freeGeometry()

Clears a p5.Geometry object from the graphics processing unit (GPU) memory.

p5.Geometry objects can contain lots of data about their vertices, surface normals, colors, and so on. Complex 3D shapes can use lots of memory which is a limited resource in many GPUs. Calling `freeGeometry()` can improve performance by freeing a p5.Geometry object's resources from GPU memory. `freeGeometry()` works with p5.Geometry objects created with beginGeometry() and endGeometry(), buildGeometry(), and loadModel().

The parameter, `geometry`, is the p5.Geometry object to be freed.

Note: A p5.Geometry object can still be drawn after its resources are cleared from GPU memory. It may take longer to draw the first time it's redrawn.

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

Examples

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

  background(200);

  // Create a p5.Geometry object.
  beginGeometry();
  cone();
  let shape = endGeometry();

  // Draw the shape.
  model(shape);

  // Free the shape's resources.
  freeGeometry(shape);
}
```

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

let button;
let particles;

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

  // Create a button to reset the particle system.
  button = createButton('Reset');

  // Call resetModel() when the user presses the button.
  button.mousePressed(resetModel);

  // Add the original set of particles.
  resetModel();
}

function draw() {
  background(50);

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

  // Turn on the lights.
  lights();

  // Style the particles.
  noStroke();

  // Draw the particles.
  model(particles);
}

function resetModel() {
```

Syntax

```
freeGeometry(geometry)
```

Parameters

geometry	p5.Geometry: 3D shape whose resources should be freed.

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.