

Reference > rotateY()

rotateY()

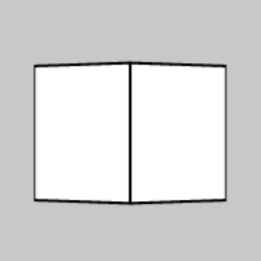
Rotates the coordinate system about the y-axis in WebGL mode.

The parameter, `angle`, is the amount to rotate. For example, calling `rotateY(1)` rotates the coordinate system about the y-axis by 1 radian. `rotateY()` interprets angle values using the current `angleMode()`.

By default, transformations accumulate. For example, calling `rotateY(1)` twice has the same effect as calling `rotateY(2)` once. The `push()` and `pop()` functions can be used to isolate transformations within distinct drawing groups.

Note: Transformations are reset at the beginning of the draw loop. Calling `rotateY(1)` inside the `draw()` function won't cause shapes to spin.

Examples



```
// Click and drag the mouse to view the scene from different angles.

function setup() {
  createCanvas(100, 100, WEBGL);

  describe('A white cube on a gray background.');
```

```
function draw() {
  background(200);

  // Enable orbiting with the mouse.
  orbitControl();

  // Rotate the coordinate system 1/8 turn.
  rotateY(QUARTER_PI);

  // Draw a box.
  box();
}
```



```
// Click and drag the mouse to view the scene from different angles.

function setup() {
  createCanvas(100, 100, WEBGL);

  describe('A white cube on a gray background.');
```

```
function draw() {
  background(200);

  // Enable orbiting with the mouse.
  orbitControl();

  // Rotate the coordinate system 1/16 turn.
  rotateY(QUARTER_PI / 2);

  // Rotate the coordinate system 1/16 turn.
  rotateY(QUARTER_PI / 2);

  // Draw a box.
  box();
}
```



```
// Click and drag the mouse to view the scene from different angles.

function setup() {
  createCanvas(100, 100, WEBGL);

  // Use degrees.
  angleMode(DEGREES);

  describe('A white cube on a gray background.');
```

```
function draw() {
  background(200);

  // Enable orbiting with the mouse.
  orbitControl();

  // Rotate the coordinate system 1/8 turn.
  rotateY(45);

  // Draw a box.
  box();
}
```



```
// Click and drag the mouse to view the scene from different angles.

function setup() {
  createCanvas(100, 100, WEBGL);

  describe('A white cube rotates slowly against a gray background.');
```

```
function draw() {
  background(200);

  // Enable orbiting with the mouse.
  orbitControl();

  // Rotate the coordinate system a little more each frame.
  let angle = frameCount * 0.01;
  rotateY(angle);

  // Draw a box.
  box();
}
```

Syntax

```
rotateY(angle)
```

Parameters

`angle` Number: angle of rotation in the current `angleMode()`.

This page is generated from the comments in `src/core/transform.js`. Please feel free to edit it and submit a pull request!

Related References

applyMatrix Applies a transformation matrix to the coordinate system.	resetMatrix Clears all transformations applied to the coordinate system.	rotate Rotates the coordinate system.	rotateX Rotates the coordinate system about the x-axis in WebGL mode.
---	--	---	---