MLHacks: Build an FAQ Bot For Your Website with Microsoft Azure

Troubleshooting Guide

Welcome to the troubleshooting guide of MLHacks: Build an FAQ Bot for your website with Microsoft Azure! This guide will help you with troubleshooting any issues you may face while following the workshop materials.

Please choose your Operating System below to see the troubleshooting guide:
WINDOWS
OSX

This is the troubleshooting guide for Windows. If that is not your operating system, please go back to the first page to choose the right troubleshooting guide.

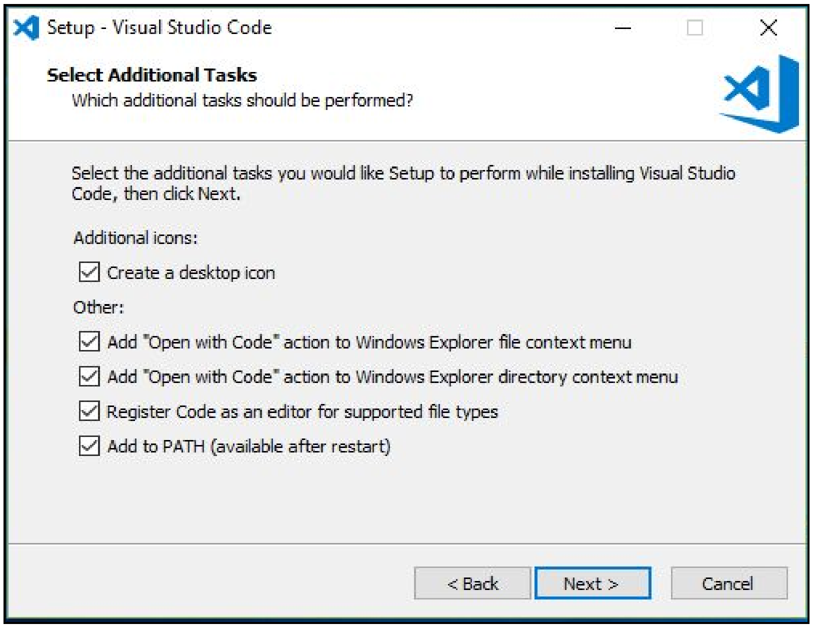
The following sections describe the possible problems that could occur or things to be aware of when following the workshop.

SECTION 1: CREATING AN AZURE ACCOUNT

1. It is possible that there are some errors while creating an Azure account. Most of these errors disappear when you **create the account in an incognito tab**.
2. During signup, you can verify your mobile number by providing an OTP. If you choose to receive the OTP via text message, note that it might take some time. Simply wait for a minute and if you do not receive an OTP by then, **click on the "I did not receive a code" button.**
3. **Do not use the OTP via call option**, as this simply redirects you to the beginning of the form.
4. If you have previously used Azure and have used your free trial credits, you will need to input your credit card information and sign up for a Pay-As-You-Go Account.
5. If you do not have a student email address, you you will need to input your credit card information and sign up for a Pay-As-You-Go Account.

SECTION 2: DOWNLOAD AND INSTALL VS CODE

1. If the site does not show the correct version for your OS, scroll down to the bottom of the site and manually select the version that matches your OS.
2. Restart your PC after installation to ensure the PATH variable is set (not essential, but may be useful).
3. During install, choose all the options in this dialog box to make it easy for you to work with VS code (not essential, but may be useful).



Setup - Visual Studio Code

Select Additional Tasks

Which additional tasks should be performed?

Select the additional tasks you would like Setup to perform while installing Visual Studio Code, then click Next.

Additional icons:

☑ Create a desktop icon

Other:

☑ Add "Open with Code" action to Windows Explorer file context menu

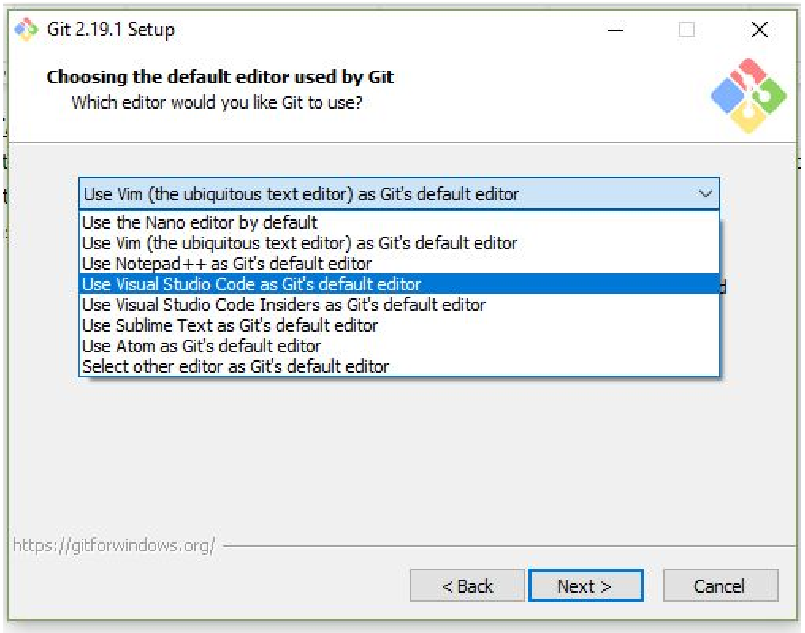
☑ Add "Open with Code" action to Windows Explorer directory context menu

☑ Register Code as an editor for supported file types

☑ Add to PATH (available after restart)

SECTION 3: INSTALLING GIT

1. Note that Git always installs the latest version for your OS automatically, so if you do not want that you need to manually select which version of git you want.
2. During install, you could choose VS code to be Git's default editor (not essential, but may be useful).



Git 2.19.1 Setup

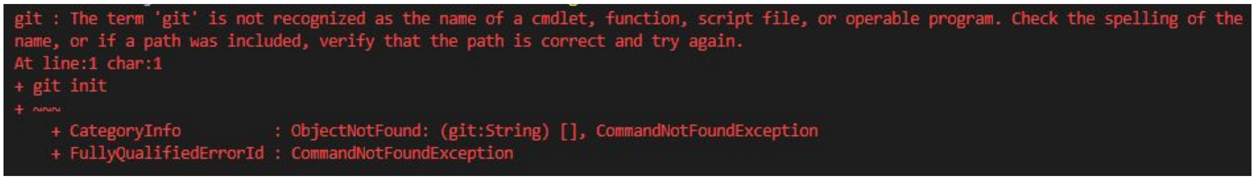
Choosing the default editor used by Git

Which editor would you like Git to use?

Use Vim (the ubiquitous text editor) as Git's default editor

- Use the Nano editor by default
- Use Vim (the ubiquitous text editor) as Git's default editor
- Use Notepad++ as Git's default editor
- Use Visual Studio Code as Git's default editor
- Use Visual Studio Code Insiders as Git's default editor
- Use Sublime Text as Git's default editor
- Use Atom as Git's default editor
- Select other editor as Git's default editor

3. After installing git, if your VS Code is open, **first close it**. Reopen VS code, create a new terminal and try to run the git --version. If that fails (shows "git is not recognized"), **kill the terminal**, create another terminal and try the command again.

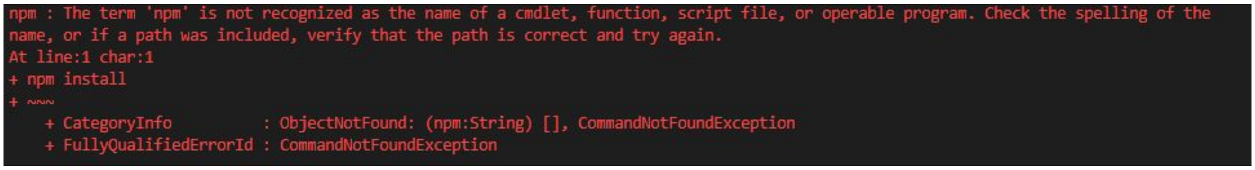


```
git : The term 'git' is not recognized as the name of a cmdlet, function, script file, or operable program. Check the spelling of the
name, or if a path was included, verify that the path is correct and try again.
At line:1 char:1
+ git init
+ ~~~
    + CategoryInfo          : ObjectNotFound: (git:String) [], CommandNotFoundException
    + FullyQualifiedErrorId : CommandNotFoundException
```

To kill the terminal, click on the icon that looks like this - 🗑 and to add a new terminal, click on the icon that looks like this - +

SECTION 4: INSTALLING NODE

1. After installing node, if your VS Code is open, **first close it**. Reopen VS code, create a new terminal and try to run the npm install (ensure that you are in the proper directory first). If that fails (shows "npm is not recognized"), **kill the terminal**, create another terminal and try the command again.



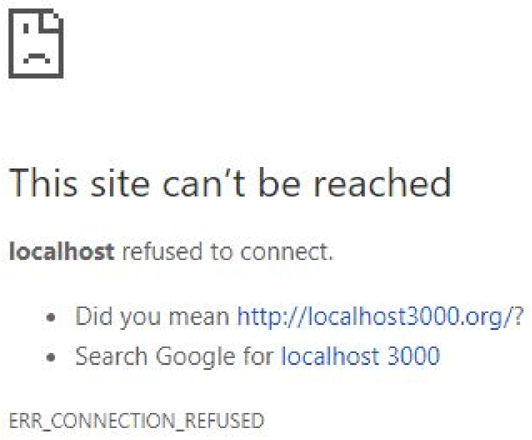
```
npm : The term 'npm' is not recognized as the name of a cmdlet, function, script file, or operable program. Check the spelling of the
name, or if a path was included, verify that the path is correct and try again.
At line:1 char:1
+ npm install
+ ~~~
    + CategoryInfo          : ObjectNotFound: (npm:String) [], CommandNotFoundException
    + FullyQualifiedErrorId : CommandNotFoundException
```

SECTION 5: ADDING CODE

1. Ensure you **delete the mlh-localhost-azure-bot-master part in the extract all dialog box**, or else there will be a nesting of folders.

SECTION 6: RUNNING THE CODE

1. When you open up a terminal, ensure you are in the correct path i.e. you should see something like \Downloads\mlh-localhost-azure-bot-master> and the cursor waiting there. If that doesn't happen, **kill the terminal and create a new one**. If that still doesn't help, **restart VS code**. Otherwise you might get an error like "ENOENT: no such file or directory, open '..\package.json'"
2. Wait till "**MLHacks app listening on port 3000!**" is shown on the console before trying to open localhost:3000 in your browser or it will say "This site can't be reached"



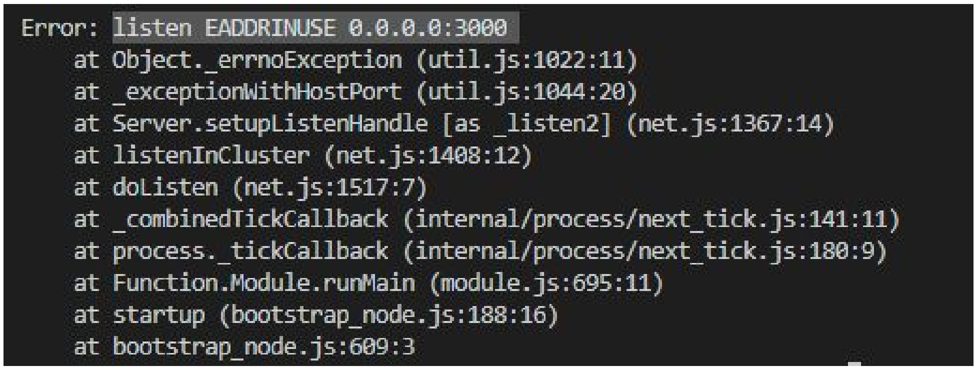
This site can't be reached

localhost refused to connect.

- Did you mean http://localhost3000.org/?
- Search Google for localhost 3000

ERR_CONNECTION_REFUSED

3. Type **clear** in your terminal to delete all the text shown, so that there is less clutter (not necessary, but recommended).
4. If you cancel the extraction midway (or if the extraction doesn't complete successfully for some reason) it will still create a folder but it might not have all the files so it may throw errors when you run npm start. If this happens to you, close VS code, **delete the extracted folder, extract again** and repeat all the steps from there. One common error is "failed to lookup..."
5. If you get the error "listen EADDRINUSE 0.0.0.0:3000", ensure there are no other terminals where the node index.js command is still running. If there are, **kill those terminals** and **re-run the command in a new terminal**.



```
Error: listen EADDRINUSE 0.0.0.0:3000
    at Object._errnoException (util.js:1022:11)
    at _exceptionWithHostPort (util.js:1044:20)
    at Server.setupListenHandle [as _listen2] (net.js:1367:14)
    at listenInCluster (net.js:1408:12)
    at doListen (net.js:1517:7)
    at _combinedTickCallback (internal/process/next_tick.js:141:11)
    at process._tickCallback (internal/process/next_tick.js:180:9)
    at Function.Module.runMain (module.js:695:11)
    at startup (bootstrap_node.js:188:16)
    at bootstrap_node.js:609:3
```

SECTION 7: TESTING LOCALLY

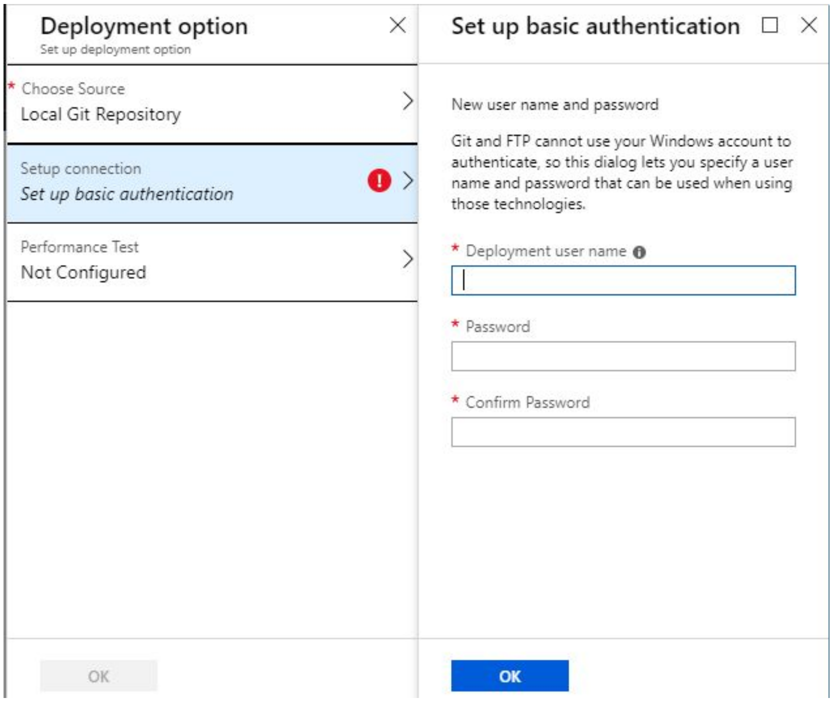
1. When you run the command “node app.js” if you get the error” EADDRINUSE 0.0.0.0:3000”**,** check if you have another terminal where you didn’t stop the node app.js command. To resolve the error simply **kill the terminal where node app.js command** is running.
2. When you type in a question on the browser and click submit if you see the error message “cannot read property replace of null” don’t worry. This is because we haven’t **set up our questions and answers** yet. Follow along from the create a KB slide to remove the error.

SECTION 8: CREATE A QNA BOT

1. When you click on “Create a QnA service” it may ask you to login to Azure. Once you have logged in, you need to click on the “Create a KB” button again and then continue the steps.
2. To hide the test menu, click on the Test button again.
3. If you get a “request failed with status 401” error message after submitting a question, one of the possible reasons is that the endpoint key variable is not initialized properly. **Edit line 10 of app.js with the correct value** and continue.

SECTION 9: DEPLOYING THE APP

1. In deployment option of the web app, you need to enter a username and a password. VS code will ask for this password every time you want to publish to this git repository so don’t forget it!



Deployment option

Set up deployment option

* Choose Source: Local Git Repository

Setup connection: *Set up basic authentication*

Performance Test: Not Configured

Set up basic authentication

New user name and password

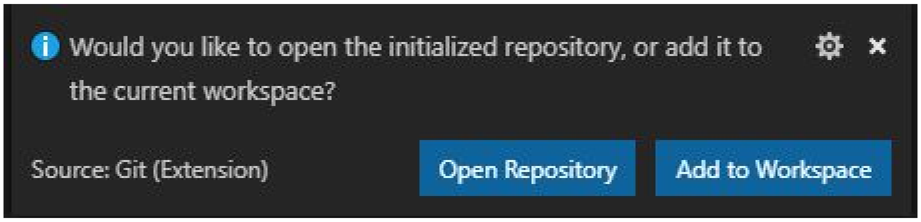
Git and FTP cannot use your Windows account to authenticate, so this dialog lets you specify a user name and password that can be used when using those technologies.

* Deployment user name

* Password

* Confirm Password

2. Give your app a name. This name must be unique across Azure. Wait for the green check mark. If you get a red exclamation mark, you must rename your app.
3. After you initialize the git repository in VS code, the following dialog box appears. Click on the add to workspace button and continue with the other steps.



Would you like to open the initialized repository, or add it to the current workspace?

Source: Git (Extension)

This is the troubleshooting guide for OSX. If that is not your operating system, please go back to the first page to choose the right troubleshooting guide.

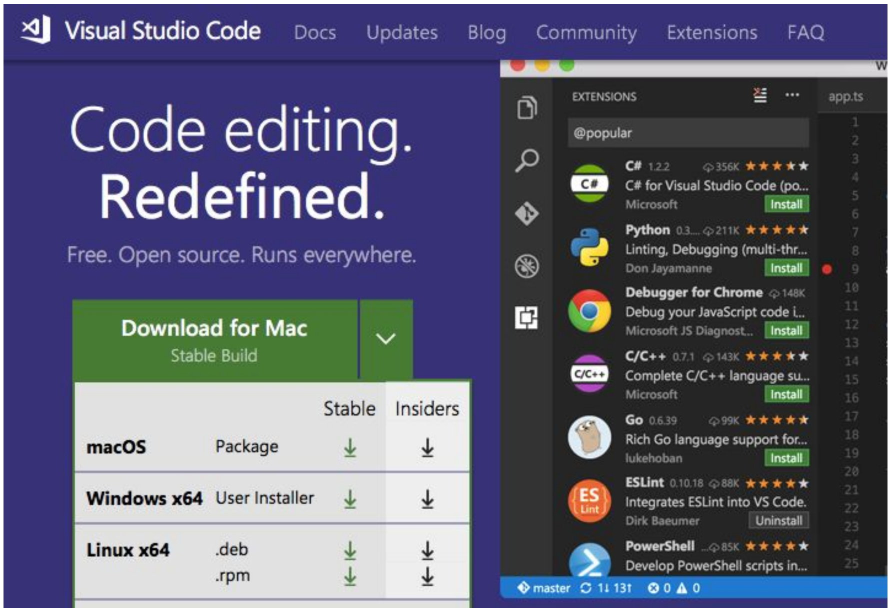
The following sections describe the possible problems that could occur or things to be aware of when following the workshop.

SECTION 1: CREATING AN AZURE ACCOUNT

1. It is possible that there are some errors while creating an Azure account. Most of these errors disappear when you **create the account in an incognito tab**.
2. During signup, you can verify your mobile number by providing an OTP. If you choose to receive the OTP via text message, note that it might take some time. Simply wait for a minute and if you do not receive an OTP by then, **click on the "I did not receive a code" button.**
3. **Do not use the OTP via call option**, as this simply redirects you to the beginning of the form.
4. If you have previously used Azure and have used your free trial credits, you will need to input your credit card information and sign up for a Pay-As-You-Go Account.
5. If you do not have a student email address, you you will need to input your credit card information and sign up for a Pay-As-You-Go Account.

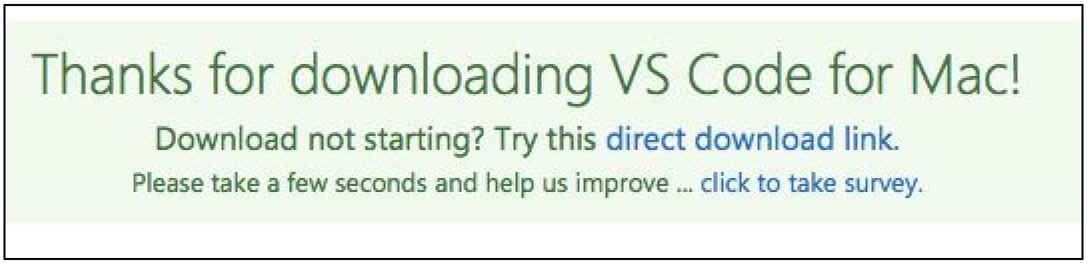
SECTION 2: DOWNLOAD AND INSTALL VS CODE

1. Select **Mac** as your operating system on Visual Studio Code's website and the Stable version as the package you want to download.



		Stable	Insiders
macOS	Package	↓	↓
Windows x64	User Installer	↓	↓
Linux x64	.deb .rpm	↓ ↓	↓ ↓

2. If the Visual Studio Code package does not start automatically downloading, you can select **Direct Download**.



Thanks for downloading VS Code for Mac!

Download not starting? Try this direct download link.

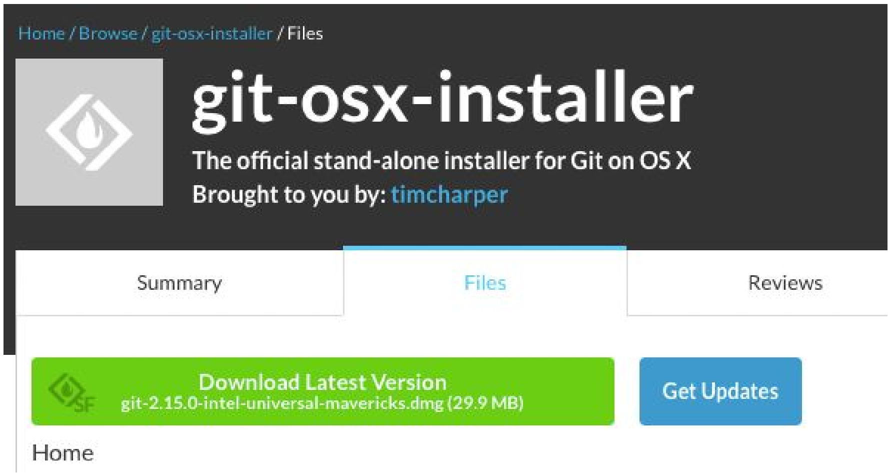
Please take a few seconds and help us improve ... click to take survey.

3. Restart your computer after installation to ensure the PATH variable is set (not essential, but may be useful).

SECTION 3: INSTALLING GIT

1. There are several ways to install Git on a Mac. The easiest way is to install the XCode Command Line tools. On Mavericks (10.9) or above you can run Git from the terminal. Open your terminal and type `$ git --version` in and then press Enter.

   If this does not work, you can go to: https://git-scm.com/download/mac to download Git via a binary installer. This will also make sure you have the most up to date version.



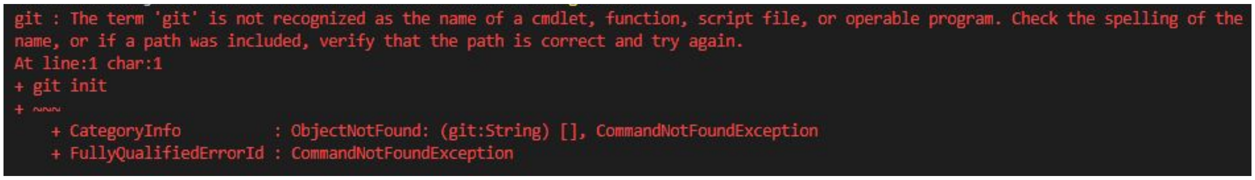
Home / Browse / git-osx-installer / Files

git-osx-installer

The official stand-alone installer for Git on OS X

Brought to you by: timcharper

2. Still not working? It might be that your computer's operating system is not compatible with the latest Git version because your operating system is out of date. Go to your **System Preferences,** then Select **Software Updates.** If there are any updates that you need to download, they will be here.
3. After installing git, if your VS Code is open, **first close it**. Reopen VS code, create a new terminal and try to run the git --version. If that fails (shows "git not recognized"), **kill the terminal** and create another one.

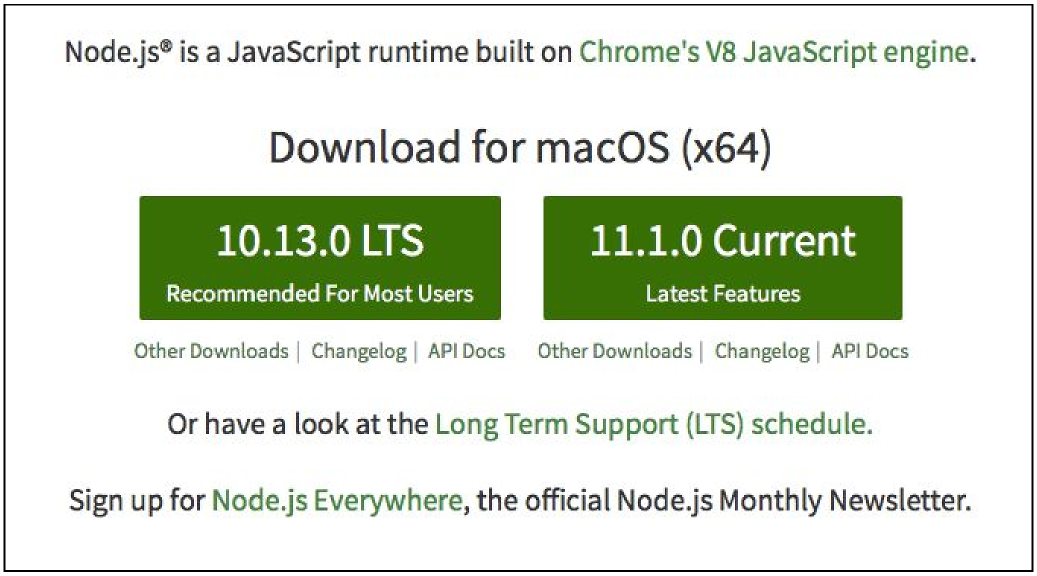


```
git : The term 'git' is not recognized as the name of a cmdlet, function, script file, or operable program. Check the spelling of the
name, or if a path was included, verify that the path is correct and try again.
At line:1 char:1
+ git init
+ ~~~
    + CategoryInfo          : ObjectNotFound: (git:String) [], CommandNotFoundException
    + FullyQualifiedErrorId : CommandNotFoundException
```

To kill the terminal, click on the icon that looks like this - 🗑 and to add a new terminal, click on the icon that looks like this - ✚

SECTION 4: INSTALLING NODE

1. Node demands two different packages before you can install it. Many developers already have these packages on their computer, but if Node is having a hard time running, it may be because these packages are not installed. You can go to NodeJS.org and download the Node installer for Mac.



Node.js® is a JavaScript runtime built on Chrome's V8 JavaScript engine.

Download for macOS (x64)

10.13.0 LTS
Recommended For Most Users
Other Downloads | Changelog | API Docs
11.1.0 Current
Latest Features
Other Downloads | Changelog | API Docs

Or have a look at the Long Term Support (LTS) schedule.

Sign up for Node.js Everywhere, the official Node.js Monthly Newsletter.

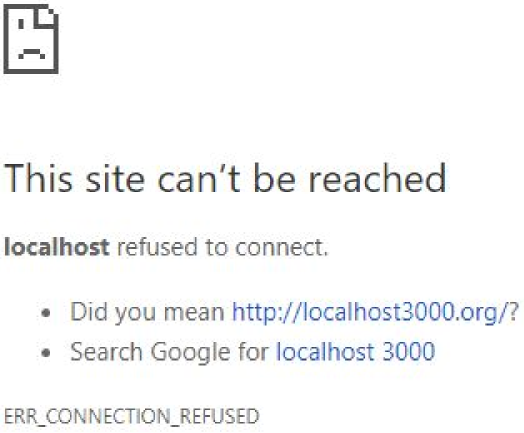
2. Or, if you are worried about any packages you download wreaking havoc on your computer, you can download Homebrew, a package manager for Mac. To install Homebrew, open your Terminal and type in:
ruby -e "$(curl -fsSL https://raw.githubusercontent.com/Homebrew/install/master/install)".
This is especially useful if you do not have administrator access in the system.

SECTION 6: RUNNING THE CODE

1. When you open up a terminal, ensure you are in the correct path i.e. you should see something like \Downloads\mlh-localhost-azure-bot-master> and the cursor waiting there. If that doesn't happen, **kill the terminal and create a new one**. If that still

doesn’t help, **restart VS code**. Otherwise you might get an error like “ENOENT: no such file or directory, open '..\package.json'”

2. Wait till “**MLHacks app listening on port 3000!**” is shown on the console before trying to open localhost:3000 in your browser or it will say “This site can’t be reached”



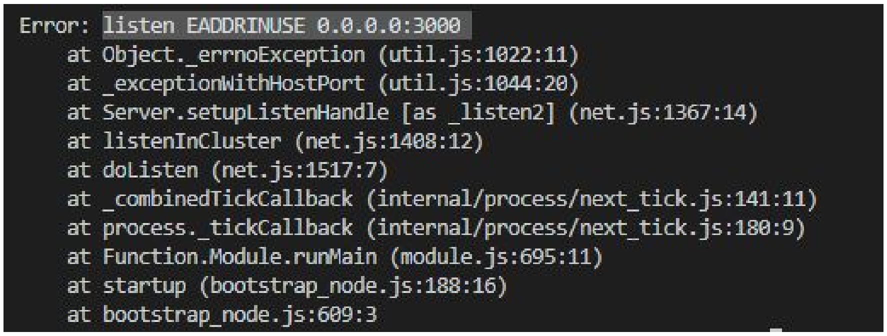
This site can’t be reached

localhost refused to connect.

- Did you mean http://localhost3000.org/?
- Search Google for localhost 3000

ERR_CONNECTION_REFUSED

3. Type **clear** in your terminal to delete all the text shown, so that there is less clutter (not necessary, but recommended).
4. If you get the error “listen EADDRINUSE 0.0.0.0:3000”, ensure there are no other terminals where the node index.js command is still running. If there are, **kill those terminals** and **re-run the command in a new terminal**.



```
Error: listen EADDRINUSE 0.0.0.0:3000
    at Object._errnoException (util.js:1022:11)
    at _exceptionWithHostPort (util.js:1044:20)
    at Server.setupListenHandle [as _listen2] (net.js:1367:14)
    at listenInCluster (net.js:1408:12)
    at doListen (net.js:1517:7)
    at _combinedTickCallback (internal/process/next_tick.js:141:11)
    at process._tickCallback (internal/process/next_tick.js:180:9)
    at Function.Module.runMain (module.js:695:11)
    at startup (bootstrap_node.js:188:16)
    at bootstrap_node.js:609:3
```

SECTION 7: TESTING LOCALLY

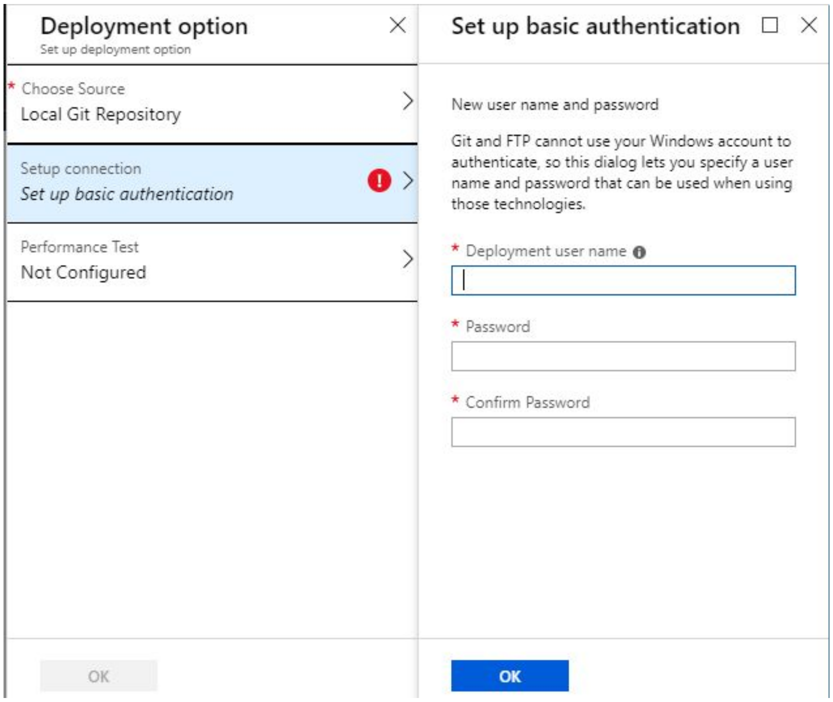
1. When you run the command “node app.js” if you get the error” EADDRINUSE 0.0.0.0:3000”**,** check if you have another terminal where you didn’t stop the node app.js command. To resolve the error simply **kill the terminal where node app.js command** is running.
2. When you type in a question on the browser and click submit if you see the error message “cannot read property replace of null” don’t worry. This is because we haven’t **set up our questions and answers** yet. Follow along from the create a KB slide to remove the error.

SECTION 8: CREATE A QNA BOT

1. When you click on “Create a QnA service” it may ask you to login to Azure. Once you have logged in, you need to click on the “Create a KB” button again and then continue the steps.
2. To hide the test menu, click on the Test button again.
3. If you get a “request failed with status 401” error message after submitting a question, one of the possible reasons is that the endpoint key variable is not initialized properly. **Edit line 10 of app.js with the correct value** and continue.

SECTION 9: DEPLOYING THE APP

1. In deployment option of the web app, you need to enter a username and a password. VS code will ask for this password every time you want to publish to this git repository so don’t forget it!



Deployment option

Set up deployment option

* Choose Source
Local Git Repository

Setup connection
Set up basic authentication

Performance Test
Not Configured

Set up basic authentication

New user name and password

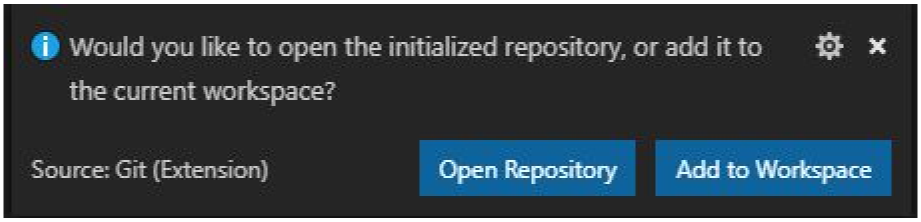
Git and FTP cannot use your Windows account to authenticate, so this dialog lets you specify a user name and password that can be used when using those technologies.

* Deployment user name

* Password

* Confirm Password

2. Give your app a name. This name must be unique across Azure. Wait for the green check mark. If you get a red exclamation mark, you must rename your app.
3. After you initialize the git repository in VS code, the following dialog box appears. Click on the add to workspace button and continue with the other steps.



Would you like to open the initialized repository, or add it to the current workspace?

Source: Git (Extension)