* [Docs](https://vscode.readthedocs.io/en/latest/) »

* Nodejs »

* Nodejs tutorial

**Node.js Tutorial in VS Code**

[Node.js](https://nodejs.org/) is a platform for building fast and scalable server applications using JavaScript. Node.js is the runtime and [NPM](https://www.npmjs.com/) is the Package Manager for Node.js modules.

Visual Studio Code has support for the JavaScript and TypeScript languages out-of-the-box as well as Node.js debugging. However, to run a Node.js application, you will need to install the Node.js runtime on your machine.

**Are you new to VS Code?** Learn more and download a [faster Node.js editor here](https://vscode.readthedocs.io/en/latest/nodejs).

To get started in this walkthrough, [install Node.js for your platform](https://nodejs.org/en/download/). The Node Package Manager is included in the Node.js distribution. You'll need to open a new terminal (command prompt) for the node and npm command line tools to be on your PATH.

**Linux**: There are specific Node.js packages available for the various flavors of Linux. See [Installing Node.js via package manager](https://nodejs.org/en/download/package-manager) to find the Node.js package and installation instructions tailored to your version of Linux.

**Tip:** To test that you've got Node.js correctly installed on your computer, open a new terminal and type node --help and you should see the usage documentation.

**Hello World**

Let's get started by creating the simplest Node.js application, "Hello World".

Create an empty folder called "hello", navigate into and open VS Code:

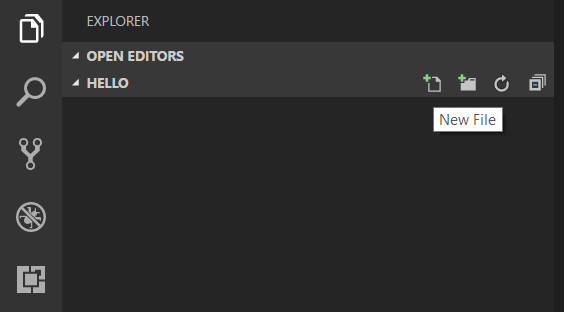
mkdir hello

cd hello

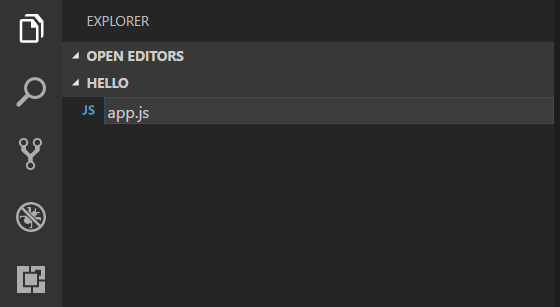
code .

**Tip:** You can open files or folders directly from the command line. The period '.' refers to the current folder, therefore VS Code will start and open the Hello folder.

From the File Explorer tool bar, press the New File button:



and name the file app.js:



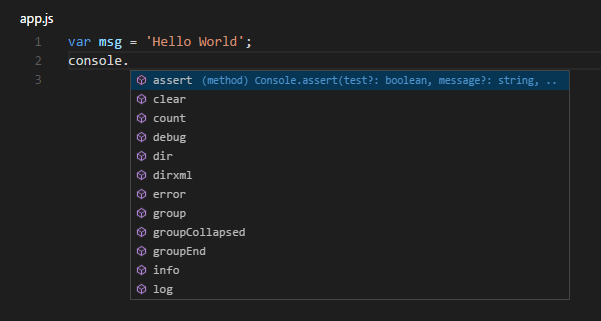
By using the .js file extension, VS Code interprets this file as JavaScript and will evaluate the contents with the JavaScript language service.

Create a simple string variable in app.js and send the contents of the string to the console:

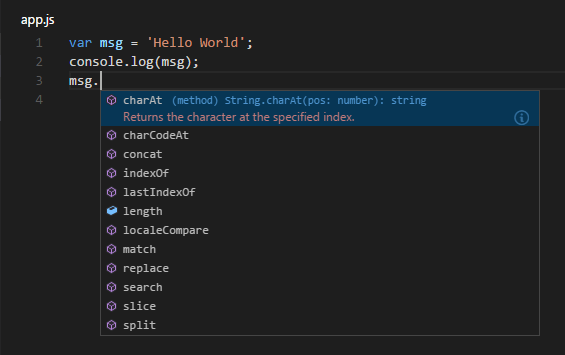
**var** msg = 'Hello World';

console.log(msg);

Note that when you typed console. [IntelliSense](https://vscode.readthedocs.io/docs/editor/intellisense.md) on the console object was automatically presented to you. When editing JavaScript files, VS Code will automatically provide you with IntelliSense for the DOM.



Also notice that VS Code knows that msg is a string based on the initialization to 'Hello World'. If you type msg. you'll see IntelliSense showing all of the string functions available on msg.



After experimenting with IntelliSense, revert any extra changes from the source code example above and save the file (kb(workbench.action.files.save)).

Running Hello World

It's simple to run app.js with Node.js. From a terminal, just type:

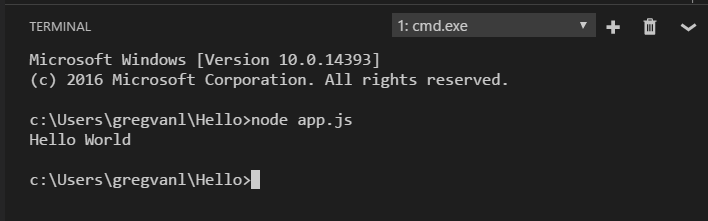
node app.js

You should see "Hello World" output to the terminal and then Node.js returns.

Integrated Terminal

VS Code has an [integrated terminal](https://vscode.readthedocs.io/docs/editor/integrated-terminal.md) which you can use to run shell commands. You can run Node.js directly from there and avoid switching out of VS Code while running command line tools.

**View** > **Integrated Terminal** (kb(workbench.action.terminal.toggleTerminal) with the backtick character) will open the integrated terminal and you can run node app.js there:

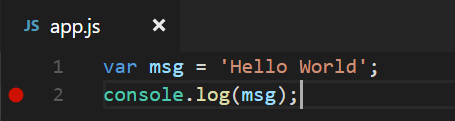


For this walkthrough, you can use either an external terminal or the VS Code integrated terminal for running the command line tools.

Debugging Hello World

As mentioned in the introduction, VS Code ships with a debugger for Node.js applications. Let's try debugging our simple Hello World application.

To set a breakpoint in app.js, put the editor cursor on the first line and press kb(editor.debug.action.toggleBreakpoint) or click in the editor left gutter next to the line numbers. A red circle will appear in the gutter.



To start debugging, select the Debug View in the Side Bar:

Debug icon

You can now click Debug tool bar green arrow or press kb(workbench.action.debug.start) to launch and debug "Hello World". Your breakpoint will be hit and you can view and step through the simple application. Notice that VS Code displays an orange Status Bar to indicate it is in Debug mode and the DEBUG CONSOLE is displayed.

Now that you've seen VS Code in action with "Hello World", the next section shows using VS Code with a full-stack Node.js web app.

**Note:** We're done with the "Hello World" example so navigate out of that folder before you create an Express app. You can delete the "Hello" folder if you wish as it is not required for the rest of the walkthrough.

