

createModel()

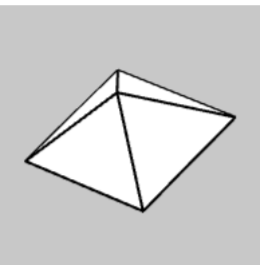
Load a 3d model from an OBJ or STL string.

OBJ and STL files lack a built-in sense of scale, causing models exported from different programs to vary in size. If your model doesn't display correctly, consider using `loadModel()` with `normalize` set to `true` to standardize its size. Further adjustments can be made using the `scale()` function.

Also, the support for colored STL files is not present. STL files with color will be rendered without color properties.

- Options can include:
- `modelString`: Specifies the plain text string of either an stl or obj file to be loaded.
- `fileType`: Defines the file extension of the model.
- `normalize`: Enables standardized size scaling during loading if set to true.
- `successCallback`: Callback for post-loading actions with the 3D model object.
- `failureCallback`: Handles errors if model loading fails, receiving an event error.
- `flipU`: Flips the U texture coordinates of the model.
- `flipV`: Flips the V texture coordinates of the model.

Examples



```
const octahedron_model = `
v 0.000000E+00 0.000000E+00 40.0000
v 22.5000 22.5000 0.000000E+00
v 22.5000 -22.5000 0.000000E+00
v -22.5000 -22.5000 0.000000E+00
v -22.5000 22.5000 0.000000E+00
v 0.000000E+00 0.000000E+00 -40.0000
f 1 2 3
f 1 3 4
f 1 4 5
f 1 5 2
f 6 5 4
f 6 4 3
f 6 3 2
f 6 2 5
`;
//draw a spinning octahedron
let octahedron;

function setup() {
  createCanvas(100, 100, WEBGL);
  octahedron = createModel(octahedron_model, '.obj');
  describe('Vertically rotating 3D octahedron.');
```

```
}

function draw() {
  background(200);
  rotateX(frameCount * 0.01);
  rotateY(frameCount * 0.01);
}
```

Syntax

```
createModel(modelString, [fileType], normalize, [successCallback], [failureCallback])
```

```
createModel(modelString, [fileType], [successCallback], [failureCallback])
```

```
createModel(modelString, [fileType], [options])
```

Parameters

modelString	String: String of the object to be loaded
fileType	String: The file extension of the model (.stl, .obj).
normalize	Boolean: If true, scale the model to a standardized size when loading
successCallback	function(p5.Geometry): Function to be called once the model is loaded. Will be passed the 3D model object.
failureCallback	Function(Event): called with event error if the model fails to load.
options	Object:

Returns

p5.Geometry: the [p5.Geometry](#) object

This page is generated from the comments in [src/webgl/loading.js](#). Please feel free to edit it and submit a pull request!

Related References

createModel

Load a 3d model from an OBJ or STL string.

loadModel

Loads a 3D model to create a p5.Geometry object.

model

Draws a p5.Geometry object to the canvas.

p5.js

Resources

Reference
Tutorials
Examples
Contribute
Community
About
Start Coding
Donate

Information

Download
Contact
Copyright
Privacy Policy
Terms of Use

Socials

GitHub ↗
Instagram ↗
X ↗
YouTube ↗
Discord ↗
Forum ↗

