

TypeIt

Documentation

Overview

TypeIt is the most versatile jQuery animated typing plugin on the planet. In simple use, it allows you to type single or multiple strings that break lines, delete & replace each other, and it even handles HTML tags & entities.

For more advanced, controlled typing effects, TypeIt comes with companion functions that can be chained to control your typing down to the smallest character, enabling you to type an dynamic narrative, with complete control over speed, characters, line breaks, deletions, pausing, everything.

Some of the Features

- Choose to start typing only when your element becomes visible on the screen.
- Loop your string(s) continuously.
- Define you strings via JSON or in the HTML (useful in case user doesn't have JavaScript enabled).
- Create several unique instances on a single page.
- Handle HTML tags (including your custom classes, ID's, etc.) and entities with ease.
- Use companion functions to chain individual commands together for fine tuning your typing.
- Compatible from jQuery 1.8.0 through 3.0.0.
- Lightweight. (~5kb, single JavaScript file)

Setup

Get the Code

Download from the following sources:

- **CDN:** The recommended way to use TypeIt.

```
https://cdn.jsdelivr.net/jquery.typeit/4.2.3/typeit.min.js
```

```
https://cdnjs.cloudflare.com/ajax/libs/typeit/4.2.3/typeit.min.js
```

- **npm:** Install with the following command.

```
npm install typeit
```

- **Github:** Clone the repository.

```
git clone https://github.com/alexmacarthur/typeit.git
```

Hook It Up

1. Load jQuery and typeit.js on your page.

```
<script src="jquery-2.1.4.min.js"></script>
<script src="typeit.js"></script>
```

2. Create an empty HTML element to select. (If you want to have a fallback for users without JavaScript, you can put a string or strings right into this element. **More on that later.**)

```
<span class="type-it"></span>
```

Simple Usage

In it's simplest use, just call `typeit()` on any element and include your **options**.

```
$('.type-it').typeit({
  strings: ['This is my string!']
});
```

About Defining Your Strings

Via JSON

Define your string(s) by using the `strings` option.

```
strings: 'Here is my string!'
```

Directly in the HTML

As a fallback for users without JavaScript, you may define strings in your HTML element. However, **this will override any strings you've defined via JSON.**

```
<span class="type-it">Here is a string.</span>
```

Typing Multiple Strings

You can also configure TypeIt to type multiple strings in a single call.

If you define your strings within your HTML element, just separate them with `
` tags:

```
<span class="type-it">Here is a string. <br>And here is another!</span>
```

If they're defined via JSON, it's possible to put them inside quotation marks separated by `
` tags, but it's recommended that you use an array.

```
$('.type-it').typeIt({  
  strings: ['Enter your string here!', 'Another string!']  
});
```

By default, multiple strings will break lines (`breakLines: true`). However, you can also set them to delete and replace each other.

```
$('.type-it').typeIt({  
  strings: ['Enter your string here!', 'Another string!'],  
  breakLines: false  
});
```

Advanced Usage

To control your typing down to the smallest character, there are five companion functions available. Simply chain them together following a `typeIt()` call on an element, and your chain will execute.

Note that when using these functions, the `strings` option should not be defined.

```
$('.type-it').typeIt({  
  speed: 900,  
  lifeLike: false,  
  autoStart: false  
})  
.tiType('I am typing slowly,')  
.tiSettings({speed: 100})  
.tiType('but now I am typing pretty fast!')  
.tiDelete(2)  
.tiType('t!');
```

Companion Functions

Function	Arguments	Description
----------	-----------	-------------

tiType()	(string)	Will type the characters.
tiDelete()	(number)	Will delete the number of characters/entities entered.
tiPause()	(number)	Will pause the specified number of milliseconds.
tiBreak()	(none)	Will break the typing to a new line.
tiEmpty()	(none)	Will instantly delete everything that's been typed.
tiSettings()	(JSON)	Will redefine your options on the fly. This will only work for updating the <i>speed</i> , <i>lifeLike</i> , and <i>html</i> options.

About HTML Handling

TypeIt is fully prepared to handle HTML in your strings.

HTML Elements

Tags must be one level deep and be inline elements.

```
// GOOD! :) $('typeit-box').typeit({
  strings: '<h1 class="your-class">This is a string!</h1>', }
```

```
// BAD! :( $('typeit-box').typeit({
  strings: '<h1 class="your-class"><span>This is a string!</span></h1>', }
```

HTML Entities

ASCII HTML entities must begin with & and end with ;.

```
$('.typeit-box').typeit({
  strings: '<h1 class="your-class">I really &hearts; Life cereal.</h1>', }
```

Options

Option	Description	Default
	(string, array) The string or strings you'd like to type. Define a single string within quotes or define multiple with an array.	
strings	strings: 'Hello! This is my string.' strings: ['Hello!', 'This is another string.']}	(nothing)
speed	(number) The speed (milliseconds) at which you want to	100

	type.	
deleteSpeed	(number) The speed (milliseconds) at which you want to delete. If left undefined, 1/3 of the speed will be used.	undefined
lifeLike	(boolean) Choose whether the plugin types at a constant pace or irregular, life-like pace. Pace is calculated by choosing a random value between a range 50% above and below the type or delete speed will be used.	true
cursor	(boolean) Choose whether you want the blinking cursor to appear when typing.	true
cursorSpeed	(number) The blinking speed (milliseconds) of the cursor.	1000
breakLines	(boolean) Choose whether you want multiple strings to be printed on top of each other (breakLines = true), or if you want each string to be deleted and replaced by the next one (breakLines = false).	true
breakDelay	(number) The amount of time (milliseconds) between typing multiple strings.	750
deleteDelay	(number) The amount of time (milliseconds) between deleting the current string and continuing to type additional strings. Only effective when breakLines is set to false.	750
startDelay	(number) The amount of time (milliseconds) before the plugin begins typing after initializing.	250
startDelete	<p>If true, strings within the element upon initialization will first be deleted before typing any strings that are defined via JSON. Requires that a string or strings are defined within the HTML element before initializing. Additionally, if you don't define more strings via JSON or by using companion functions, no further typing will take place after these strings are deleted.</p> <p><code>This will first be deleted...</code></p> <pre>\$('#element').typeIt({ startDelete: true, strings: 'And then this will be typed.' });</pre>	true
loop	(boolean) Have your string or strings continuously loop after completing.	false

loopDelay	(number) The amount of time (milliseconds) between looping over a string or set of strings again. Half of this delay will occur after the remaining strings have been successfully deleted, and half will occur before the strings begin to type again.	750
html	Whether HTML tags and entities will be rendered when typing.	true
autoStart	Whether TypeIt will begin typing automatically on page load (true), or only when the element becomes visible in the viewport (false).	true
callback	<p>A function that executes after your typing has completed. Must be passed as a function. Example:</p> <pre>// BAD :(callback: alert('done!'); // GOOD :) callback: function() { alert('done!'); }</pre>	(nothing)

Contributions

Gulp is configured to check & minify the JavaScript. Just use the following command to make it happen (make sure Node.js, npm, and Gulp are installed on your computer).

```
npm install
gulp develop
```