

createVector()

Creates a new `p5.Vector` object.

A vector can be thought of in different ways. In one view, a vector is like an arrow pointing in space. Vectors have both magnitude (length) and direction. This view is helpful for programming motion.

A vector's components determine its magnitude and direction. For example, calling `createVector(3, 4)` creates a new `p5.Vector` object with an x-component of 3 and a y-component of 4. From the origin, this vector's tip is 3 units to the right and 4 units down.

`p5.Vector` objects are often used to program motion because they simplify the math. For example, a moving ball has a position and a velocity. Position describes where the ball is in space. The ball's position vector extends from the origin to the ball's center. Velocity describes the ball's speed and the direction it's moving. If the ball is moving straight up, its velocity vector points straight up. Adding the ball's velocity vector to its position vector moves it, as in `pos.add(vel)`. Vector math relies on methods inside the `p5.Vector` class.

Examples

```
•   ▶   ◻
      .   .   .
```

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

  background(200);

  // Create p5.Vector objects.
  let p1 = createVector(25, 25);
  let p2 = createVector(50, 50);
  let p3 = createVector(75, 75);

  // Draw the dots.
  strokeWeight(5);
  point(p1);
  point(p2);
  point(p3);

  describe('Three black dots form a diagonal line from top left to bottom right.');
}
```

```
▶   ◻
      .
```

```
let pos;
let vel;

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

  // Create p5.Vector objects.
  pos = createVector(50, 100);
  vel = createVector(0, -1);

  describe('A black dot moves from bottom to top on a gray square. The dot reappears at the bottom when it reaches the top.');
}

function draw() {
  background(200);

  // Add velocity to position.
  pos.add(vel);

  // If the dot reaches the top of the canvas,
  // restart from the bottom.
  if (pos.y < 0) {
    pos.y = 100;
  }

  // Draw the dot.
  strokeWeight(5);
  point(pos);
}
```

```
▶   ◻
      .
```

```
let pos;
let vel;

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

  // Create p5.Vector objects.
  pos = createVector(50, 100);
  vel = createVector(0, -1);

  describe('A black dot moves from bottom to top on a gray square. The dot reappears at the bottom when it reaches the top.');
}

function draw() {
  background(200);

  // Add velocity to position.
  pos.add(vel);

  // If the dot reaches the top of the canvas,
  // restart from the bottom.
  if (pos.y < 0) {
    pos.y = 100;
  }

  // Draw the dot.
  strokeWeight(5);
  point(pos);
}
```

```
▶   ◻
      .
```

Syntax

```
createVector([x], [y], [z])
```

Parameters

x Number: x component of the vector.
y Number: y component of the vector.
z Number: z component of the vector.

Returns

`p5.Vector`: new `p5.Vector` object.

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

Related References

<code>add</code>	<code>angleBetween</code>	<code>array</code>	<code>clampToZero</code>
Adds to a vector's x, y, and z components.	Calculates the angle between two vectors.	Returns the vector's components as an array of numbers.	Replaces the components of a <code>p5.Vector</code> that are very close to zero with zero.

<code>p5.js</code>	<code>Resources</code>	<code>Information</code>	<code>Socials</code>
	Reference Tutorials Examples Contribute Community About Start Coding Donate	Download Contact Copyright Privacy Policy Terms of Use	GitHub  Instagram  X  YouTube  Discord  Forum 

	Reference Tutorials Examples Contribute Community About Start Coding Donate	Download Contact Copyright Privacy Policy Terms of Use	GitHub  Instagram  X  YouTube  Discord  Forum 
---	--	--	---

	Reference Tutorials Examples Contribute Community About Start Coding Donate	Download Contact Copyright Privacy Policy Terms of Use	GitHub  Instagram  X  YouTube  Discord  Forum 
---	--	--	---

	Reference Tutorials Examples Contribute Community About Start Coding Donate	Download Contact Copyright Privacy Policy Terms of Use	GitHub  Instagram  X  YouTube  Discord  Forum 
---	--	--	---

	Reference Tutorials Examples Contribute Community About Start Coding Donate	Download Contact Copyright Privacy Policy Terms of Use	GitHub  Instagram  X  YouTube  Discord  Forum 
---	--	--	---

	Reference Tutorials Examples Contribute Community About Start Coding Donate	Download Contact Copyright Privacy Policy Terms of Use	GitHub Instagram X YouTube Discord Forum
---	--	--	---

	Reference Tutorials Examples Contribute Community About Start Coding Donate	Download Contact Copyright Privacy Policy Terms of Use	GitHub Instagram X YouTube Discord <img alt="Discord icon" data-bbox="865
---	--	--	---