normal()

Sets the normal vector for vertices in a custom 3D shape.

3D shapes created with beginShape() and endShape() are made by connecting sets of points called vertices. Each vertex added with vertex() has a normal vector that points away from it. The normal vector controls how light reflects off the shape.

`normal()` can be called two ways with different parameters to define the normal vector's components.

The first way to call `normal()` has three parameters, `x`, `y`, and `z`. If `Number`s are passed, as in `normal(1, 2, 3)`, they set the x-, y-, and z-components of the normal vector.

The second way to call `normal()` has one parameter, `vector`. If a p5.Vector object is passed, as in `normal(myVector)`, its components will be used to set the normal vector.

`normal()` changes the normal vector of vertices added to a custom shape with vertex(). `normal()` must be called between the beginShape() and endShape() functions, just like vertex(). The normal vector set by calling `normal()` will affect all following vertices until `normal()` is called again:

```
beginShape();

// Set the vertex normal.
normal(-0.4, -0.4, 0.8);

// Add a vertex.
vertex(-30, -30, 0);

// Set the vertex normal.
normal(0, 0, 1);

// Add vertices.
vertex(30, -30, 0);
vertex(30, 30, 0);

// Set the vertex normal.
normal(0.4, -0.4, 0.8);

// Add a vertex.
vertex(-30, 30, 0);

endShape();
```

Examples

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

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

  describe(
    'A colorful square on a black background. The square
changes color and rotates when the user drags the mouse. Parts
of its surface reflect light in different directions.'
  );
}

function draw() {
  background(0);

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

  // Style the shape.
  normalMaterial();
  noStroke();

  // Draw the shape.
  beginShape();
  vertex(-30, -30, 0);
  vertex(30, -30, 0);
  vertex(30, 30, 0);
  vertex(-30, 30, 0);
  endShape();
}
```

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

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

  describe(
    'A colorful square on a black background. The square
changes color and rotates when the user drags the mouse. Parts
of its surface reflect light in different directions.'
  );
}

function draw() {
  background(0);

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

  // Style the shape.
  normalMaterial();
  noStroke();

  // Draw the shape.
  // Use normal() to set vertex normals.
  beginShape();
  normal(-0.4, -0.4, 0.8);
  vertex(-30, -30, 0);

  normal(0, 0, 1);
```

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

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

  describe(
    'A colorful square on a black background. The square
changes color and rotates when the user drags the mouse. Parts
of its surface reflect light in different directions.'
  );
}

function draw() {
  background(0);

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

  // Style the shape.
  normalMaterial();
  noStroke();

  // Create p5.Vector objects.
  let n1 = createVector(-0.4, -0.4, 0.8);
  let n2 = createVector(0, 0, 1);
  let n3 = createVector(0.4, -0.4, 0.8);

  // Draw the shape.
```

Syntax

```
normal(vector)
```

```
normal(x, y, z)
```

Parameters

vector	p5.Vector: vertex normal as a p5.Vector object.
x	Number: x-component of the vertex normal.
y	Number: y-component of the vertex normal.
z	Number: z-component of the vertex normal.

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

Related References

beginContour
Begins creating a hole within a flat shape.

beginShape
Begins adding vertices to a custom shape.

bezierVertex
Adds a Bézier curve segment to a custom shape.

curveVertex
Adds a spline curve segment to a custom shape.