

4ID3 - IoT Devices and Networks

Lab 0

Pre-Lab Setup

Adam Sokacz, Ishwar Singh, and Salman Bawa

Sponsored by *Future Skills Center, Canada* and *McMaster W Booth SEPT*

Group Number:

Name, Student Number:

Name, Student Number:

Name, Student Number:

Submission Date:



Objective

To install the software that will be used in upcoming labs.

Contents

Objective	2
Feedback	3
Pre-Lab Questions	4
Post-Lab Questions	4
Setting Up a Software Portfolio	5
Installing the Arduino IDE	16
Installing Arduino Sensor Libraries	19
Installing NodeJS	21
Installing NodeRED	24
Installing NodeRED Dashboard	28
Mosquitto MQTT Broker Installation	31

Lab 0 - IoT Pre-Lab Setup

Feedback

Q1 - What would you rate the difficulty of this lab?

(1 = easy, 5 = difficult)

1

2

3

4

5

Comments about the difficulty of the lab:

Q2 - Did you have enough time to complete the lab within the designated lab time?

YES

NO

Q3 - How easy were the lab instructions to understand?

(1 = easy, 5 = unclear)

1

2

3

4

5

List any unclear steps:

Q4 - Could you see yourself using the skills learned in this lab to tackle future engineering challenges?

(1 = no, 5 = yes)

1

2

3

4

5

Pre-Lab Questions

Post-Lab Questions

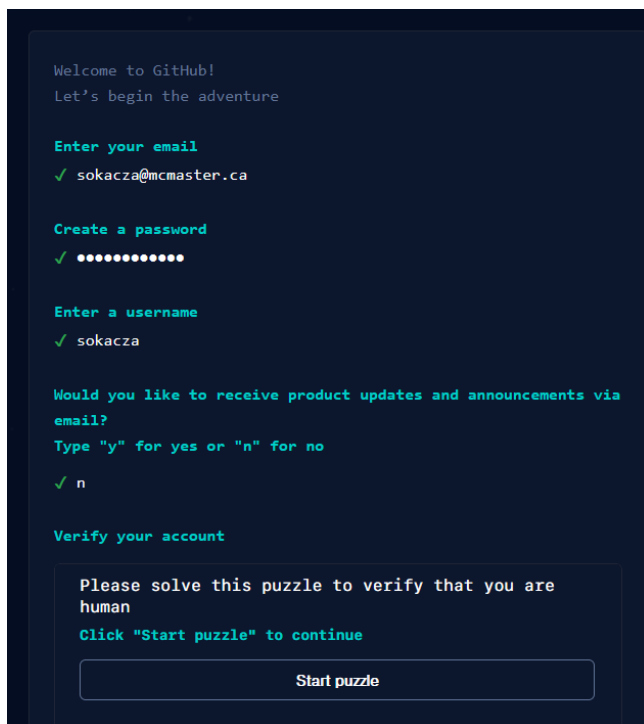
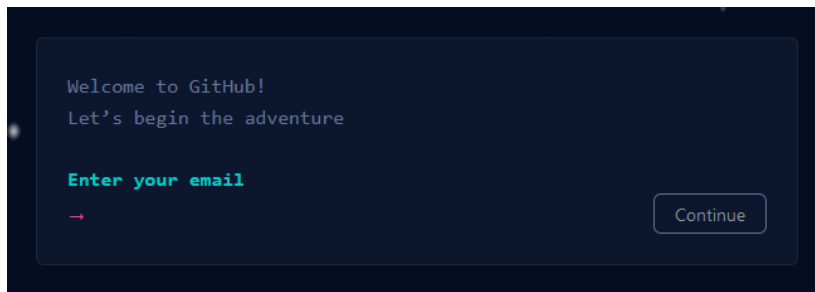
Setting Up a Software Portfolio

To gain a competitive advantage over other applicants when pursuing software positions, it is important to be able to demonstrate and articulate your past experiences, projects, and skills. During this class and other classes, it is highly recommended that you collect all of your coding assignments and projects into a GitHub that you can showcase on your resume and LinkedIn profile. This section will guide you step-by-step to setting up a GitHub account, initializing a repository, cloning an existing repository, and pushing your local changes to the remote.

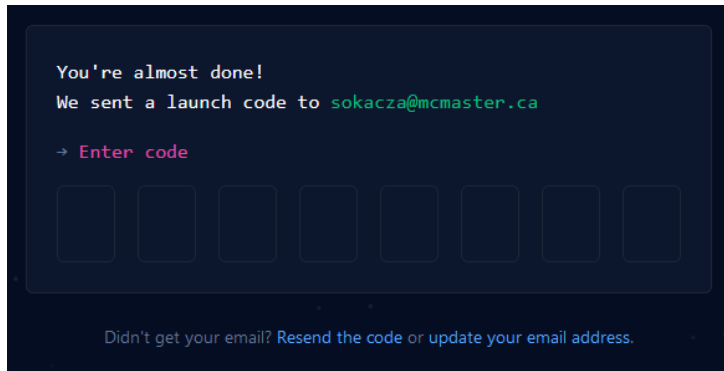
Navigate to the following website:

<https://github.com/signup?source=login>

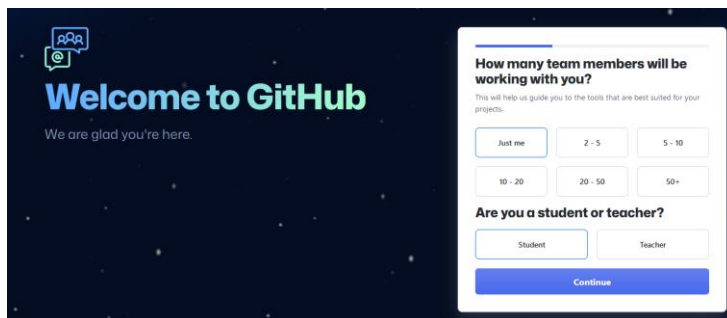
Follow the step-by-step instructions to creating an account using your McMaster email address.



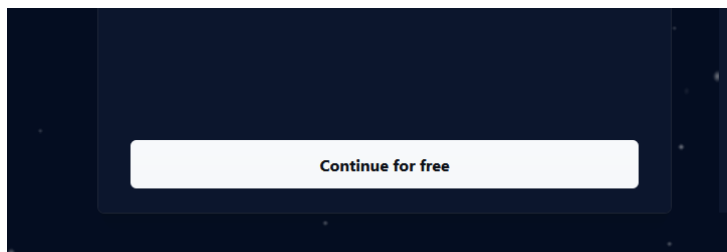
Lab 0 - IoT Pre-Lab Setup



Press **continue**.

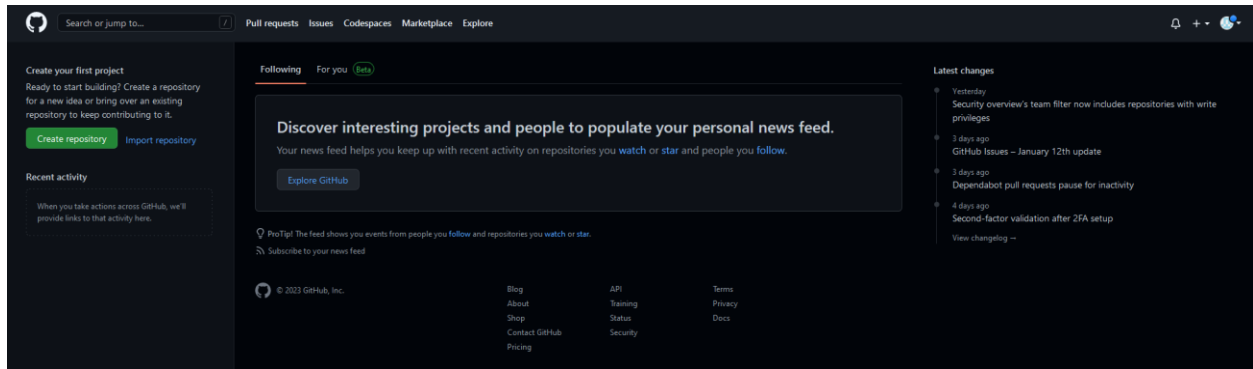


Press **continue for free**.



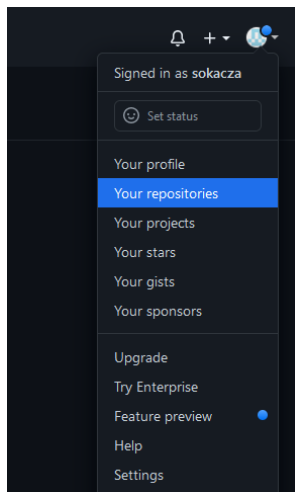
You should be shown to the account homepage.

Lab 0 - IoT Pre-Lab Setup

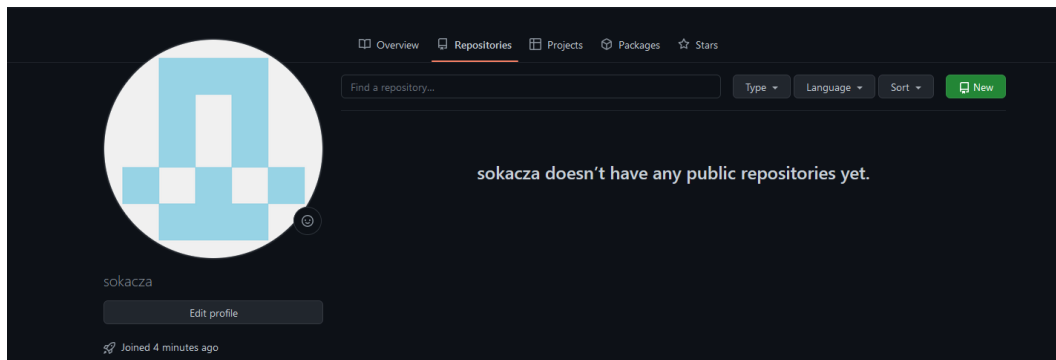


GitHub is **free with restrictions**, see their documentation for more information.

Next, we will be creating a remote repository. A remote repository is a software project that is stored in GitHub's servers. To do this, click on your account icon in the top right corner of the screen and select **Your repositories**.

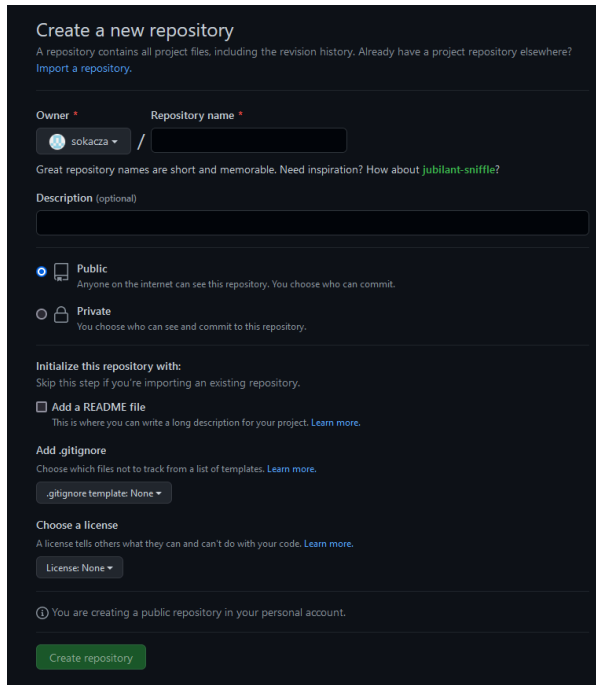


Press **New**.



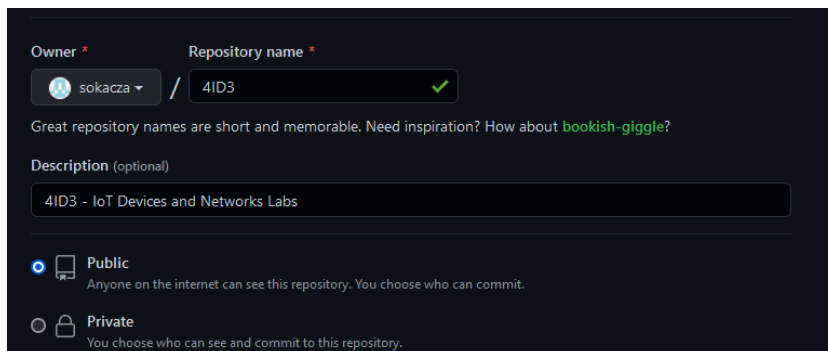
Lab 0 - IoT Pre-Lab Setup

Fill in the requested fields. Set your repository to **public** so anyone can view it without requesting access. Ensure that there are no spaces or uncommon characters in the repository name because this will make it difficult to search for.



The screenshot shows the 'Create a new repository' page on GitHub. The 'Owner' is set to 'sokacza'. The 'Repository name' field is empty. A hint suggests 'jubilant-sniffle?'. The 'Description (optional)' field is empty. The 'Public' option is selected, with a note that anyone on the internet can see the repository. The 'Private' option is also visible. Under 'Initialize this repository with:', the 'Add a README file' checkbox is checked. The 'Add .gitignore' section shows a dropdown for '.gitignore template' set to 'None'. The 'Choose a license' section shows a dropdown for 'License' set to 'None'. A note at the bottom states 'You are creating a public repository in your personal account.' A green 'Create repository' button is at the bottom.

We can set up one repository for the entire course, and you can resync your code after completing each lab assignment.

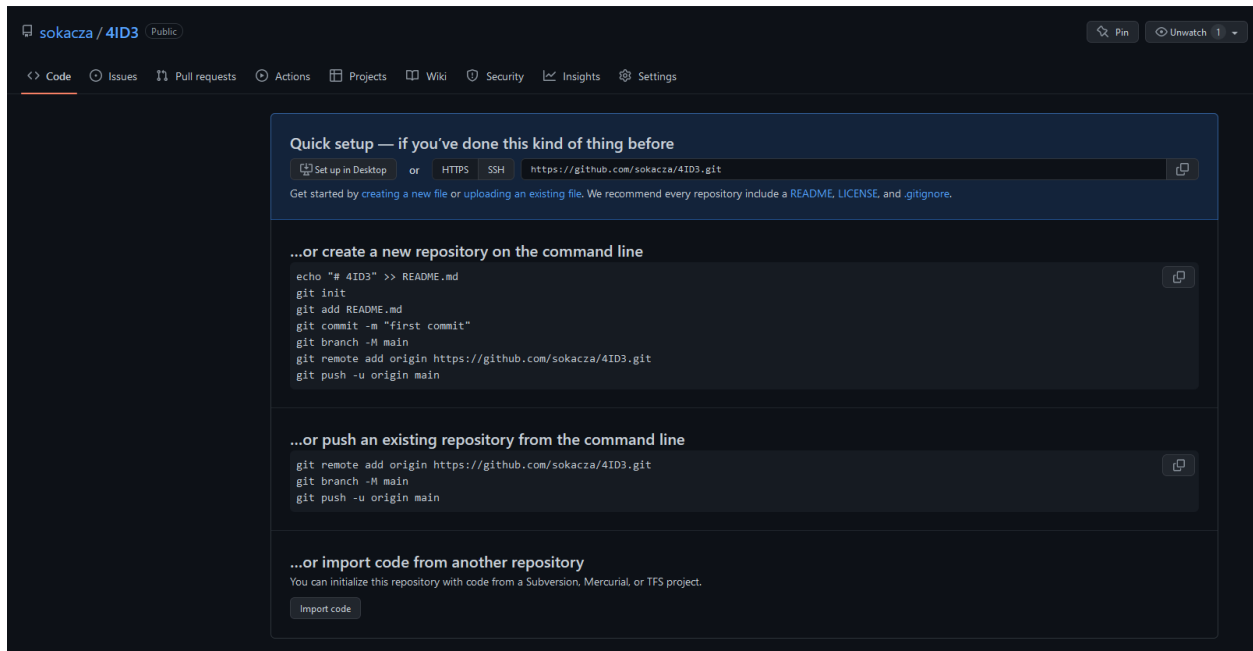


This screenshot shows the same 'Create a new repository' page, but with the 'Repository name' field filled with '4ID3', which is marked with a green checkmark. The 'Description (optional)' field is now filled with '4ID3 - IoT Devices and Networks Labs'. The 'Public' option remains selected. The 'Add a README file' checkbox is still checked. The 'Add .gitignore' and 'Choose a license' sections are unchanged. The 'Create repository' button is still at the bottom.

Press **Create Repository**.

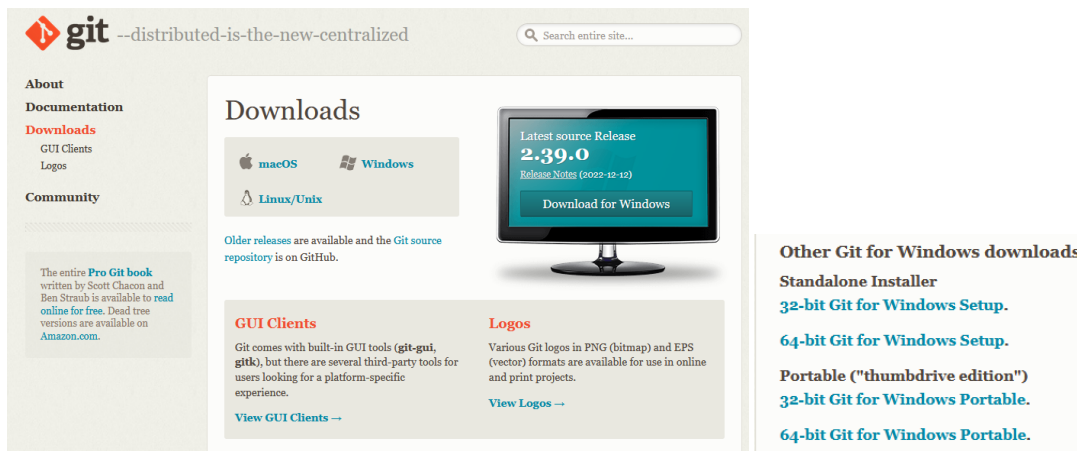
GitHub will now generate a list of commands for further setup. **Do not exit the page** just yet.

Lab 0 - IoT Pre-Lab Setup



The next step is to download **Git**. Git Bash is a command line tool to create, download, and sync repositories. It can be downloaded at the following link:

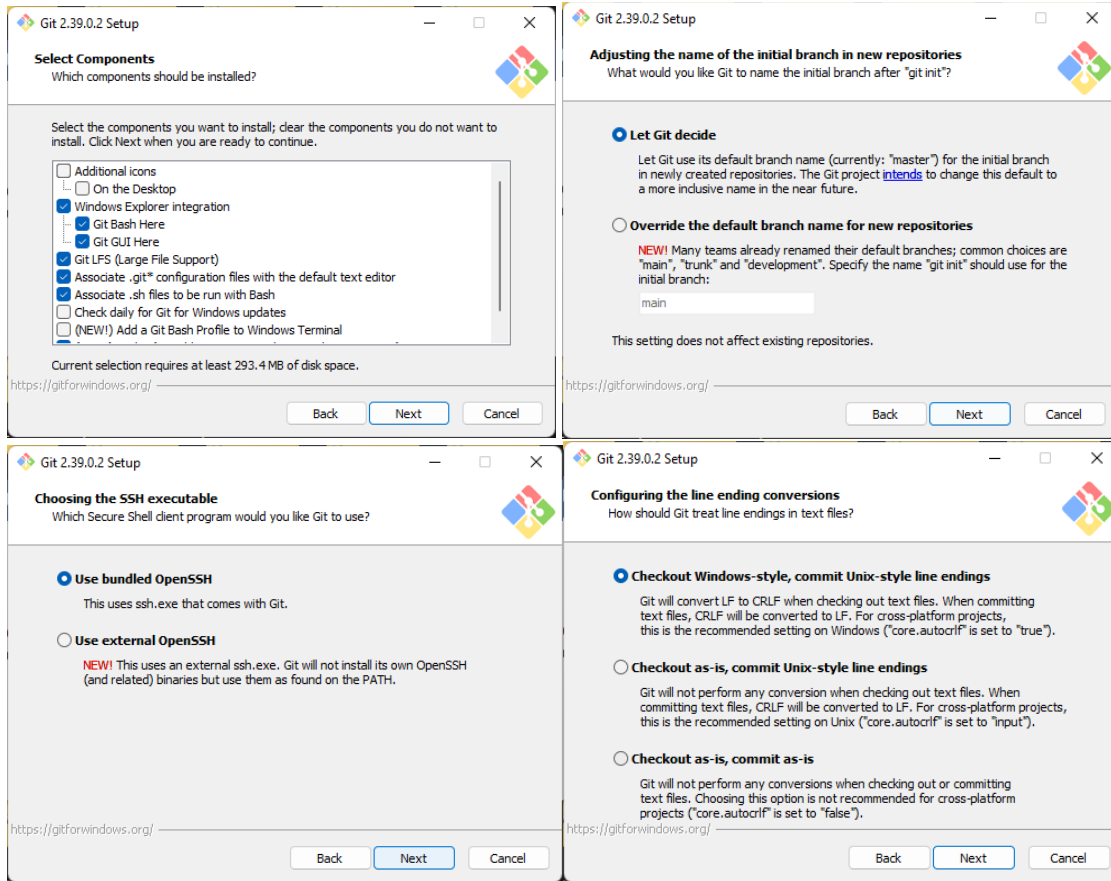
<https://git-scm.com/downloads>



Download the 64 bit version for your operating system.

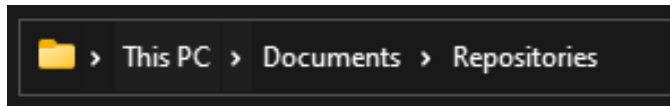
Leave the default install options.

Lab 0 - IoT Pre-Lab Setup

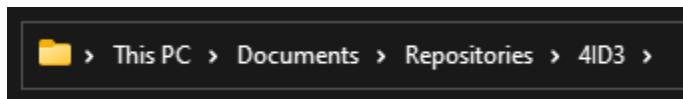


Once installed, we now need to create a local repository and connect it to our remote repository.

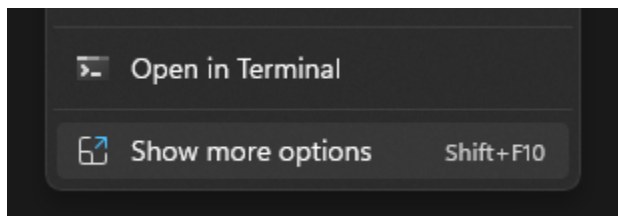
In your **Documents** folder, create a new folder called **Repositories**.



Within the **Repositories** folder, create another folder called **4ID3**.

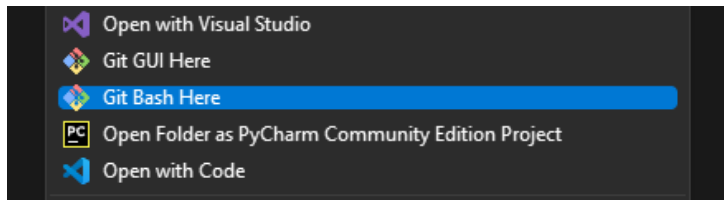


If you are on Windows 11, right click on the folder and select **Show more options**.



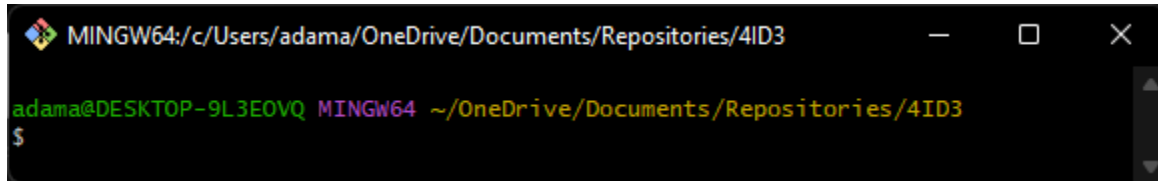
Select **Git Bash here**.

Lab 0 - IoT Pre-Lab Setup

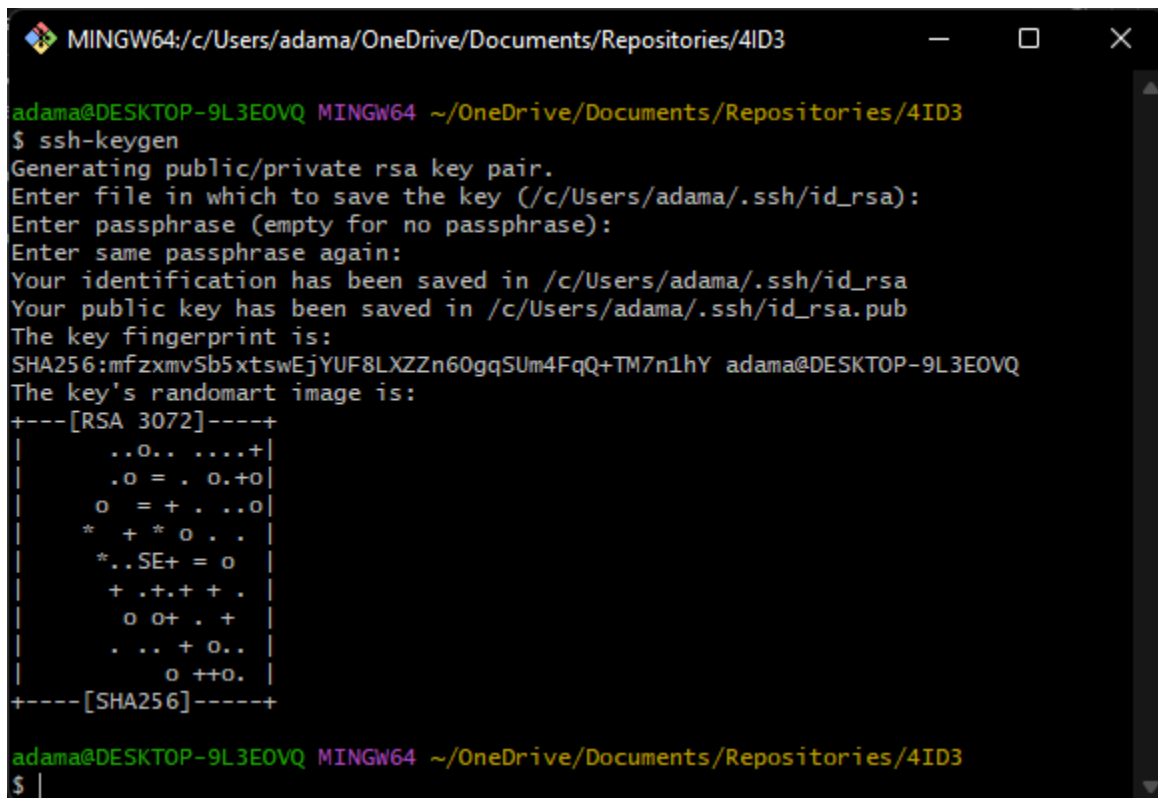


If this option is unavailable, a) Restart your PC, b) find the program in the Start Menu and navigate to this directory.

A terminal window should open.

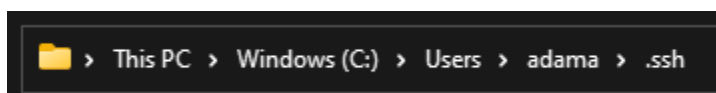


Now, we'll be going back to those commands that GitHub gave us earlier and pasting them here one-by-one. But before that, we need to set up SSH so that the file transfer is encrypted.

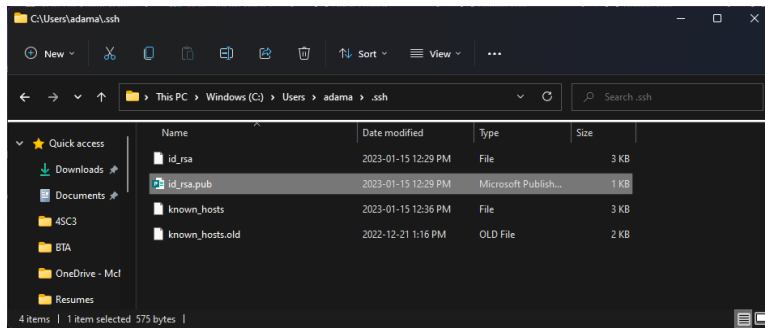


Press **space** to not use a password.

Navigate to the output directory in **file explorer**.



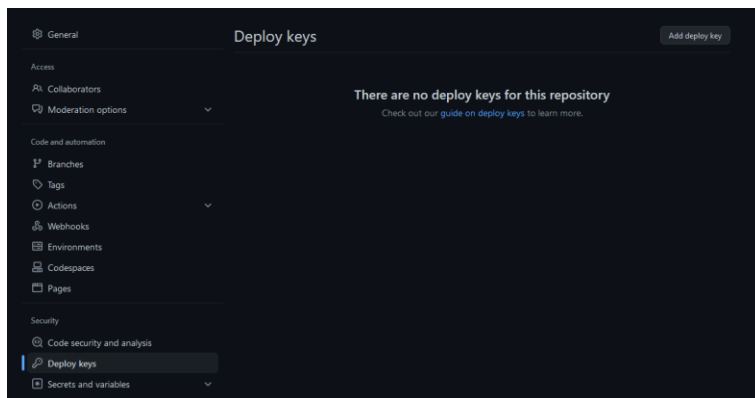
Lab 0 - IoT Pre-Lab Setup



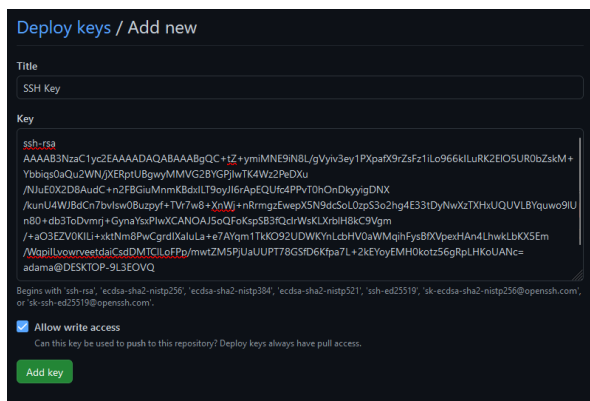
Open **id_rsa.pub** using **Notepad** and copy all the text.

Inside your GitHub repository, select **Settings > Deploy Keys**.

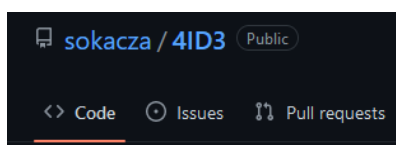
Press **Add deploy key**.



Name the key, **allow write access**, and press **Add key**.

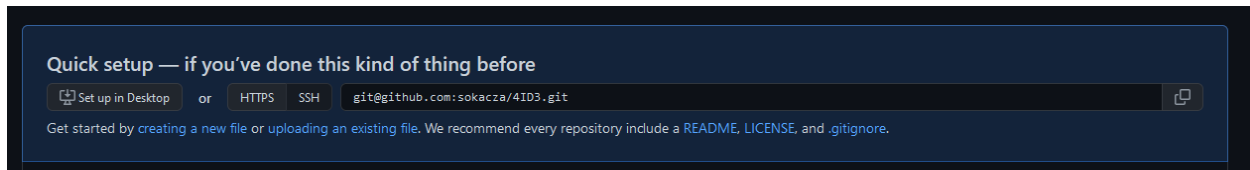


Once the key is added, navigate back to the **Code** tab.



Lab 0 - IoT Pre-Lab Setup

Press **SSH** in the instructions to generate the commands that use SSH.



Copy and paste the listed commands into Git Bash.

```
MINGW64:/c/Users/adama/OneDrive/Documents/Repositories/4ID3
adama@DESKTOP-9L3E0VQ MINGW64 ~/OneDrive/Documents/Repositories/4ID3
$ echo "# 4ID3" >> README.md

adama@DESKTOP-9L3E0VQ MINGW64 ~/OneDrive/Documents/Repositories/4ID3
$ git init
Initialized empty Git repository in C:/Users/adama/OneDrive/Documents/Repositories/4ID3/.git/

adama@DESKTOP-9L3E0VQ MINGW64 ~/OneDrive/Documents/Repositories/4ID3 (master)
$ git add README.md
warning: in the working copy of 'README.md', LF will be replaced by CRLF the next time Git touches it

adama@DESKTOP-9L3E0VQ MINGW64 ~/OneDrive/Documents/Repositories/4ID3 (master)
$ git commit -m "Initial commit"
[master (root-commit) 2fbfd03] Initial commit
1 file changed, 1 insertion(+)
create mode 100644 README.md

adama@DESKTOP-9L3E0VQ MINGW64 ~/OneDrive/Documents/Repositories/4ID3 (master)
$
```

```
MINGW64:/c/Users/adama/OneDrive/Documents/Repositories/4ID3
adama@DESKTOP-9L3E0VQ MINGW64 ~/OneDrive/Documents/Repositories/4ID3 (main)
$ git config --global user.email "sokacza@mcmaster.ca"

adama@DESKTOP-9L3E0VQ MINGW64 ~/OneDrive/Documents/Repositories/4ID3 (main)
$ git config --global user.name "Adam"
```

```
MINGW64:/c/Users/adama/OneDrive/Documents/Repositories/4ID3
adama@DESKTOP-9L3E0VQ MINGW64 ~/OneDrive/Documents/Repositories/4ID3 (master)
$ git branch -M main

adama@DESKTOP-9L3E0VQ MINGW64 ~/OneDrive/Documents/Repositories/4ID3 (main)
$ git remote add origin git@github.com:sokacza/4ID3.git
```

Lab 0 - IoT Pre-Lab Setup

```
MINGW64:/c/Users/adama/OneDrive/Documents/Repositories/4ID3
adama@DESKTOP-9L3E0VQ MINGW64 ~/OneDrive/Documents/Repositories/4ID3 (main)
$ git push origin main
The authenticity of host 'github.com (140.82.113.3)' can't be established.
ED25519 key fingerprint is SHA256:+DiY3wvV6TuJJhbpZisF/zLDA0zPMSvHdkr4UvCOqU.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added 'github.com' (ED25519) to the list of known hosts.
Enumerating objects: 3, done.
Counting objects: 100% (3/3), done.
Writing objects: 100% (3/3), 216 bytes | 216.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0), pack-reused 0
To github.com:sokacza/4ID3.git
 * [new branch]      main -> main

adama@DESKTOP-9L3E0VQ MINGW64 ~/OneDrive/Documents/Repositories/4ID3 (main)
$ |
```

If you get an error, delete all the files in the SSH key folder and regenerate the key. Upload the new key to GitHub.

echo - prints text to a new file

git init - initializes a local repository

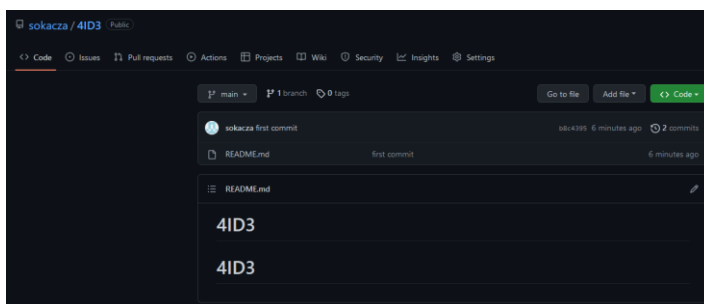
add - tells git that this file has been changed and needs to be synced with GitHub

branch - tells git which version of the software project you wish to sync to. If in doubt, use **main**

remote add - tells git which remote repository to sync to

push - syncs the changes

Now, you should see a file called README.md in your GitHub. While the setup of GitHub might be tedious, using it from here on out is straightforward.



Common problems:

1. GitHub does not recognize your PC

Follow the instructions printed to the terminal on how to add your email and name.

Lab 0 - IoT Pre-Lab Setup

```
git config --global user.email "my@email.com"
```

```
git config --global user.name "my name"
```

2. Some refs failed to push

Regenerate the repo by doing the following:

```
git init
```

```
git add .
```

```
git commit -m "First commit"
```

```
git push origin main
```

3. sokacza/4ID3 not found

In this situation, you used my repo as the origin url instead of your own. Do the following:

```
git remote remove origin
```

```
git remote add your_remote_url
```

4. You cannot find git bash in your toolbar

Ensure that git bash is installed. If it is, launch it from the start menu and use cd to navigate to the project directory.

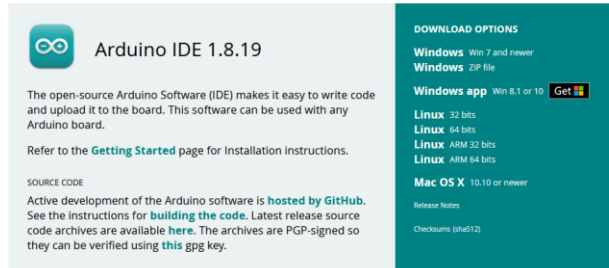
Installing the Arduino IDE

The Arduino IDE is an integrated development environment created by the company Arduino Inc. An integrated development environment provides software developers the tools to efficiently edit code, manage libraries, and troubleshoot issues.

Navigate to the following website and download the latest version of Arduino IDE **version 1.XX**.

<https://www.arduino.cc/en/software>

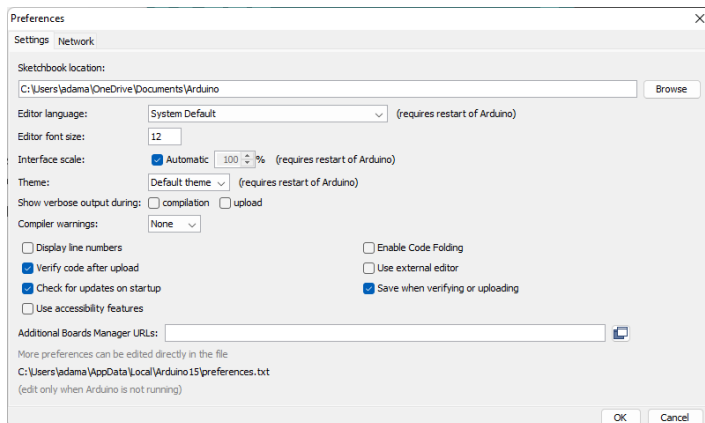
Legacy IDE (1.8.X)



Navigate through the installer, selecting the default options.

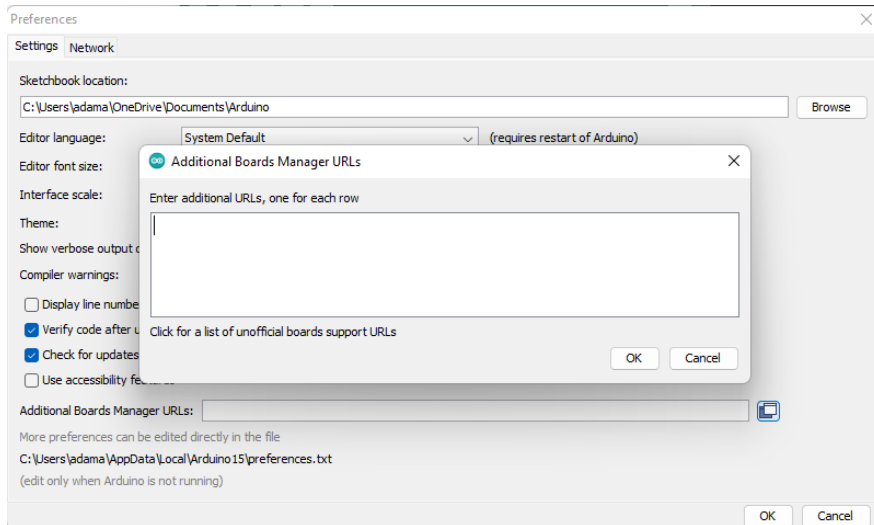
The ESP32 and ESP8266 will be used frequently throughout this course. In order to compile for those boards and see them in the board manager, a custom URL must be added into the preferences.

Select **Preferences** and see the **Additional Board Manager URL's** field.



Press the button to the right of the field.

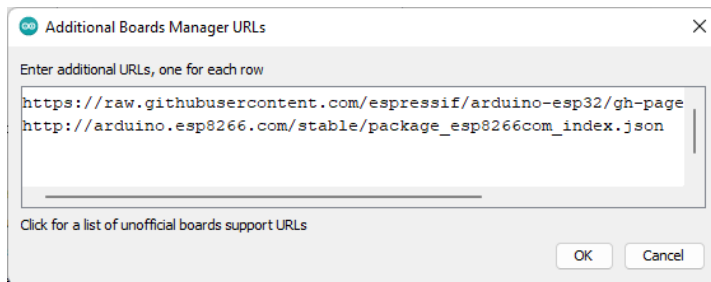
Lab 0 - IoT Pre-Lab Setup



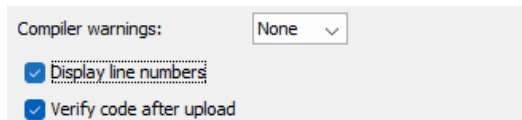
Paste the following **two** url's on separate lines:

https://raw.githubusercontent.com/espressif/arduino-esp32/gh-pages/package_esp32_index.json

http://arduino.esp8266.com/stable/package_esp8266com_index.json



Press **OK**. Also, within preferences, check **Display line numbers**.



Press **OK** to exit.

Install the ESP8266 board by ESP8266 Community.

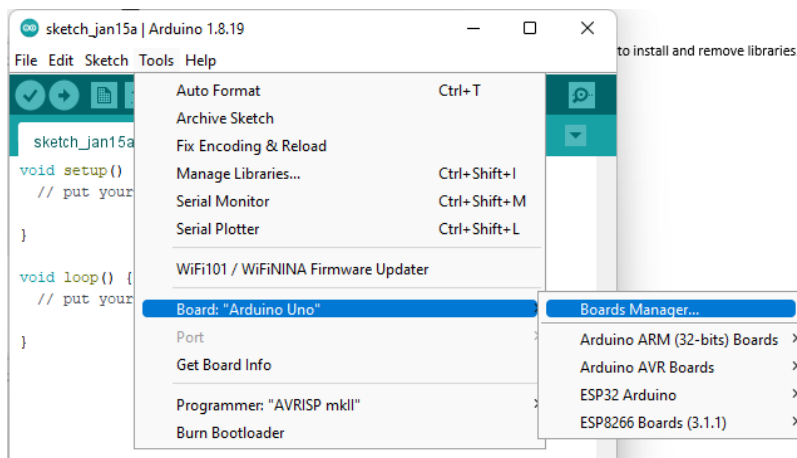
Lab 0 - IoT Pre-Lab Setup



Lastly, install the ESP32 board by Espressif Systems.



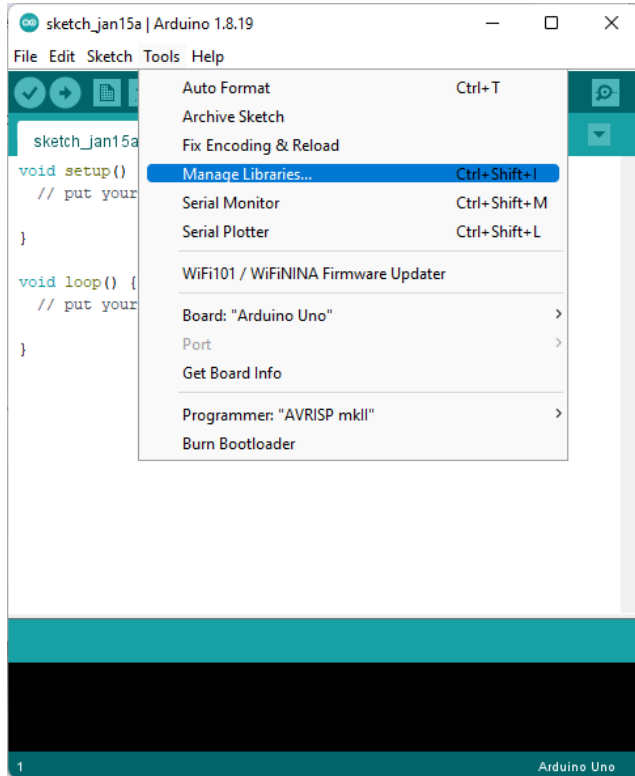
Ensure that you can see the boards available in your **Board Manager**.



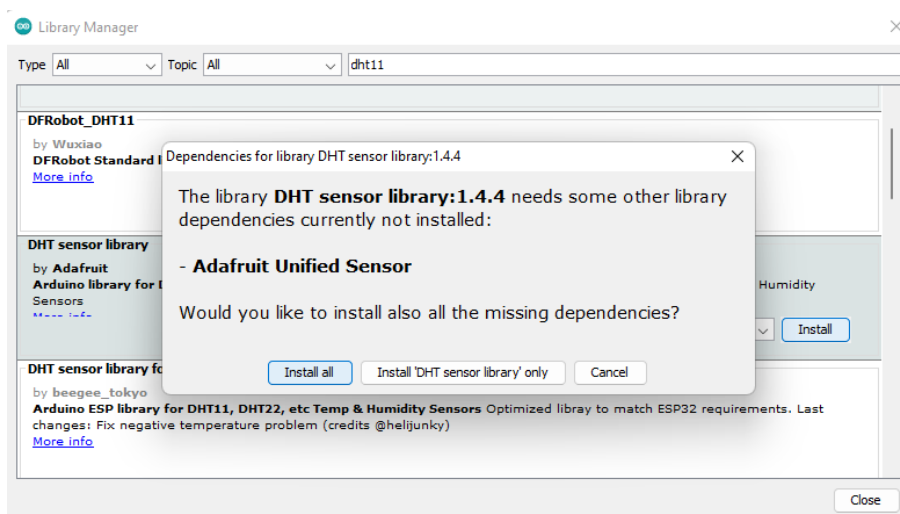
Installing Arduino Sensor Libraries

The Arduino IDE has a built-in library manager that can be used to install and remove libraries.

Navigate to **Tools > Manage Libraries**.



Search for the DHT sensor library by Adafruit and install the latest version. If prompted to install the unified library, choose **Install all**.



Lab 0 - IoT Pre-Lab Setup

Next, install the BMP Unified sensor library by Adafruit.

Adafruit BMP085 Unified

by Adafruit Version 1.1.1 **INSTALLED**

Unified sensor driver for Adafruit's BMP085 & BMP180 breakouts Unified sensor driver for Adafruit's BMP085 & BMP180 breakouts

[More info](#)

Next, install the BH1750 library by Stefan Armbrorst.

hp_BH1750

by Stefan Armbrorst Version 1.0.2 **INSTALLED**

Digital light sensor breakout boards containing the BH1750FVI IC high performance non-blocking BH1750 library

[More info](#)

Lastly, install the PubSubClient MQTT library by Nick O'Leary.

PubSubClient

by Nick O'Leary Version 2.8.0 **INSTALLED**

A client library for MQTT messaging. MQTT is a lightweight messaging protocol ideal for small devices. This library allows you to send and receive MQTT messages. It supports the latest MQTT 3.1.1 protocol and can be configured to use the older MQTT 3.1 if needed. It supports all Arduino Ethernet Client compatible hardware, including the Intel Galileo/Edison, ESP8266 and TI CC3000.

[More info](#)

Installing NodeJS

NodeJS is a JavaScript runtime. This means that it allows JavaScript software to run outside of the web browser and have access to the filesystem and other computing resources. NodeJS is a prerequisite to installing NodeRED, which is a graphical data-driven programming environment that can be used to rapidly build and deploy IoT applications and dashboards.

Navigate to the following url:

<https://nodejs.org/en/download/>

Downloads

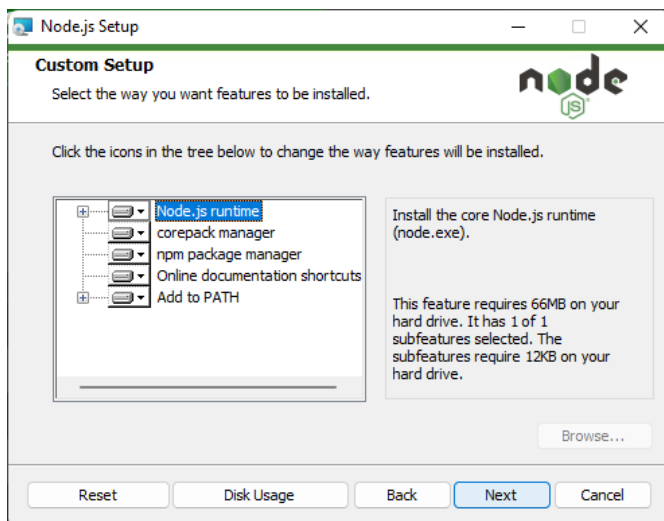
Latest LTS Version: **18.13.0** (includes npm 8.19.3)

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

LTS Recommended For Most Users	Current Latest Features	
 Windows Installer node-v18.13.0-x64.msi	 macOS Installer node-v18.13.0.pkg	 Source Code node-v18.13.0.tar.gz

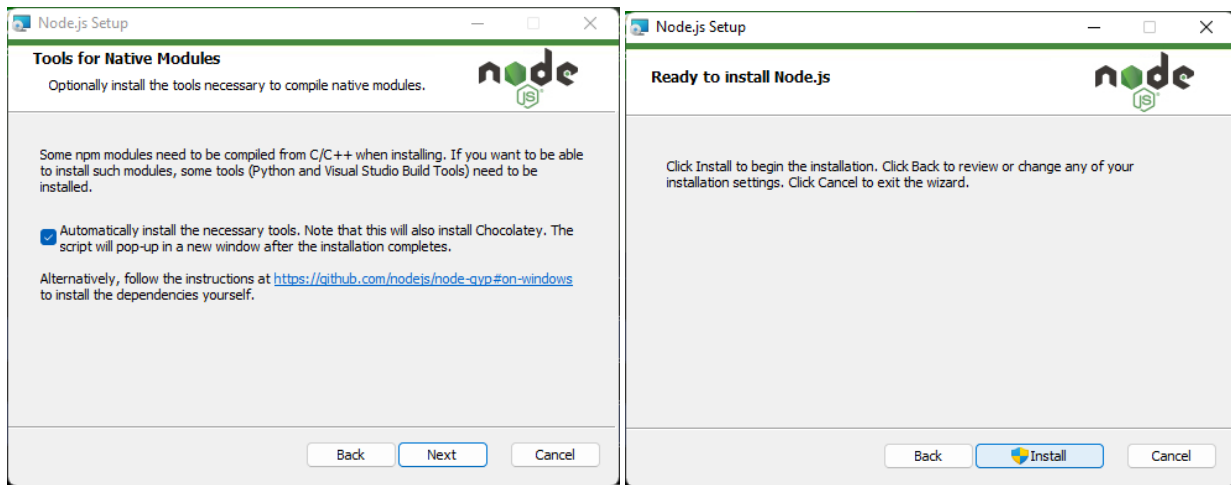
Windows Installer (.msi)	32-bit	64-bit
Windows Binary (.zip)	32-bit	64-bit
macOS Installer (.pkg)	64-bit / ARM64	
macOS Binary (.tar.gz)	64-bit	ARM64
Linux Binaries (x64)	64-bit	
Linux Binaries (ARM)	ARMv7	ARMv8
Source Code	node-v18.13.0.tar.gz	

Select the correct installer for your operating system.



Lab 0 - IoT Pre-Lab Setup

Have it automatically install the necessary tools and press **Install**.



Allow the additional tools to install. This may take 2 minutes.

```
Administrator: Windows PowerShell
WARNING: 'choco' was found at 'C:\ProgramData\chocolatey\bin\choco.exe'.
WARNING: An existing Chocolatey installation was detected. Installation will not continue.
For security reasons, this script will not overwrite existing installations.

Please use choco upgrade chocolatey to handle upgrades of Chocolatey itself.
If the existing installation is not functional or a prior installation did not complete, follow these steps:
- Backup the files at the path listed above so you can restore your previous installation if needed
- Remove the existing installation manually
- Rerun this installation script
- Reinstall any packages previously installed, if needed (refer to the lib folder in the backup)

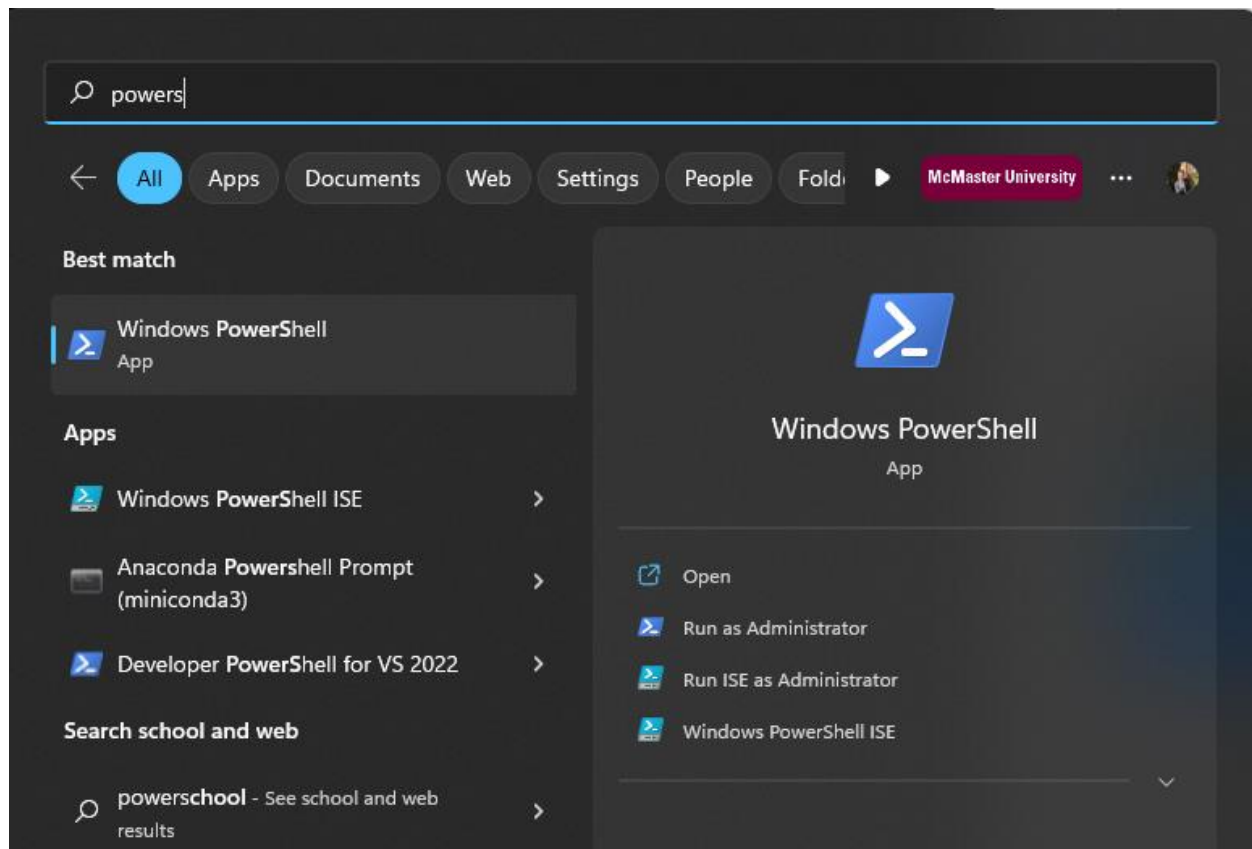
Once installation is completed, the backup folder is no longer needed and can be deleted.
Chocolatey v1.1.0
Upgrading the following packages:
python:visualstudio2019-workload-vctools
By upgrading, you accept licenses for the packages.

You have python v3.10.4 installed. Version 3.11.0 is available based on your source(s).
Progress: Downloading python3 3.11.0... 100%
Progress: Downloading python 3.11.0... 100%

python3 v3.11.0 [Approved]
python3 package files upgrade completed. Performing other installation steps.
```

To verify that it has been installed correctly, using the **Start Button**, search for **Powershell**.

Lab 0 - IoT Pre-Lab Setup



Run the node command with the version flag to echo out the current installed version of NodeJS.

```
Windows PowerShell
PS C:\Users\adama> node --version
v18.13.0
PS C:\Users\adama>
```

node --version

Installing NodeRED

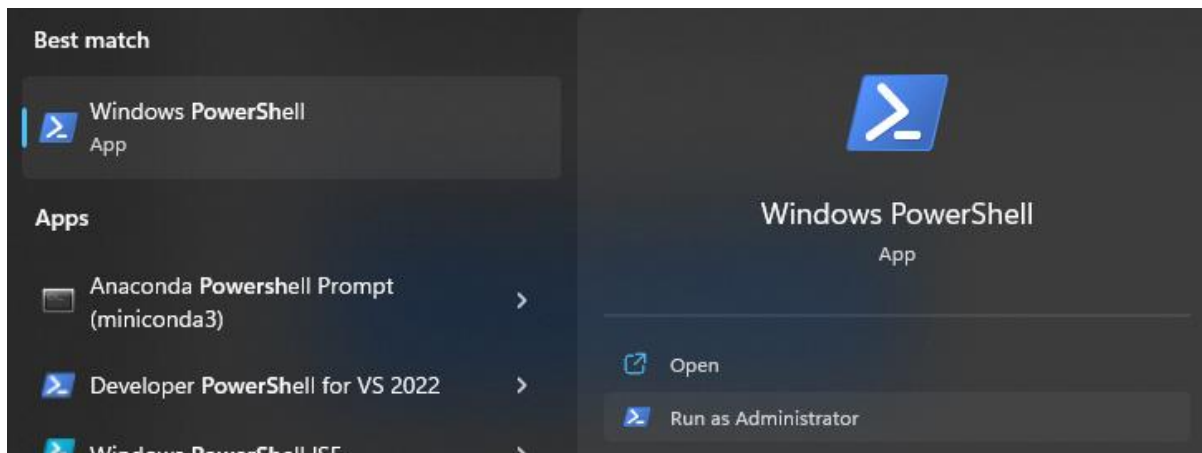
NodeRED will be installed by NPM. NPM is an acronym for Node Package Manager. When programming, sometimes different programs need to use the same libraries or frameworks. A package manager ensures the same libraries are not installed twice, they remain up-to-date, and they have the correct dependencies installed for them to work properly. NPM will be used to install NodeRED.

The official install documentation can be found here:

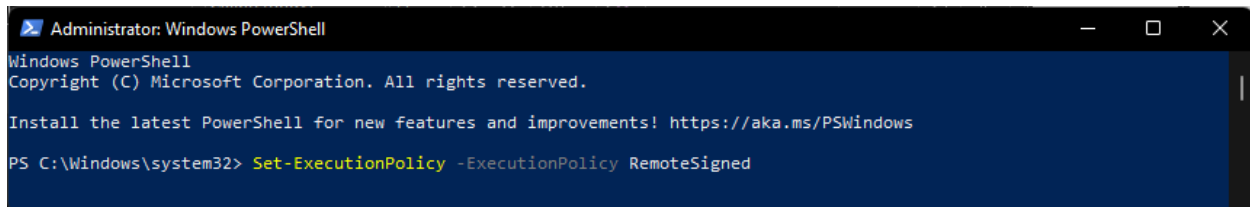
<https://nodered.org/docs/getting-started/windows>

Open **Powershell**.

Launch PowerShell as **Administrator**.



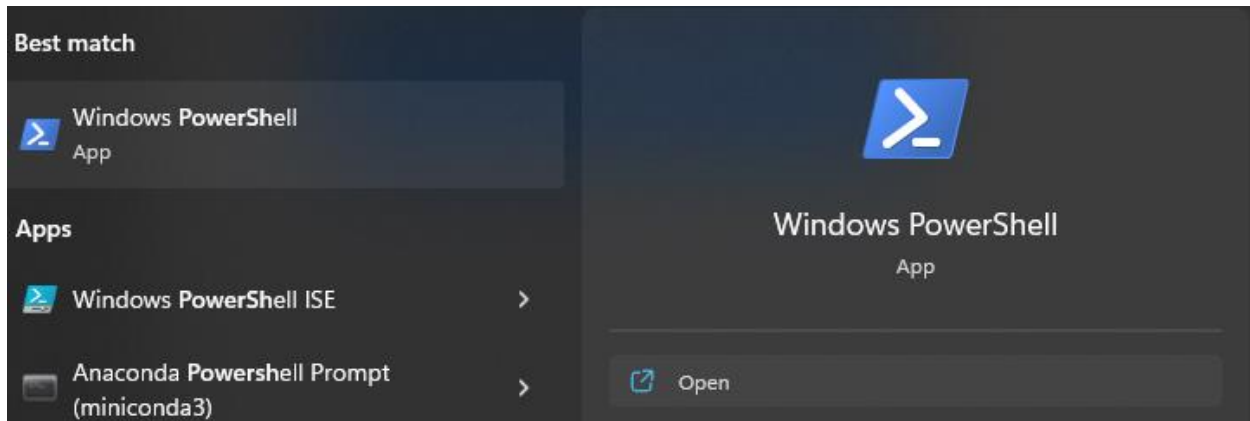
Use the following command to change the execution policy.



Set-ExecutionPolicy -ExecutionPolicy RemoteSigned

Next, re-open Powershell as a normal user.

Lab 0 - IoT Pre-Lab Setup



Check to ensure that NPM has been installed correctly.

```
Windows PowerShell
PS C:\Users\adama> npm --version
8.19.3
PS C:\Users\adama>
```

`npm --version`

Install NodeRED using the following command:

```
Windows PowerShell
PS C:\Users\adama> npm install -g --unsafe-perm node-red

added 292 packages, and audited 293 packages in 12s

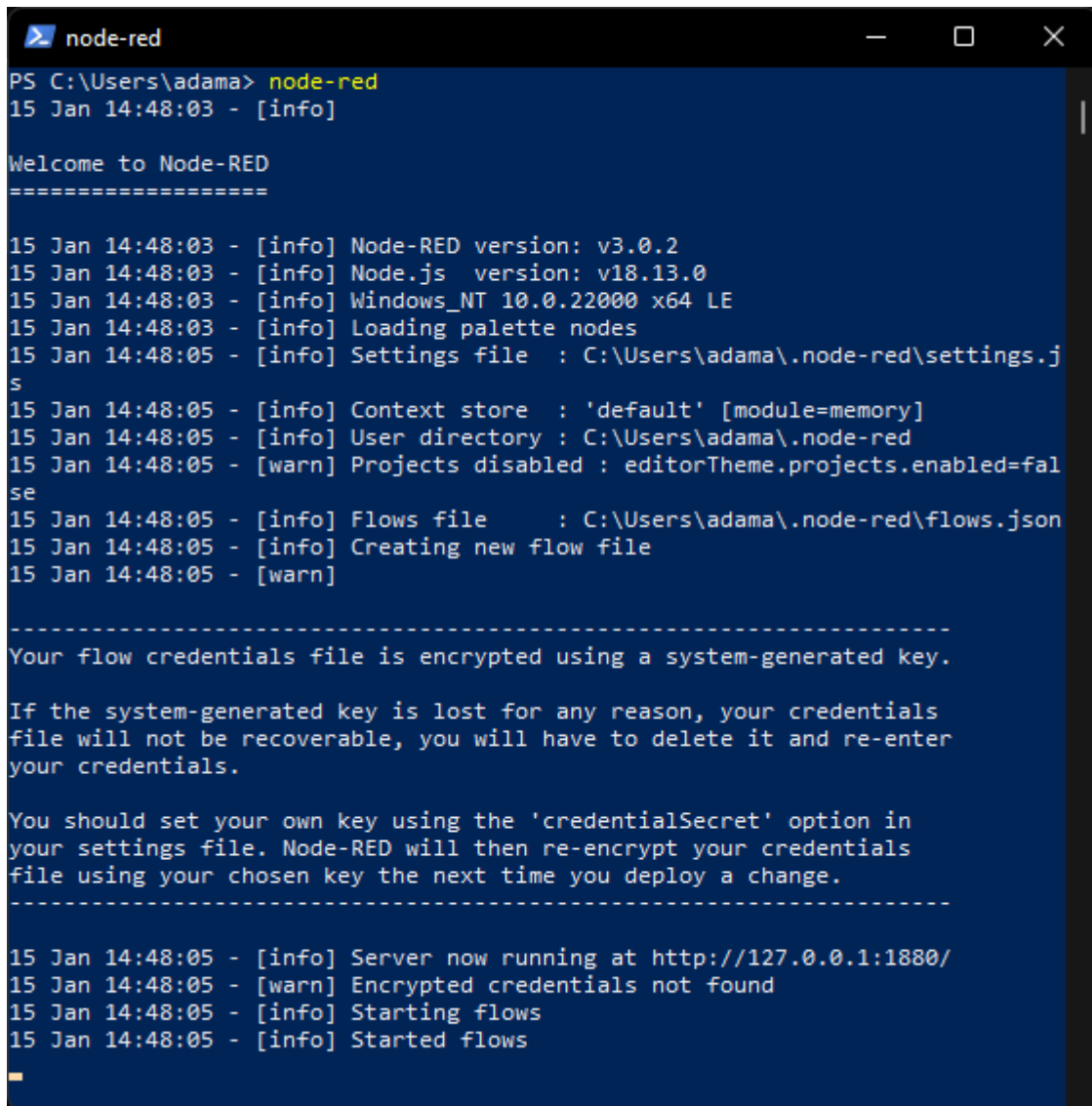
39 packages are looking for funding
  run `npm fund` for details

5 vulnerabilities (4 low, 1 moderate)
To address issues that do not require attention, run:
  npm audit fix
To address all issues (including breaking changes), run:
  npm audit fix --force
Run `npm audit` for details.
npm notice
npm notice New major version of npm available! 8.19.3 -> 9.3.0
npm notice Changelog: https://github.com/npm/cli/releases/tag/v9.3.0
npm notice Run `npm install -g npm@9.3.0` to update!
npm notice
PS C:\Users\adama>
PS C:\Users\adama>
```

`npm install -g --unsafe-perm node-red`

Lab 0 - IoT Pre-Lab Setup

To start NodeRED, use the following command:



```
node-red
PS C:\Users\adama> node-red
15 Jan 14:48:03 - [info]

Welcome to Node-RED
=====

15 Jan 14:48:03 - [info] Node-RED version: v3.0.2
15 Jan 14:48:03 - [info] Node.js version: v18.13.0
15 Jan 14:48:03 - [info] Windows_NT 10.0.22000 x64 LE
15 Jan 14:48:03 - [info] Loading palette nodes
15 Jan 14:48:05 - [info] Settings file : C:\Users\adama\.node-red\settings.js
15 Jan 14:48:05 - [info] Context store : 'default' [module=memory]
15 Jan 14:48:05 - [info] User directory : C:\Users\adama\.node-red
15 Jan 14:48:05 - [warn] Projects disabled : editorTheme.projects.enabled=false
15 Jan 14:48:05 - [info] Flows file : C:\Users\adama\.node-red\flows.json
15 Jan 14:48:05 - [info] Creating new flow file
15 Jan 14:48:05 - [warn]

-----
Your flow credentials file is encrypted using a system-generated key.

If the system-generated key is lost for any reason, your credentials
file will not be recoverable, you will have to delete it and re-enter
your credentials.

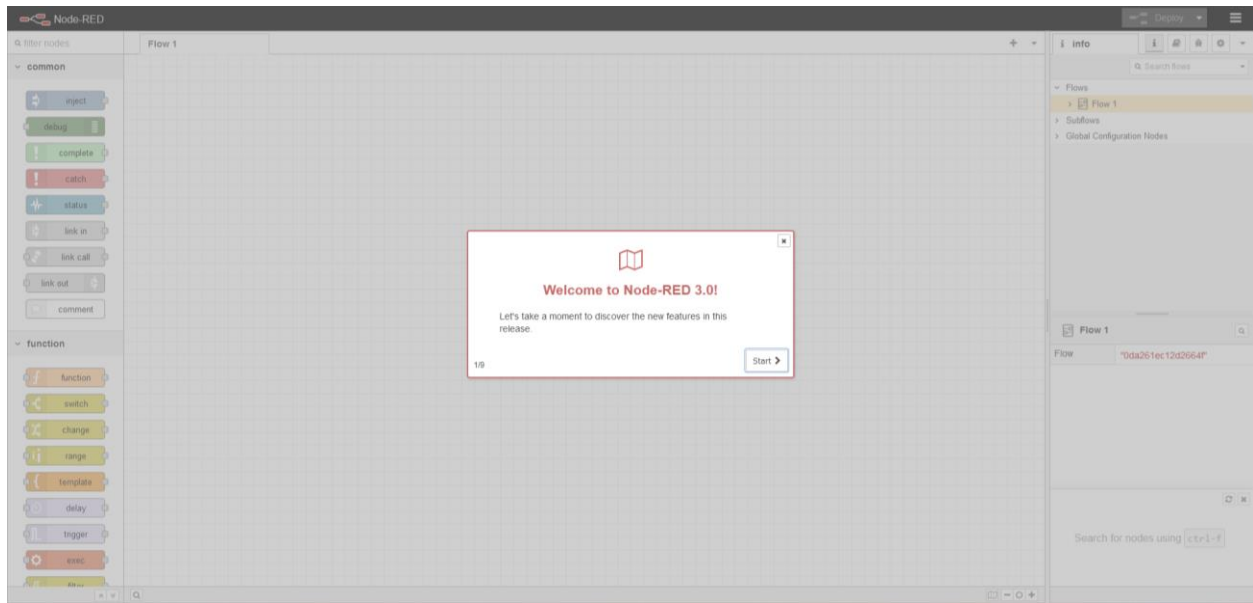
You should set your own key using the 'credentialSecret' option in
your settings file. Node-RED will then re-encrypt your credentials
file using your chosen key the next time you deploy a change.
-----

15 Jan 14:48:05 - [info] Server now running at http://127.0.0.1:1880/
15 Jan 14:48:05 - [warn] Encrypted credentials not found
15 Jan 14:48:05 - [info] Starting flows
15 Jan 14:48:05 - [info] Started flows
```

Notice the line **Server now running at...**

Type that **IP address** into your **web browser**. Do not close this terminal window until you are done using NodeRED. Ensure that you see the **Welcome to Node-RED** screen.

Lab 0 - IoT Pre-Lab Setup

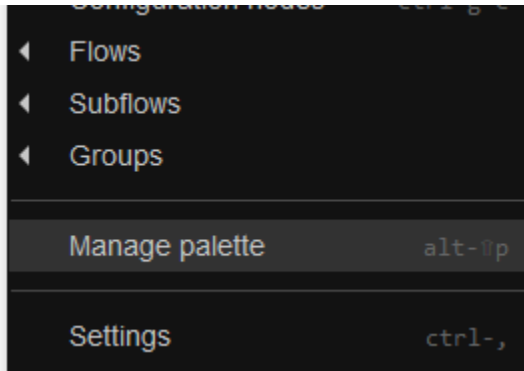


To safely close NodeRED, open the Powershell window and press **CTRL + C**.

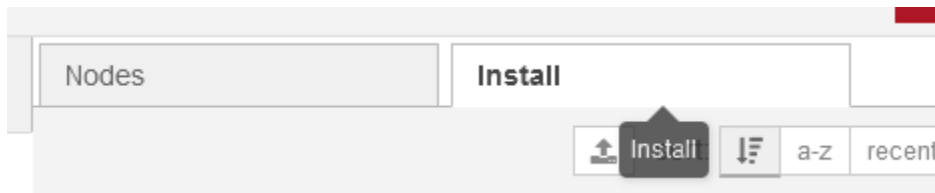
Installing NodeRED Dashboard

NodeRED Dashboard provides us a ready-to-use interface for visualizing our NodeRED applications.

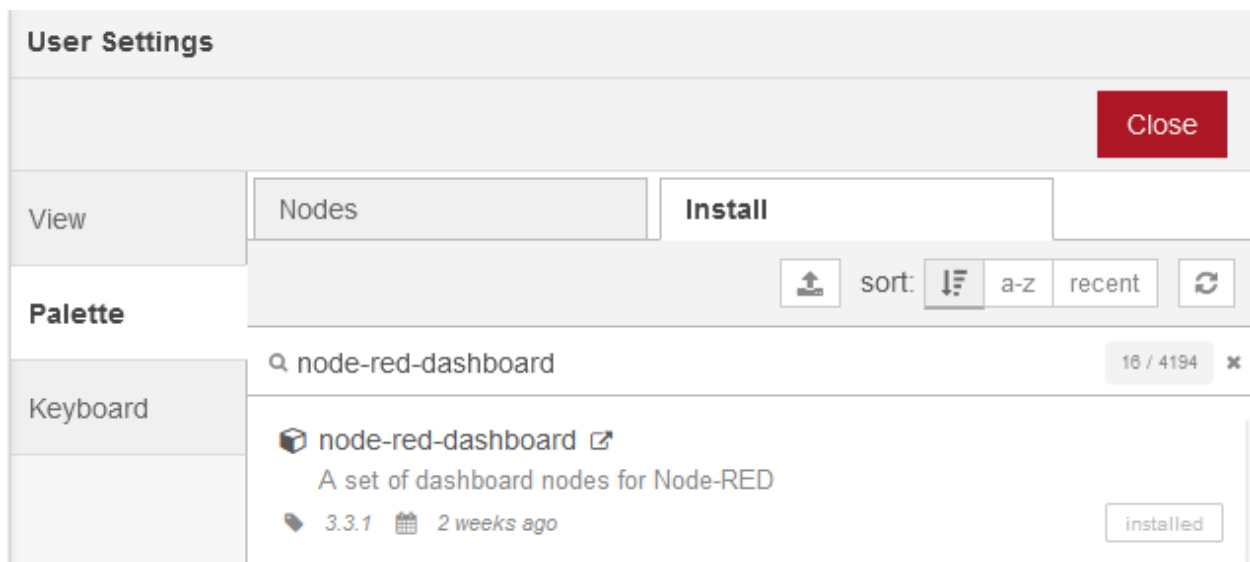
To add the **node-red-dashboard** add-on, navigate to **Manage palette** by pressing on the hamburger menu on the top right corner of the NodeRED interface.



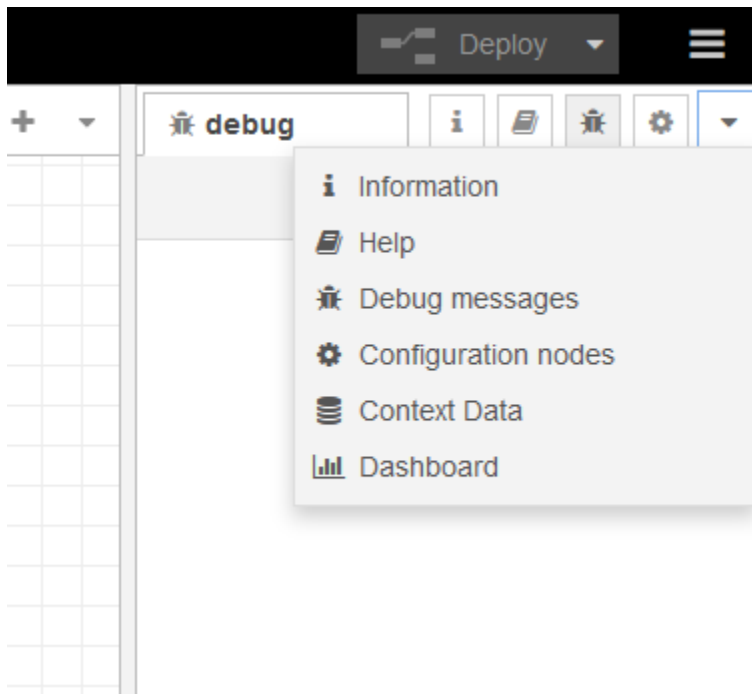
Click the **Install** tab.



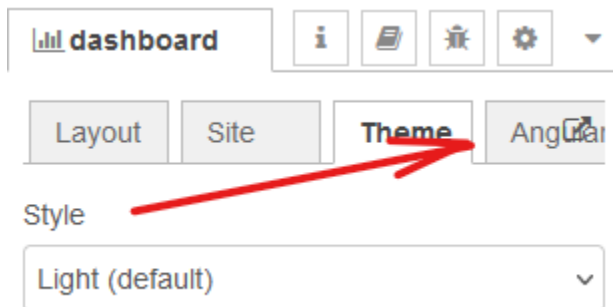
Filter for **node-red-dashboard** and press **Install**.



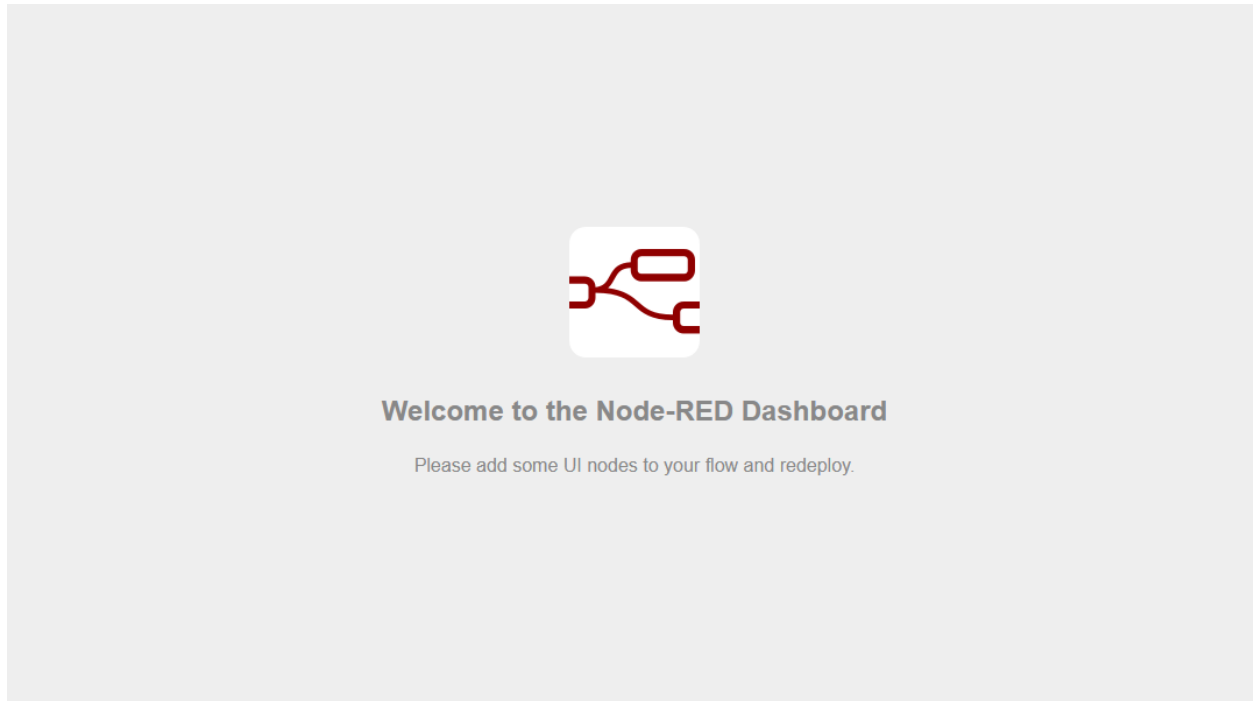
Using the **dropdown menu**, click on **Dashboard**.



Press the hard to see button to open the dashboard.

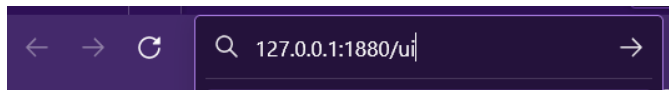


Lab 0 - IoT Pre-Lab Setup



If you want to check if it successfully installed, to get to the dashboard without pressing the **Open Dashboard Icon**, you can instead launch **node-red** from Powershell and in your web browser, use the following url:

127.0.0.1:1880/ui



Mosquitto MQTT Broker Installation

Mosquitto is a free MQTT broker from the Eclipse foundation.

Navigate to the following downloads page:

<https://mosquitto.org/download/>

Select the version for your operating system.

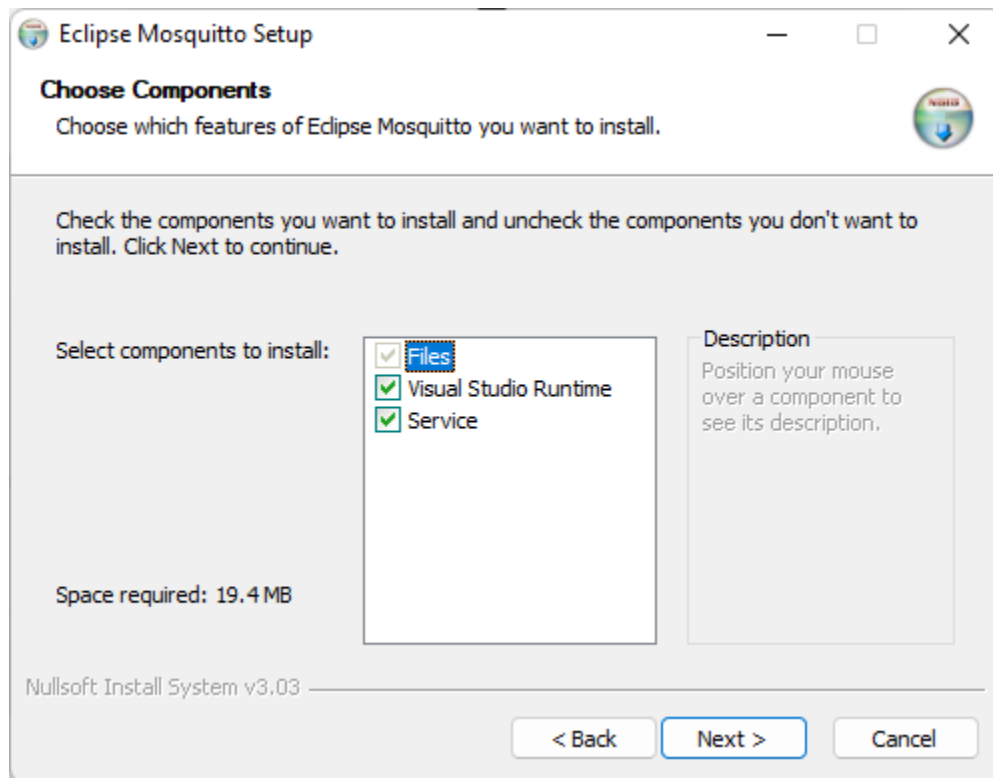
Windows

- `mosquitto-2.0.15-install-windows-x64.exe` (64-bit build, Windows Vista and up, built with Visual Studio Community 2019)
- `mosquitto-2.0.15-install-windows-x32.exe` (32-bit build, Windows Vista and up, built with Visual Studio Community 2019)

Older installers can be found at <https://mosquitto.org/files/binary/>.

See also README-windows.md after installing.

Install using the default options.



Once installed, open Powershell and navigate to the install directory.

Lab 0 - IoT Pre-Lab Setup

```
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Install the latest PowerShell for new features and improvements! https://aka.ms/PSWindows

PS C:\Users\adama> cd '..\..\Program Files\mosquitto\'
PS C:\Program Files\mosquitto> ls

    Directory: C:\Program Files\mosquitto

Mode                LastWriteTime         Length Name
----                -
d-----          2023-01-15  7:23 PM             devel
-a-----          2022-08-16  9:34 AM              230 aclfile.example
-a-----          2022-08-16  9:34 AM         135368 ChangeLog.txt
-a-----          2022-08-16  9:34 AM          1568 edl-v10
-a-----          2022-08-16  9:34 AM         14197 epl-v20
-a-----          2022-07-05  5:43 PM        3415552 libcrypto-1_1-x64.dll
-a-----          2022-07-05  5:43 PM        685056 libssl-1_1-x64.dll
-a-----          2022-08-16  9:34 AM          40449 mosquitto.conf
-a-----          2022-08-16  9:35 AM          87040 mosquitto.dll
-a-----          2022-08-16  9:41 AM        382464 mosquitto.exe
-a-----          2022-08-16  9:35 AM          18432 mosquitto_top.dll
-a-----          2022-08-16  9:35 AM          76288 mosquitto_ctrl.exe
-a-----          2022-08-16  9:37 AM        122880 mosquitto_dynamic_security.dll
-a-----          2022-08-16  9:34 AM          22528 mosquitto_passwd.exe
-a-----          2022-08-16  9:35 AM          51712 mosquitto_pub.exe
-a-----          2022-08-16  9:35 AM          79872 mosquitto_rr.exe
-a-----          2022-08-16  9:35 AM          81920 mosquitto_sub.exe
-a-----          2022-08-16  9:34 AM           1886 NOTICE.md
-a-----          2022-08-16  9:34 AM           355 pwfile.example
-a-----          2022-08-16  9:34 AM           939 README-letsencrypt.md
-a-----          2022-08-16  9:34 AM          2453 README-windows.txt
-a-----          2022-08-16  9:34 AM          3768 README.md
-a-----          2023-01-15  7:23 PM        66085 Uninstall.exe

PS C:\Program Files\mosquitto>
```

`cd 'C:\Program Files\mosquitto\'`

Start the broker using the following command:

```
Windows PowerShell

PS C:\Program Files\mosquitto> .\mosquitto.exe -v
1673829026: mosquitto version 2.0.15 starting
1673829026: Using default config.
1673829026: Starting in local only mode. Connections will only be possible from clients running on this machine.
1673829026: Create a configuration file which defines a listener to allow remote access.
1673829026: For more details see https://mosquitto.org/documentation/authentication-methods/
1673829026: Opening ipv4 listen socket on port 1883.
1673829026: Opening ipv6 listen socket on port 1883.
1673829026: mosquitto version 2.0.15 running
```

`.\mosquitto.exe -v`

If the server starts up without errors then the installation has been successful.

End