camera()

Sets the position and orientation of the camera.

`myCamera.camera()` allows objects to be viewed from different angles. It has nine parameters that are all optional.

The first three parameters, `x`, `y`, and `z`, are the coordinates of the camera's position in "world" space. For example, calling `myCamera.camera(0, 0, 0)` places the camera at the origin `(0, 0, 0)`. By default, the camera is placed at `(0, 0, 800)`.

The next three parameters, `centerX`, `centerY`, and `centerZ` are the coordinates of the point where the camera faces in "world" space. For example, calling `myCamera.camera(0, 0, 0, 10, 20, 30)` places the camera at the origin `(0, 0, 0)` and points it at `(10, 20, 30)`. By default, the camera points at the origin `(0, 0, 0)`.

The last three parameters, `upX`, `upY`, and `upZ` are the components of the "up" vector in "local" space. The "up" vector orients the camera's y-axis. For example, calling `myCamera.camera(0, 0, 0, 10, 20, 30, 0, -1, 0)` places the camera at the origin `(0, 0, 0)`, points it at `(10, 20, 30)`, and sets the "up" vector to `(0, -1, 0)` which is like holding it upside-down. By default, the "up" vector is `(0, 1, 0)`.

Examples

```
// Double-click to toggle between cameras.

let cam1;
let cam2;
let isDefaultCamera = true;

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

  // Create the first camera.
  // Keep its default settings.
  cam1 = createCamera();

  // Create the second camera.
  cam2 = createCamera();

  // Place it at the top-right: (1200, -600, 100)
  // Point it at the row of boxes: (-10, -10, 400)
  // Set its "up" vector to the default: (0, 1, 0)
  cam2.camera(1200, -600, 100, -10, -10, 400, 0, 1, 0);

  // Set the current camera to cam1.
  setCamera(cam1);

  describe(
    'A row of white cubes against a gray background. The
camera toggles between a frontal and an aerial view when the
user double-clicks.'
```

```
// Double-click to toggle between cameras.

let cam1;
let cam2;
let isDefaultCamera = true;

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

  // Create the first camera.
  // Keep its default settings.
  cam1 = createCamera();

  // Create the second camera.
  cam2 = createCamera();

  // Place it at the right: (1200, 0, 100)
  // Point it at the row of boxes: (-10, -10, 400)
  // Set its "up" vector to the default: (0, 1, 0)
  cam2.camera(1200, 0, 100, -10, -10, 400, 0, 1, 0);

  // Set the current camera to cam1.
  setCamera(cam1);

  describe(
    'A row of white cubes against a gray background. The
camera toggles between a static frontal view and an orbiting
view when the user double-clicks.'
```

Syntax

```
camera([x], [y], [z], [centerX], [centerY], [centerZ], [upX], [upY], [upZ])
```

Parameters

x	Number: x-coordinate of the camera. Defaults to 0.
y	Number: y-coordinate of the camera. Defaults to 0.
z	Number: z-coordinate of the camera. Defaults to 800.
centerX	Number: x-coordinate of the point the camera faces. Defaults to 0.
centerY	Number: y-coordinate of the point the camera faces. Defaults to 0.
centerZ	Number: z-coordinate of the point the camera faces. Defaults to 0.
upX	Number: x-component of the camera's "up" vector. Defaults to 0.
upY	Number: x-component of the camera's "up" vector. Defaults to 1.
upZ	Number: z-component of the camera's "up" vector. Defaults to 0.

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.