CSC 240 Computer Graphics
Day 14: Animation & 3D Transformations

Nick Howe
Smith College
Q. Can we go over the differences between each "render" in the code?

A. Let’s compare them side by side.

```javascript
// Render the scene.
function render() {
    renderer.render(scene, camera);
}

// Render the scene.
function render() {
    cube.position.z -= 0.1;
    renderer.render(scene, camera);
    requestAnimationFrame(render);
}

// Render the scene.
function render(now) {
    var elapsed = now - then;
    console.log(elapsed);
    then = now;
    cube.position.z -= elapsed / 1000;
    renderer.render(scene, camera);
    requestAnimationFrame(render);
}
Your Questions

Q. If we are using elapsed/some value to ensure smoother animation, does this mean that the animation might be noticeably slower if our machine is rendering more slowly?

A. If the machine is rendering slowly, the motion will be jerky.

Q. Why does renderer.render take scene and camera as parameters? Will this always be the same?

A. This will always be the same in our work. In theory, you might have more than one camera for taking different shots of a scene. You might even have different scenes prepared, and want to switch between them. We’ll typically stick to one scene, one camera.
Q. What does it mean that the 'look at' method uses prop 'up' to get the twist around the line of sight? I'm having trouble visualizing this.

A. lookAt makes the camera point at the target. But you can roll the camera around this axis, so to fix the orientation we have to say which way is “up”.

Q. What does applyMatrix() do? What do you mean when you say it records the modeling transform of an object?

A. applyMatrix4() is the 3D equivalent of setTransform(): it replaces the old transformation with a completely new one.

Q. What is Q? Can you please go over these questions and answers slowly?

A. Certainly! (Q is just a hypothetical variable name.)
Questions

1. What number would be on top after each rotation, always starting from the position shown?
   a. rotation.x = Math.PI/2; 3
   b. rotation.y = Math.PI/2; 6
   c. rotation.z = Math.PI/2; 5

2. What transformations are performed by the matrices below?
   a. \[
   \begin{bmatrix}
   1 & 0 & 0 & 3 \\
   0 & 1 & 0 & 2 \\
   0 & 0 & 1 & 1 \\
   0 & 0 & 0 & 1
   \end{bmatrix}
   \]
   b. \[
   \begin{bmatrix}
   0 & 0 & -1 & 0 \\
   0 & 1 & 0 & 0 \\
   1 & 0 & 0 & 0 \\
   0 & 0 & 0 & 1
   \end{bmatrix}
   \]
   c. \[
   \begin{bmatrix}
   1 & 0 & 0 & 5 \\
   0 & 5 & 0 & 0 \\
   0 & 0 & 1 & 0 \\
   0 & 0 & 0 & 1
   \end{bmatrix}
   \]
   d. \[
   \begin{bmatrix}
   1 & 0 & 0 & 0 \\
   0 & .71 & -0.71 & 0 \\
   0 & .71 & .71 & 0 \\
   0 & 0 & 0 & 1
   \end{bmatrix}
   \]

3. What WebGL code would generate such a transformation for object Q?

   Rotation around Y
   Q.rotation.y = Math.PI/2;

   Translation
   Q.position.set(3,2,1);

   Rotation around X
   Q.rotation.x = Math.PI/4;

   Scale & Translation
   Q.scale.y = 5; Q.position.x = 5;
Your Questions