slerp()

Calculates a new heading and magnitude that are between two vectors.

The `amt` parameter is the amount to interpolate between the old vector and the new vector. 0.0 keeps the heading and magnitude equal to the old vector's, 0.5 sets them halfway between, and 1.0 sets the heading and magnitude equal to the new vector's.

`slerp()` differs from lerp() because it interpolates magnitude. Calling `v0.slerp(v1, 0.5)` sets `v0`'s magnitude to a value halfway between its original magnitude and `v1`'s. Calling `v0.lerp(v1, 0.5)` makes no such guarantee.

The static version of `slerp()`, as in `p5.Vector.slerp(v0, v1, 0.5)`, returns a new p5.Vector object and doesn't change the original.

Examples

```
function setup() {
  // Create a p5.Vector object.
  let v0 = createVector(3, 0);

  // Prints "3" to the console.
  print(v0.mag());

  // Prints "0" to the console.
  print(v0.heading());

  // Create a p5.Vector object.
  let v1 = createVector(0, 1);

  // Prints "1" to the console.
  print(v1.mag());

  // Prints "1.570..." to the console.
  print(v1.heading());

  // Interpolate halfway between v0 and v1.
  v0.slerp(v1, 0.5);

  // Prints "2" to the console.
  print(v0.mag());

  // Prints "0.785..." to the console.
  print(v0.heading());
}
```

```
function setup() {
  // Create a p5.Vector object.
  let v0 = createVector(3, 0);

  // Prints "3" to the console.
  print(v0.mag());

  // Prints "0" to the console.
  print(v0.heading());

  // Create a p5.Vector object.
  let v1 = createVector(0, 1);

  // Prints "1" to the console.
  print(v1.mag());

  // Prints "1.570..." to the console.
  print(v1.heading());

  // Create a p5.Vector that's halfway between v0 and v1.
  let v3 = p5.Vector.slerp(v0, v1, 0.5);

  // Prints "2" to the console.
  print(v3.mag());

  // Prints "0.785..." to the console.
  print(v3.heading());
}
```

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

  describe('Three arrows extend from the center of a gray
square. A red arrow points to the right, a blue arrow points
to the left, and a purple arrow points down.');
}

function draw() {
  background(200);

  // Create p5.Vector objects.
  let v0 = createVector(50, 50);
  let v1 = createVector(20, 0);
  let v2 = createVector(-40, 0);

  // Create a p5.Vector that's halfway between v1 and v2.
  let v3 = p5.Vector.slerp(v1, v2, 0.5);

  // Draw the red arrow.
  drawArrow(v0, v1, 'red');

  // Draw the blue arrow.
  drawArrow(v0, v2, 'blue');

  // Draw the purple arrow.
  drawArrow(v0, v3, 'purple');
}

// Draws an arrow between two vectors.
```

Syntax

```
slerp(v, amt)
```

```
slerp(v1, v2, amt, [target])
```

Parameters

v	p5.Vector: p5.Vector to slerp toward.
amt	Number: amount of interpolation between 0.0 (old vector) and 1.0 (new vector). 0.5 is halfway between.
v1	p5.Vector: old vector.
v2	p5.Vector: new vector.
target	p5.Vector: vector to receive the result.

Returns

p5.Vector:

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

Related References

add
Adds to a vector's x, y, and z components.

angleBetween
Calculates the angle between two vectors.

array
Returns the vector's components as an array of numbers.

clampToZero
Replaces the components of a p5.Vector that are very close to zero with zero.