perspective()

Sets a perspective projection for the current camera in a 3D sketch.

In a perspective projection, shapes that are further from the camera appear smaller than shapes that are near the camera. This technique, called foreshortening, creates realistic 3D scenes. It's applied by default in WebGL mode.

`perspective()` changes the camera's perspective by changing its viewing frustum. The frustum is the volume of space that's visible to the camera. Its shape is a pyramid with its top cut off. The camera is placed where the top of the pyramid should be and views everything between the frustum's top (near) plane and its bottom (far) plane.

The first parameter, `fovy`, is the camera's vertical field of view. It's an angle that describes how tall or narrow a view the camera has. For example, calling `perspective(0.5)` sets the camera's vertical field of view to 0.5 radians. By default, `fovy` is calculated based on the sketch's height and the camera's default z-coordinate, which is 800. The formula for the default `fovy` is `2 * atan(height / 2 / 800)`.

The second parameter, `aspect`, is the camera's aspect ratio. It's a number that describes the ratio of the top plane's width to its height. For example, calling `perspective(0.5, 1.5)` sets the camera's field of view to 0.5 radians and aspect ratio to 1.5, which would make shapes appear thinner on a square canvas. By default, aspect is set to `width / height`.

The third parameter, `near`, is the distance from the camera to the near plane. For example, calling `perspective(0.5, 1.5, 100)` sets the camera's field of view to 0.5 radians, its aspect ratio to 1.5, and places the near plane 100 pixels from the camera. Any shapes drawn less than 100 pixels from the camera won't be visible. By default, near is set to `0.1 * 800`, which is 1/10th the default distance between the camera and the origin.

The fourth parameter, `far`, is the distance from the camera to the far plane. For example, calling `perspective(0.5, 1.5, 100, 10000)` sets the camera's field of view to 0.5 radians, its aspect ratio to 1.5, places the near plane 100 pixels from the camera, and places the far plane 10,000 pixels from the camera. Any shapes drawn more than 10,000 pixels from the camera won't be visible. By default, far is set to `10 * 800`, which is 10 times the default distance between the camera and the origin.

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

Examples

```
// Double-click to squeeze the box.

let isSqueezed = false;

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

  describe('A white rectangular prism on a gray background.
The box appears to become thinner when the user double-
clicks.');
}

function draw() {
  background(200);

  // Place the camera at the top-right.
  camera(400, -400, 800);

  if (isSqueezed === true) {
    // Set fovy to 0.2.
    // Set aspect to 1.5.
    perspective(0.2, 1.5);
  }

  // Draw the box.
  box();
}

// Change the camera's perspective when the user double-clicks.
function doubleClicked() {
  isSqueezed = true;
```

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

  describe('A white rectangular prism on a gray background.
The prism moves away from the camera until it disappears.');
}

function draw() {
  background(200);

  // Place the camera at the top-right.
  camera(400, -400, 800);

  // Set fovy to 0.2.
  // Set aspect to 1.5.
  // Set near to 600.
  // Set far to 1200.
  perspective(0.2, 1.5, 600, 1200);

  // Move the origin away from the camera.
  let x = -frameCount;
  let y = frameCount;
  let z = -2 * frameCount;
  translate(x, y, z);

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

Syntax

```
perspective([fovy], [aspect], [near], [far])
```

Parameters

fovy	Number: camera frustum vertical field of view. Defaults to `2 * atan(height / 2 / 800)`.
aspect	Number: camera frustum aspect ratio. Defaults to `width / height`.
near	Number: distance from the camera to the near clipping plane. Defaults to `0.1 * 800`.
far	Number: distance from the camera to the far clipping plane. Defaults to `10 * 800`.

This page is generated from the comments in src/webgl/p5.Camera.js . Please feel free to edit it and submit a pull request!

Related References

camera
Sets the position and orientation of the camera.

centerX
The x-coordinate of the place where the camera looks.

centerY
The y-coordinate of the place where the camera looks.

centerZ
The y-coordinate of the place where the camera looks.