Let's verify your QLSC 612 software setup

A PDF of these slides is available in the QLS-course-materials GitHub repository, under Lectures/2024/00_installation_clinic

1. Bash shell

Open a **terminal** and type: `echo $SHELL`

```
# Expected output
/bin/bash
```

Not working? Tips:

- Mac/Linux: You may have to type `bash` first to access the bash shell.
- Windows: Ensure you are in the WSL2 Ubuntu terminal.

1.1. Install the `tree` program

This is an optional step, but will be useful in the Terminal and Bash lecture to view directory structures.

Check if you have `tree` already installed:

```
tree --version
```

If you get a message `tree: command not found`, install it using the package manager for your OS (will take a few seconds):

Linux or WSL

```
sudo apt-get update
sudo apt-get install tree
```

macOS

```
brew install tree
```

2. Git

2.1. Type: `git --version`

```
# Expected output
git version 2.xx.x
```

2.2. Now, let's do some basic Git configuration.
Type the following commands, and make sure to use the name and email associated with your GitHub account.

```
git config --global user.name "Jane Doe"
git config --global user.email "janedoe@example.com"
git config --global core.autocrlf true
```
Tip: You can review your configuration at any time with: `git config --list`

2.3. Create a personal access token (PAT) for GitHub

To authenticate to GitHub from the command line, you will use a PAT instead of your GitHub account password.

Log in to your GitHub account, and follow these instructions to create a **fine-grained PAT**: https://docs.github.com/en/authentication/keeping-your-account-and-data-secure/managing-your-personal-access-tokens#creating-a-fine-grained-personal-access-token

- Use an informative name for the token, such as "QLSC612_PAT"
- Under **Expiration**, select **7 days**
- Under **Resource owner**, select your username
- Under **Repository access**, select **All repositories**
- Under **Permissions**, navigate to **Contents** and change the access level to **Read and write**

Then, click **Generate token**, and copy and store the resulting token value in a secure location*. In this course, when using a `git` command that prompts you for your password, you will enter this token.

***DISCLAIMER**

PATs are like passwords, and **it is not best practice to save them in plain text on your computer!!** As an exception for this course, you are using a very short-term PAT with limited permissions so that you can push to GitHub repositories from the command line without downloading additional tools.

To use git/GitHub for your own purposes after this course, you can use the GitHub CLI or Git Credential manager instead of creating a PAT. For more info, see:

- https://docs.github.com/en/authentication/keeping-your-account-and-data-secure/managing-your-personal-access-tokens#keeping-your-personal-access-tokens-secure
- https://docs.github.com/en/authentication/keeping-your-account-and-data-secure/about-authentication-to-github#authenticating-with-the-command-line

A note for VS Code users

If you use the terminal in VS Code, when you run a Git action that requires GitHub authentication, you may instead see a popup prompt to sign in using GitHub.

If you get this popup, you can can select "Allow" to sign in using your browser, in which case you should be able to authenticate using your regular username and password with two-factor authentication.

3. Python

3.1. Let's check for the conda environment that you created during the course setup steps:

```
conda env list
```

You should see `qlsc612` in the list of available conda environments.

3.2. Activate the `qlsc612` conda environment:

```
conda activate qlsc612
```

```
# Your shell should now display:
(qlsc612) USERNAME@YOURMACHINE:
```

3.3. Confirm that the path of the Python interpreter being used is correct:

```
which python
```

```
# Expected output
home/USERNAME/miniconda3/envs/qlsc612/bin/python
```

If you are using Anaconda, the path should end in something like `anaconda3/envs/qlsc612/bin/python` instead.

Tip: To deactivate the active conda environment after you are finished with it, type:

```
conda deactivate
```

4. Docker

4.1. If you are on a Mac or Windows machine, first start the Docker Desktop application.

4.2. In the terminal, type:

```
docker run hello-world
```

(You will learn more about Docker commands in the Containers module!)

After a few seconds, you should see a message that starts with:

```
Hello from Docker!
This message shows that your installation appears to be working correctly.
...
```

Getting the course materials

Let's get a copy of the materials we will need for the lectures from GitHub onto our own computer.

For convenience, let's store the course materials in our home directory (represented by `~`). In the terminal, type:

```
cd
git clone https://github.com/neurodatascience/QLS-course-materials.git --depth 1
```

(You'll learn more about `git` commands in the Git and GitHub module!)

This will take a few seconds to complete. Once finished, type `ls`. You should see among the output `QLS-course-materials`. This is the folder containing all the course materials.

Note: If you have already cloned the repo elsewhere, we recommend deleting that copy and running the above commands to get the latest version of the materials in your home directory.

From now on, to ensure you have the latest version of the materials for a lecture from GitHub, simply type:

```
cd
cd QLS-course-materials
git pull
```

A note for Windows users

You will not be able to easily find this new folder `QLS-course-materials` in your normal file explorer, because the WSL2 file system is separate from your normal Windows file system (which is under `/mnt` inside WSL).

However, if you *really want* to be able to view the course materials directory in your file explorer, type `explorer.exe .` while you are in your home directory (`~`) in the WSL2 Ubuntu terminal (type `cd` first if you are not sure). This will open a file explorer window showing your WSL2 home directory files and folders, including `QLS-course-materials`. *You should not directly modify any files in the WSL2 file system from outside the terminal*, unless you know what you are doing. However, if you wanted to, you could *copy* the directory (`QLS-course-materials`) into a Windows location you are familiar with (e.g., a folder in your D drive), to be able to access the contents as you would your usual files for review outside of class, etc.

For the purposes of the lectures and exercises, we will only be working with the `QLS-course-materials` through the WSL2 Ubuntu.