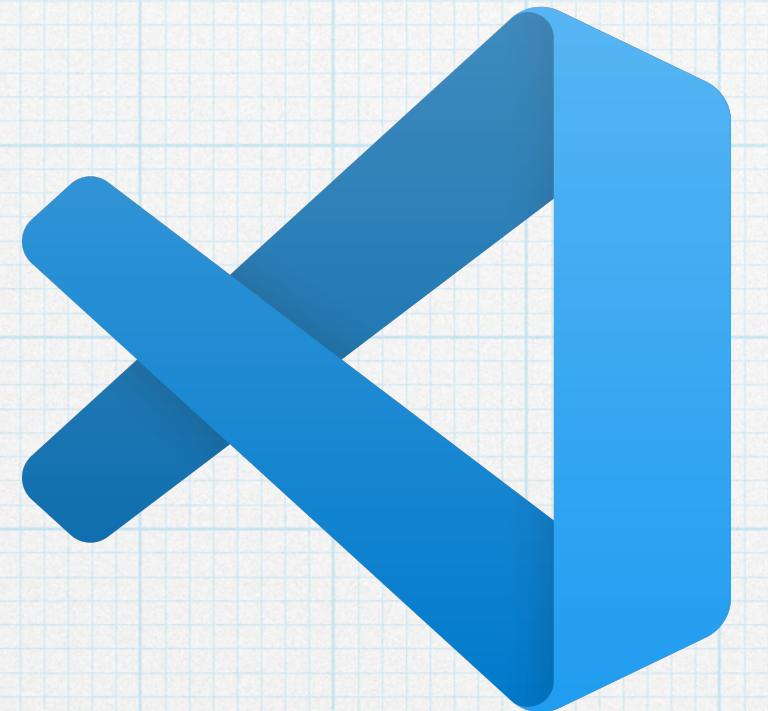
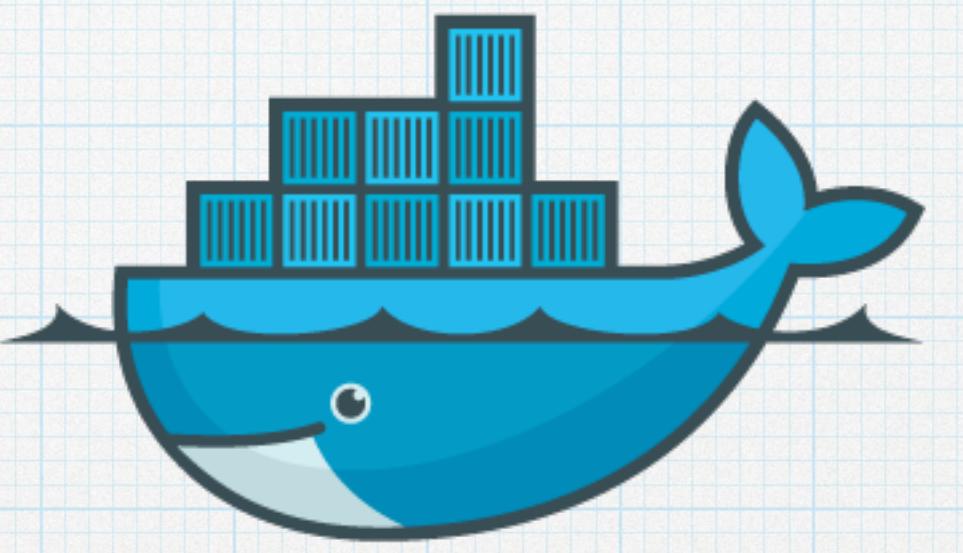


# Lab Software Prerequisites



Visual Studio Code



docker

## Instructor:

John Rofrano ([rofrano@cs.nyu.edu](mailto:rofrano@cs.nyu.edu))

Senior Technical Staff Member @ IBM T.J. Watson Research

@JohnRofrano 

# Why Prerequisites?

<https://github.com/nyu-devops/lab-starter>

- \* Because of the size of the installable downloads required for the hands-on labs, it is strongly recommended that you perform these steps at home or your dorm on a fast network before attending the session:
  - \* Download and Install Docker Desktop and Visual Studio Code with Remote Containers extension, and git
  - \* Clone the git repo that has the lesson material
  - \* Bring up the development environment to make sure that everything is working

# Some Assembly Required

\* Tools you will need to complete the labs in class:

- Computer running macOS, Linux, or Windows
- Internet Access to download software
- GitHub Account
- Install prerequisite software





# Computer running macOS, Linux, or Windows

- \* While any modern computer should do, here are some specs:
- \* Memory: 4 GB min (8 GB recommended)
- \* Intel CPU: Core i5 Dual Core (minimum)
- \* Apple M1 Silicon (perfect)
- \* Disk Space: 20 GB free space
- \* The more, the better. A Core i3 may work but it will be slow





# Internet Access

- \* A lot of our labs download packages from the internet
- \* Any broadband connection should be fine
- \* While DSL might work, it will be sloooooowwww...
- \* If your internet can handle Zoom you should be good to go



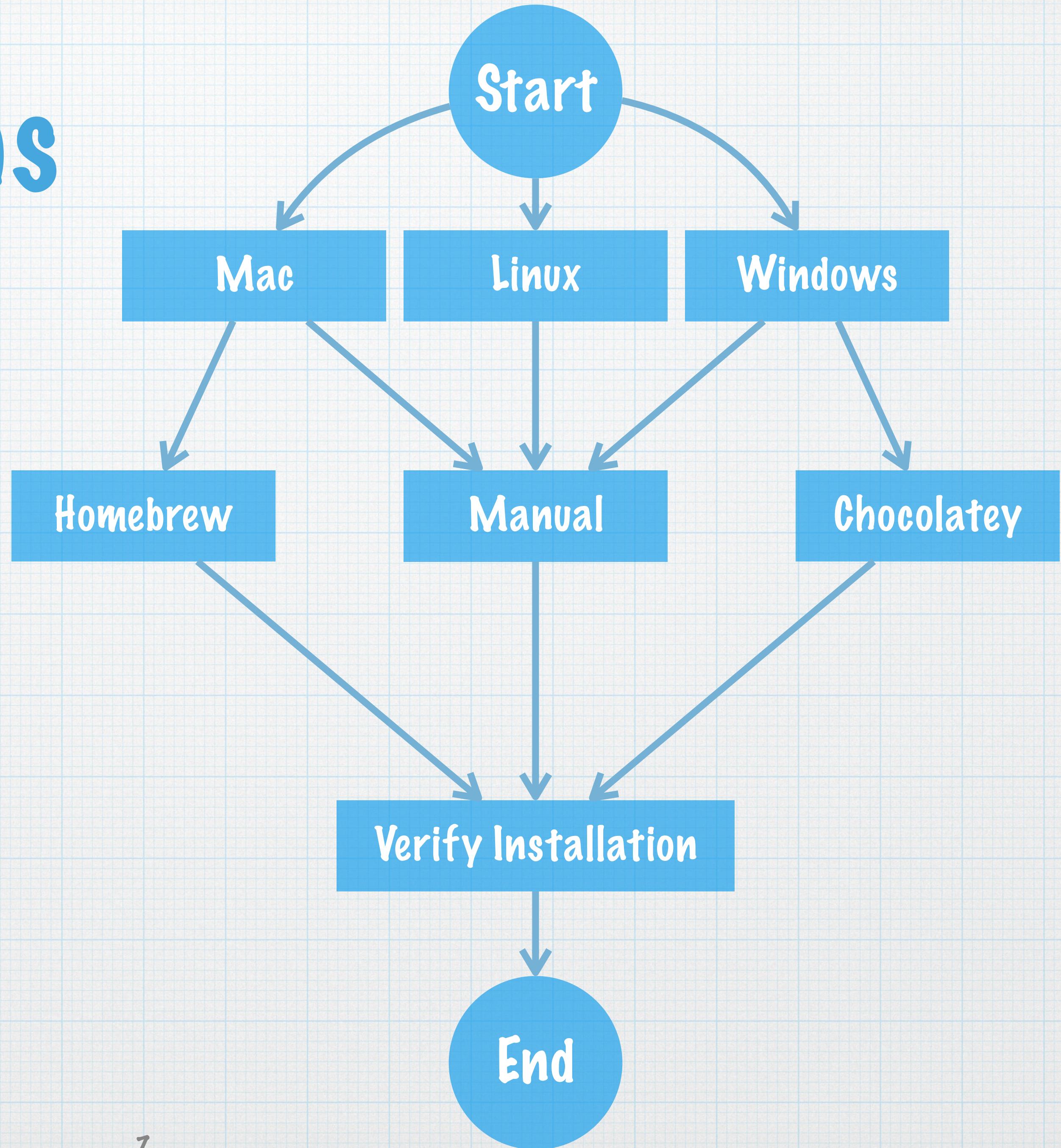
# GitHub Account

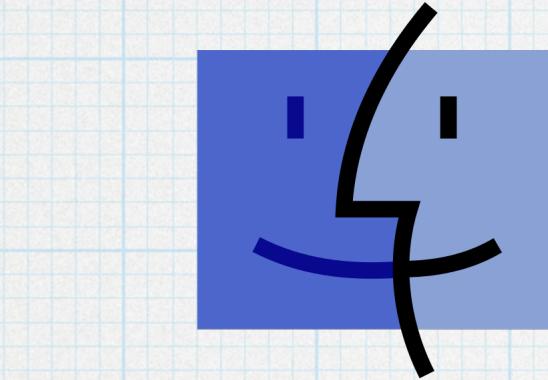
- \* If you don't have a GitHub Account, now is the time to sign up for one
- \* Goto <http://www.github.com> and create a free account
- \* You don't have to use your edu.nyu email. Just remember which email you used.
- \* This is the email you will need to give to your classmates if they have to invite you to their GitHub Organization



# Installation Paths

- \* If you have a Mac, the Homebrew path is recommended (one step)
- \* If you have a Windows, the Chocolatey path is recommended (one step)
- \* Otherwise choose the manual path





Mac OS

# macOS Users

If you are using a Mac you can use Homebrew\*



**Install Homebrew:** From (<https://brew.sh>)

```
/bin/bash -c "$(curl -fsSL https://raw.githubusercontent.com/Homebrew/install/HEAD/install.sh)"
```

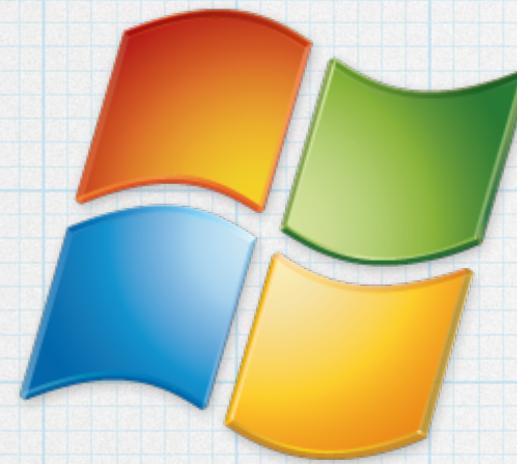
**Install Git, Docker Desktop, and Visual Studio Code using Homebrew:**

```
brew install git  
brew install --cask docker  
brew install --cask visual-studio-code
```

Note: while macOS already has git, brew will install the latest version

**Go to Slide 15** (you're done installing 😎)

\* You must open a terminal (shell) to execute these commands



# Windows Users

If you are using a Windows you can use Chocolatey\*



## Install Chocolatey:

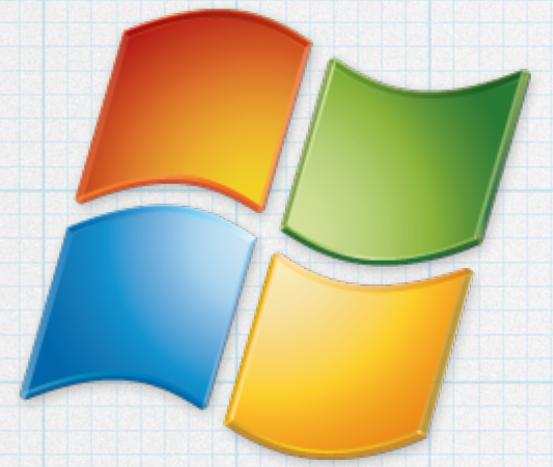
Instructions are at: <https://chocolatey.org/install>

## Install Git, Docker Desktop, and Visual Studio Code using Chocolatey:

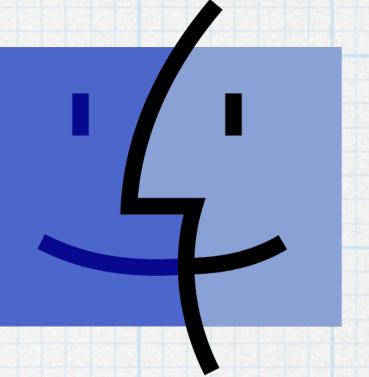
```
choco install git  
choco install docker-desktop  
choco install vscode  
code --install-extension ms-vscode-remote.remote-containers
```

Goto Slide 17 (you're done installing 😎)

\* You must open a command shell to execute these commands



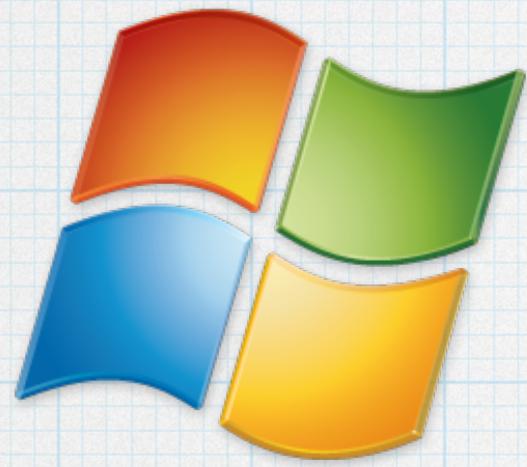
# Manual Installations



Mac OS

- \* If you are on Windows or Mac and you don't want to install homebrew (Mac) or chocolatey (Windows), then these instructions are for you
- \* If you are on Linux, go to Slide 13 and follow instructions to download Docker and Visual Studio Code.
- \* Download and install each software package on your computer manually



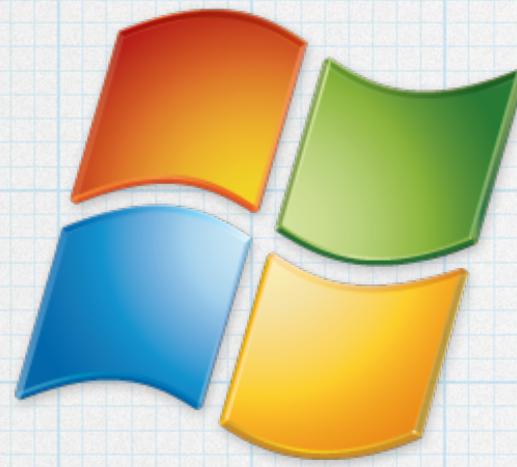


# Download Git

<https://git-scm.com/downloads>

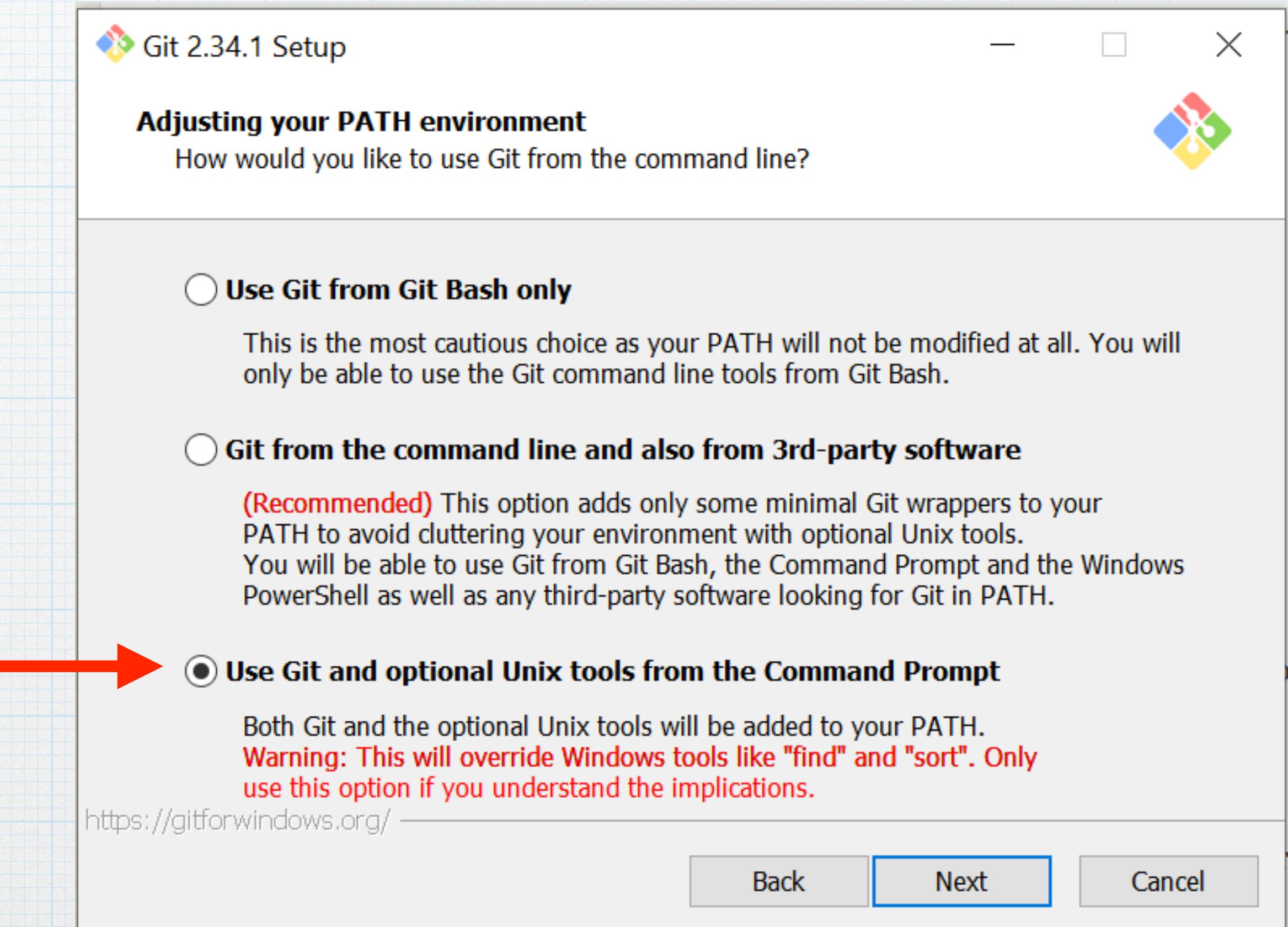
Note: macOS comes with Git so this is optional for Mac although it will give you the latest version

The screenshot shows the official Git website's download page. At the top left is the Git logo with the tagline "local-branching-on-the-cheap". On the right is a search bar. Below the header, there are navigation links for "About", "Documentation", "Downloads" (which is highlighted in red), and "Community".  
  
The main content area features a large "Downloads" heading. Below it, there are three download buttons: "macOS" (with an Apple icon), "Windows" (with a Windows icon), and "Linux/Unix" (with a Linux icon). A red arrow points from a large orange callout box containing the text "Select the appropriate download for your OS" towards the "macOS" button.  
  
To the right of the download buttons, there is a section for "Older releases" and a link to the "Git source repository" on GitHub. Further down, there are sections for "GUI Clients" (with a brief description and a "View GUI Clients →" link), "Logos" (with a description and a "View Logos →" link), and "Git via Git" (with a command-line link and a note about using Git itself to get the latest version).  
  
At the bottom, there is a note about browsing the git repository using the "web interface".



# Windows Install GIT

- \* Windows users should select the 3rd option:
- \* Use Git and optional Unix tools from the Windows Command Prompt



# Download and Install Docker Desktop

<https://www.docker.com/products/docker-desktop>

The screenshot shows the Docker Desktop landing page. At the top, there is a navigation bar with links for Products, Developers, Pricing, Blog, About us, Partners, a search icon, Sign In, and a Get Started button. The main heading is "Docker Desktop" with the subtext "Install Docker Desktop – the fastest way to containerize applications." Below this, there are two blue buttons: "Mac with Intel Chip" (labeled "MOST COMMON") and "Mac with Apple Chip". A red callout box with the text "Select the appropriate download for your OS" has an arrow pointing to the "Mac with Apple Chip" button.

**Docker Desktop**

Install Docker Desktop – the fastest way to containerize applications.

Mac with Intel Chip  
MOST COMMON

Mac with Apple Chip

Also available for [Windows](#) and [Linux](#)

Select the appropriate download for your OS

The Docker Subscription Service Agreement has been updated.

# Download and Install Visual Studio Code

<https://code.visualstudio.com>

The screenshot shows the official Visual Studio Code website and a running instance of the VS Code editor.

**Website Header:**

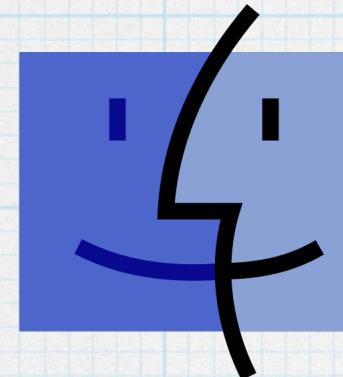
- Visual Studio Code logo
- Docs, Updates, Blog, API, Extensions, FAQ links
- Search Docs input field
- Download button

**Website Content:**

- A large headline: "Code editing. Redefined."
- A sub-headline: "Free. Built on open source. Runs everywhere."
- Download buttons for "Download for Mac" (Stable Build) and "Other platforms and Insiders Edition".
- A note about terms of service: "By using VS Code, you agree to its license and privacy statement."

**VS Code Editor:**

- Shows a code editor with multiple tabs: "blog-post.js — gatsby-graphql-app", "index.js", and "utils.js".
- The "blog-post.js" tab contains JavaScript code for a blog post component.
- The "utils.js" tab shows a snippet of code involving "data" and "dataFormat".
- The "PROBLEMS" and "TERMINAL" tabs are visible at the bottom.
- The status bar at the bottom shows: "master", "0 0 1 1", "0 0 0 0", "Gatsby Develop (gatsby-graphql-app)", "Ln 6, Col 21", "Spaces: 2", "UTF-8", "LF", "JavaScript", and a few icons.

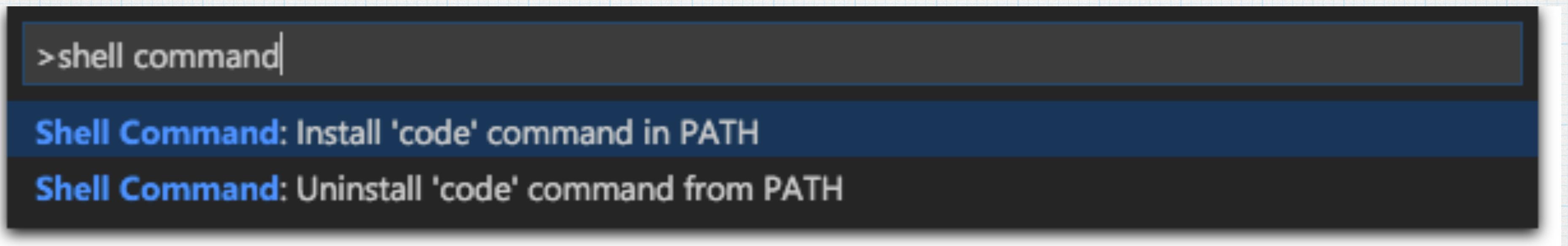


# Install the Code Command for Mac

## Mac OS

<https://code.visualstudio.com/docs/setup/mac>

- \* Launch VS Code:
- \* Open the Command Palette ( $\text{↑ } \text{⌘P}$ ) and type 'shell command' to find the Shell Command: Install 'code' command in PATH command.



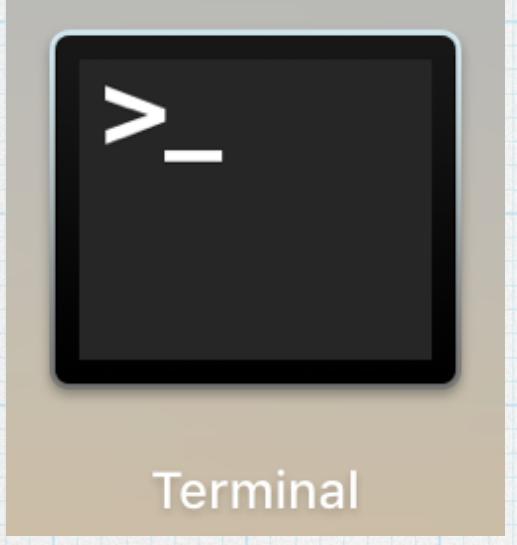
- \* Restart the terminal for the new \$PATH value to take effect. You'll be able to type 'code .' in any folder to start editing files in that folder.

# Visual Studio Code Extensions

- \* Open a Terminal (mac) or Command Prompt (win)
- \* Copy this text and paste it into a terminal window

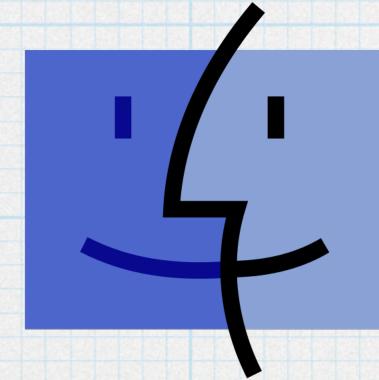
```
code --install-extension ms-vscode-remote.remote-containers
```

- \* It will install the remote container extension



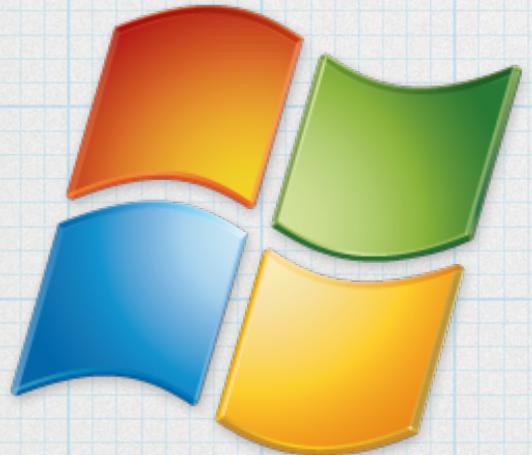
Terminal

# Validation: Open a Bash Shell



Mac OS

- \* If you are on macOS open a shell with the terminal app
  - \* Go to: Dock -> Launch Pad -> Other -> Terminal
- 
- \* If you are on Windows open the Bash Shell app that you just downloaded in the previous step
  - \* Go to: Start Programs -> Git -> Git Bash



# Clone the Repo into your DevOps folder

- \* It's a good idea to place all of your labs in a DevOps folder

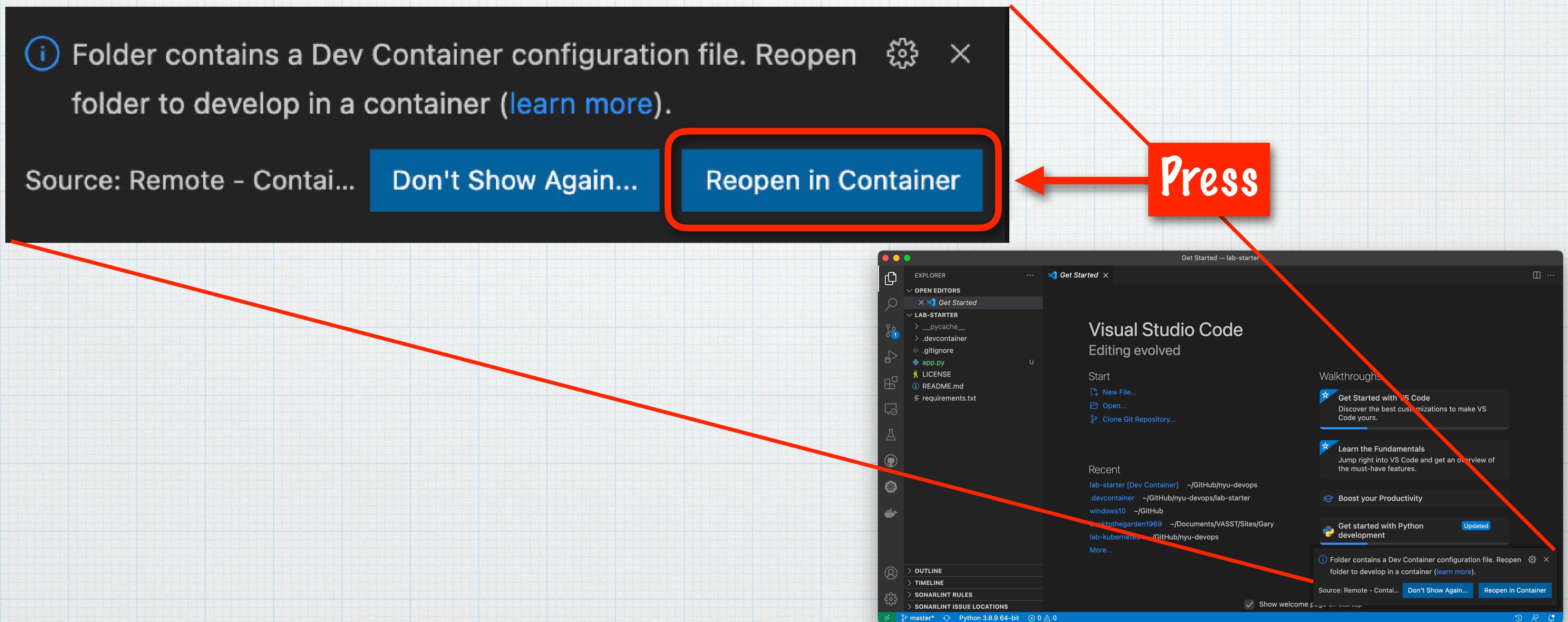
```
mkdir DevOps  
cd DevOps
```

- \* Clone the project repository

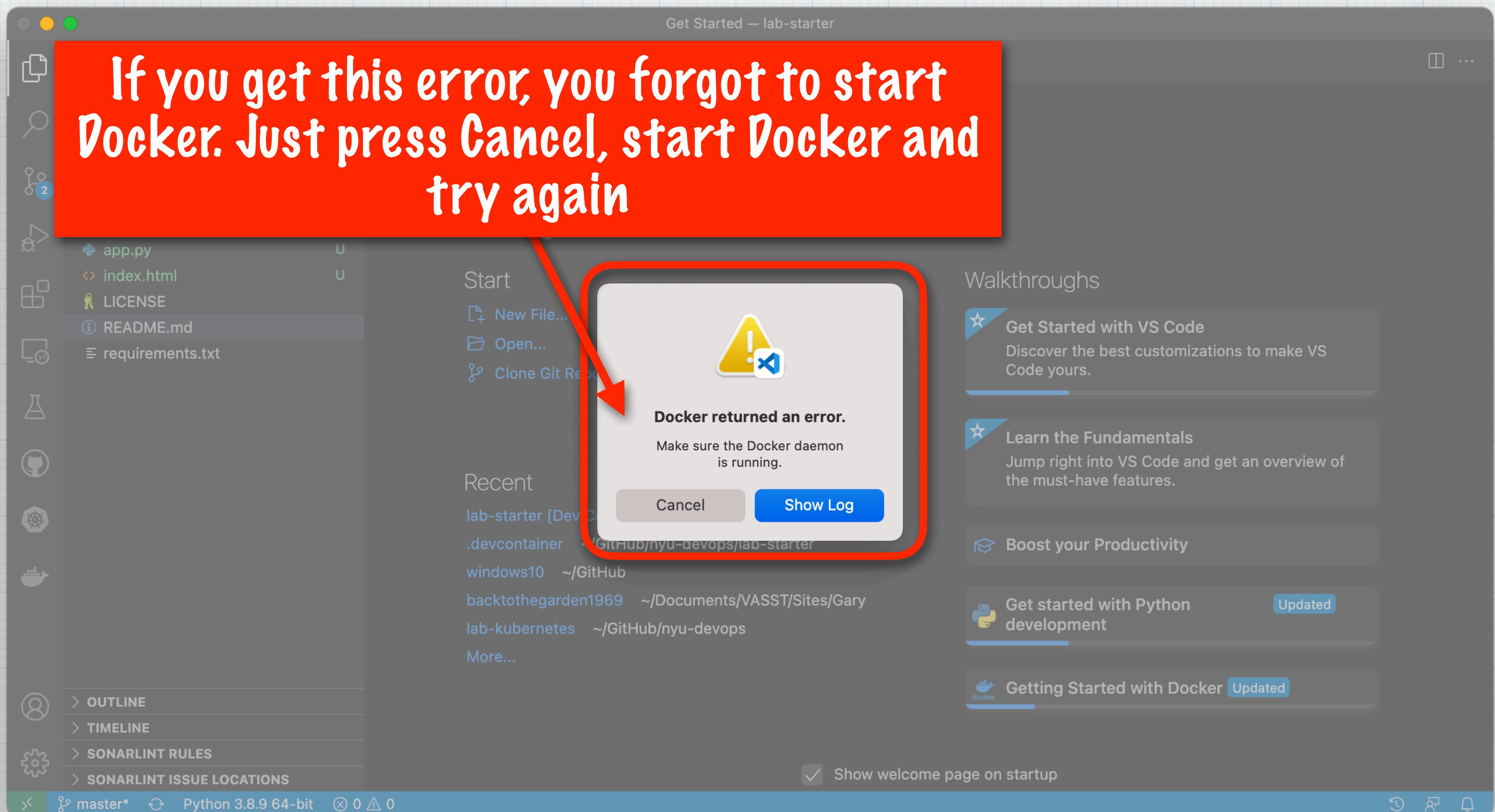
```
git clone https://github.com/nyu-devops/lab-starter  
cd lab-starter  
code .
```

# Reopen in Container

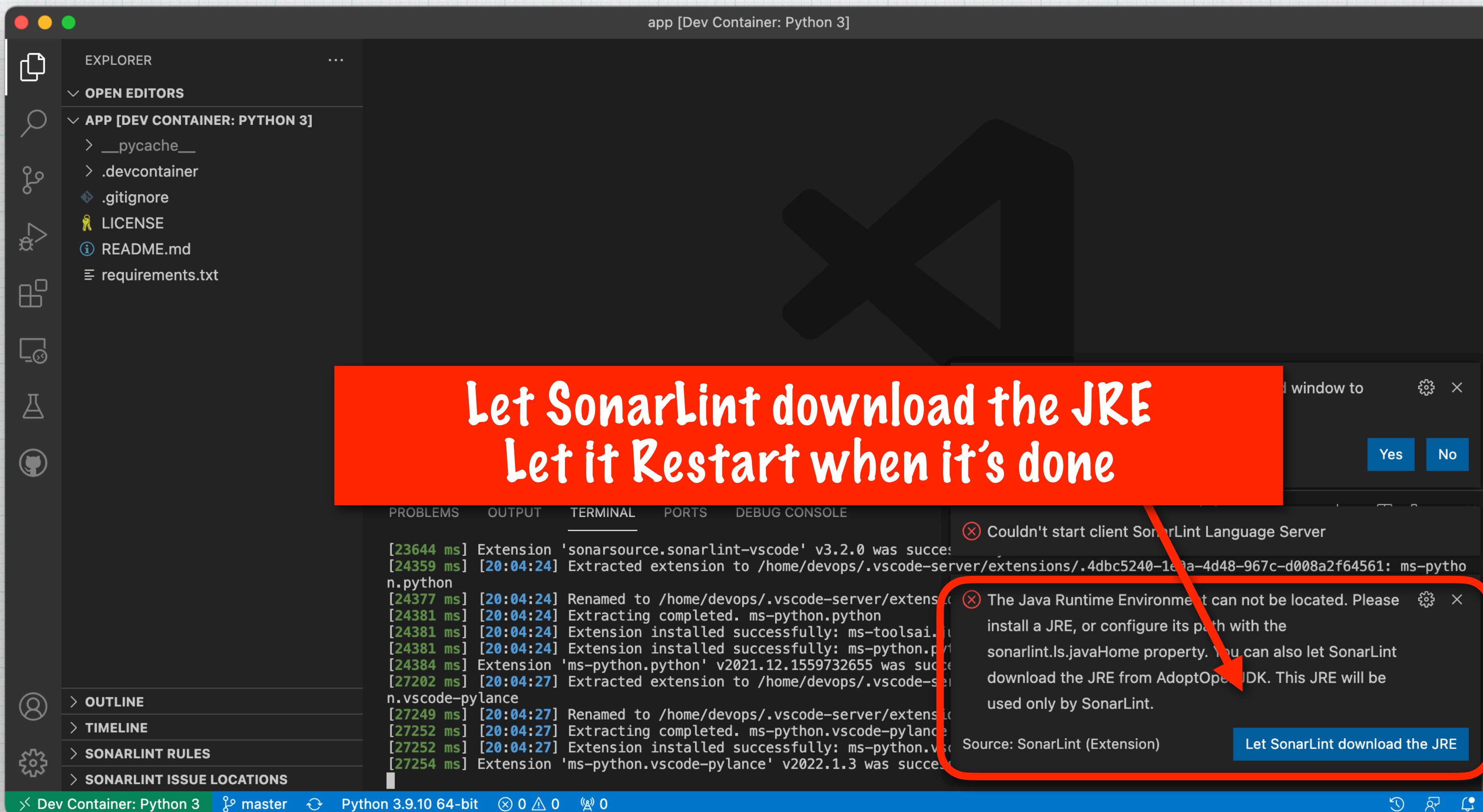
- \* When prompted to Reopen in Container press the button



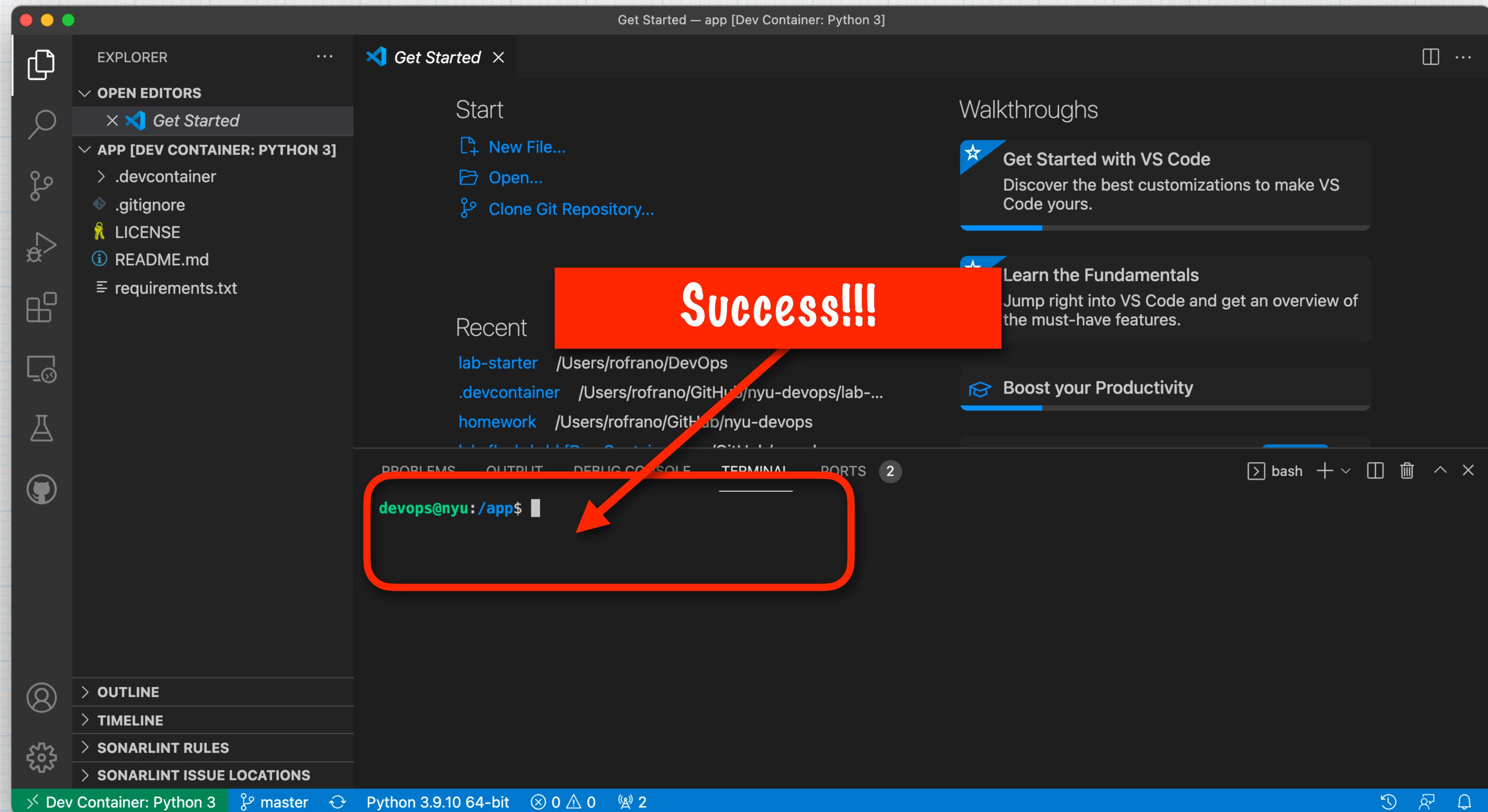
# Docker not Started Error



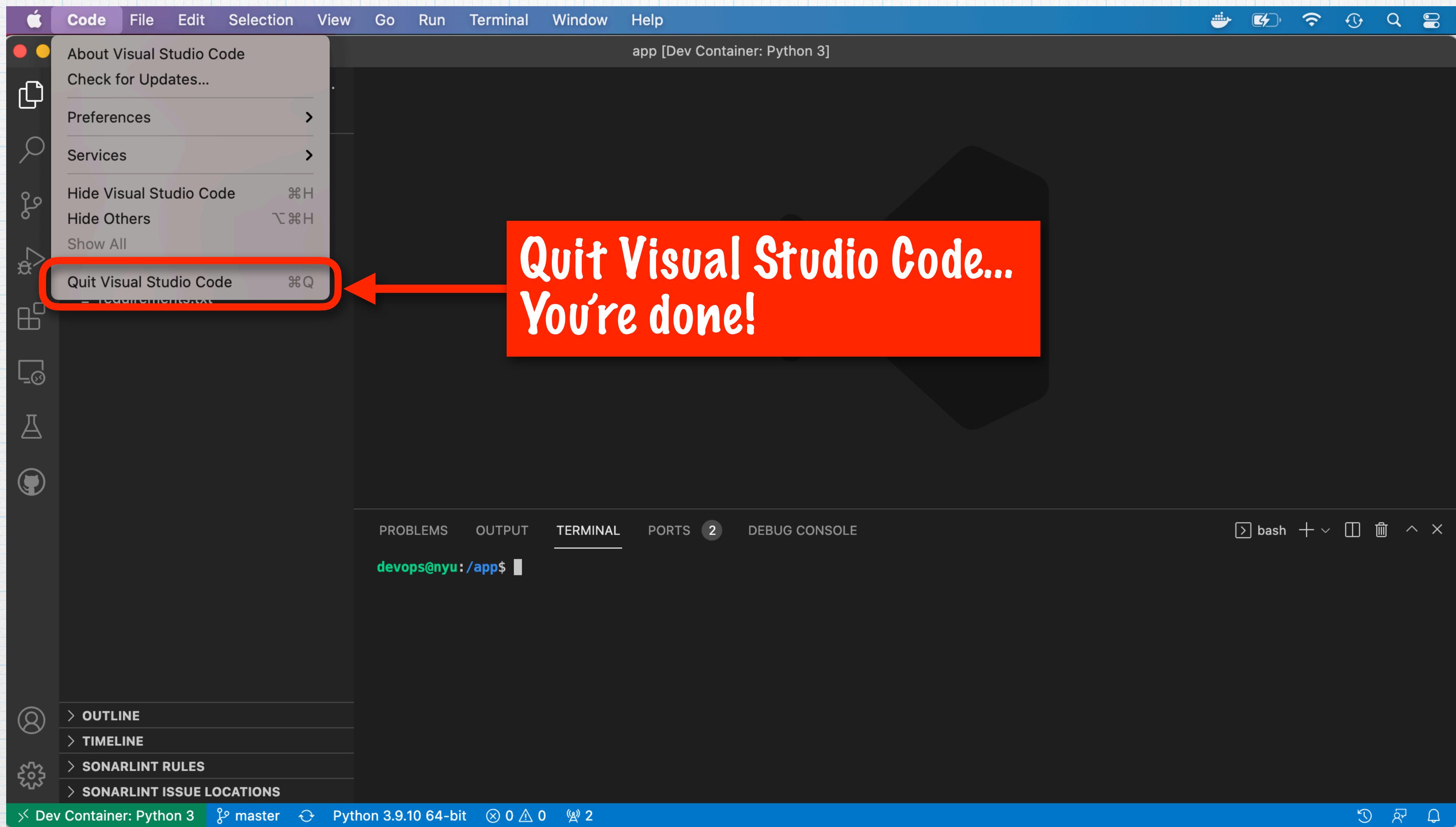
# SonarLint JRE



# Prompt: devops@nyu:/app



# Quit Visual Studio Code



# What did remote containers do?

- \* When you pressed Reopen in Container, the remote containers extension:
  - \* Downloaded a Debian + Python 3.9 Docker image
  - \* Build a new Docker image with all of your development tools
  - \* Created a container from the image with a development user
  - \* Downloaded and installed all of the packages needed for the lab
  - \* Shared the /app folder with your current directory
  - \* Placed you in a bash shell in the folder

# Visual Studio Code Extensions

- \* The development environment contains some of my favorite extensions
- \* You will have the exact same environment that I have
- \* So will everyone on the team ;-)

```
"VisualStudioExptTeam.vscodeintellicode",
"alexkrechik.cucumberautocomplete",
"bierner.github-markdown-preview",
"cstrap.flask-snippets",
"DavidAnson.vscode-markdownlint",
"donjayamanne.githistory",
"GitHub.vscode-pull-request-github",
"hbenl.vscode-test-explorer",
"hnw.vscode-auto-open-markdown-preview",
"ms-python.python",
"ms-python.vscode-pylance",
"njpwerner.autodocstring",
"redhat.vscode-yaml",
"SonarSource.sonarlint-vscode",
"streetsidesoftware.code-spell-checker",
"wholroyd.jinja",
"yzhang.markdown-all-in-one"
```

# Questions?

#troubleshooting

- \* Post any questions in the #troubleshooting channel of our NYU DevOps Spring 2022 Slack team

That's It !!!

