

# String

A sequence of text characters.

The `String` data type is helpful for working with text. For example, a string could contain a welcome message:

```
// Use a string literal.  
text('Hello!', 10, 10);
```

```
// Create a string variable.  
let message = 'Hello!';  
  
// Use the string variable.  
text(message, 10, 10);
```

The most common way to create strings is to use some form of quotations as follows:

```
text("hi", 50, 50);
```

```
text('hi', 50, 50);
```

```
text(`hi`, 50, 50);
```

`"hi"`, `'hi'`, and `hi` are all string literals. A "literal" means a value was actually written, as in `text('hi', 50, 50)`. By contrast, `text(message, 50, 50)` uses the variable `message`, so it isn't a string literal.

Single quotes `' '` and double quotes `" "` mean the same thing. It's nice to have the option for cases when a string contains one type of quote:

```
text("What's up?", 50, 50);
```

```
text('Air quotes make you look "cool."', 50, 50);
```

Backticks `` `` create template literals. Template literals have many uses. For example, they can contain both single and double quotes as needed:

```
text(`Don't you forget about me`, 10, 10);
```

Template literals are helpful when strings are created from variables like so:

```
let size = random(10, 20);  
circle(50, 50, size);  
  
text(`The circle's diameter is ${size} pixels.`, 10, 10);
```

The `size` variable's value will replace `${size}` when the string is created. `${}` is a placeholder for any value. That means an expression can be used, as in `${round(PI, 3)}`. All of the following are valid template literals:

```
text(`π is about ${round(PI, 2)} pixels.`, 10, 10);  
text(`It's ${mouseX < width / 2} that I'm on the left half of the canvas.`, 10, 30);
```

Template literals can include several variables:

```
let x = random(0, 100);  
let y = random(0, 100);  
let size = random(10, 20);  
circle(x, y, size);  
  
text(`The circle at (${x}, ${y}) has a diameter of ${size} pixels.`, 10, 10);
```

Template literals are also helpful for creating multi-line text like so:

```
let poem = `My sketch doesn't run;  
it waits for me patiently  
while bugs point the way.`;  
  
text(poem, 10, 10);
```

## Examples

▶

■

```
function setup() {  
  createCanvas(100, 100);  
  
  background(200);  
  
  // Style the text.  
  textAlign(CENTER, CENTER);  
  textSize(20);  
  
  // Display a welcome message.  
  text('Hello!', 50, 50);  
  
  describe('The text "Hello!" written on a gray background.');
```

▶

■

```
function setup() {  
  createCanvas(100, 100);  
  
  background(200);  
  
  // Style the text.  
  textAlign(CENTER, CENTER);  
  textSize(20);  
  
  // Create a string variable.  
  let world = '🌍';  
  
  // Display a welcome message using a template string.  
  text(`Hello, ${world}!`, 50, 50);  
  
  describe('The text "Hello, 🌍!" written on a gray background.');
```

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

## Related References

<b>class</b> A template for creating objects of a particular type.	<b>console</b> Prints a message to the web browser's console.	<b>for</b> A way to repeat a block of code when the number of iterations is known.	<b>function</b> A named group of statements.
---	--	---	---

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 ↗

