
34
35     mat4 sc = mat4( 0.5, 0.0, 0.0, 0.0,
36                   0.0, 0.5, 0.0, 0.0,
37                   0.0, 0.0, 0.5, 0.0,
38                   0.0, 0.0, 0.0, 1.0 );
39
40     mat4 tr = mat4( 1.0, 0.0, 0.0, 0.0,
41                   0.0, 1.0, 0.0, 0.0,
42                   0.0, 0.0, 1.0, 0.0,
43                   displ.x, displ.y, displ.z, 1.0 );
44
45     gl_Position = tr * rz * ry * rx * sc * vPosition;
46     fColor = vColor;

```



File Edit Selection View Go Run Terminal Helptransform.js - Visual Studio Code

colorCube.htmlJS colorCube.jshexapyramid.htmlJS hexapyramid.jstransform.htmlJS transform.js X

C: > Users > Sun-Jeong Kim > Desktop > CG > Week06 > JS transform.js > ...

```
1  var gl;
2  var points = [];
3  var colors = [];
4
5  var axis = 0;
6  var theta = [0, 0, 0];
7  var thetaLoc;
8  var displ = [0, 0, 0];
9  var displLoc;
10
11 var rotation = false;
12
13 window.onload = function init()
14 {
15     var canvas = document.getElementById("gl-canvas");
16
17     gl = WebGLUtils.setupWebGL(canvas);
18     if( !gl ) {
19         alert("WebGL isn't available!");
20     }
21
22     generateColorCube();
23     generateHexaPyramid();
24
25     // Configure WebGL
26     gl.viewport(0, 0, canvas.width, canvas.height);
27     gl.clearColor(0.9, 0.9, 0.9, 1.0);
28
29     // Enable hidden-surface removal
30     gl.enable(gl.DEPTH_TEST);
31
32     // Load shaders and initialize attribute buffers
33     var program = initShaders(gl, "vertex-shader", "fragment-shader");
34     gl.useProgram(program);
35 }
```

13

Restricted Mode 0 0 Ln 9, Col 14 Spaces: 4 UTF-8 CRLF JavaScript

File Edit Selection View Go Run Terminal Helptransform.js - Visual Studio Code

colorCube.htmlJS colorCube.jshexapyramid.htmlJS hexapyramid.jstransform.htmlJS transform.js X

C: > Users > Sun-Jeong Kim > Desktop > CG > Week06 > JS transform.js > init

```
36 // Load the data into the GPU
37 var bufferId = gl.createBuffer();
38 gl.bindBuffer(gl.ARRAY_BUFFER, bufferId);
39 gl.bufferData(gl.ARRAY_BUFFER, flatten(points), gl.STATIC_DRAW);
40
41 // Associate our shader variables with our data buffer
42 var vPosition = gl.getAttribLocation(program, "vPosition");
43 gl.vertexAttribPointer(vPosition, 4, gl.FLOAT, false, 0, 0);
44 gl.enableVertexAttribArray(vPosition);
45
46 // Create a buffer object, initialize it, and associate it with
47 // the associated attribute variable in our vertex shader
48 var cBufferId = gl.createBuffer();
49 gl.bindBuffer(gl.ARRAY_BUFFER, cBufferId);
50 gl.bufferData(gl.ARRAY_BUFFER, flatten(colors), gl.STATIC_DRAW);
51
52 var vColor = gl.getAttribLocation(program, "vColor");
53 gl.vertexAttribPointer(vColor, 4, gl.FLOAT, false, 0, 0);
54 gl.enableVertexAttribArray(vColor);
55
56 thetaLoc = gl.getUniformLocation(program, "theta");
57 //gl.uniform3fv(thetaLoc, theta);
58 displLoc = gl.getUniformLocation(program, "displ");
59 //gl.uniform3fv(displLoc, displ);
60
61 // Event listeners for buttons
62 document.getElementById("xButton").onclick = function () {
63 |   axis = 0;
64 | };
65 document.getElementById("yButton").onclick = function () {
66 |   axis = 1;
67 | };
68 document.getElementById("zButton").onclick = function () {
69 |   axis = 2;
70 | };
```

14

Restricted Mode 0 0 Ln 59, Col 36 Spaces: 4 UTF-8 CRLF JavaScript

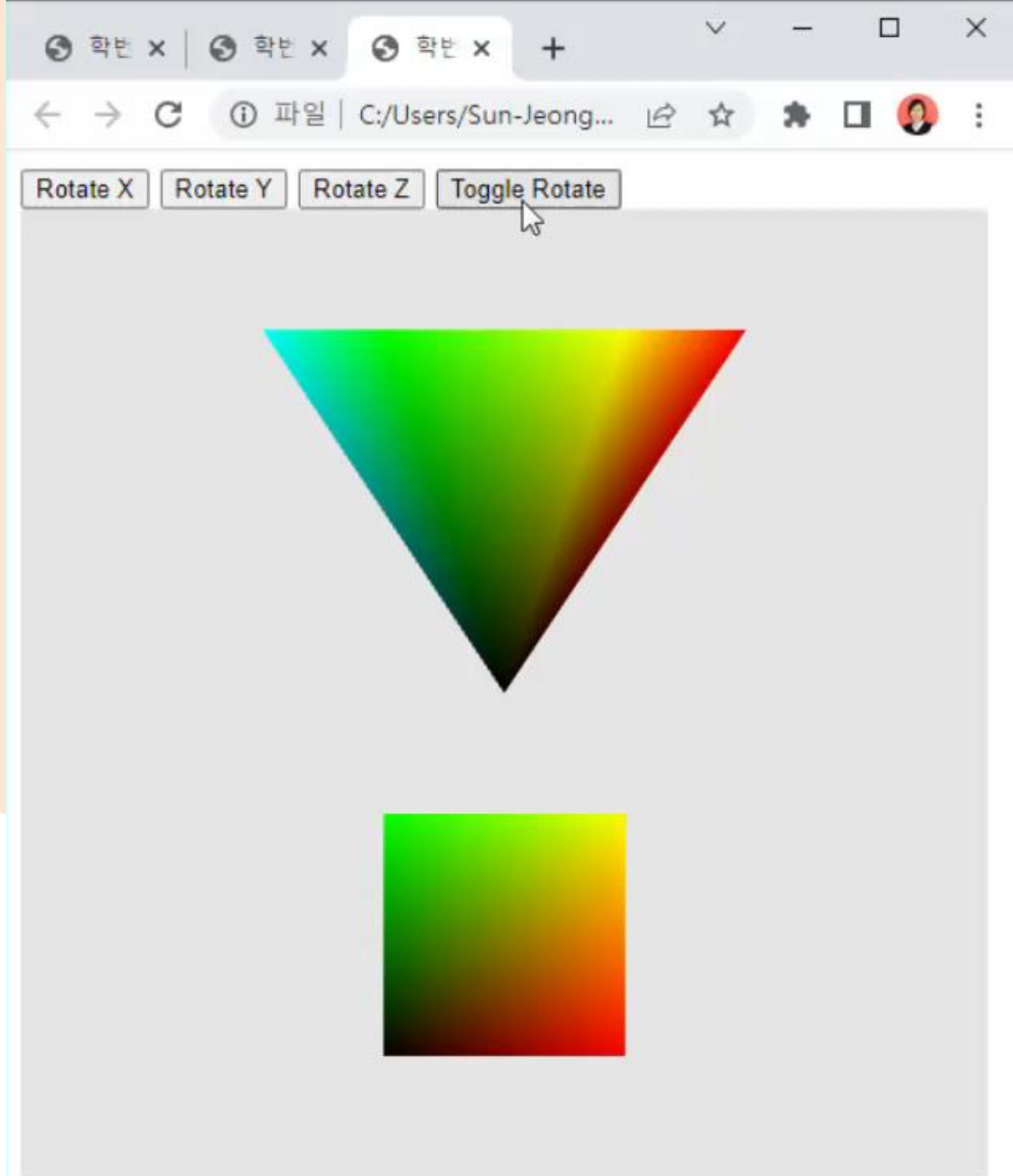
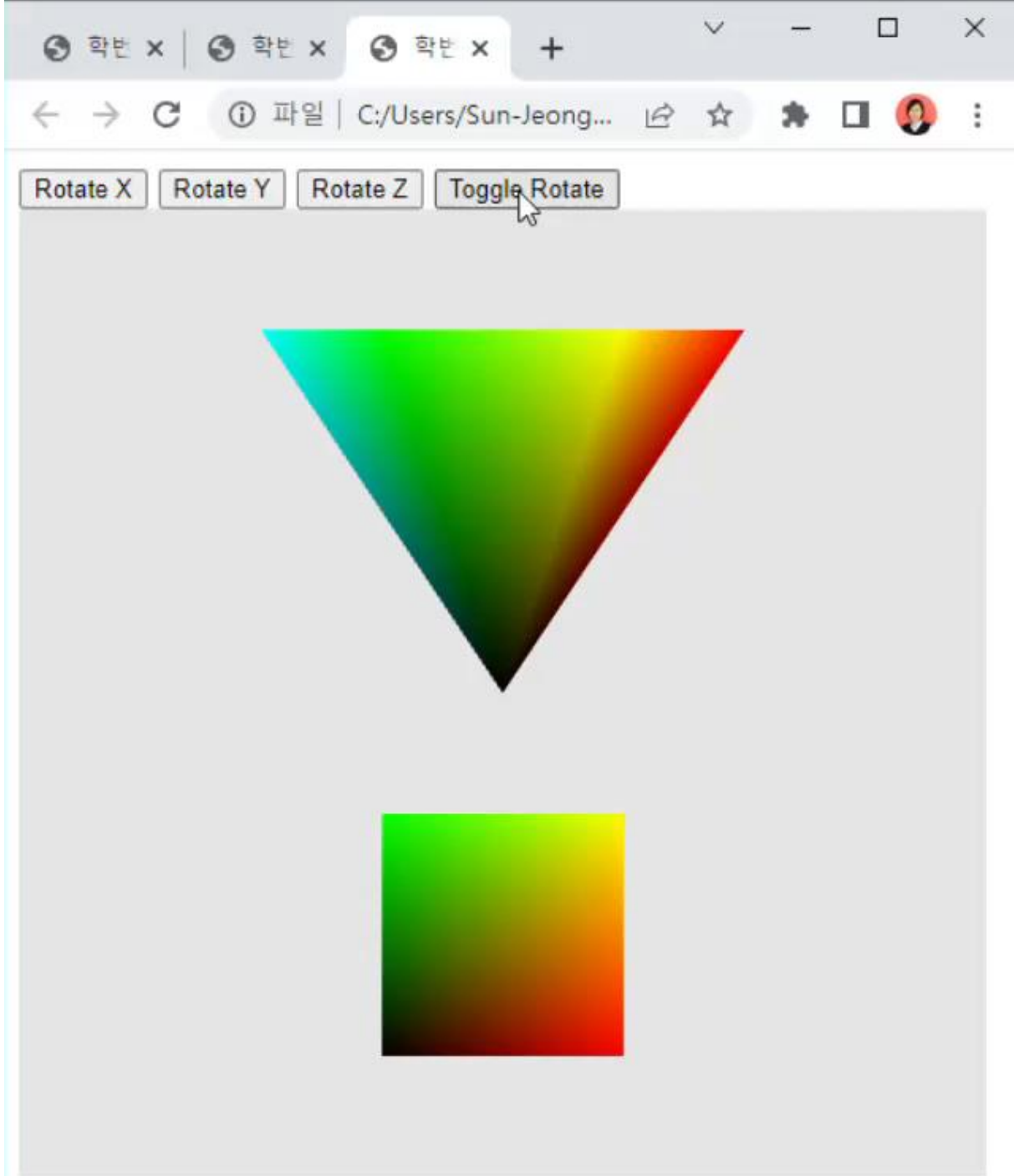
File Edit Selection View Go Run Terminal Helptransform.js - Visual Studio Code

colorCube.htmlJS colorCube.jshexapyramid.htmlJS hexapyramid.jstransform.htmlJS transform.js X

C: > Users > Sun-Jeong Kim > Desktop > CG > Week06 > JS transform.js > render

```
71     document.getElementById("toggleButton").onclick = function () {
72         rotation = !rotation;
73     };
74
75     render();
76 };
77
78 function render() {
79     gl.clear(gl.COLOR_BUFFER_BIT | gl.DEPTH_BUFFER_BIT);
80
81     if( rotation ) {
82         theta[axis] += 2.0;
83     }
84     gl.uniform3fv(thetaLoc, theta)
85
86     // Draw a color cube (12 triangles * 3 = 36 vertices)
87     displ[1] = -0.5;
88     gl.uniform3fv(displLoc, displ);
89     gl.drawArrays(gl.TRIANGLES, 0, 36);
90
91     // Draw a hexagonal pyramid (12 triangles * 3 = 36 vertices)
92     displ[1] = 0.5;
93     gl.uniform3fv(displLoc, displ);
94     gl.drawArrays(gl.TRIANGLES, 36, 36);
95
96     window.requestAnimationFrame(render);
97 }
98
99 function generateColorCube() {
100     quad(1, 0, 3, 2);
101     quad(2, 3, 7, 6);
102     quad(3, 0, 4, 7);
103     quad(4, 5, 6, 7);
104     quad(5, 4, 0, 1);
105     quad(6, 5, 1, 2);
```

Ln 94, Col 35 Spaces: 4 UTF-8 CRLF JavaScript



C: > Users > Sun-Jeong Kim > Desktop > CG > Week06 > <> transform.html > html > head > script#vertex-shader

```

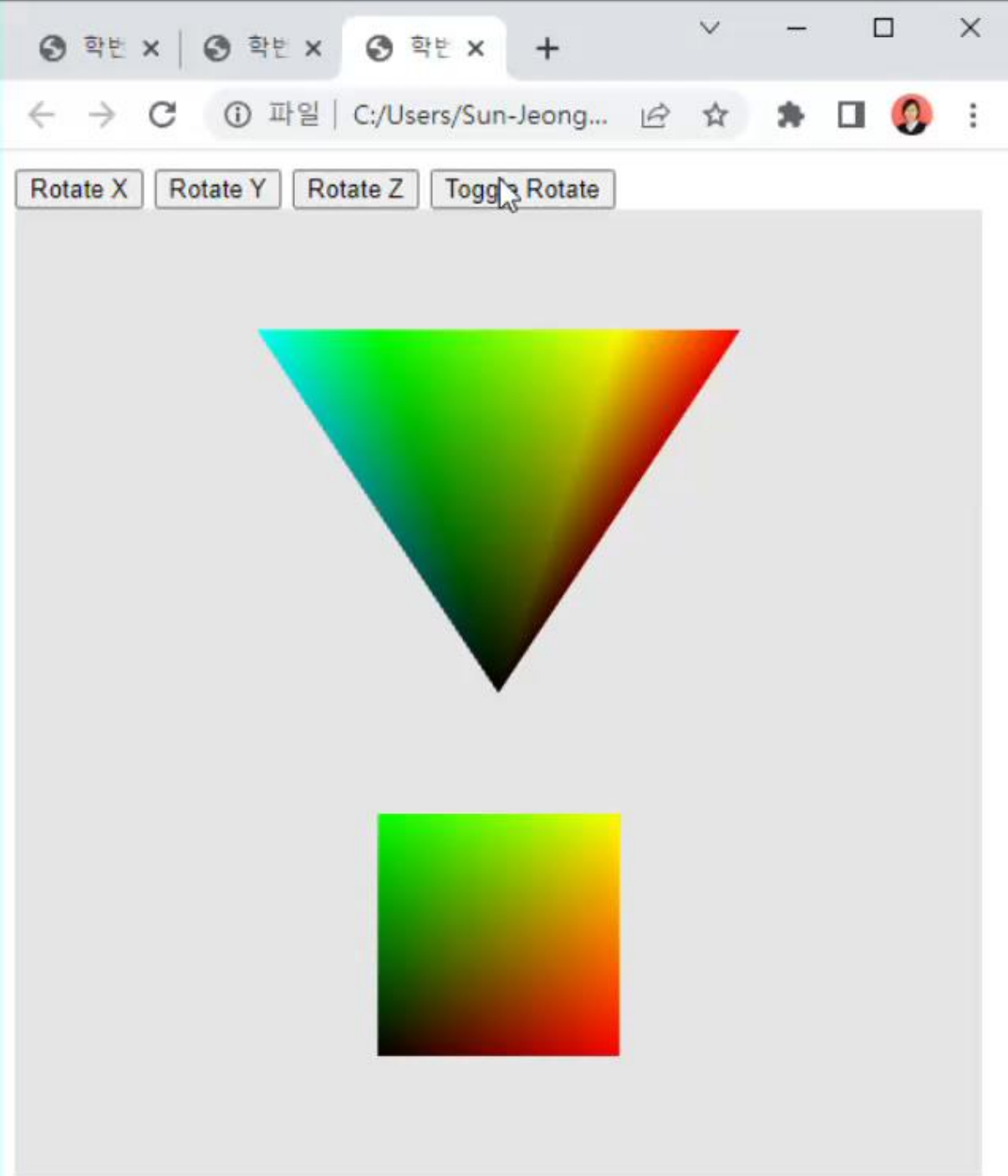
12 void main() {
13     // Compute the sines and cosines of theta for each of
14     // the three axes in one computation
15     vec3 angles = radians(theta);
16     vec3 c = cos(angles);
17     vec3 s = sin(angles);
18
19     // Remember: these matrices are column-major
20     mat4 rx = mat4( 1.0, 0.0, 0.0, 0.0,
21                   0.0, c.x, s.x, 0.0,
22                   0.0, -s.x, c.x, 0.0,
23                   0.0, 0.0, 0.0, 1.0 );
24
25     mat4 ry = mat4( c.y, 0.0, -s.y, 0.0,
26                   0.0, 1.0, 0.0, 0.0,
27                   s.y, 0.0, c.y, 0.0,
28                   0.0, 0.0, 0.0, 1.0 );
29
30     mat4 rz = mat4( c.z, s.z, 0.0, 0.0,
31                   -s.z, c.z, 0.0, 0.0,
32                   0.0, 0.0, 1.0, 0.0,
33                   0.0, 0.0, 0.0, 1.0 );
34
35     mat4 sc = mat4( 0.5, 0.0, 0.0, 0.0,
36                   0.0, 0.5, 0.0, 0.0,
37                   0.0, 0.0, 0.5, 0.0,
38                   0.0, 0.0, 0.0, 1.0 );
39
40     mat4 tr = mat4( 1.0, 0.0, 0.0, 0.0,
41                   0.0, 1.0, 0.0, 0.0,
42                   0.0, 0.0, 1.0, 0.0,
43                   displ.x, displ.y, displ.z, 1.0 );
44
45     gl_Position = rz * ry * rx * tr * sc * vPosition;
46     fColor = vColor;

```



연습 문제 (5)

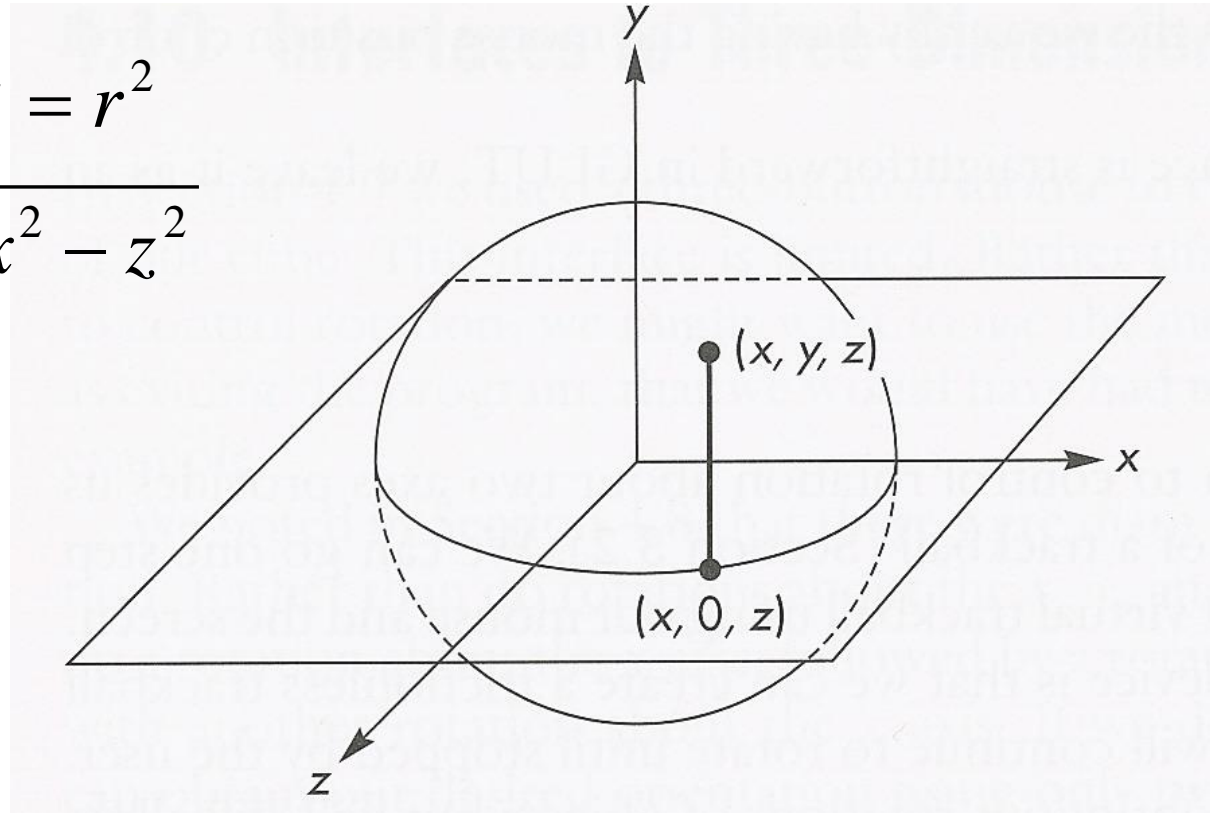
- 두 오브젝트가 서로 반대 방향으로 회전하도록 구현하시오.



Rotation with a Virtual Trackball (1)

- Projection of the trackball position to the plane

$$x^2 + y^2 + z^2 = r^2$$
$$\therefore y = \sqrt{r^2 - x^2 - z^2}$$



Rotation with a Virtual Trackball (2)

- Determination of the orientation of a plane

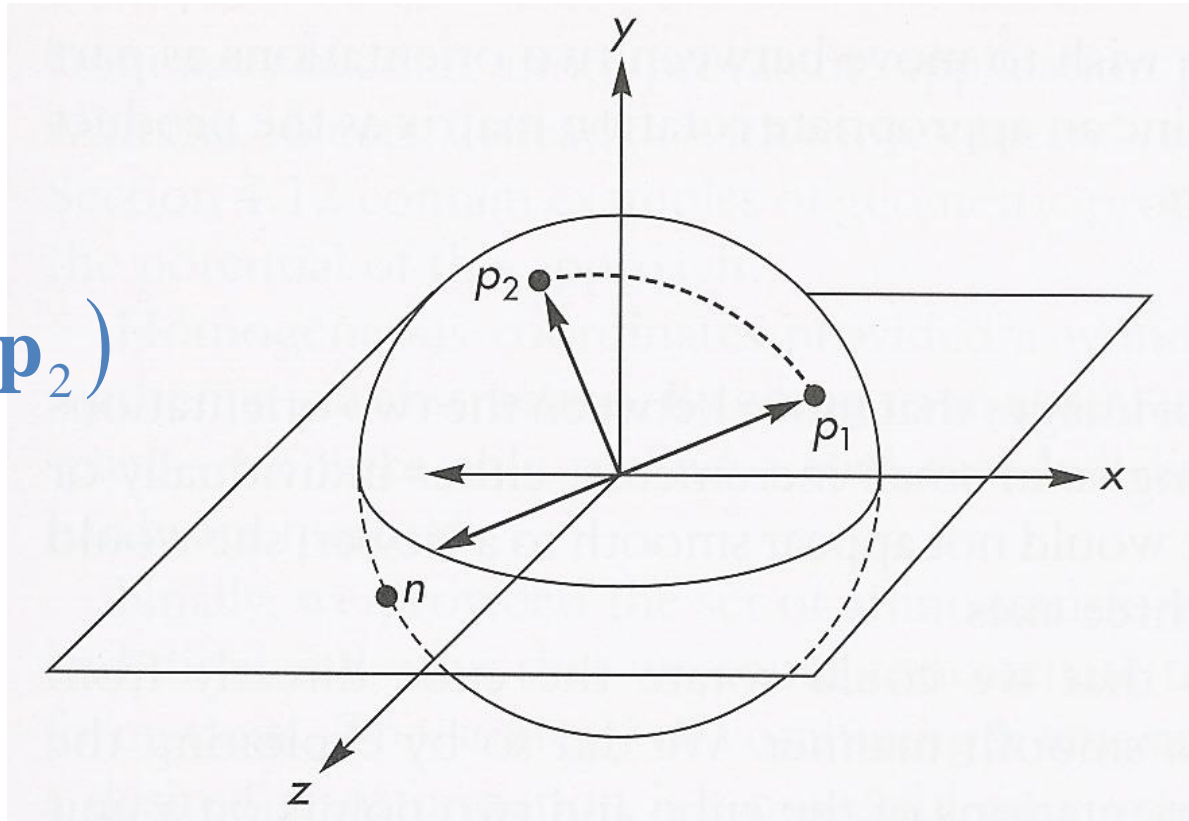
$$\mathbf{n} = \mathbf{p}_1 \times \mathbf{p}_2$$

- Rotation angle

$$\theta = \cos^{-1}(\mathbf{p}_1 \cdot \mathbf{p}_2)$$



Quaternions



Rotations with Quaternions (1)

- Rotation about an arbitrary axis

- Setting up a unit quaternion (**u**: unit vector)

$$s = \cos \frac{\theta}{2}, \quad \mathbf{v} = \mathbf{u} \sin \frac{\theta}{2} = (a, b, c)$$

- Representing any point position **P** in quaternion notation (**p** = (x, y, z))

$$\mathbf{P} = (0, \mathbf{p})$$

- Carrying out with the quaternion operation ($q^{-1} = (s, -\mathbf{v})$)

$$\mathbf{P}' = q\mathbf{P}q^{-1}$$

- Producing the new quaternion

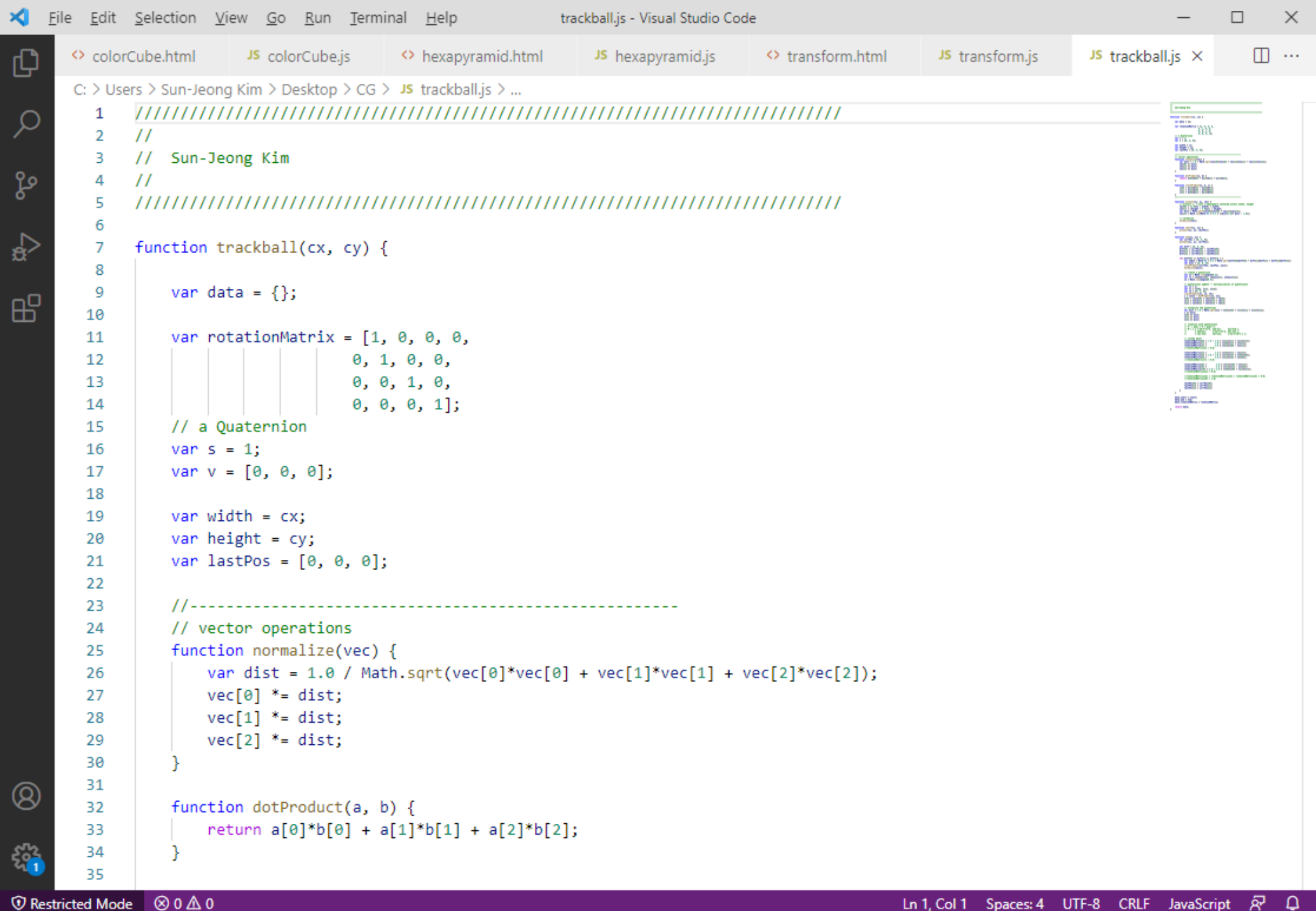
$$\mathbf{P}' = (0, \mathbf{p}')$$

$$\mathbf{p}' = s^2\mathbf{p} + \mathbf{v}(\mathbf{p} \cdot \mathbf{v}) + 2s(\mathbf{v} \times \mathbf{p}) + \mathbf{v} \times (\mathbf{v} \times \mathbf{p})$$

Rotations with Quaternions (2)

- Obtaining the rotation matrix by quaternion multiplication

$$\begin{aligned}\mathbf{M}_R(\theta) &= \begin{bmatrix} 1-2b^2-2c^2 & 2ab-2sc & 2ac+2sb \\ 2ab+2sc & 1-2a^2-2c^2 & 2bc-2sa \\ 2ac-2sb & 2bc+2sa & 1-2a^2-2b^2 \end{bmatrix} \\ &= \mathbf{R}_x(-\theta_x)\mathbf{R}_y(-\theta_y)\mathbf{R}_z(\theta)\mathbf{R}_y(\theta_y)\mathbf{R}_x(\theta_x)\end{aligned}$$



File Edit Selection View Go Run Terminal Help

trackball.js - Visual Studio Code

colorCube.html JS colorCube.js hexapyramid.html JS hexapyramid.js transform.html JS transform.js JS trackball.js X

C: > Users > Sun-Jeong Kim > Desktop > CG > JS trackball.js > ...

```
36 function crossProduct(a, b, c) {
37     c[0] = a[1]*b[2] - a[2]*b[1];
38     c[1] = a[2]*b[0] - a[0]*b[2];
39     c[2] = a[0]*b[1] - a[1]*b[0];
40 }
41 //-----
42
43 function project(xi, yi, vec) {
44     // project x, y onto a hemisphere centered within width, height
45     vec[0] = (2.0*xi - width) / width;
46     vec[1] = (height - 2.0*yi) / height;
47     var dist = Math.sqrt(vec[0]*vec[0] + vec[1]*vec[1]);
48     vec[2] = Math.cos(Math.PI * 0.5 * ((dist<1.0)? dist : 1.0));
49
50     // normalize
51     normalize(vec);
52 }
53
54 function start(xi, yi) {
55     project(xi, yi, lastPos);
56 }
57
58 function end(xi, yi) {
59     var currPos = [0, 0, 0];
60     project(xi, yi, currPos);
61
62     var diff = [0, 0, 0];
63     diff[0] = currPos[0] - lastPos[0];
64     diff[1] = currPos[1] - lastPos[1];
65     diff[2] = currPos[2] - lastPos[2];
66
67     if( diff[0] || diff[1] || diff[2] ) {
68         var angle = Math.PI * 0.5 * Math.sqrt(diff[0]*diff[0] + diff[1]*diff[1] + diff[2]*diff[2]);
69         var axis = [0, 0, 0];
70         crossProduct(currPos, lastPos, axis);
```

Ln 1, Col 1 Spaces: 4 UTF-8 CRLF JavaScript

<> colorCube.html

JS colorCube.js

<> hexapyramid.html

JS hexapyramid.js

<> transform.html

JS transform.js

JS trackball.js ×

□ ...

C: > Users > Sun-Jeong Kim > Desktop > CG > JS trackball.js > ...

```
67     if( diff[0] || diff[1] || diff[2] ) {
68         var angle = Math.PI * 0.5 * Math.sqrt(diff[0]*diff[0] + diff[1]*diff[1] + diff[2]*diff[2]);
69         var axis = [0, 0, 0];
70         crossProduct(currPos, lastPos, axis);
71         normalize(axis);
72
73         // create a quaternion
74         var s2 = Math.sin(angle*0.5);
75         var v2 = [s2*axis[0], s2*axis[1], s2*axis[2]];
76         s2 = Math.cos(angle*0.5);
77
78         // quaternions update -- multiplication of quaternions
79         var s1 = s;
80         var v1 = [v[0], v[1], v[2]];
81         var v3 = [0, 0, 0];
82         crossProduct(v1, v2, v3);
83         s = s1*s2 - dotProduct(v1, v2);
84         v[0] = s1*v2[0] + s2*v1[0] + v3[0];
85         v[1] = s1*v2[1] + s2*v1[1] + v3[1];
86         v[2] = s1*v2[2] + s2*v1[2] + v3[2];
87
88         // normalize the quaternion
89         var dist = 1.0 / Math.sqrt(s*s + v[0]*v[0] + v[1]*v[1] + v[2]*v[2]);
90         s *= dist;
91         v[0] *= dist;
92         v[1] *= dist;
93         v[2] *= dist;
94
95         // rotation with quaternions
96         // P' = quat * P * quat^-1
97         // M = { { 1-2b^2-2c^2, 2ab-2sc,      2ac+2sb },
98               { 2ab+2sc,      1-2a^2-2c^2, 2bc-2sa },
99               { 2ac-2sb,      2bc+2sa,      1-2a^2-2b^2 } };
100
101         // column major
```



File Edit Selection View Go Run Terminal Help

trackball.js - Visual Studio Code

colorCube.html JS colorCube.js hexapyramid.html JS hexapyramid.js transform.html JS transform.js JS trackball.js

C: > Users > Sun-Jeong Kim > Desktop > CG > JS trackball.js > ...

```
100
101     // column major
102     rotationMatrix[0] = 1.0 - 2.0 * (v[1]*v[1] + v[2]*v[2]);
103     rotationMatrix[1] =      2.0 * (v[0]*v[1] + s*v[2]);
104     rotationMatrix[2] =      2.0 * (v[2]*v[0] - s*v[1]);
105     //rotationMatrix[3] = 0.0;
106
107     rotationMatrix[4] =      2.0 * (v[0]*v[1] - s*v[2]);
108     rotationMatrix[5] = 1.0 - 2.0 * (v[2]*v[2] + v[0]*v[0]);
109     rotationMatrix[6] =      2.0 * (v[1]*v[2] + s*v[0]);
110     //rotationMatrix[7] = 0.0;
111
112     rotationMatrix[8] =      2.0 * (v[2]*v[0] + s*v[1]);
113     rotationMatrix[9] =      2.0 * (v[1]*v[2] - s*v[0]);
114     rotationMatrix[10] = 1.0 - 2.0 * (v[0]*v[0] + v[1]*v[1]);
115     //rotationMatrix[11] = 0.0;
116
117     //rotationMatrix[12] = rotationMatrix[13] = rotationMatrix[14] = 0.0;
118     //rotationMatrix[15] = 1.0;
119
120     lastPos[0] = currPos[0];
121     lastPos[1] = currPos[1];
122     lastPos[2] = currPos[2];
123 }
124 }
125
126 data.start = start;
127 data.end = end;
128 data.rotationMatrix = rotationMatrix;
129
130 return data;
131 }
132
```

```
100
101
102
103
104
105
106
107
108
109
110
111
112
113
114
115
116
117
118
119
120
121
122
123
124
125
126
127
128
129
130
131
132
```

Ln 1, Col 1 Spaces: 4 UTF-8 CRLF JavaScript

Restricted Mode 0 0

C: > Users > Sun-Jeong Kim > Desktop > CG > Week06 > <> transform.html > html > head > script#vertex-shader

```

1  <!DOCTYPE html>
2  <html>
3    <head>
4      <title>학번 이름 - Transformations</title>
5      <script id="vertex-shader" type="x-shader/x-vertex">
6        attribute vec4 vPosition;
7        attribute vec4 vColor;
8        uniform vec3 theta;
9        uniform vec3 displ;
10       uniform mat4 trMatrix;
11       varying vec4 fColor;
12
13       void main() {
14         // Compute the sines and cosines of theta for each of
15         // the three axes in one computation
16         vec3 angles = radians(theta);
17         vec3 c = cos(angles);
18         vec3 s = sin(angles);
19
20         // Remember: these matrices are column-major
21         mat4 rx = mat4( 1.0, 0.0, 0.0, 0.0,
22                        0.0, c.x, s.x, 0.0,
23                        0.0, -s.x, c.x, 0.0,
24                        0.0, 0.0, 0.0, 1.0 );
25
26         mat4 ry = mat4( c.y, 0.0, -s.y, 0.0,
27                        0.0, 1.0, 0.0, 0.0,
28                        s.y, 0.0, c.y, 0.0,
29                        0.0, 0.0, 0.0, 1.0 );
30
31         mat4 rz = mat4( c.z, s.z, 0.0, 0.0,
32                        -s.z, c.z, 0.0, 0.0,
33                        0.0, 0.0, 1.0, 0.0,
34                        0.0, 0.0, 0.0, 1.0 );
35

```



File Edit Selection View Go Run Terminal Helptransform.html - Visual Studio Code

colorCube.htmlJS colorCube.js<> hexapyramid.htmlJS hexapyramid.js<> transform.html XJS transform.jsJS trackball.js

C: > Users > Sun-Jeong Kim > Desktop > CG > Week06 > <> transform.html > html > head > script

```
31 mat4 rz = mat4( c.z, s.z, 0.0, 0.0,
32             -s.z, c.z, 0.0, 0.0,
33             0.0, 0.0, 1.0, 0.0,
34             0.0, 0.0, 0.0, 1.0 );
35
36 mat4 sc = mat4( 0.5, 0.0, 0.0, 0.0,
37             0.0, 0.5, 0.0, 0.0,
38             0.0, 0.0, 0.5, 0.0,
39             0.0, 0.0, 0.0, 1.0 );
40
41 mat4 tr = mat4( 1.0, 0.0, 0.0, 0.0,
42             0.0, 1.0, 0.0, 0.0,
43             0.0, 0.0, 1.0, 0.0,
44             displ.x, displ.y, displ.z, 1.0 );
45
46 gl_Position = tr * rz * ry * rx * sc * trMatrix * vPosition;
47 fColor = vColor;
48 }
49 </script>
50
51 <script id="fragment-shader" type="x-shader/x-fragment">
52 precision mediump float;
53 varying vec4 fColor;
54
55 void main() {
56     gl_FragColor = fColor;
57 }
58 </script>
59
60 <script type="text/javascript" src="../Common/webgl-utils.js"></script>
61 <script type="text/javascript" src="../Common/initShaders.js"></script>
62 <script type="text/javascript" src="../Common/MV.js"></script>
63 <script type="text/javascript" src="../trackball.js"></script>
64 <script type="text/javascript" src="transform.js"></script>
65 </head>
```

58

Restricted Mode 0 0 0 Ln 63, Col 62 Spaces: 4 UTF-8 CRLF HTML

transform.js - Visual Studio Code

File Edit Selection View Go Run Terminal Help

colorCube.html JS colorCube.js hexapyramid.html JS hexapyramid.js transform.html JS transform.js X JS trackball.js

C: > Users > Sun-Jeong Kim > Desktop > CG > Week06 > JS transform.js > ...

```
1  var gl;
2  var points = [];
3  var colors = [];
4
5  var axis = 0;
6  var theta = [0, 0, 0];
7  var thetaloc;
8  var displ = [0, 0, 0];
9  var displloc;
10 var trMatrixLoc;
11
12 var rotation = false;
13
14 window.onload = function init()
15 {
16     var canvas = document.getElementById("gl-canvas");
17
18     gl = WebGLUtils.setupWebGL(canvas);
19     if( !gl ) {
20         alert("WebGL isn't available!");
21     }
22
23     generateColorCube();
24     generateHexaPyramid();
25
26     // virtual trackball
27     var trball = trackball(canvas.width, canvas.height);
28     var bMouseDown = false;
29
30     canvas.addEventListener("mousedown", function(event) {
31         trball.start(event.clientX, event.clientY);
32
33         bMouseDown = true;
34     });
35     canvas.addEventListener("mouseup", function(event) {
```

Ln 10, Col 17 Spaces: 4 UTF-8 CRLF JavaScript

Restricted Mode 0 0 0

File Edit Selection View Go Run Terminal Help

transform.js - Visual Studio Code

colorCube.html JS colorCube.js hexapyramid.html JS hexapyramid.js transform.html JS transform.js X JS trackball.js

C: > Users > Sun-Jeong Kim > Desktop > CG > Week06 > JS transform.js > init > canvas.addEventListener("mousemove") callback

```
25
26 // virtual trackball
27 var trball = trackball(canvas.width, canvas.height);
28 var bMouseDown = false;
29
30 canvas.addEventListener("mousedown", function(event) {
31     trball.start(event.clientX, event.clientY);
32
33     bMouseDown = true;
34 });
35 canvas.addEventListener("mouseup", function(event) {
36     bMouseDown = false;
37 });
38 canvas.addEventListener("mousemove", function(event) {
39     if( bMouseDown ) {
40         trball.end(event.clientX, event.clientY);
41
42         gl.uniformMatrix4fv(trMatrixLoc, false, trball.rotationMatrix);
43
44     }
45 });
46
47 // Configure WebGL
48 gl.viewport(0, 0, canvas.width, canvas.height);
49 gl.clearColor(0.9, 0.9, 0.9, 1.0);
50
51 // Enable hidden-surface removal
52 gl.enable(gl.DEPTH_TEST);
53
54 // Load shaders and initialize attribute buffers
55 var program = initShaders(gl, "vertex-shader", "fragment-shader");
56 gl.useProgram(program);
57
58 // Load the data into the GPU
59 var bufferId = gl.createBuffer();
60 gl.bindBuffer(gl.ARRAY_BUFFER, bufferId);
```

60

Restricted Mode 0 0 0 Ln 42, Col 76 Spaces: 4 UTF-8 CRLF JavaScript

FileEditSelectionViewGoRunTerminalHelptransform.js - Visual Studio Code

colorCube.htmlJS colorCube.js<> hexapyramid.htmlJS hexapyramid.js<> transform.htmlJS transform.js XJS trackball.js

C: > Users > Sun-Jeong Kim > Desktop > CG > Week06 > JS transform.js > init

767778798081828384858687888990919293949596979899100101102103104105106107108109110

```
thetaLoc = gl.getUniformLocation(program, "theta");
//gl.uniform3fv(thetaLoc, theta);
displLoc = gl.getUniformLocation(program, "displ");
//gl.uniform3fv(displLoc, displ);
trMatrixLoc = gl.getUniformLocation(program, "trMatrix");
gl.uniformMatrix4fv(trMatrixLoc, false, trball.rotationMatrix);

// Event listeners for buttons
document.getElementById("xButton").onclick = function () {
    axis = 0;
};
document.getElementById("yButton").onclick = function () {
    axis = 1;
};
document.getElementById("zButton").onclick = function () {
    axis = 2;
};
document.getElementById("toggleButton").onclick = function () {
    rotation = !rotation;
};

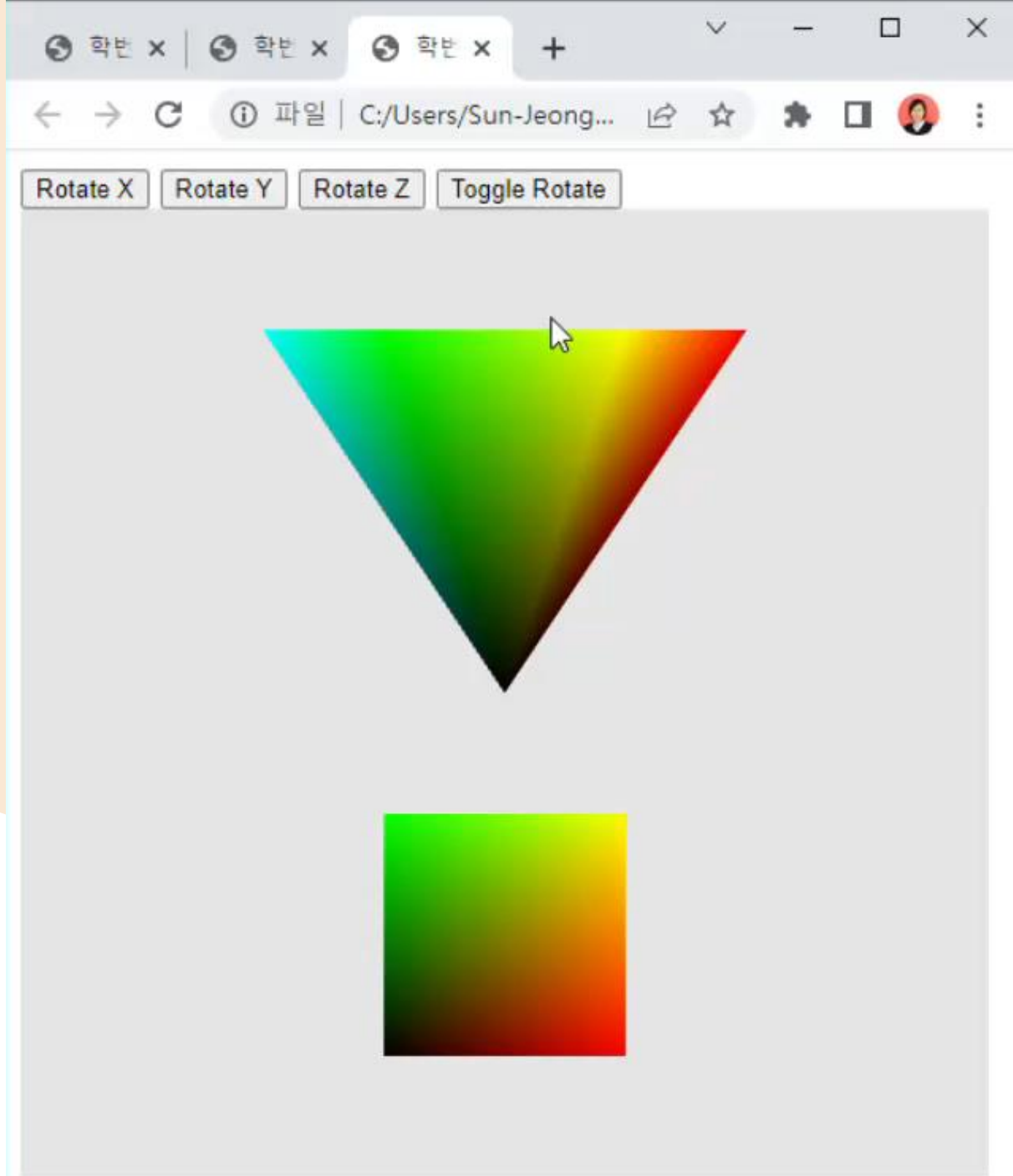
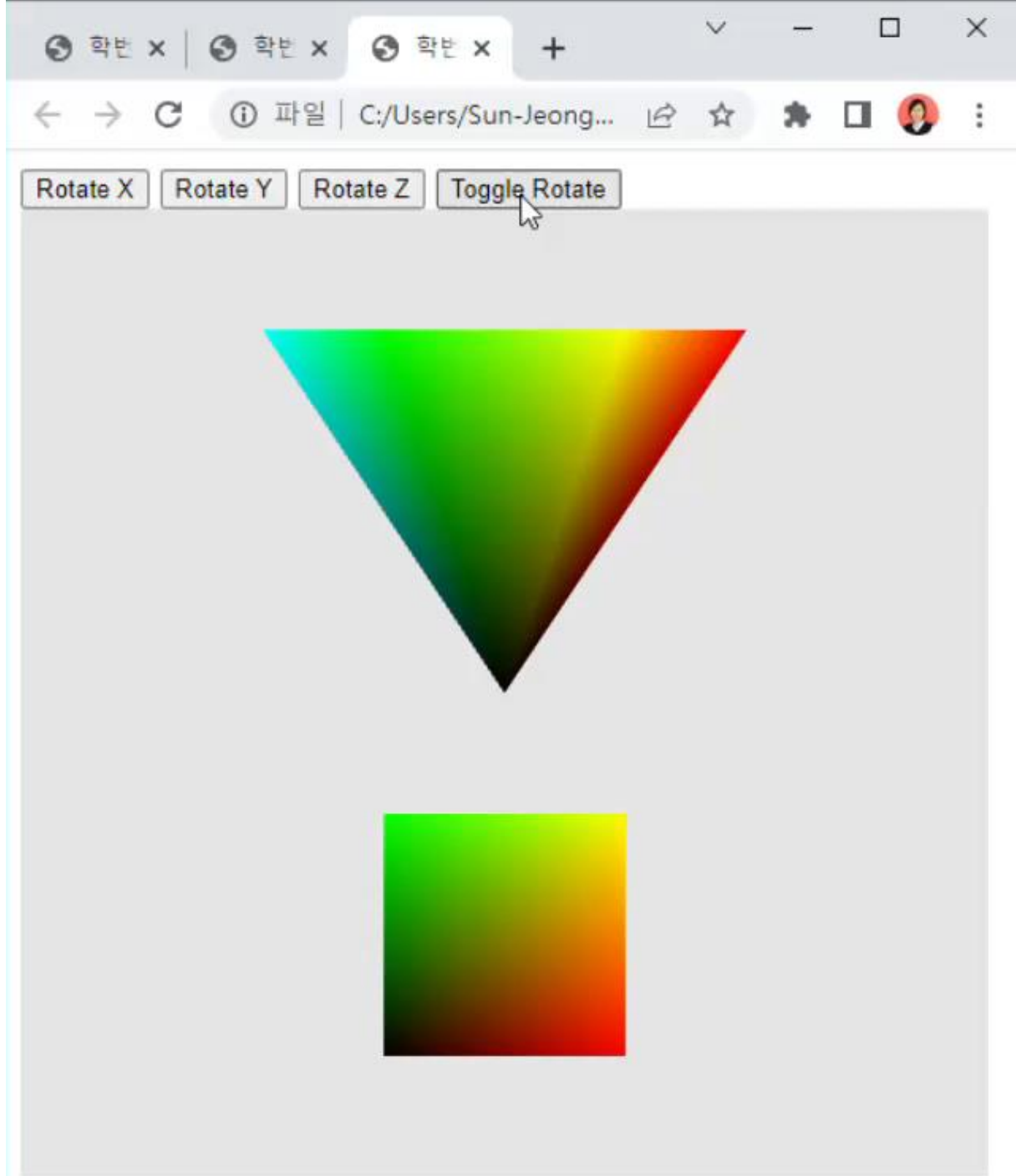
render();
};

function render() {
    gl.clear(gl.COLOR_BUFFER_BIT | gl.DEPTH_BUFFER_BIT);

    if( rotation ) {
        theta[axis] += 2.0;
    }
    gl.uniform3fv(thetaLoc, theta);

    // Draw a color cube (12 triangles * 3 = 36 vertices)
    displ[1] = -0.5;
```

1111121314151617181920212223242526272829303132333435363738394041424344454647484950515253545556575859606162636465666768697071727374757677787980818283848586878889909192939495969798991001011021031041051061071081091101111121131141151161171181191201211221231241251261271281291301311321331341351361371381391401411421431441451461471481491501511521531541551561571581591601611621631641651661671681691701711721731741751761771781791801811821831841851861871881891901911921931941951961971981992002012022032042052062072082092102112122132142152162172182192202212222232242252262272282292302312322332342352362372382392402412422432442452462472482492502512522532542552562572582592602612622632642652662672682692702712722732742752762772782792802812822832842852862872882892902912922932942952962972982993003013023033043053063073083093103113123133143153163173183193203213223233243253263273283293303313323333343353363373383393403413423433443453463473483493503513523533543553563573583593603613623633643653663673683693703713723733743753763773783793803813823833843853863873883893903913923933943953963973983994004014024034044054064074084094104114124134144154164174184194204214224234244254264274284294304314324334344354364374384394404414424434444454464474484494504514524534544554564574584594604614624634644654664674684694704714724734744754764774784794804814824834844854864874884894904914924934944954964974984995005015025035045055065075085095105115125135145155165175185195205215225235245255265275285295305315325335345355365375385395405415425435445455465475485495505515525535545555565575585595605615625635645655665675685695705715725735745755765775785795805815825835845855865875885895905915925935945955965975985996006016026036046056066076086096106116126136146156166176186196206216226236246256266276286296306316326336346356366376386396406416426436446456466476486496506516526536546556566576586596606616626636646656666676686696706716726736746756766776786796806816826836846856866876886896906916926936946956966976986997007017027037047057067077087097107117127137147157167177187197207217227237247257267277287297307317327337347357367377387397407417427437447457467477487497507517527537547557567577587597607617627637647657667677687697707717727737747757767777787797807817827837847857867877887897907917927937947957967977987998008018028038048058068078088098108118128138148158168178188198208218228238248258268278288298308318328338348358368378388398408418428438448458468478488498508518528538548558568578588598608618628638648658668678688698708718728738748758768778788798808818828838848858868878888898908918928938948958968978988999009019029039049059069079089099109119129139149159169179189199209219229239249259269279289299309319329339349359369379389399409419429439449459469479489499509519529539549559569579589599609619629639649659669679689699709719729739749759769779789799809819829839849859869879889899909919929939949959969979989991000100110021003100410051006100710081009101010111012101310141015101610171018101910201021102210231024102510261027102810291030103110321033103410351036103710381039104010411042104310441045104610471048104910501051105210531054105510561057105810591060106110621063106410651066106710681069107010711072107310741075107610771078107910801081108210831084108510861087108810891090109110921093109410951096109710981099110011011102110311041105110611071108110911101111111211131114111511161117111811191120112111221123112411251126112711281129113011311132113311341135113611371138113911401141114211431144114511461147114811491150115111521153115411551156115711581159116011611162116311641165116611671168116911701171117211731174117511761177117811791180118111821183118411851186118711881189119011911192119311941195119611971198119912001201120212031204120512061207120812091210121112121213121412151216121712181219122012211222122312241225122612271228122912301231123212331234123512361237123812391240124112421243124412451246124712481249125012511252125312541255125612571258125912601261126212631264126512661267126812691270127112721273127412751276127712781279128012811282128312841285128612871288128912901291129212931294129512961297129812991300130113021303130413051306130713081309131013111312131313141315131613171318131913201321132213231324132513261327132813291330133113321333133413351336133713381339134013411342134313441345134613471348134913501351135213531354135513561357135813591360136113621363136413651366136713681369137013711372137313741375137613771378137913801381138213831384138513861387138813891390139113921393139413951396139713981399140014011402140314041405140614071408140914101411141214131414141514161417141814191420142114221423142414251426142714281429143014311432143314341435143614371438143914401441144214431444144514461447144814491450145114521453145414551456145714581459146014611462146314641465146614671468146914701471147214731474147514761477147814791480148114821483148414851486148714881489149014911492149314941495149614971498149915001501150215031504150515061507150815091510151115121513151415151516151715181519152015211522152315241525152615271528152915301531153215331534153515361537153815391540154115421543154415451546154715481549155015511552155315541555155615571558155915601561156215631564156515661567156815691570157115721573157415751576157715781579158015811582158315841585158615871588158915901591159215931594159515961597159815991600160116021603160416051606160716081609161016111612161316141615161616171618161916201621162216231624162516261627162816291630163116321633163416351636163716381639164016411642164316441645164616471648164916501651165216531654165516561657165816591660166116621663166416651666166716681669167016711672167316741675167616771678167916801681168216831684168516861687168816891690169116921693169416951696169716981699170017011702170317041705170617071708170917101711171217131714171517161717171817191720172117221723172417251726172717281729173017311732173317341735173617371738173917401741174217431744174517461747174817491750175117521753175417551756175717581759176017611762176317641765176617671768176917701771177217731774177517761777177817791780178117821783178417851786178717881789179017911792179317941795179617971798179918001801180218031804180518061807180818091810181118121813181418151816181718181819182018211822182318241825182618271828182918301831183218331834183518361837183818391840184118421843184418451846184718481849185018511852185318541855185618571858185918601861186218631864186518661867186818691870187118721873187418751876187718781879188018811882188318841885188618871888188918901891189218931894189518961897189818991900190119021903190419051906190719081909191019111912191319141915191619171918191919201921192219231924192519261927192819291930193119321933193419351936193719381939194019411942194319441945194619471948194919501951195219531954195519561957195819591960196119621963196419651966196719681969197019711972197319741975197619771978197919801981198219831984198519861987198819891990199119921993199419951996199719981999200020012002200320042005200620072008200920102011201220132014201520162017201820192020202120222023202420252026202720282029203020312032203320342035203620372038203920402041204220432044204520462047204820492050205120522053205420552056205720582059206020612062206320642065206620672068206920702071207220732074207520762077207820792080208120822083208420852086208720882089209020912092209320942095209620972098209921002101210221032104210521062107210821092110211121122113211421152116211721182119212021212122212321242125212621272128212921302131213221332134213521362137213821392140214121422143214421452146214721482149215021512152215321542155215621572158215921602161216221632164216521662167216821692170217121722173217421752176217721782179218021812182218321842185218621872188218921902191219221932194219521962197219821992200220122022203220422052206220722082209221022112212221322142215221622172218221922202221222222232224222522262227222822292230223122322233223422352236223722382239224022412242224322442245224622472248224922502251225222532254225522562257225822592260226122622263226422652266226722682269227022712272227322742275227622772278227922802281228222832284228522862287228822892290229122922293229422952296229722982299230023012302230323042305230623072308230923102311231223132314231523162317231823192320232123222323232423252326232723282329233023312332233323342335233623372338233923402341234223432344234523462347234823492350235123522353235423552356235723582359236023612362236323642365236623672368236923702371237223732374237523762377237823792380238123822383238423852386238723882389239023912392239323942395239623972398239924002401240224032404240524062407240824092410241124122413241424152416241724182419242024212422242324242425242624272428242924302431243224332434243524362437243824392440244124422443244424452446244724482449245024512452245324542455245624572458245924602461246224632464246524662467246824692470247124722473247424752476247724782479248024812482248324842485248624872488248924902491249224932494249524962497249824992500250125022503250425052506250725082509251025112512251325142515251625172518251925202521252225232524252525262527252825292530253125322533253425352536253725382539254025412542254325442545254625472548254925502551255225532554255525562557255825592560256125622563256425652566256725682569257025712572257325742575257625772578257925802581258225832584258525862587258825892590259125922593259425952596259725982599260026012602260326042605260626072608260926102611261226132614261526162617261826192620262126222623262426252626262726282629263026312632263326342635263626372638263926402641264226432644264526462647264826492650265126522653265426552656265726582659266026612662266326642665266626672668266926702671267226732674267526762677267826792680268126822683268426852686268726882689269026912692269326942695269626972698269927002701270227032704270527062707270827092710271127122713271427152716271727182719272027212722272327242725272627272728272927302731273227332734273527362737273827392740274127422743274427452746274727482749275027512752275327542755275627572758275927602761276227632764276527662767276827692770277127722773277427752776277727782779278027812782278327842785278627872788278927902791279227932794279527962797279827992800280128022803280428052806280728082809281028112812281328142815281628172818281928202821282228232824282528262827282828292830283128322833283428352836283728382839284028412842284328442845284628472848284928502851285228532854285528562857285828592860286128622863286428652866286728682869287028712872287328742875287628772878287928802881288228832884288528862887288828892890289128922893289428952896289728982899290029012902290329042905290629072908290929102911291229132914291529162917291829192920292129222923292429252926292729282929293029312932293329342935293629372938293929402941294229432944294529462947294829492950295129522953295429552956295729582959296029612962296329642965296629672968296929702971297229732974297529762977297829792980298129822983298429852986298729882989299029912992299329942995299629972998299930003001300230033004300530063007300830093010301130123013301430153016301730183019302030213022302330243025302630273028302930303031303230333034303530363037303830393040304130423043304430453046304730483049305030513052305330543055305630573058305930603061306230633064306530663067306830693070307130723073307430753076307730783079308030813082308330843085308630873088308930903091309230933094309530963097309830993100310131023103310431053106310731083109311031113112311331143115311631173118311931203121312231233124312531263127312831293130313131323133313431353136313731383139314031413142314331443145314631473148314931503151315231533154315531563157315831593160316131623163316431653166316731683169317031713172317331743175317631773178317931803181318231833184318531863187318831893190319131923193319431953196319731983199320032013202320332043205320632073208320932103211321232133214321532163217321832193220322132223223322343225322632273228322932303231323232333234323532363237323832393240324132423243324432453246324732483249325032513252




File Edit Selection View Go Run Terminal Help

transform.html - Visual Studio Code

colorCube.html JS colorCube.js hexapyramid.html JS hexapyramid.js transform.html X JS transform.js JS trackball.js

C: > Users > Sun-Jeong Kim > Desktop > CG > Week06 > <> transform.html > html > head > script#vertex-shader

```
31 mat4 rz = mat4( c.z, s.z, 0.0, 0.0,
32               -s.z, c.z, 0.0, 0.0,
33               0.0, 0.0, 1.0, 0.0,
34               0.0, 0.0, 0.0, 1.0 );
35
36 mat4 sc = mat4( 0.5, 0.0, 0.0, 0.0,
37               0.0, 0.5, 0.0, 0.0,
38               0.0, 0.0, 0.5, 0.0,
39               0.0, 0.0, 0.0, 1.0 );
40
41 mat4 tr = mat4( 1.0, 0.0, 0.0, 0.0,
42               0.0, 1.0, 0.0, 0.0,
43               0.0, 0.0, 1.0, 0.0,
44               displ.x, displ.y, displ.z, 1.0 );
45
46 gl_Position = trMatrix * tr * rz * ry * rx * sc * vPosition;
47 fColor = vColor;
48
49 </script>
50
51 <script id="fragment-shader" type="x-shader/x-fragment">
52 precision mediump float;
53 varying vec4 fColor;
54
55 void main() {
56     gl_FragColor = fColor;
57 }
58 </script>
59
60 <script type="text/javascript" src="../Common/webgl-utils.js"></script>
61 <script type="text/javascript" src="../Common/initShaders.js"></script>
62 <script type="text/javascript" src="../Common/MV.js"></script>
63 <script type="text/javascript" src="../trackball.js"></script>
64 <script type="text/javascript" src="transform.js"></script>
65 </head>
```



Ln 46, Col 35 Spaces: 4 UTF-8 CRLF HTML

연습 문제 (6)

- 4개 오브젝트를 그리시오.
 - 상하 오브젝트들은 서로 반대 방향으로 회전
 - 좌우 오브젝트들도 서로 반대 방향으로 회전

