Phys441/PCSE 503 Development Environment Setup

Disclaimer: All of this works on MacOS, Windows, or pretty much any modern Linux distribution (Fedora, Centos, Debian, AlmaLinux, Ubuntu for sure).

Step 1: GitHub.com Setup

Account Setup (if you do not already have a GitHub account)

1. Go to www.github.com
2. Click on "Sign Up"
3. Choose a username (can be anything), use your CNU email address, and choose a password. Verify and create your account.
4. Verify your email address (the email will be sent to you by GitHub).
5. Complete the profile questions

PHYS441 Fork Repository Setup

1. In your GitHub account, in a browser, in the Search bar in the upper left corner, search for "owner:brash99". This will take you to my GitHub repo pages
2. Click on the Repositories tab
3. Find the repository called phys441, and click on that
4. Near the upper right corner of the page, click on "Fork". Accept the default choices, and then click on the green "Create Fork" button to create your own fork of my PHYS 441 repository.
5. Now, you have a copy (fork) or my PHYS 441 repository. See below for how to keep this up-to-date!

Personal PHYS 441 Repository Setup

1. In your GitHub account, in a browser, go to your list of repositories, by clicking on your profile picture in the upper right corner, and choosing "Your repositories".
2. Click on 'New'
3. Choose 'phys441FirstnameLastname' as the Repository name, where you replace 'FirstnameLastname' with your first and last name.
4. Put something appropriate in the Description
5. Choose to add a README file
6. Click on 'Create repository'

Configuring GitHub.com Security

1. In your GitHub account, in a browser, go the Settings, by clicking on your profile picture in the upper right corner, and choosing "Settings".
2. In the long menu shown on the left side, choose "Developer Settings", which is all the way at the bottom of the list of options.
3. Choose "Personal access tokens" -> Tokens(classic) from the new menu on the left.
4. Choose Generate new token -> Generate new token (classic)
5. In the "Note" field, enter phys441
6. Choose "No expiration" for the Expiration time
7. Select ALL of the radio buttons in the "Select scopes" region
8. Click on "Generate token"
9. This will generate a new token that looks like 'ghp_******'. Copy this token to the clipboard!!! This is important, as you are going to need this in the next steps!
10. On your local computer, open up a simple text file, and copy this token into that file. Save the file, giving it some name of your choice.

Step 2a: Download and install Anaconda

Go to https://www.anaconda.com and download the installer for your operating system (MacOS or Linux)

Linux:

```
bash ~/Downloads/Anaconda3-2023.....sh
```

Accept license agreement, accept default installation location, accept all other defaults. This should result in anaconda being installed in /home/<username>/anaconda3

Open a new terminal window, to verify that the shell startup scripts have been modified - you should see a prompt like:

```
(base) [brash@localhost ~]$
```

```
Check the conda and python versions with:
```

```
(base) [brash@localhost ~]$ conda --version
```

```
conda 23.5.2
(base) [brash@localhost ~]$ python --version
Python 3.11.3
```

MacOS and Windows:

```
Run the downloaded installer and follow instructions ...
the main thing is to make sure to choose installation for
JUST YOU!
```

Step 2b: Configure PHYS 441 Conda Environment

```
(Note:  this is the same for all operating systems ... the
commands can be executed from a terminal window on Linux
or MacOS, or from an Anaconda terminal on Windows)
```

Create a new conda environment for PHYS441 (choosing the appropriate python version), with:

```
conda create –name phys441 python=3.11.3
conda activate phys441

You should now see the prompt as:

(phys441) [brash@localhost ~]$
```

Step 3: Open, Configure, and Test PyCharm Professional

Linux:

Download PyCharm for Linux from: https://www.jetbrains.com/pycharm/download/?section=linux

This will download a pycharm-professional-2023.1.4.tar.gz file to your ~/Downloads folder. Install with the following:

```
cd
tar –xvzf ~/Downloads/pycharm–professional–2023.1.4.tar.gz
```

This will create a new pycharm-2023.1.4/ directory under your home directory.

Start up PyCharm for the first time with:

./pycharm-2023.1.4/bin/pycharm.sh

Activate the license by logging into your JetBrains account.

MacOS and Windows:

Install and launch PyCharm Professional from the Anaconda Navigator application

Configuring PyCharm

Step 3a: Clone your GitHub.com forked repository, created above

1. In the "Welcome to PyCharm" window, click on: Get from VCS. Or alternatively, you can choose Git->Clone from the top-level menubar.
2. In the URL area, enter the location of your GitHub.com repo … the format is as follows: https://edwardbrash:ghp_UZJZ2GUZORidXcg3wAfsUP1tUbCRt0Hdt5@github.com/edwardbrash/phys441.git

   Where you need to:

   - Replace 'edwardbrash' with YOUR GitHub.com username, in both places!
   - Replace ghp_UZJZ2GUZORidXcg3wAfsUP1tUbCRt0Hdt5' with YOUR Personal access token
   - If necessary, replace 'phys441' with whatever you called your repository

3. Choose to Trust this repository, and choose to trust all repositories in PyCharmProjects, when asked

Step 3b: Create a new "Hello World" program, for testing

1. Highlight "phys441" in the Project window (left side of the main window)
2. Open a new file with File -> New -> File, and call it 'helloworld.py'
3. You should see that this new file has been opened in the editor window. Add the line: print ("Hello World!") to this new file

Step 3c: Configure Local Interpreter

1. In the bottom left corner of the PyCharm window, you may (probably!) see that it says <No interpreter> … this means we have to tell PyCharm where the Python interpreter is located (the one that you installed using conda above).
2. Click on <No interpreter> -> Add New Interpreter -> Add Local Interpreter
3. Choose Conda Environment
4. Make sure the 'Use existing environment' radio button is checked, and then choose 'phys441' from the pull-down menu. Click OK
5. Be patient … you will see things going on at the bottom of the Pycharm window

Step 3d: Install useful Python packages

1. Click on View -> Tool Windows -> Python Packages
2. In the new bottom left window that opens, in the search area, look for: matplotlib … click on 'Install with conda', in the bottom right window
3. Repeat Step 2 for the following: numpy, pandas, scipy
4. Again .. be patient … wait until all of this finishes!
5. We may need other packages at some point, and any package can be installed in this way! :)

Step 3e: Test the helloworld.py program that you created above by clicking on the green forward arrow in the top menu bar. You should see the following in the output window:

/home/<username>/anaconda3/envs/phys441/bin/python /home/<username>/PyCharmProjects/phys441/helloworld.py
Hello World!

Process finished with exit code 0

Step 3f: Keeping your fork up-to-date

The following will pull all changes from my GitHub.com phys441 repository into your forked copy:

1. Choose Git -> GitHub -> Sync Fork
2. The first time that you do this, you will have to authorize PyCharm (a JetBrains app) to access your GitHub account. Just follow the instructions when this happens, and once you have provided this authorization, it should work seamlessly going forward.

Step 3g: Test the installation of the various module packages in the following way: Open up the Python file called phys441/JupyterNotebooks/Fitting/graph_air_density.py. Click on Run, and verify that the program runs and produces an appropriate looking plot!

Step 4: Install Jupyter Server, and test ROOT installation

Open up the Jupyter Notebook file called phys441/JupyterNotebooks/Fitting/graph_air_density.ipynb

Begin to execute each cell one by one … when you try to execute the first code cell, you will be asked to install the Jupyter package. Choose to do this, and then wait patiently until it is in installed, and the interpreter is re-configured, etc.