



## JavaScript (ES6) code snippets

charalampos karypidis | 2,955,434 installs | ★★★★★ (22) | Free

Code snippets for JavaScript in ES6 syntax

[Install](#)[Trouble Installing?](#)[Overview](#)[Q & A](#)[Rating & Review](#)

### JavaScript

### VS Code JavaScript (ES6) snippets

Visual Studio Marketplace **v1.7.2** | installs **2955336** | rating average: 4.77/5 (22 ratings)

This extension contains code snippets for JavaScript in ES6 syntax for [Vs Code](#) editor (supports both JavaScript and TypeScript).

#### Note

All the snippets include the final semicolon ; There is a fork of those snippets [here](#) made by @jmsv where semicolons are not included. So feel free to use them according to your needs.

#### Installation

In order to install an extension you need to launch the Command Palette (Ctrl + Shift + P or Cmd + Shift + P) and type Extensions. There you have either the option to show the already installed snippets or install new ones. Search for *JavaScript (ES6) code snippets* and install it.

#### Supported languages (file extensions)

- JavaScript (.js)
- TypeScript (.ts)
- JavaScript React (.jsx)
- TypeScript React (.tsx)
- Html (.html)
- Vue (.vue)

#### Snippets

Below is a list of all available snippets and the triggers of each one. The → means the TAB key.

#### Import and export

Trigger	Content
imp→	imports entire module <code>import fs from 'fs';</code>
imn→	imports entire module without module name <code>import 'animate.css'</code>
imd→	imports only a portion of the module using destructuring <code>import {rename} from 'fs';</code>
ime→	imports everything as alias from the module <code>import * as localAlias from 'fs';</code>
ima→	imports only a portion of the module as alias <code>import { rename as localRename } from 'fs';</code>
rqr→	require package <code>require('');</code>
mde→	default module.exports <code>module.exports = {};</code>
enf→	exports name function <code>export const log = (parameter) =&gt; { console.log(parameter);};</code>

#### Categories

[Snippets](#)

#### Tags

[javascript](#)[snippet](#)

#### Resources

[Repository](#)[Download Extension](#)

#### Project Details

[xabikos/vscode-javascript](#)

Last Commit: a year ago

7 Pull Requests

11 Open Issues

#### More Info

Version 1.7.2  
Released on 11/18/2015, 2:58:11 AM  
Last updated 12/24/2018, 7:37:47 PM  
Publisher charalampos karypidis  
Unique Identifier xabikos.JavaScriptSnippets  
Report [Report Abuse](#)



Trigger	Content
edf→	exports default function export default (parameter) => { console.log(parameter);};
ec1→	exports default class export default class Calculator { };
ece→	exports default class by extending a base one export default class Calculator extends BaseClass { };

### Class helpers

Trigger	Content
con→	adds default constructor in the class constructor() {}
met→	creates a method inside a class add() {}
pge→	creates a getter property get propertyName() {return value;}
pse→	creates a setter property set propertyName(value) {}

### Various methods

Trigger	Content
fre→	forEach loop in ES6 syntax array.forEach(currentItem => {})
fof→	for ... of loop for(const item of object) {}
fin→	for ... in loop for(const item in object) {}
anfn→	creates an anonymous function (params) => {}
nfn→	creates a named function const add = (params) => {}
dob→	destructuring object syntax const {rename} = fs
dar→	destructuring array syntax const [first, second] = [1,2]
sti→	set interval helper method setInterval(() => {});
sto→	set timeout helper method setTimeout(() => {});
prom→	creates a new Promise return new Promise((resolve, reject) => {});
thenc→	adds then and catch declaration to a promise .then((res) => {}).catch((err) => {});

### Console methods

Trigger	Content
cas→	console alert method console.assert(expression, object)
ccl→	console clear console.clear()
cco→	console count console.count(label)
cdi→	console dir console.dir
cer→	console error console.error(object)
cgr→	console group console.group(label)
cge→	console groupEnd console.groupEnd()
clg→	console log console.log(object)
clo→	console log object with name console.log('object :', object);
ctr→	console trace console.trace(object)
cwa→	console warn console.warn
cin→	console info console.info
clt→	console table console.table
cti→	console time console.time
cte→	console timeEnd console.timeEnd