cone()

Draws a cone.

A cone is a 3D shape with triangular faces that connect a flat bottom to a single point. Cones with few faces look like pyramids. Cones with many faces have smooth surfaces.

The first parameter, `radius`, is optional. If a `Number` is passed, as in `cone(20)`, it sets the radius of the cone's base. By default, `radius` is 50.

The second parameter, `height`, is also optional. If a `Number` is passed, as in `cone(20, 30)`, it sets the cone's height. By default, `height` is set to the cone's `radius`.

The third parameter, `detailX`, is also optional. If a `Number` is passed, as in `cone(20, 30, 5)`, it sets the number of edges used to form the cone's base. Using more edges makes the base look more like a circle. By default, `detailX` is 24.

The fourth parameter, `detailY`, is also optional. If a `Number` is passed, as in `cone(20, 30, 5, 7)`, it sets the number of triangle subdivisions to use along the y-axis connecting the base to the tip. All 3D shapes are made by connecting triangles to form their surfaces. By default, `detailY` is 1.

The fifth parameter, `cap`, is also optional. If a `false` is passed, as in `cone(20, 30, 5, 7, false)` the cone's base won't be drawn. By default, `cap` is `true`.

Note: `cone()` can only be used in WebGL mode.

Examples

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

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

  describe('A white cone on a gray background.');
}

function draw() {
  background(200);

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

  // Draw the cone.
  cone();
}
```

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

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

  describe('A white cone on a gray background.');
}

function draw() {
  background(200);

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

  // Draw the cone.
  // Set its radius and height to 30.
  cone(30);
}
```

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

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

  describe('A white cone on a gray background.');
}

function draw() {
  background(200);

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

  // Draw the cone.
  // Set its radius to 30 and height to 50.
  cone(30, 50);
}
```

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

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

  describe('A white cone on a gray background.');
}

function draw() {
  background(200);

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

  // Draw the cone.
  // Set its radius to 30 and height to 50.
  // Set its detailX to 5.
  cone(30, 50, 5);
}
```

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

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

  describe('A white pyramid on a gray background.');
}

function draw() {
  background(200);

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

  // Draw the cone.
  // Set its radius to 30 and height to 50.
  // Set its detailX to 5.
  cone(30, 50, 5);
}
```

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

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

  describe('A white cone on a gray background.');
}

function draw() {
  background(200);

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

  // Draw the cone.
  // Set its radius to 30 and height to 50.
  // Set its detailX to 24 and detailY to 2.
  cone(30, 50, 24, 2);
}
```

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

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

  describe('A white cone on a gray background. Its base is
missing.');
}

function draw() {
  background(200);

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

  // Draw the cone.
  // Set its radius to 30 and height to 50.
  // Set its detailX to 24 and detailY to 1.
  // Don't draw its base.
  cone(30, 50, 24, 1, false);
}
```

Syntax

```
cone([radius], [height], [detailX], [detailY], [cap])
```

Parameters

radius	Number: radius of the cone's base. Defaults to 50.
height	Number: height of the cone. Defaults to the value of `radius`.
detailX	Integer: number of edges used to draw the base. Defaults to 24.
detailY	Integer: number of triangle subdivisions along the y-axis. Defaults to 1.
cap	Boolean: whether to draw the cone's base. Defaults to `true`.

This page is generated from the comments in src/webgl/3d_primitives.js . Please feel free to edit it and submit a pull request!

Related References

calculateBoundingBox
Calculates the position and size of the smallest box that contains the geometry.

clearColors
Removes the geometry's internal colors.

computeFaces
Computes the geometry's faces using its vertices.

computeNormals
Calculates the normal vector for each vertex on the geometry.