

# Introduction to Big Data & Data Engineering Workshop

**3-5 April 2023**

**Instructor:** [Dr Adam Hill](#)

---

I look forward to meeting everyone at our virtual workshop at the beginning of April. The goal of this 3-day workshop is to give you a whistle stop tour of some of the concepts and technologies available to process, analyse, and utilise "big data". Additionally we will explore the challenges and (some) solutions for deploying data science solutions.

The workshop will be part-lecture series and part practical workshop, with the intention of giving you a sufficient introduction to know how to get started with these tools. This is very much a starting-point and the sessions will be quite interactive. So depending on how we get on we may spend more/less time on some of the topics. The core topics will include:

- The Python ecosystem specifically utilising the Pandas library
- The JupyterLab environment
- Docker containers
- SQL and/or NoSQL databases
- Apache Spark

Stretch topics that we can hopefully spend a little bit of time on include:

- Data streaming
- Dashboards and interactive visualisations
- Workflow orchestration

## Pre-course setup

To smooth the start of the course please install the following software on the computer that you will be using on the course. All of this software definitely runs and is available for Mac OSX and Linux (I have tested everything on Ubuntu); if you are using a Windows machine hopefully most things will work, but you may wish to consider setting up a virtual machine with something like [Virtual Box](#) in-order to give yourself a Linux or Mac environment.

You will need to sign-up for a couple of different accounts as well. The software and tools you will need are as follows, detailed instructions are later in this document:

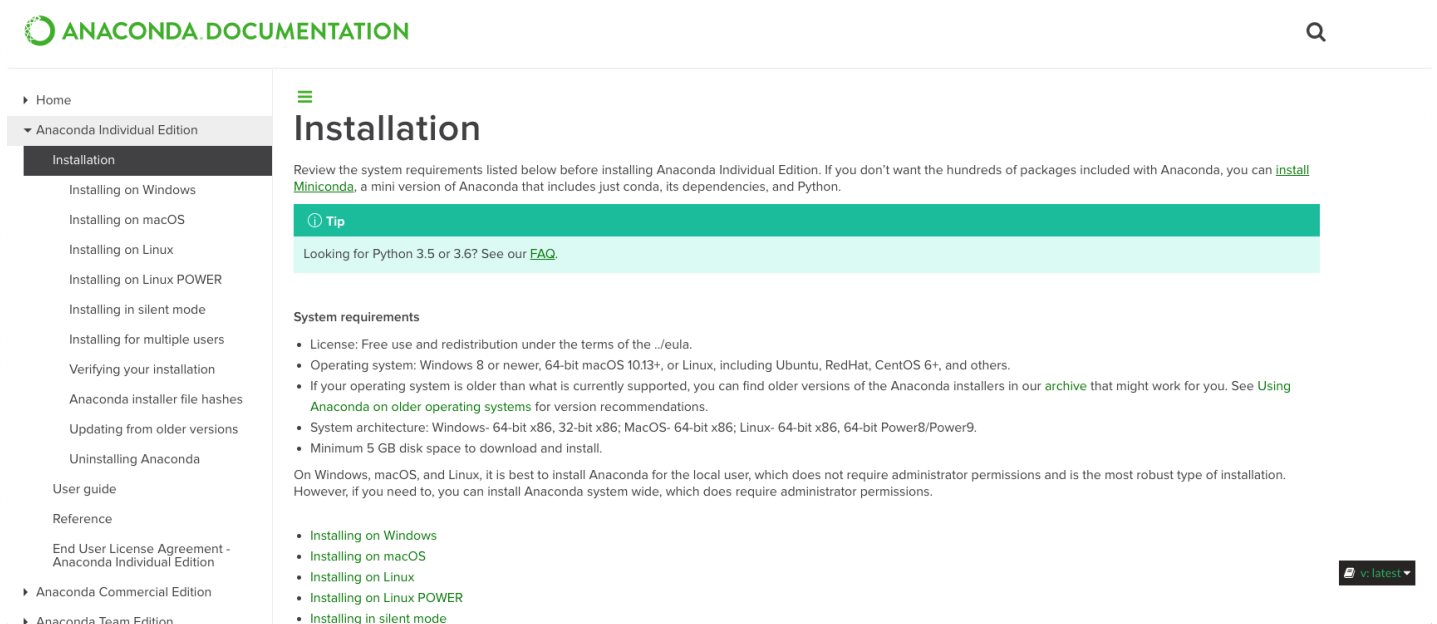
- Anaconda, <https://docs.anaconda.com/anaconda/install/>
- Docker, <https://docs.docker.com/get-docker/>
- Register for Databricks Community Edition: [Databricks registration](#)
- Setup a [GitHub](#) account
- Setup a [Docker Hub](#) account
- Make sure you have a text editor or IDE installed that you like to use; I'm a fan of [Visual Studio Code](#) and [Atom](#)

If you already have any of this software please check to see that any updates that are due have been installed.

## Anaconda Installation

We will be using a lot of Python in this workshop and one of the main ways we will interact with Python is via the Anaconda environment. I will be encouraging people to create dedicated Python environments during the workshop so that we know that we are all using compatible and equivalent tools.

Go to <https://docs.anaconda.com/anaconda/install/> to find installation instructions for the latest version of Anaconda that is appropriate for your operating system.



The screenshot shows the Anaconda Documentation website. The header includes the Anaconda logo and the text "ANACONDA DOCUMENTATION". A search icon is in the top right. The left sidebar shows a navigation menu with "Home" at the top, followed by "Anaconda Individual Edition" (which is expanded to show "Installation", "User guide", "Reference", and "End User License Agreement - Anaconda Individual Edition"). Below this are "Anaconda Commercial Edition" and "Anaconda Team Edition". The main content area is titled "Installation" and contains a tip box about Miniconda, system requirements, and links to installation guides for various operating systems.

**ANACONDA DOCUMENTATION**

Q

Home

Anaconda Individual Edition

Installation

Installing on Windows

Installing on macOS

Installing on Linux

Installing on Linux POWER

Installing in silent mode

Installing for multiple users

Verifying your installation

Anaconda installer file hashes

Updating from older versions

Uninstalling Anaconda

User guide

Reference

End User License Agreement - Anaconda Individual Edition

Anaconda Commercial Edition

Anaconda Team Edition

### Installation

Review the system requirements listed below before installing Anaconda Individual Edition. If you don't want the hundreds of packages included with Anaconda, you can [install Miniconda](#), a mini version of Anaconda that includes just conda, its dependencies, and Python.

**Tip**

Looking for Python 3.5 or 3.6? See our [FAQ](#).

**System requirements**

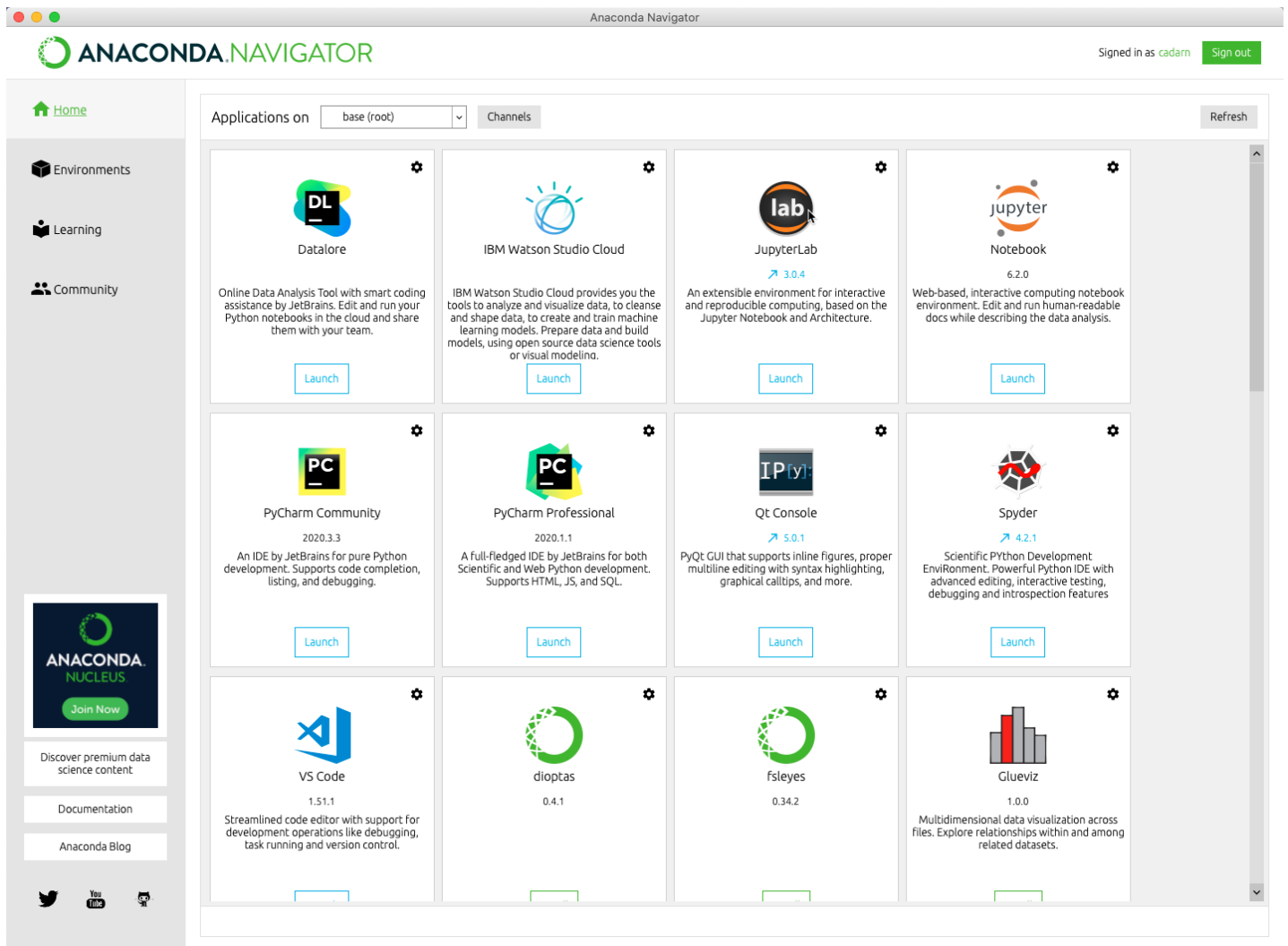
- License: Free use and redistribution under the terms of the [./eula](#).
- Operating system: Windows 8 or newer, 64-bit macOS 10.13+, or Linux, including Ubuntu, RedHat, CentOS 6+, and others.
- If your operating system is older than what is currently supported, you can find older versions of the Anaconda installers in our [archive](#) that might work for you. See [Using Anaconda on older operating systems](#) for version recommendations.
- System architecture: Windows- 64-bit x86, 32-bit x86; MacOS- 64-bit x86; Linux- 64-bit x86, 64-bit Power8/Power9.
- Minimum 5 GB disk space to download and install.

On Windows, macOS, and Linux, it is best to install Anaconda for the local user, which does not require administrator permissions and is the most robust type of installation. However, if you need to, you can install Anaconda system wide, which does require administrator permissions.

- [Installing on Windows](#)
- [Installing on macOS](#)
- [Installing on Linux](#)
- [Installing on Linux POWER](#)
- [Installing in silent mode](#)

v. latest

Once installed open new terminal window. If you type `which conda` you should see something like `/Users/myname/opt/anaconda3/bin/conda`. You will likely have installed the Anaconda Navigator GUI as well as this should appear in the Applications (Mac & Linux) or Start (Windows) menu. We will be using the terminal during the course but feel free to familiarise yourself with the GUI if it is your preferred way to manage Anaconda.



## Docker Installation

Docker will be the tool we will use to make it easy for us to install and test new technologies and software packages.

To install the software go to the [Get Started with Docker](#) page there should be a straight forward link to download the Docker Desktop software installer tool for Mac or Windows; for Linux there are different instructions depending on your Linux version (I tested the Ubuntu instructions).

# Get Started with Docker

We have a complete container solution for you – no matter who you are and where you are on your containerization journey.



## Docker Desktop

Developer productivity tools and a local Kubernetes environment.

Download for Mac



## Docker Hub

Cloud-based application registry and development team collaboration services.

Signup



## Play with Docker

Cloud-based docker environment to try out docker and learn the ropes.

Play with Docker

## Linux

For linux there are are few more hoops to jump through to get Docker installed. The link above should guide you to the instructions for your "Flavour" of linux here I will summarise the case for Ubuntu.

The detailed instructions are [here](#) and follow all the instructions in the section labelled "**Install using the repository**", we will only be using the **stable** release so ignore anything that is only needed for the 'nightly' or 'test' versions!

Once complete you should be able to run the following command in the terminal:

```
sudo docker run hello-world
```

if you see the following then Docker has installed successfully

```
Hello from Docker!  
This message shows that your installation appears to be working correctly.  
  
To generate this message, Docker took the following steps:  
1. The Docker client contacted the Docker daemon.  
2. The Docker daemon pulled the "hello-world" image from the Docker Hub.  
   (amd64)  
3. The Docker daemon created a new container from that image which runs the  
   executable that produces the output you are currently reading.  
4. The Docker daemon streamed that output to the Docker client, which sent it  
   to your terminal.  
  
To try something more ambitious, you can run an Ubuntu container with:  
$ docker run -it ubuntu bash  
  
Share images, automate workflows, and more with a free Docker ID:  
https://hub.docker.com/  
  
For more examples and ideas, visit:  
https://docs.docker.com/get-started/
```

## Post installation steps

After completing the initial Linux install there are a few tidying up steps to be followed so that Docker doesn't need sudo permissions. The instructions are [here](#) and should be followed by all Linux users; you only need to complete the instructions in the sections on:

- "Manage Docker as a non-root user"
- "Configure Docker to start on boot"

## Testing for all OS installations

In a terminal everyone should now be able to execute the command `docker run hello-world` and get an output similar to that above.

As an extra test first run `docker image ls`, you should see a single image listed called "hello-world".

If you now run `docker pull mongo` you should see a number of progress bars while Docker downloads the latest image for the MongoDB database. If you run `docker image ls` again you should see an additional image listed.

## Docker Compose - Stretch exercise

---

This isn't essential pre-course but if you have the time it would be good to try and become a little familiar with Docker Compose as we may end up using it to support some elements of the workshop.

Docker Compose was installed when you installed Docker so there no additional installation is required. Try and follow along with the [Getting Started](#) exercise. Don't worry if you get stuck we will come back to it in the workshop.

# Databricks Community Edition

As part of the workshop we will explore the use of Apache Spark as part of this we will do some work in the cloud using the Databricks Community Edition. This requires an account so please register via this [link](#).

After submitting your details on the registration page a new web page will open that looks like,

Try Databricks

AN OPEN AND UNIFIED DATA ANALYTICS PLATFORM FOR DATA ENGINEERING, MACHINE LEARNING, AND ANALYTICS

From the original creators of Apache Spark™, Delta Lake, MLflow, and Koalas

Select a platform

### DATABRICKS PLATFORM - FREE TRIAL

For businesses

- Collaborative environment for Data teams to build solutions together
- Unlimited clusters that can scale to any size, processing data in your own account
- Job scheduler to execute jobs for production pipelines
- Fully collaborative notebooks with multi-language support, dashboards, REST APIs
- Native integration with the most popular ML frameworks (scikit-learn, TensorFlow, Keras,...), Apache Spark™, Delta Lake, and MLflow
- Advanced security, role-based access controls, and audit logs
- Single Sign On support

### COMMUNITY EDITION

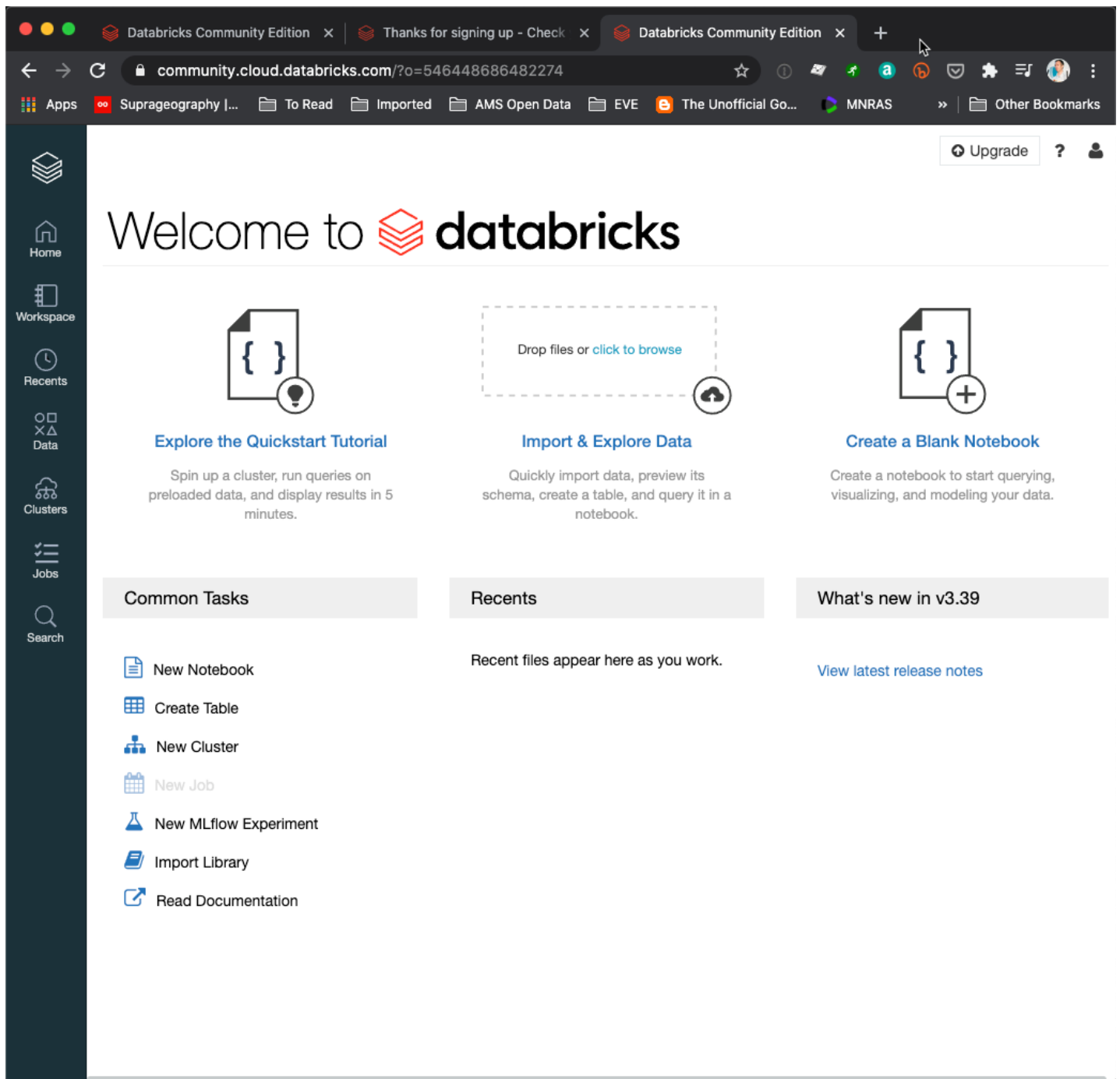
For students and educational institutions

- Single cluster limited to 6GB and no worker nodes
- Basic notebooks without collaboration
- Limited to 3 max users
- Public environment to share your work

**GET STARTED**

**Make sure to click on the link on the RIGHT hand side to "GET STARTED" with the Community Edition. DO NOT opt for the "Free Trial"!**

You will then receive an email confirming your registration and that link should lead you to the Community Edition portal that should look like,



Feel free to try the "Explore the Quickstart Tutorial" notebook that should appear in the upper left of the portal, if you want to. We will be taking a deeper look during the workshop.

## Register for a GitHub account

If you don't already have a GitHub account or the Github Student Pack then go to <https://education.github.com> to register.

## Register for a Docker Hub account

Similarly, if you do not already have a Docker Hub account then go to <https://hub.docker.com/> to register for

free.

# **Congratulations you're ready!**

Thank you for reading through everything and preparing for the workshop, see you soon.