TensorFlow+Docker Setup Guide for Windows

Hi, guys! I'm Justin. I drafted this quick setup guide to help you guys prepare for our upcoming workshop on TensorFlow. I encourage you all to do this as early as possible so we can immediately start with our session.

There are actually many ways to install Tensorflow, but for purpose of the workshop, this guide will only show you how to install TensorFlow using Docker.

What is this thing called Docker, you might wonder? Docker is a software container tool that will do most of the heavy-lifting and handling of dependencies for us. Any piece of software along with everything it needs to run are packaged into an 'isolated container'. Using that container or image on Docker, the software is guaranteed to run the same way every time, and on any machine.

Note:

This guide will cover the Docker installation process for Windows only. That means the same instructions might not necessarily work for Mac or Linux. So if you do operate on a different machine, please follow the specific guides for installing Docker on your OS from the respective links below. Once you have installed Docker successfully, you can start to follow along with the guide from `Step 2`.

- Docker for Mac OS X: https://docs.docker.com/docker-for-mac/ (https://docs.docker.com/docker-for-mac/)
- Docker for Linux: https://www.tensorflow.org/versions/r0.12/get_started/os_setup.html#docker-installation (https://www.tensorflow.org/versions/r0.12/get_started/os_setup.html#docker-installation)
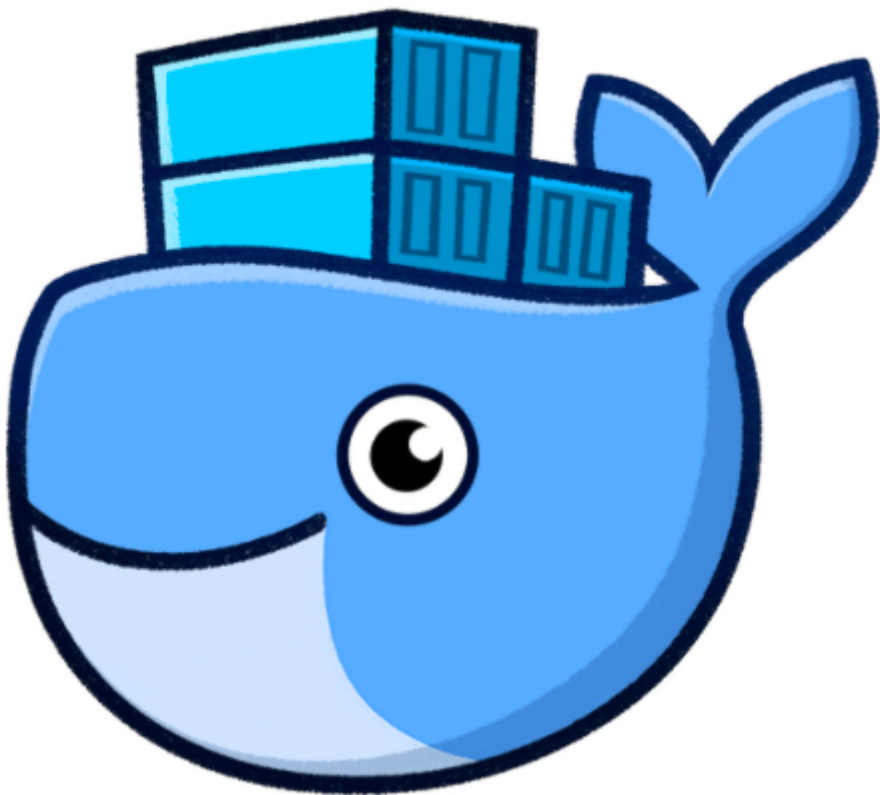
[WARNING] Oh, one more thing! Before you scroll down any further, Docker actually requires a 64-bit OS running in your machine.

Awww -- sad, I know. :(

But that's okay, you can still share with a classmate! :)

Start Here!

So let's get right to it! To make sure you guys are all set up, simply follow the 3 main steps below...

Step 1. Download & Install Docker

Step 1.1. If you already have Docker installed, you can skip ahead to **Step 2 (Run Docker's Terminal & Install Image)**. Otherwise, go download Docker for your Windows version from the links below:

- Windows 7 (and higher): https://www.docker.com/products/docker-toolbox (https://www.docker.com/products/docker-toolbox) (FYI, this is what I use for my Win7)
- Windows 10: https://store.docker.com/editions/community/docker-ce-desktop-windows (https://store.docker.com/editions/community/docker-ce-desktop-windows)

> ### Note:
>
> If you happen to have Git in your machine already, I suggest you uncheck the Git checkbox during the Docker installation and hopefully the installer should be able to locate and map Docker to your existing Git installation. If you don't do this, your machine would re-install Git and it might result to a path conflict.
>
> > *Tip: Not sure if you have Git installed already?* Open up your Windows terminal, Command Prompt, and run the command, `git --version`.
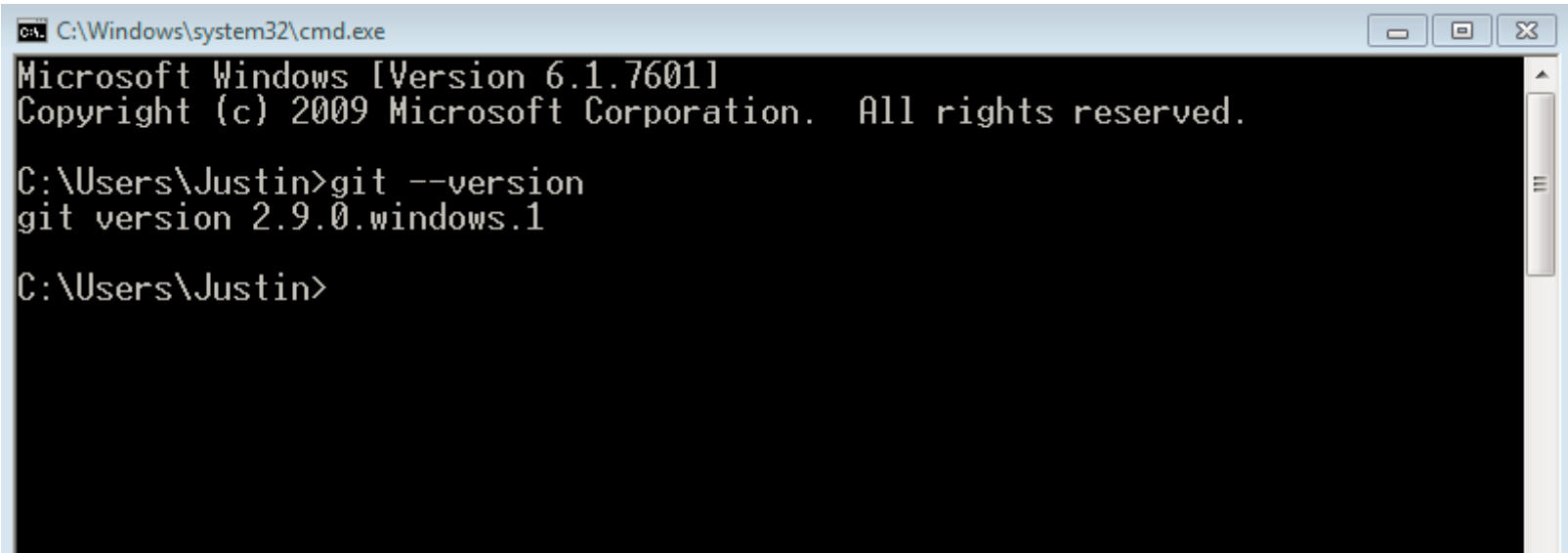
Step 2. Run Docker's Terminal & Install Image

Once you have installed Docker successfully, the next step is to download the TensorFlow image we will be using.

- **Step 2.1.** Go ahead and run the **Docker Quickstart Terminal** from your All Programs list.
- **Step 2.2. (Optional)** You can test it by entering the command `docker run hello-world`. If it looks like the image below, then it works!
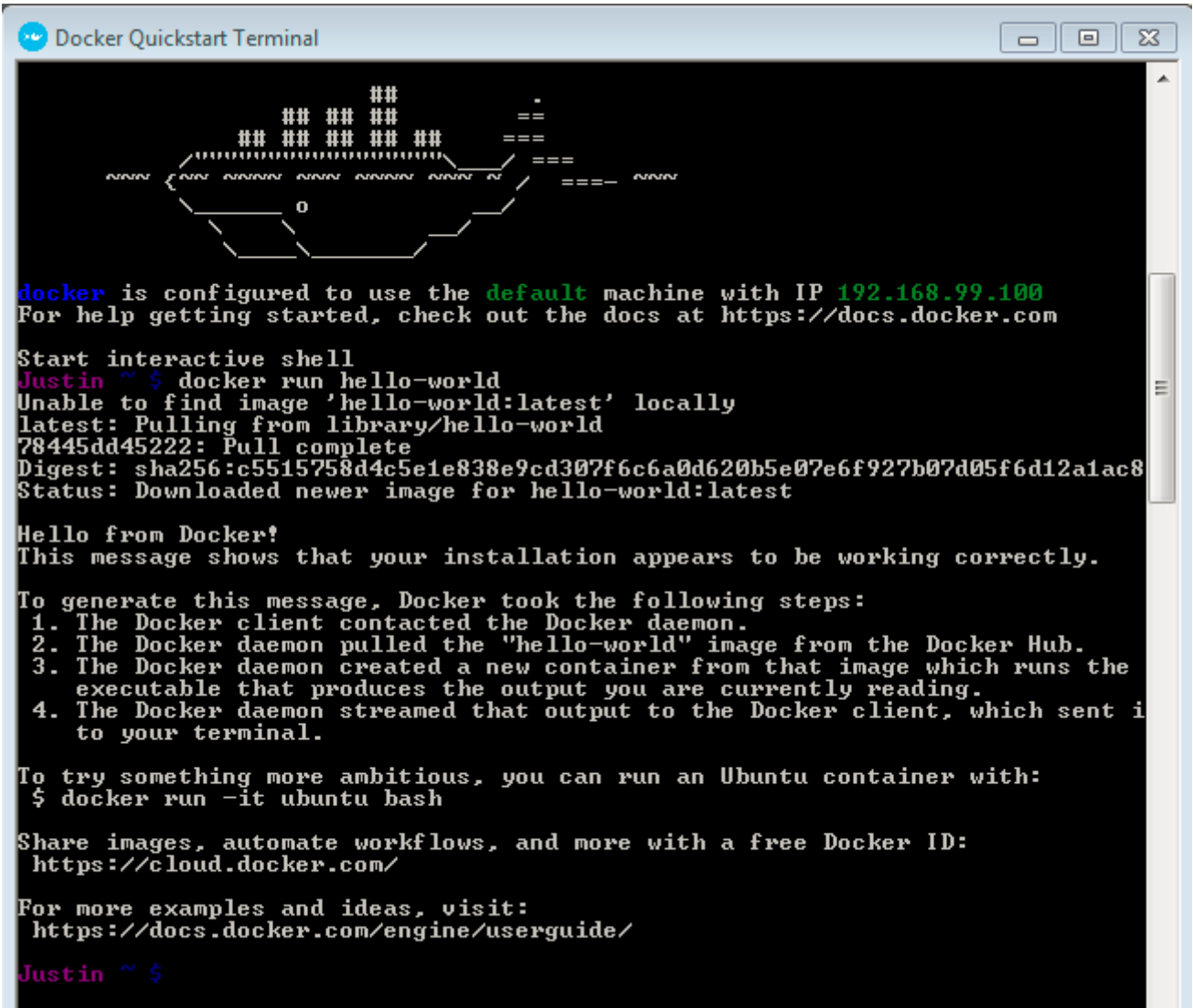
- **Step 2.3.** To download & install the TensorFlow image from the Docker terminal, enter the command: `docker run -it gcr.io/tensorflow/tensorflow:latest-devel`
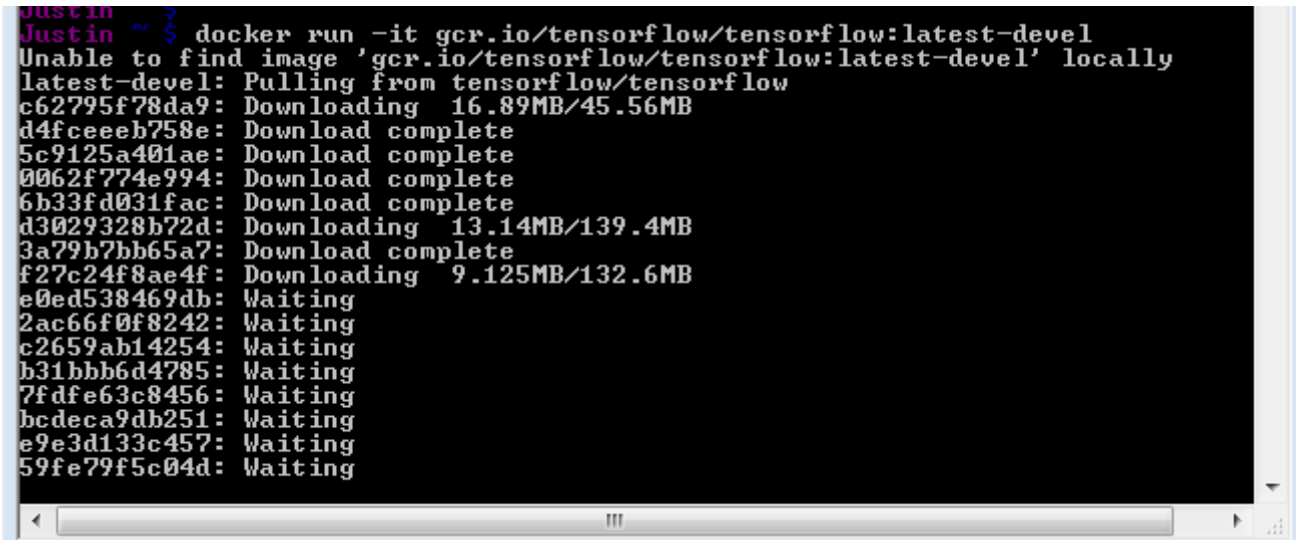
Note:

This might take a while depending on your Internet connection. You might want to grab a cup of coffee or tea for this. Maybe go read a book or check your Facebook? Or you could just watch the download process all the way... it's up to you (:

Additional Note:

You might run into a problem with Git, like I did. If Docker still asks where your Git-bash (`bash.exe`) application path is, even after installation, then just help locate it manually and it should work just fine afterwards.

> *Tip: Don't know where your `bash.exe` path is?*
>
> - Go to Windows Task Manager's Processes tab.
> - Locate `bash.exe` in the list.
> - Right-click on it, then select 'Open File Location'. That should be it.

Try running your Docker Quickstart Terminal again and give it your 'newfound path'. If you want, you can also do this manually or check the current path by opening up the Docker Quickstart Terminal Properties (see an example output below). Then once you're able to load it, repeat **Step 2.3.**.
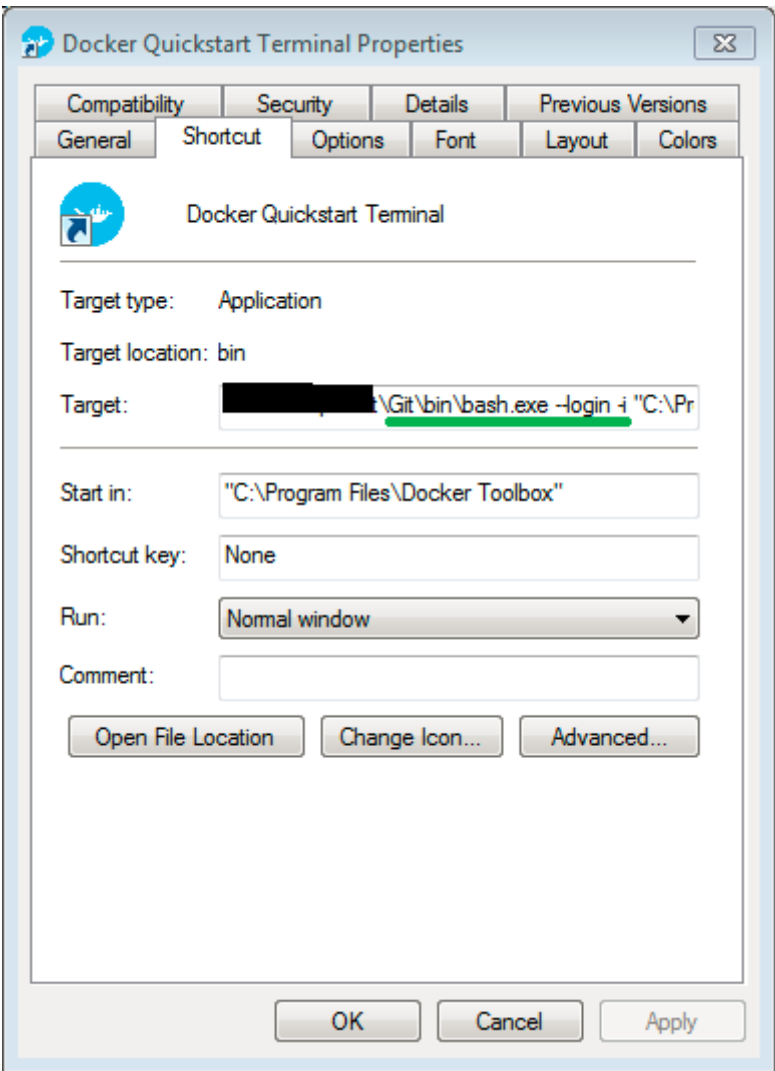
- **Step 2.4.** Let's just do a Sanity Check. To see if you have installed the TensorFlow image successfully, let's run some TensorFlow code with Python on the Docker Terminal. We can launch the Python interpreter by entering **python**.
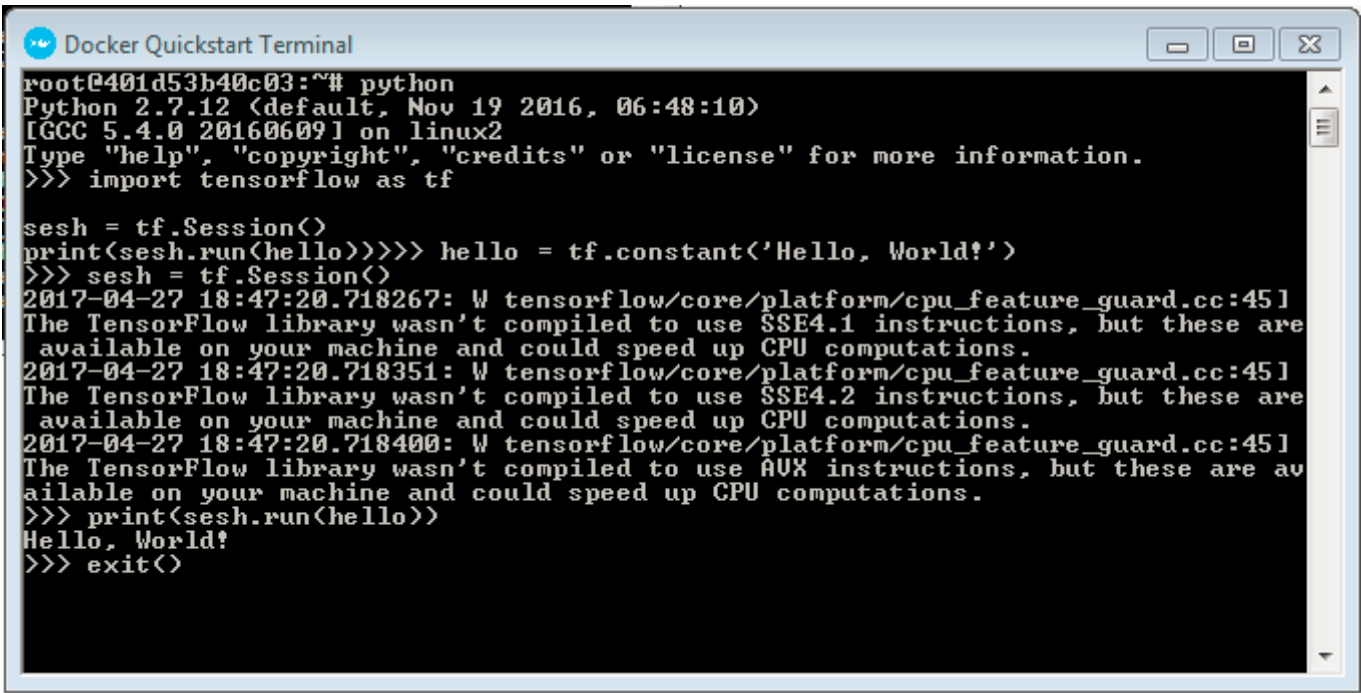

```
# type out or copy-paste this block of code on your terminal and run it
import tensorflow as tf
hello = tf.constant('Hello, World!')
sesh = tf.Session()
print(sesh.run(hello))

# to exit python, enter the line below
exit()
```

If you see the "Hello, World!" output, then you know the image actually works! See the beauty of Docker right there? You were able to use TensorFlow already without having to manually install Python, the TensorFlow library and the other libraries. They are all packaged into the TensorFlow image which Docker handles very nicely so you can focus on what matters and start building.

(BTW, don't mind the warnings you see. They're not harmful.)

So yay! -- on to the last step...

Step 3. Download Dataset

Before we can do some actual TensorFlow work, we will need some data (images) first.

- **Step 3.1** Exit from your Docker Terminal (Press Ctrl+D or type **exit**).
- **Step 3.2** Navigate to your Users directory via Command Prompt: `cd C:/Users/_your-username_`
- **Step 3.3** Create a directory with the name tf_files: `mkdir tf_files`
- **Step 3.4** To download the images in this directory, enter: `curl -O http://download.tensorflow.org/example_images/flower_photos.tgz`
- **Step 3.5** To unpack the downloaded file, enter: `tar xzf flower_photos.tgz` (Or unpack/unzip them as you normally would in Windows Explorer)

In the end, your directory should contain the unpacked folder (`flower_photos`) and a intially compressed tgz file (`flower_photos.tgz`) just like the image below.
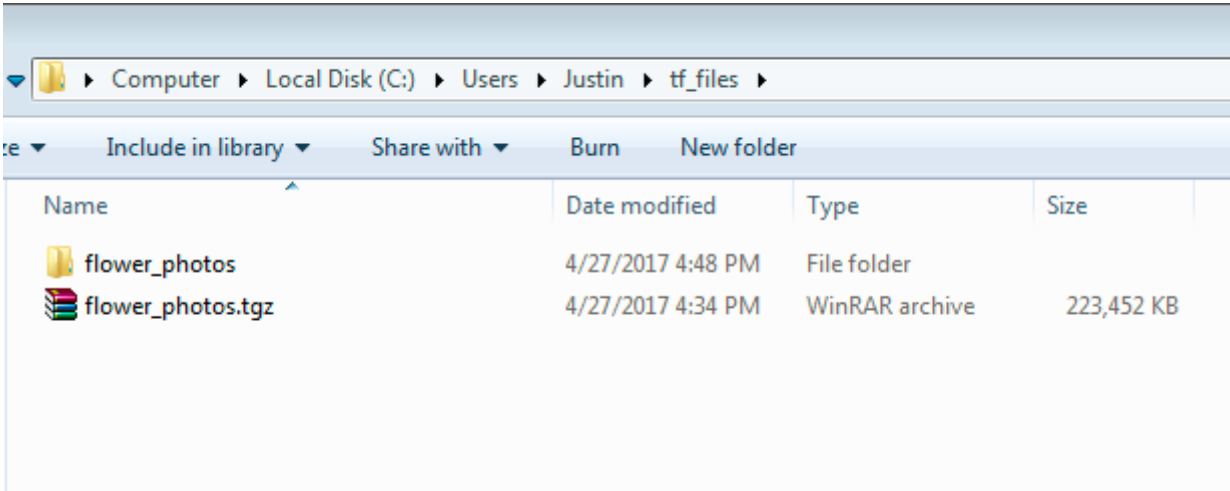
Et, voila!

That's it! Hopefully, you already have Docker, the TensorFlow image, and the flower_photos dataset at this point.

If you do, then you are now ready for the workshop! We'll be doing some cool stuff with all of it. Using TensorFlow, we will learn how to train the machine to automatically recognize photos it hasn't seen before, all by itself. In other words, we will specifically be teaching an AI! :)

See you soon!

> P.S. If you still run into any trouble, do feel free to let us know thru Sir Ilao. :)