

# Node.js Mac

## Node/NPM/Validator Setup - Mac

### Command Line Reference

#### Key Controls/Shortcuts

<b>Enter/Return</b>	Executes the current command
<b>Up/Down Arrow</b>	Auto-fills previously/next run commands
<b>Control+C</b>	Stops the process currently running
<b>Tab</b>	Autocomplete a file or directory name

#### File System Commands

<b>ls</b>	Lists all of the files/folders inside of the current directory
<b>cd</b>	Changes directory to the folder you specify (not including the <>)
<b>cd ..</b>	Goes up one to the parent of the current directory
<b>pwd</b>	Displays the full path to the current directory

## 1) Installing Node.js and NPM



### Preface/Explanation

Programming languages are just like real languages. They are used to communicate ideas in a structured way. In this course, we will be using the JavaScript programming language to both communicate to your browser how to handle the behavior of your website and to communicate what we want our server to do and respond to requests with.

On the client-side, your JavaScript is interpreted by your browser, so it knows how to respond to user events like button clicks and pages loads. On the server-side, your JavaScript is interpreted by a piece of software called Node.js, so that it knows how to respond to user requests over the network. This includes responding with the client-side files (HTML/CSS/Client-side JavaScript) for the client to display, or responding with information stored or calculated on the server. While developing for this course, we will be running both the client-side browser and the server on the same machine (your computer). (We will go more in depth into the server-side later on in the course).

There are a wide array of community-made modules that expand upon the default features of Node.js. We will be using some of these modules in this class. In order to install and manage these modules, we will use a program called npm that is included when installing Node.js. Follow the steps below to install both Node.js and npm.

### Steps

1. Start off by downloading the LTS (version number  $\geq 14.15.0$ ) Node.js/NPM  installer from <https://nodejs.org/en/download/>  (<https://nodejs.org/en/download/>). Then, run the installer to start the

installation process.

**Downloads**

Latest LTS Version: **12.13.0** (includes npm 6.12.0)

Download the Node.js source code or a pre-built installer for your platform, and start developing today.

**1)** **LTS**  
Recommended For Most Users

**2)** **Current**  
Latest Features

**Windows Installer**  
node-v12.13.0-x86.msi

**macOS Installer**  
node-v12.13.0.pkg

**Source Code**  
node-v12.13.0.tar.gz

**Windows Installer (.msi)**

**Windows Binary (.zip)**

**macOS Installer (.pkg)**

**macOS Binary (.tar.gz)**

**Linux Binaries (x64)**

**Linux Binaries (ARM)**

**Source Code**

**Additional Platforms**

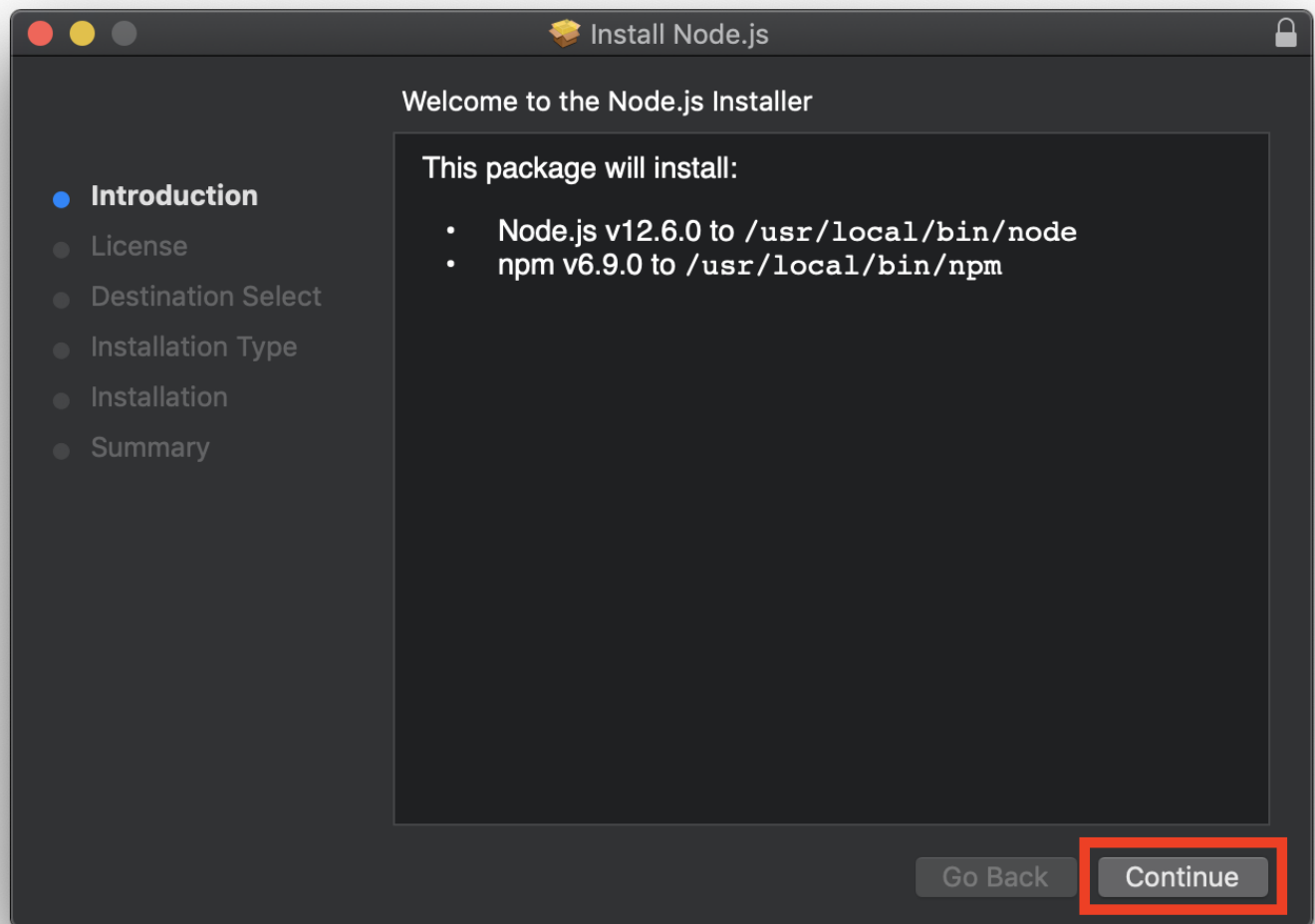
**SmartOS Binaries**

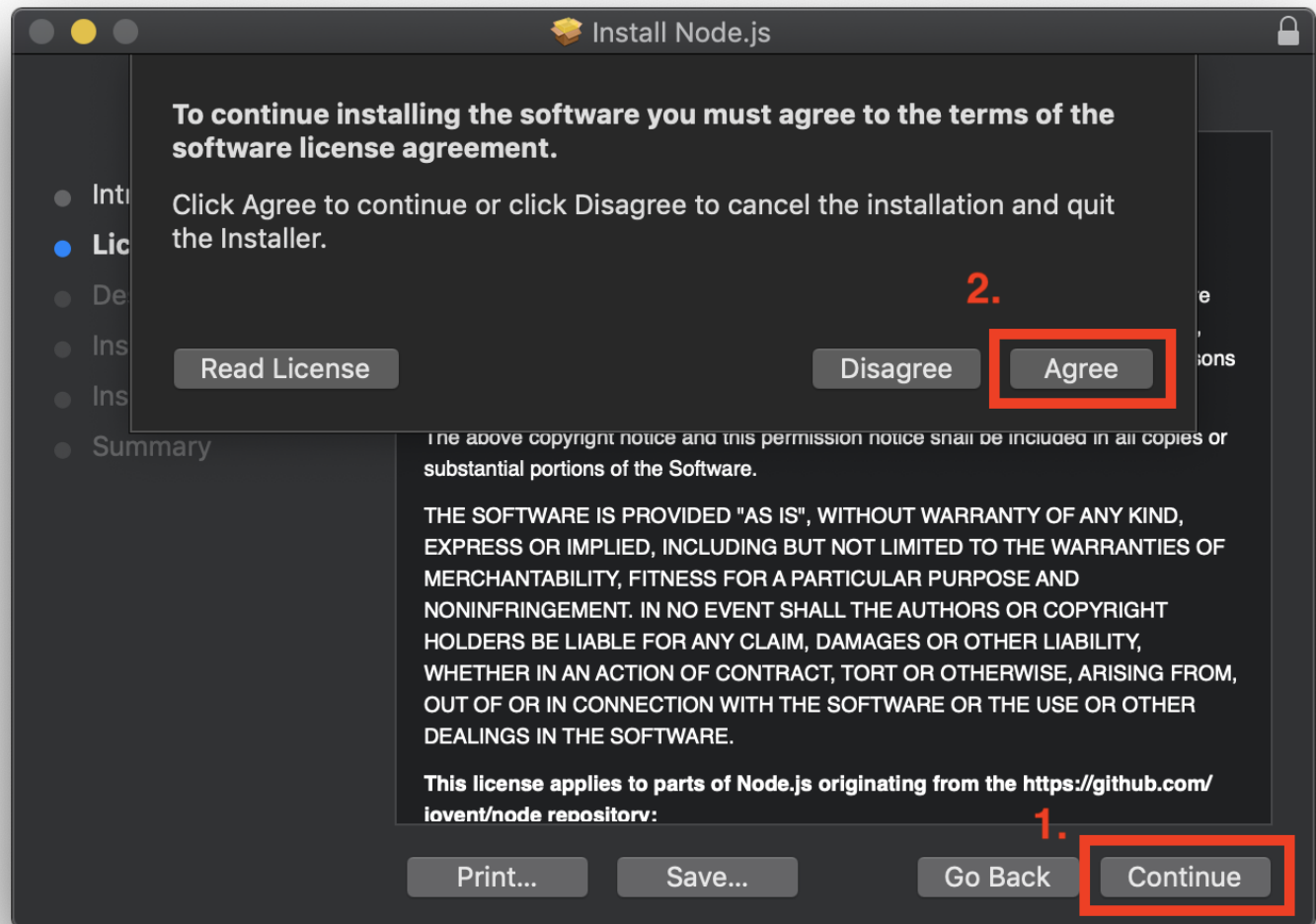
**Docker Image**

node-v12.13.0.pkg

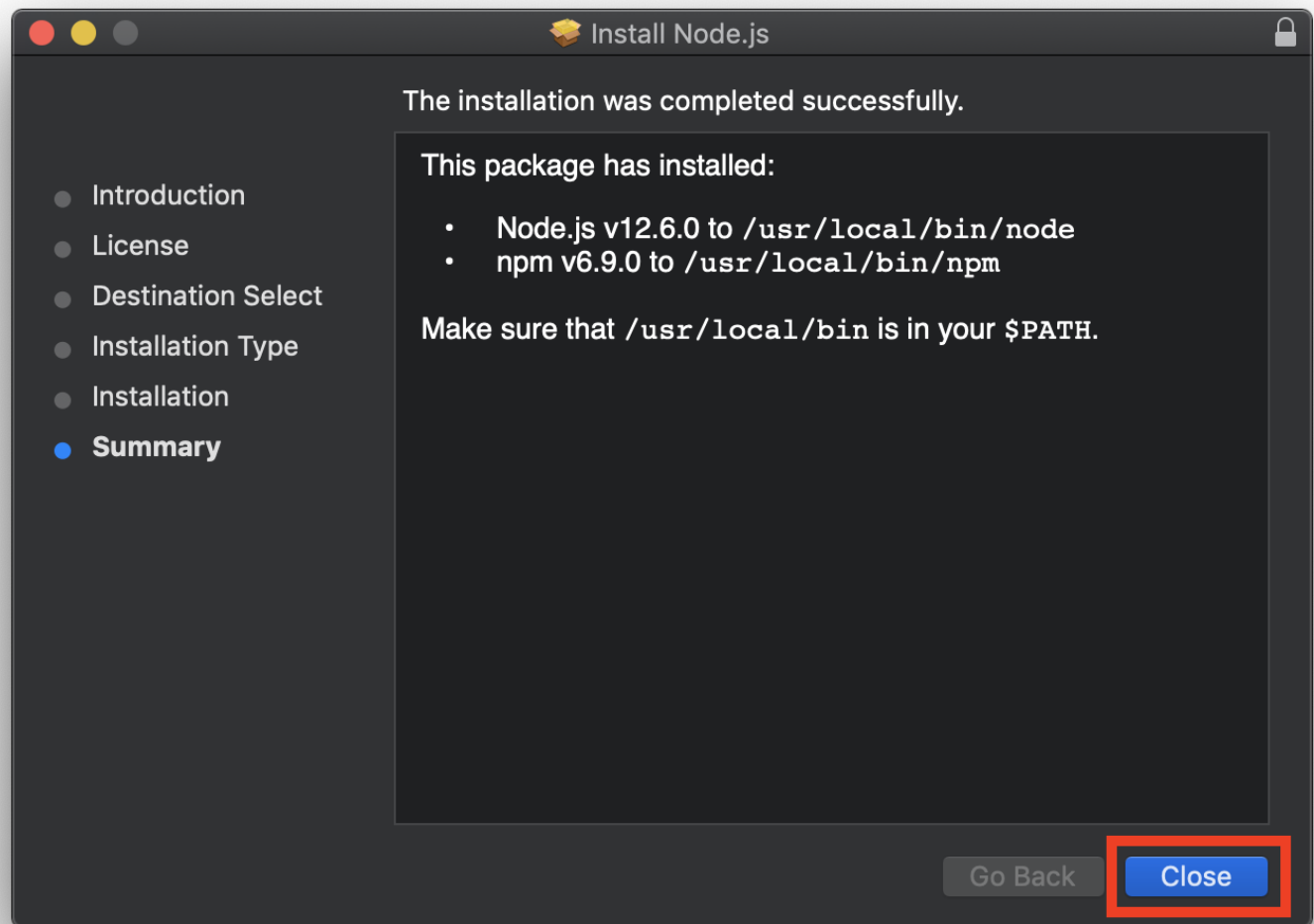
Show All

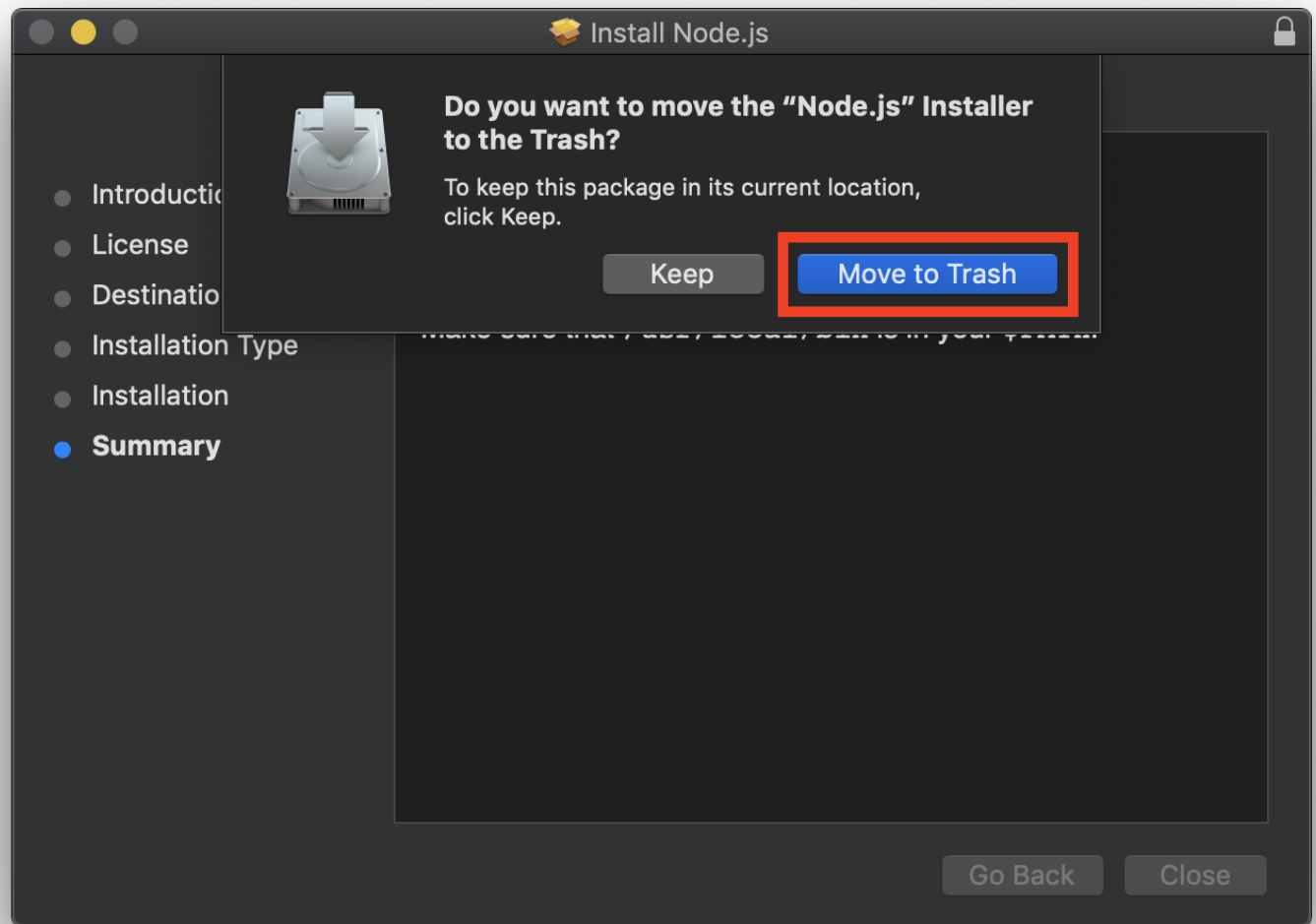
2. Go through the prompts hitting **continue** / **install** each time with the default options.



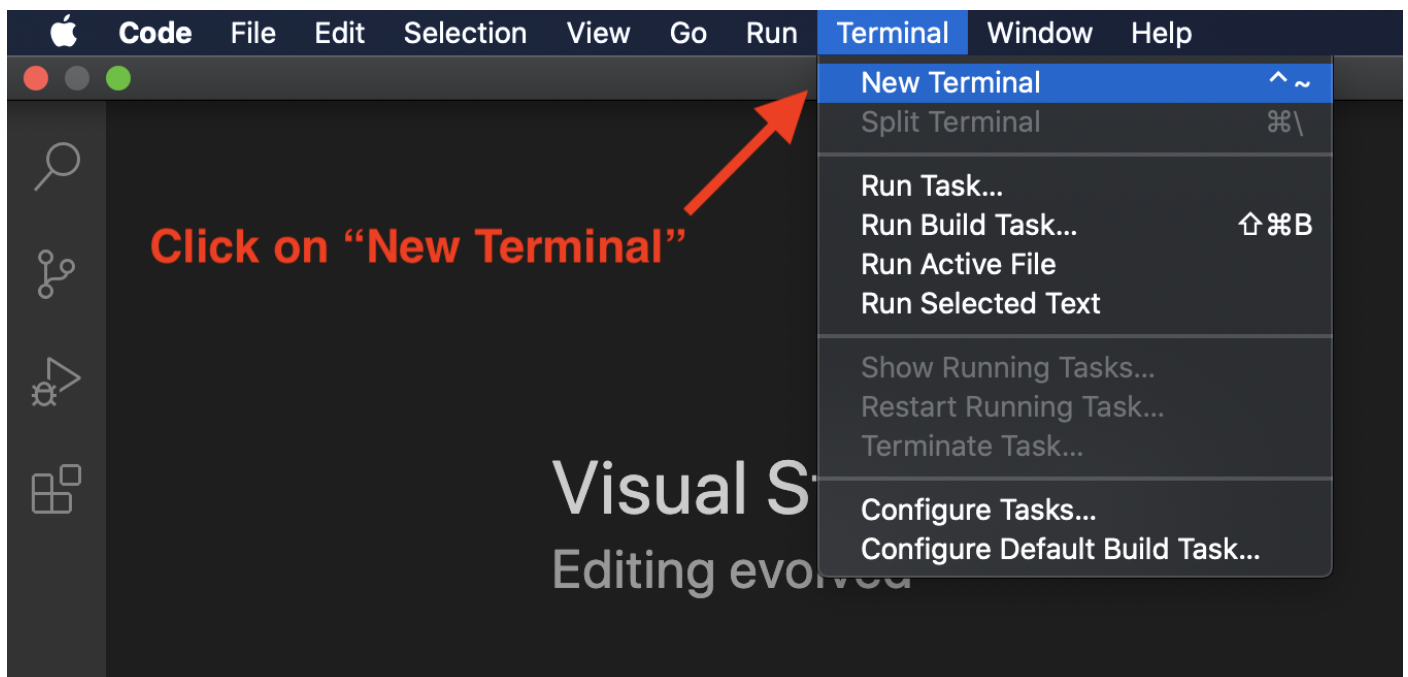








3. Check to make sure that Node and NPM are installed correctly by opening up VSCode terminal. You can do this by either clicking `Terminal > New Terminal` or pressing `Ctrl + ~`.



4. Then, run both `node -v` and `npm -v` to see the version numbers and check that both commands are installed and working.



```
(base) MacBook-Pro-Jack:cse344-hw4-jvenberg jackv$ node -v
v12.13.0
(base) MacBook-Pro-Jack:cse344-hw4-jvenberg jackv$ npm -v
6.12.0
(base) MacBook-Pro-Jack:cse344-hw4-jvenberg jackv$
```

## 2) Installing Global Node Modules

### Preface/Explanation

Node modules, by default, are installed locally within a particular project, so they are only accessible and importable within a particular Node project directory. However, some modules are able to be installed globally which can allow them to create useful command-line commands. Here are a few global modules used in this course:

<code>http-server</code>	Allows you to quickly run a local server that hosts the files inside of the current directory you run the <code>http-server</code> command in.
<code>nodemon</code>	A wrapper command that acts like if you run <code>node</code> , but it detects any changes you have made to your files, and restarts the <code>node</code> process so your changes are immediately reflected without manually restarting.
<code>eslint</code>	Javascript linter commandline module that we use to run the local linters.
<code>stylelint</code>	CSS linter commandline module that we use to run the local linters.
<code>@linthtml/linthtml</code>	HTML linter commandline module that we use to run the local linters.