**DNN Workshop**

**Objectives**

In this hands-on lab, you will learn how to:

* Create a Deep Learning Virtual Machine (DLVM)
* Connect to the DLVM
* Download the workshop content from GitHub
* Setup the Anaconda environment
* Access lab notebooks from jupyter notebooks

**Prerequisites**

The following are required to complete this hands-on lab:

* An active Microsoft Azure subscription. If you don't have one, [sign up for a free trial](http://aka.ms/WATK-FreeTrial).
* A Deep Learning VM uses GPUs which – if your azure subscription is new – may be set to a quota of ‘0’. To request a quota increase, [open a Support Request from your azure portal](https://docs.microsoft.com/en-us/azure/azure-supportability/resource-manager-core-quotas-request).

**Exercises**

This hands-on lab includes the following exercises:

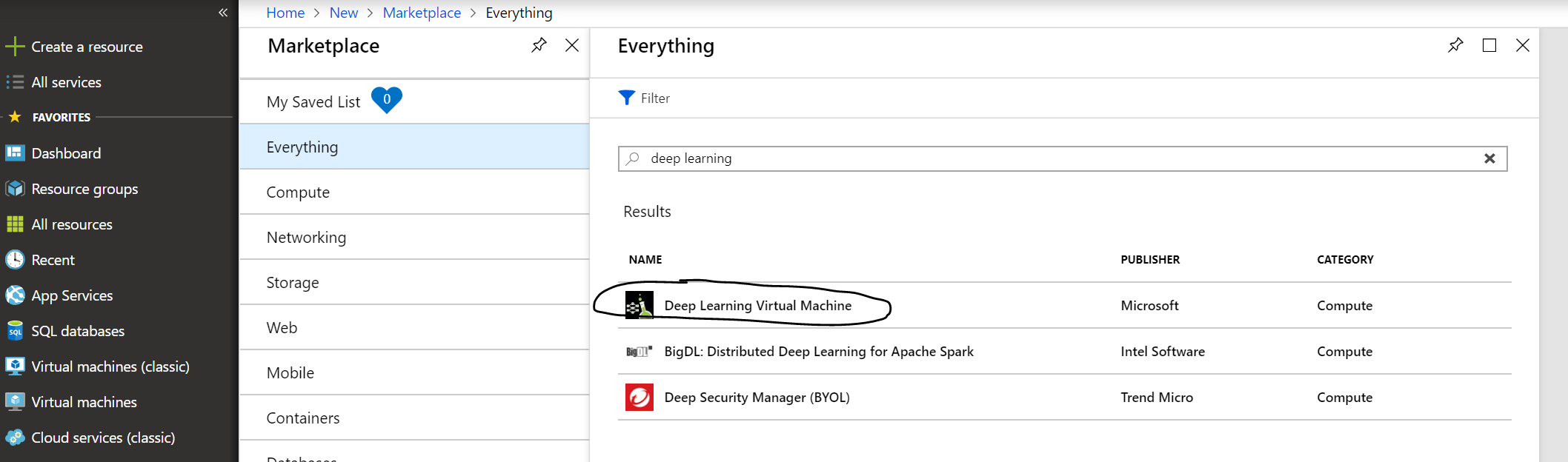
* [Exercise 1: Create an Deep Learning Virtual Machine (DLVM)](#Excersize1)
* [Exercise 2: Connect to the DLVM](#Excersize2)
* [Exercise 3: Clone (download) the workshop content from GitHub repo](#Excersize3)
* [Exercise 4: Setup Anaconda environment](#Excersize4)
* [Exercise 5: Install jupyter notebook and connect to the labs](#Excersize5)

Estimated time to complete this lab: **60-90 mins.**

**Exercise 1: Create a Deep Learning VM (DLVM)**

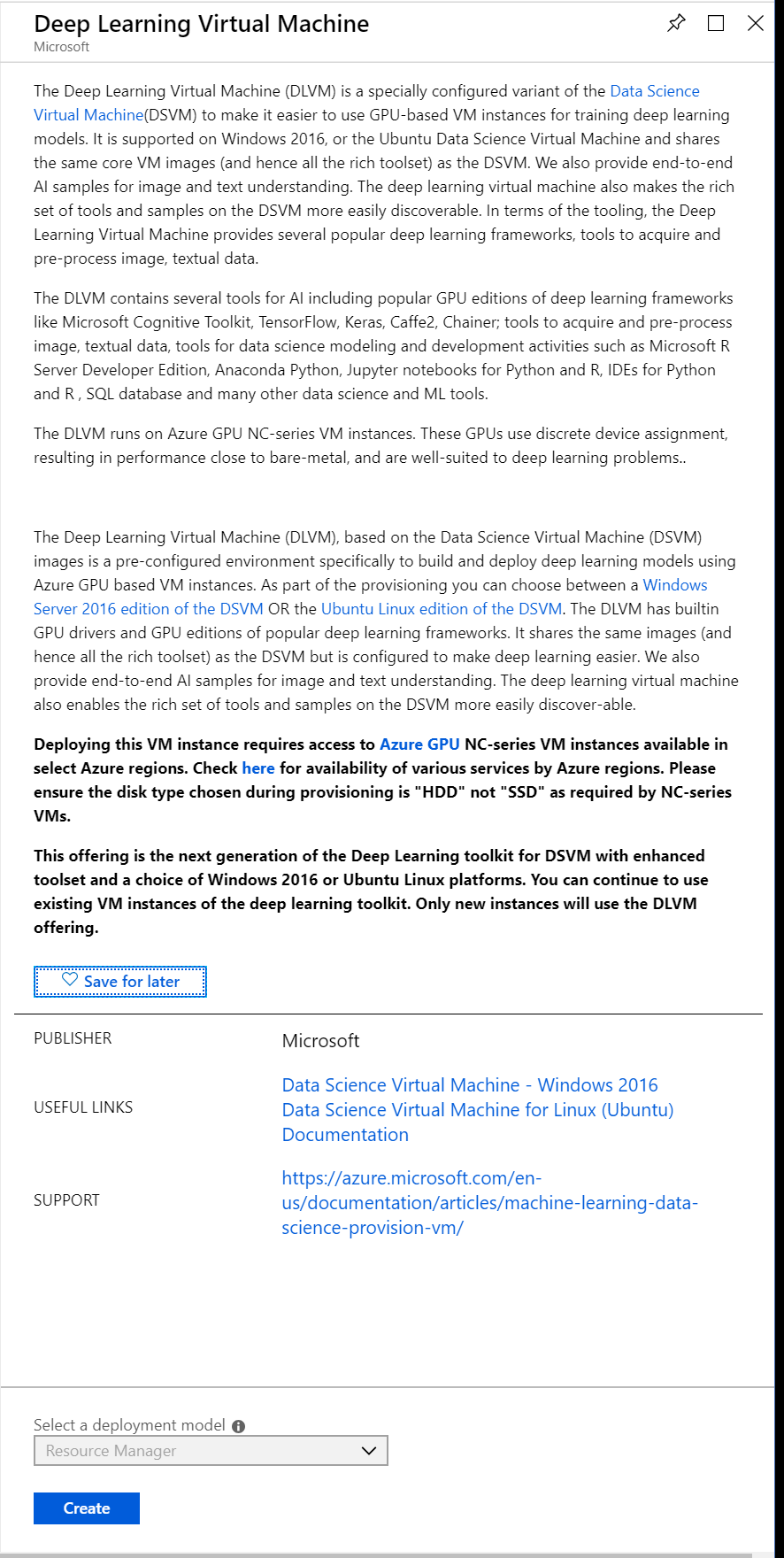
The Deep Learning Virtual Machine is a virtual-machine image that makes it easy to get started with data science. Multiple tools are already built, installed, and configured to get you up and running quickly. The NVIDIA GPU driver, [NVIDIA CUDA](https://developer.nvidia.com/cuda-downloads), and [NVIDIA CUDA Deep Neural Network](https://developer.nvidia.com/cudnn) library (cuDNN) are also included, as are [Jupyter](http://jupyter.org/) and several sample Jupyter notebooks. All installed frameworks are GPU-enabled but work on CPUs as well. In this exercise, you will create an instance of the Data Science Virtual Machine for Windows in Azure.

1. Open the [Azure Portal](https://portal.azure.com) in your browser. If asked to log in, do so using your Azure account.
2. Click **+ Create a resource** in the menu on the left side of the portal, and then type "deep learning” (without quotation marks) into the search box. Select **Deep Learning Virtual Machine** from the results list.



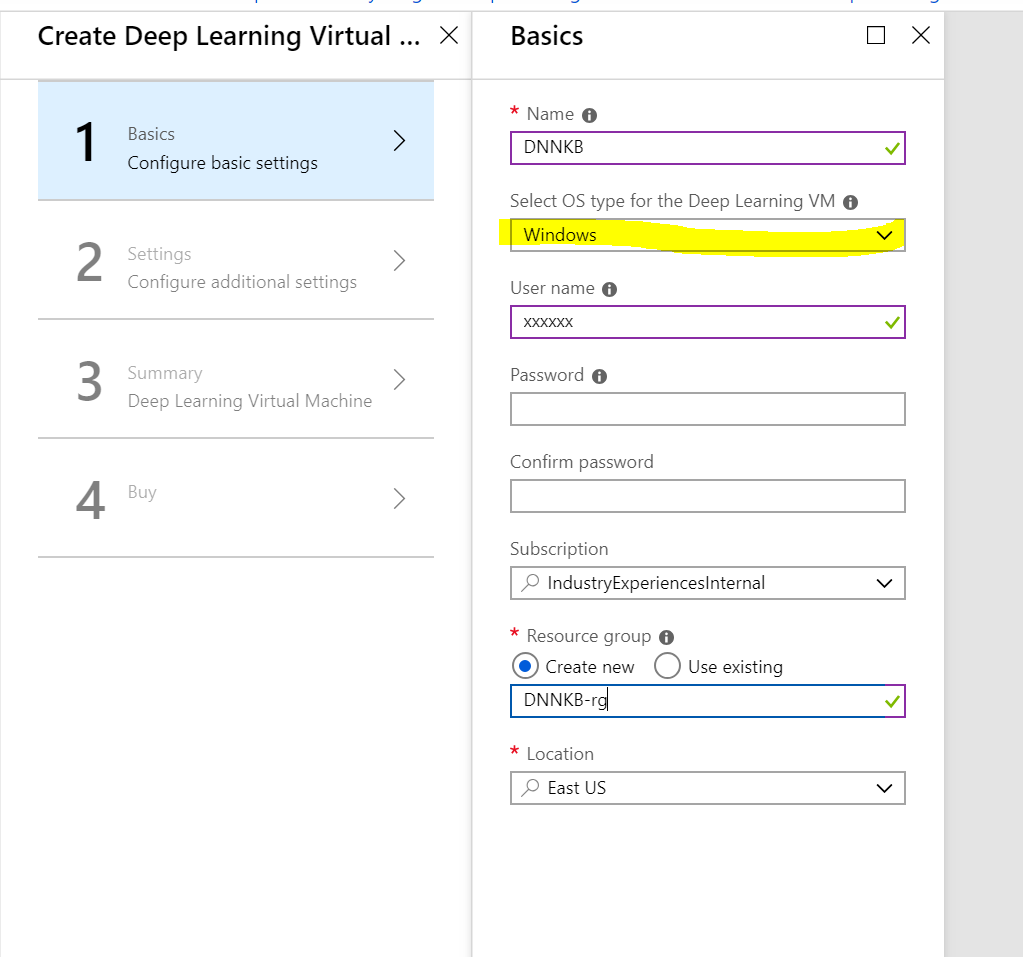
*Finding the Deep Learning VM*

1. Take a moment to review the list of tools included in the VM. Then click **Create**.



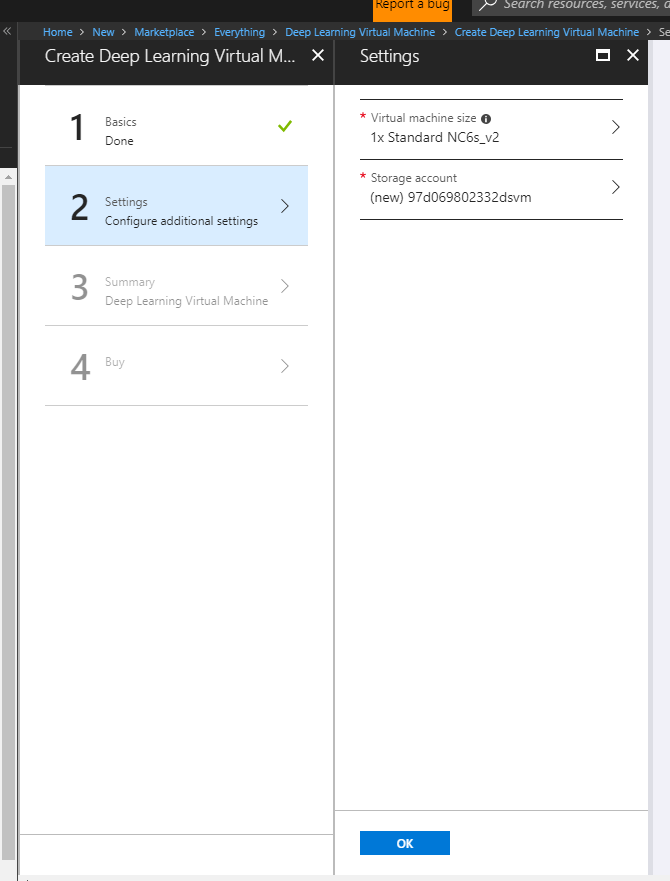
*Creating a Data Science VM*

1. Enter a name for the virtual machine, select OS type to be Windows, and enter a user name for logging into it. Set **Authentication type** to **Password** and enter a password. *Be sure to remember or note the user name and password that you enter*, because you will need them to access the VM. Select **Create new** under **Resource group** and enter a resource-group name such as "data-science-rg." Select the **Location** nearest you, and then click **OK**.



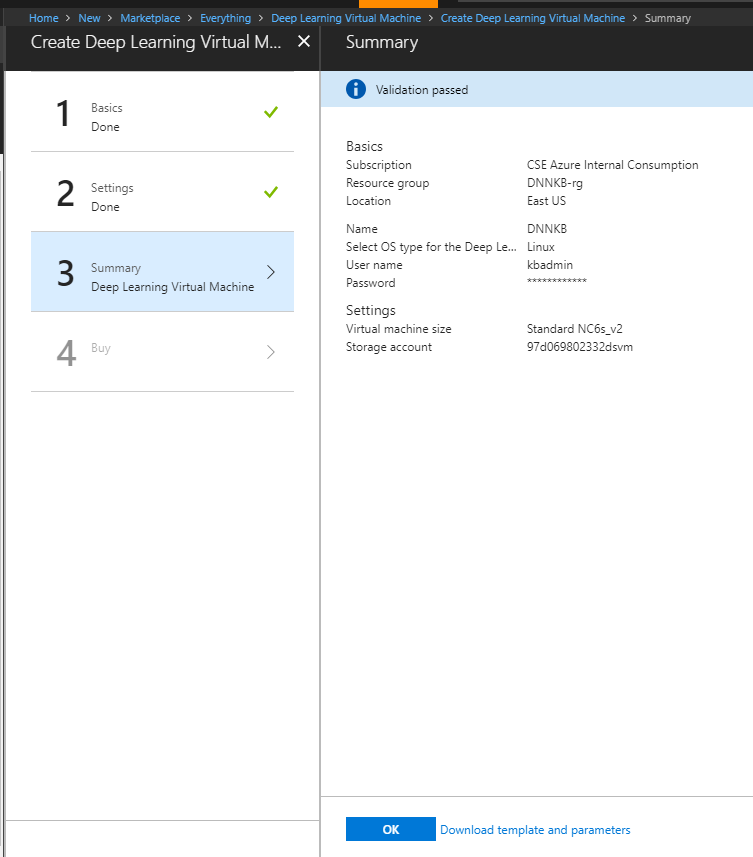
*Enter basic settings*

1. Accept the defaults for the virtual machine size and storage account. Then click the **OK** button at the bottom of the blade.

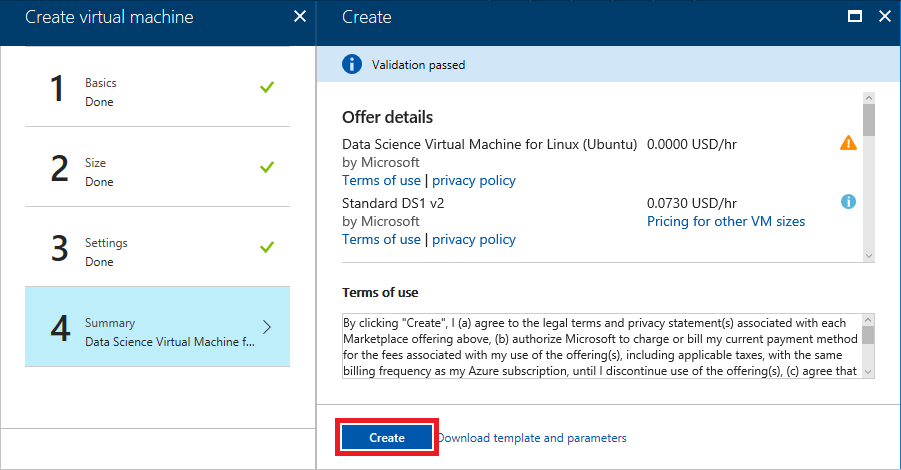


*Choosing a VM size*

1. Click **OK** once the configuration validation has passed:

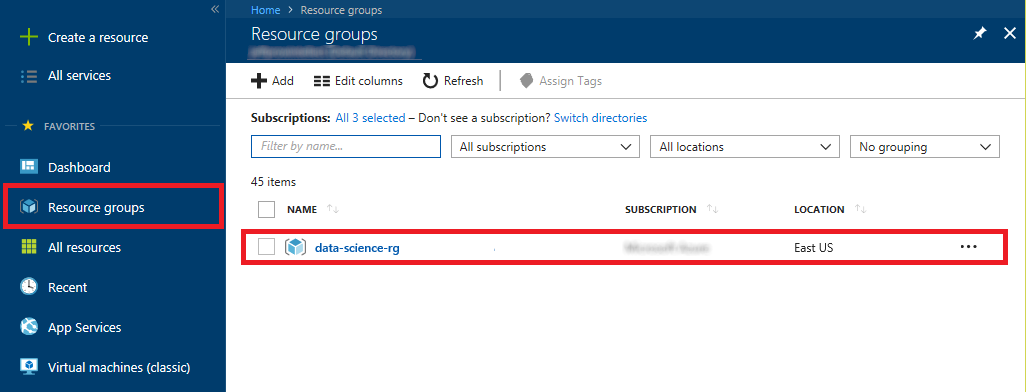


1. In the "Create" blade, take a moment to review the options you selected for the VM, and click **Create** to start the VM creation process.

[](https://github.com/Microsoft/computerscience/blob/6fe13af1b7c684b47db404ce9fd06c4542bb9147/Labs/Deep%20Learning/200%20-%20Machine%20Learning%20in%20Python/1%20-%20Ingest/Images/create-data-science-vm-4.png)

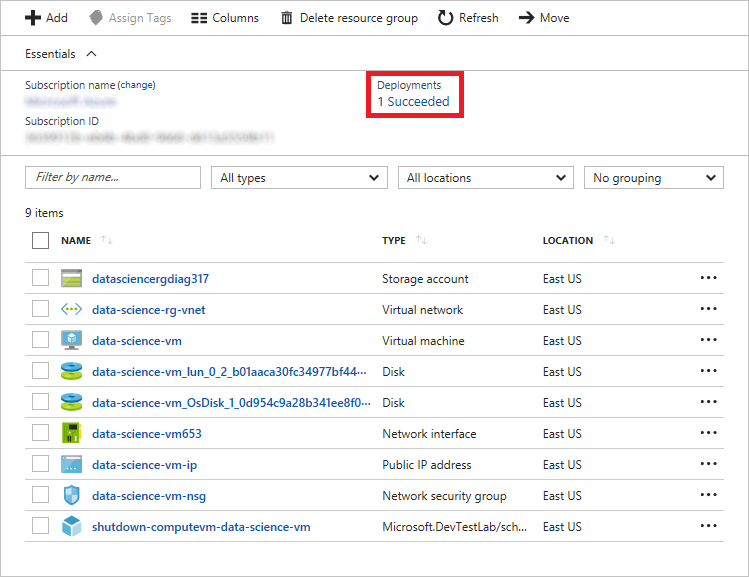
*Creating the VM*

1. Click **Resource groups** in the menu on the left side of the portal. Then click the resource group whose name you specified in Step 4.

[](https://github.com/Microsoft/computerscience/blob/6fe13af1b7c684b47db404ce9fd06c4542bb9147/Labs/Deep%20Learning/200%20-%20Machine%20Learning%20in%20Python/1%20-%20Ingest/Images/open-resource-group.png)

*Opening the resource group*

1. Wait until "Deploying" changes to "Succeeded" indicating that deployment has completed. Deployment typically takes 5 minutes or less. Periodically click **Refresh** at the top of the blade to refresh the deployment status.

[](https://github.com/Microsoft/computerscience/blob/6fe13af1b7c684b47db404ce9fd06c4542bb9147/Labs/Deep%20Learning/200%20-%20Machine%20Learning%20in%20Python/1%20-%20Ingest/Images/deployment-succeeded.png)

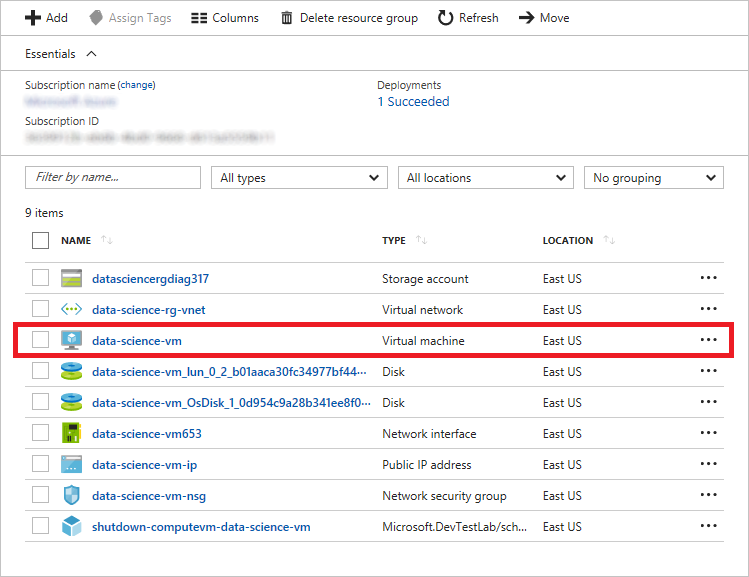
*Monitoring the deployment status*

The VM has been created. The next step is to connect remotely so you can work with the VM's Windows desktop.

**Exercise 2: Connect to the DLVM**

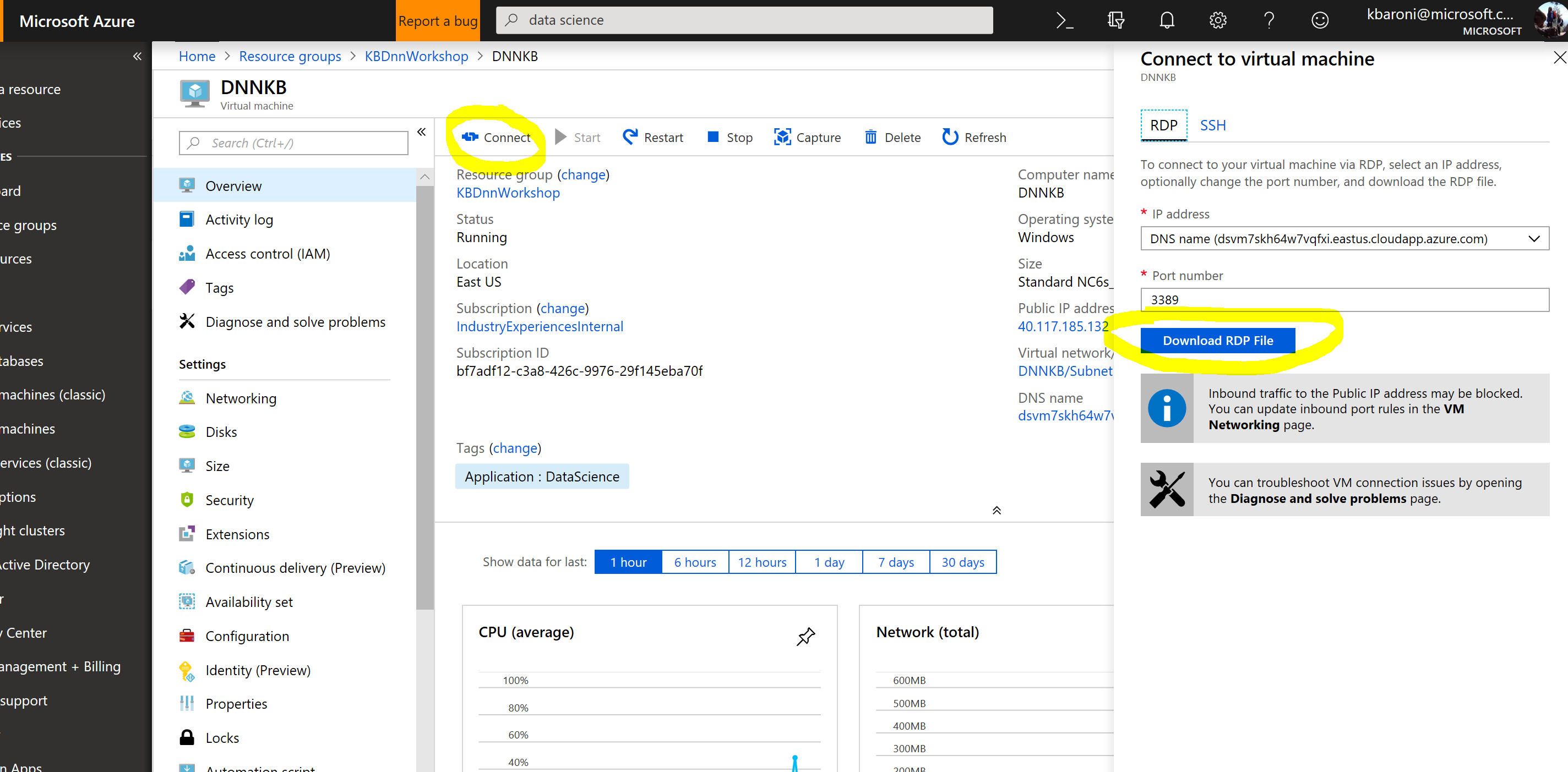
In this exercise, you will connect remotely to the Windows desktop in the VM that you created in the previous exercise.

Return to the Azure Portal and the blade for the resource group containing the Data Science VM. Then click the VM.

[](https://github.com/Microsoft/computerscience/blob/6fe13af1b7c684b47db404ce9fd06c4542bb9147/Labs/Deep%20Learning/200%20-%20Machine%20Learning%20in%20Python/1%20-%20Ingest/Images/open-data-science-vm.png)

*Opening the Data Science VM*

1. Click the Connect button at the top of the screen to download the RDP file and save it to your local machine:

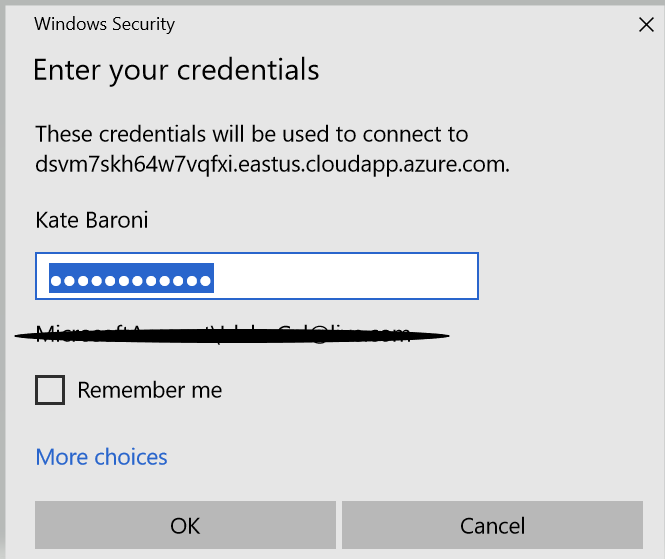


*Download the RDP file*

1. Update the inbound port rules in the VM Networking page for Port 3389:

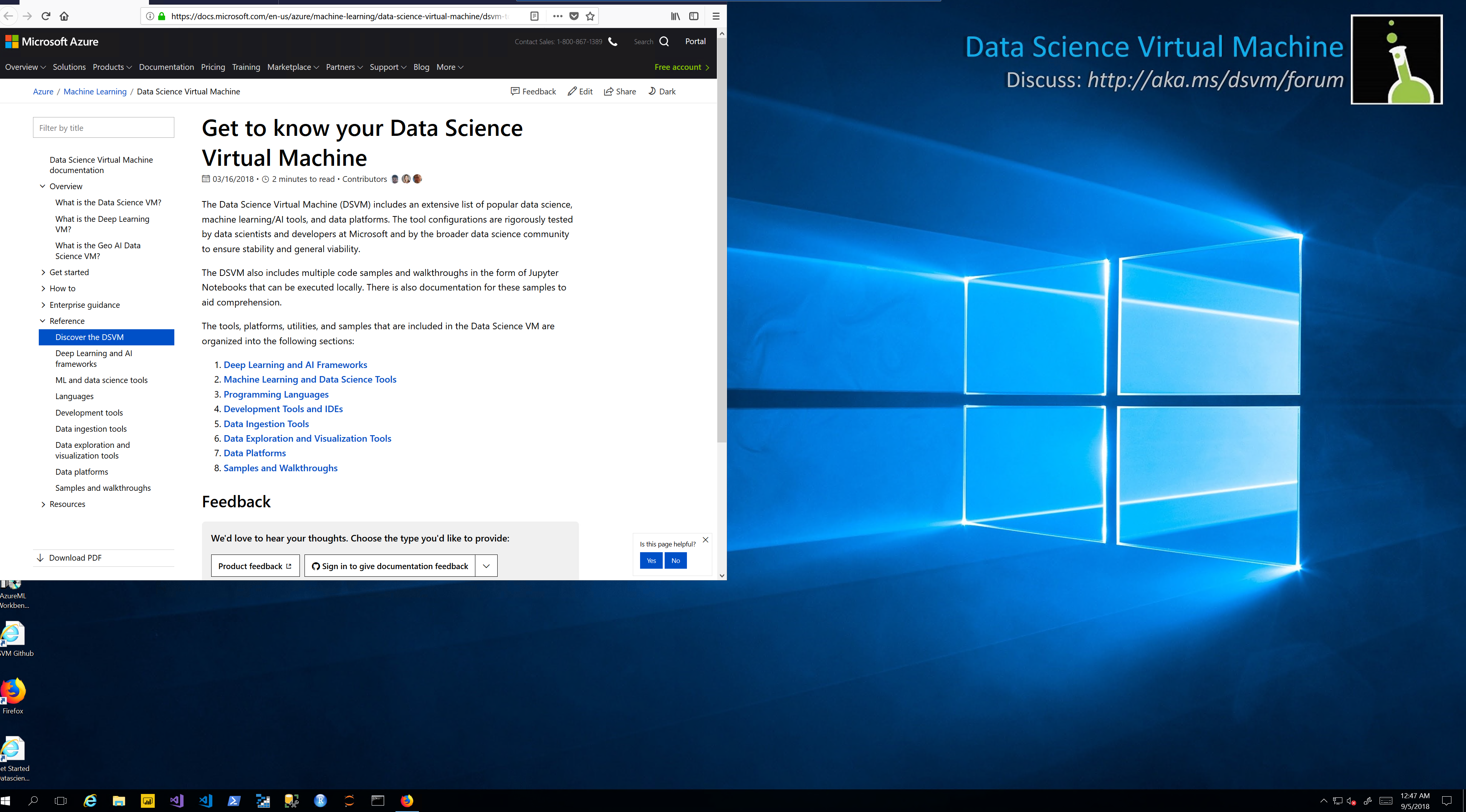
*Logging into the VM*

1. Open the RDP file and connect to the VM after entering in the password you created for the VM:



*Log into the VM*

1. Wait for the remote desktop to appear and confirm that it resembles the one below.



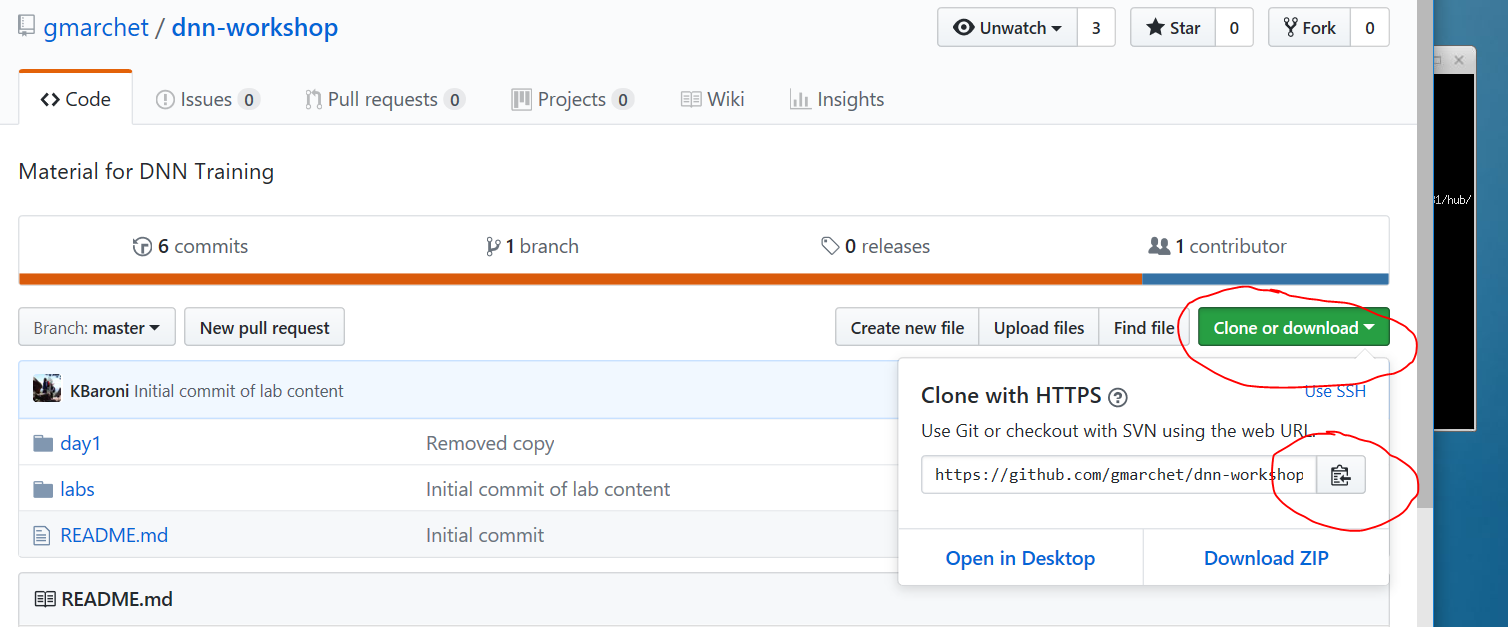
*Connected!*

Now that you are connected, take a moment to explore the shortcuts on the desktop. These are shortcuts to the numerous data-science tools preinstalled in the VM, which include [Jupyter](http://jupyter.org/), [R Studio](https://www.rstudio.com/), and the [Microsoft Azure Storage Explorer](https://azure.microsoft.com/en-us/features/storage-explorer/), among others.

**Exercise 3: Clone (download) the workshop content from GitHub repo**

In this exercise, you will download the content from the lab repo on GitHub. The content includes notebooks that we will use in labs for the workshop.

1. Open the Documents folder. You will be downloading the git repo to the ***notebooks*** folder.
2. In this step we are going to clone the lab repository in GitHub. Bring up a browser in your VM and navigate to the lab’s git repository here: <https://github.com/gmarchet/dnn-workshop> and click on ‘Clone or download’. Select the icon to copy the web URL.

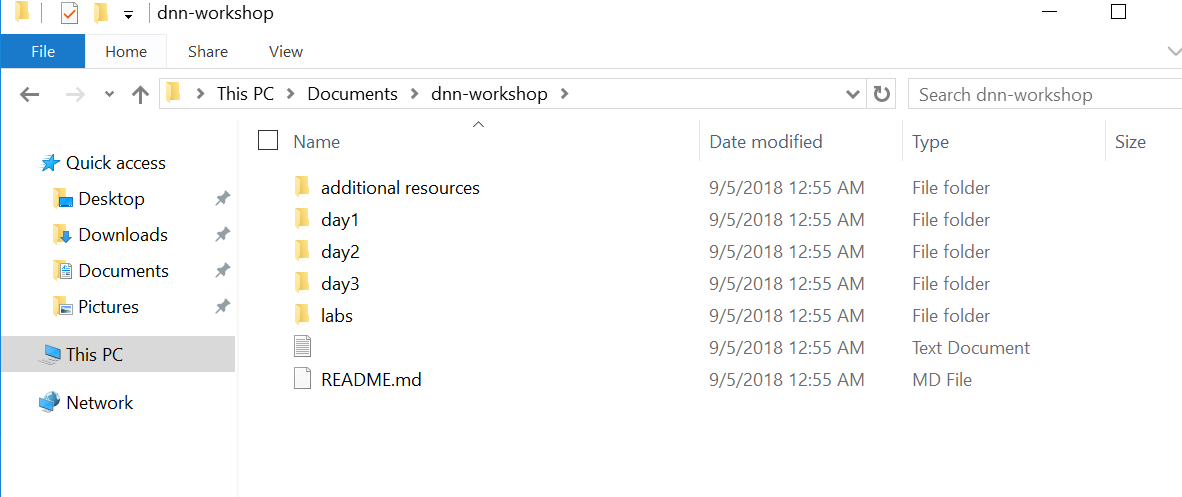


Bring up a new command window and type in

cd Documents and then type in

git clone <paste in the URL from github>

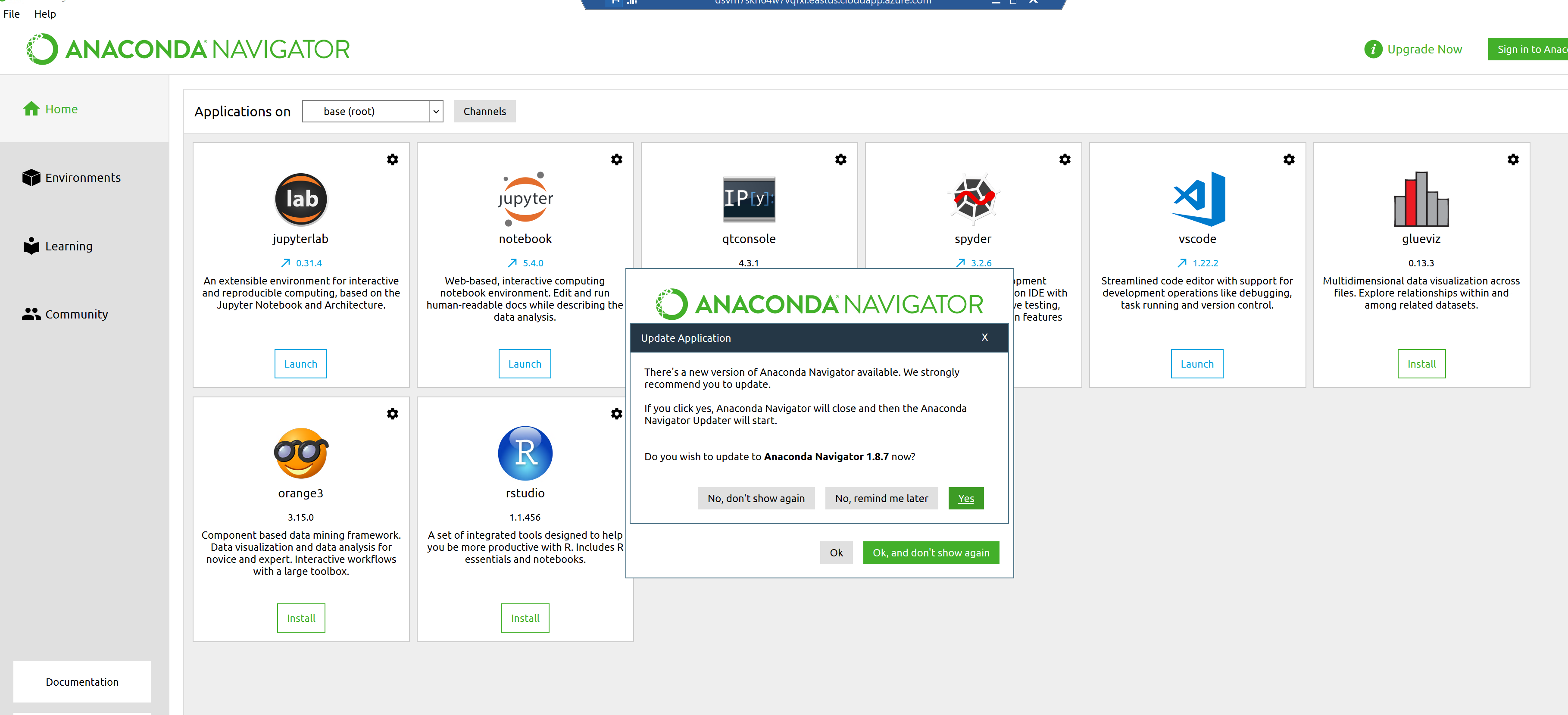
This will clone the *dnn-workshop* repo to your **Document**s folder:



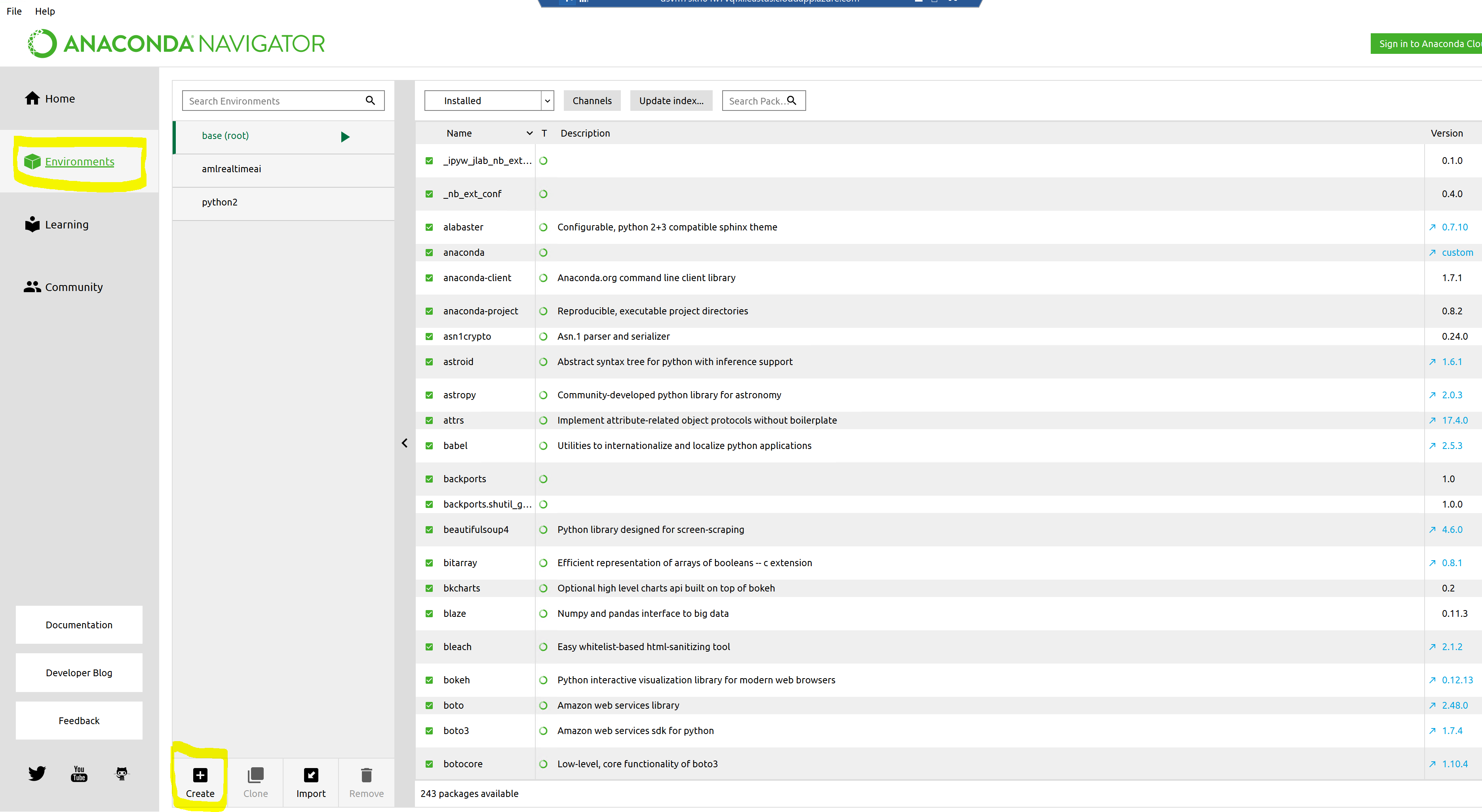
**Exercize 4: Setup Anaconda Environment**

In this exercise, you will setup the Anaconda environment needed for the labs. Anaconda and Anaconda Navigator are already installed on the VM.

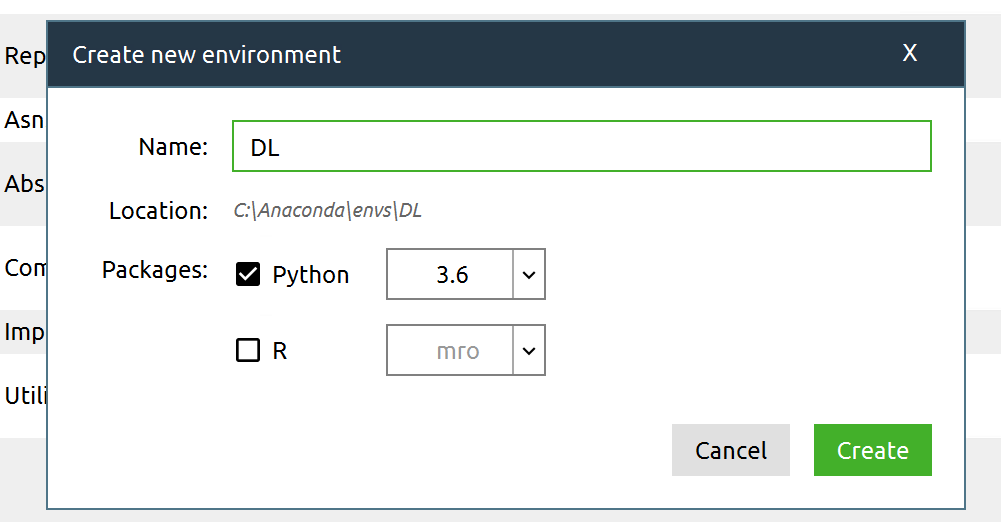
* 1. In the search bar, enter ‘’anaconda and select *Anaconda Navigator*. It may take a minute to load. When it does, update the application if prompted to do so:



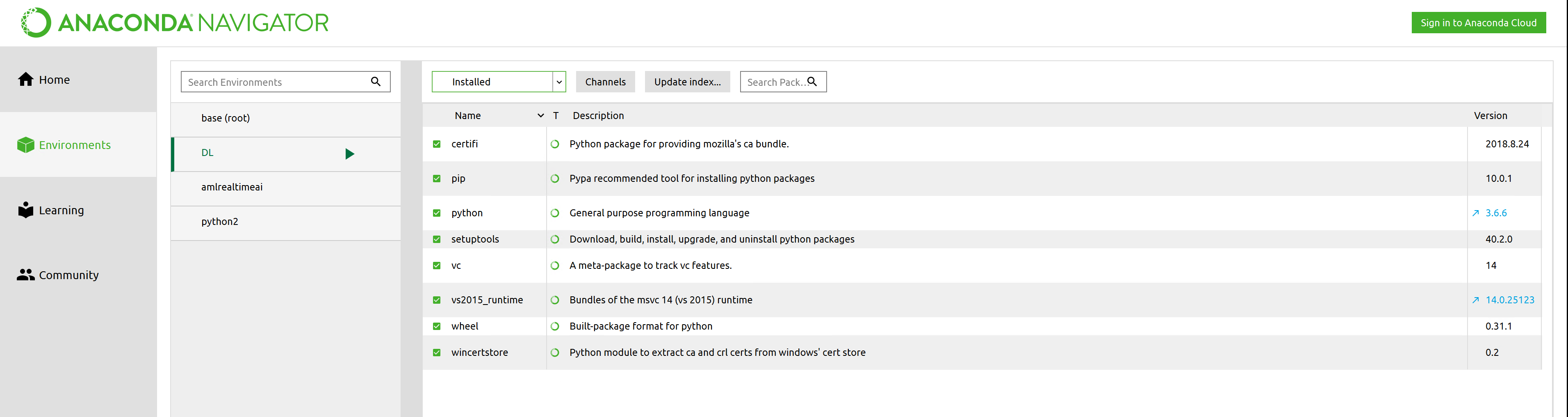
1. Re-launch *Anaconda Navigator* if it was updated and create the environment we will use for the lab. Select **Environments** from the left panel and the **Create** button on the bottom screen:



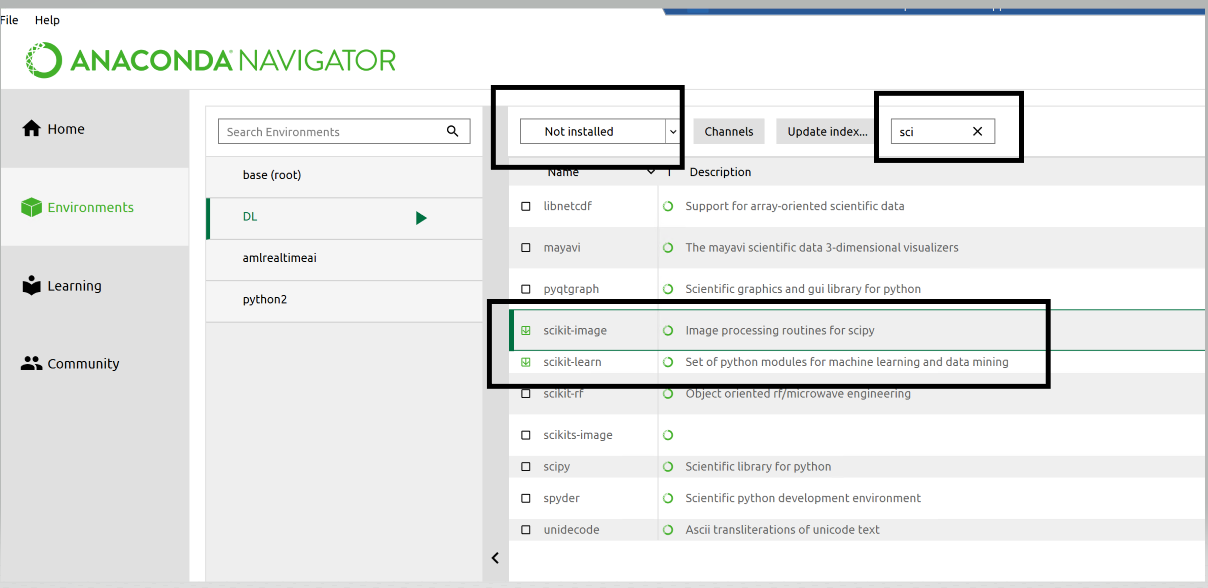
1. Name the Environment **DL** and select **Create**:



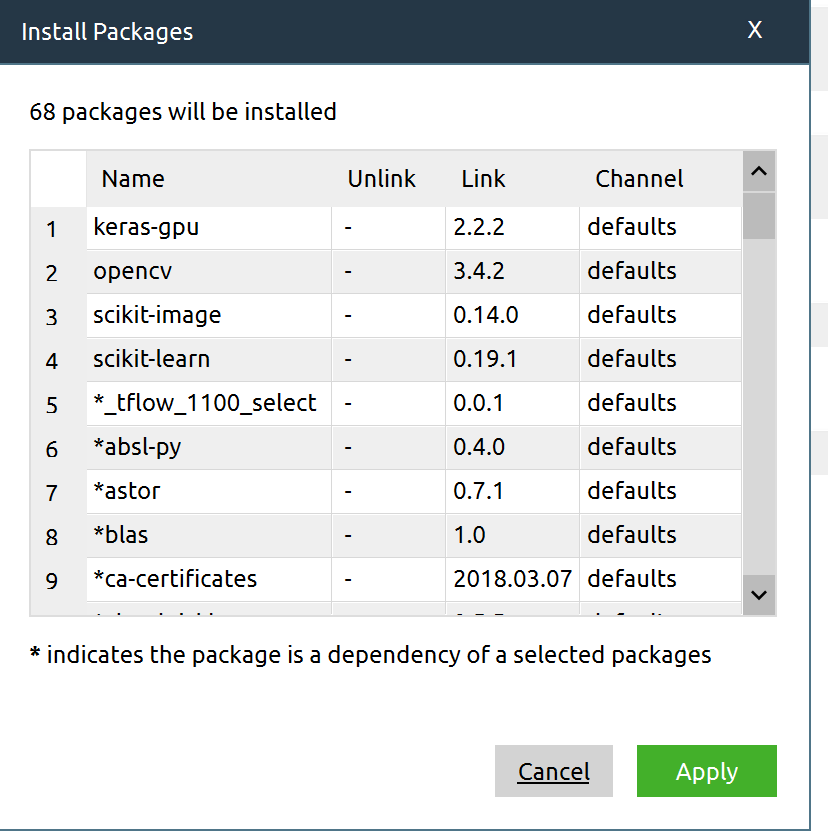
1. When the environment creates, review the packages that come pre-installed into the basic environment:



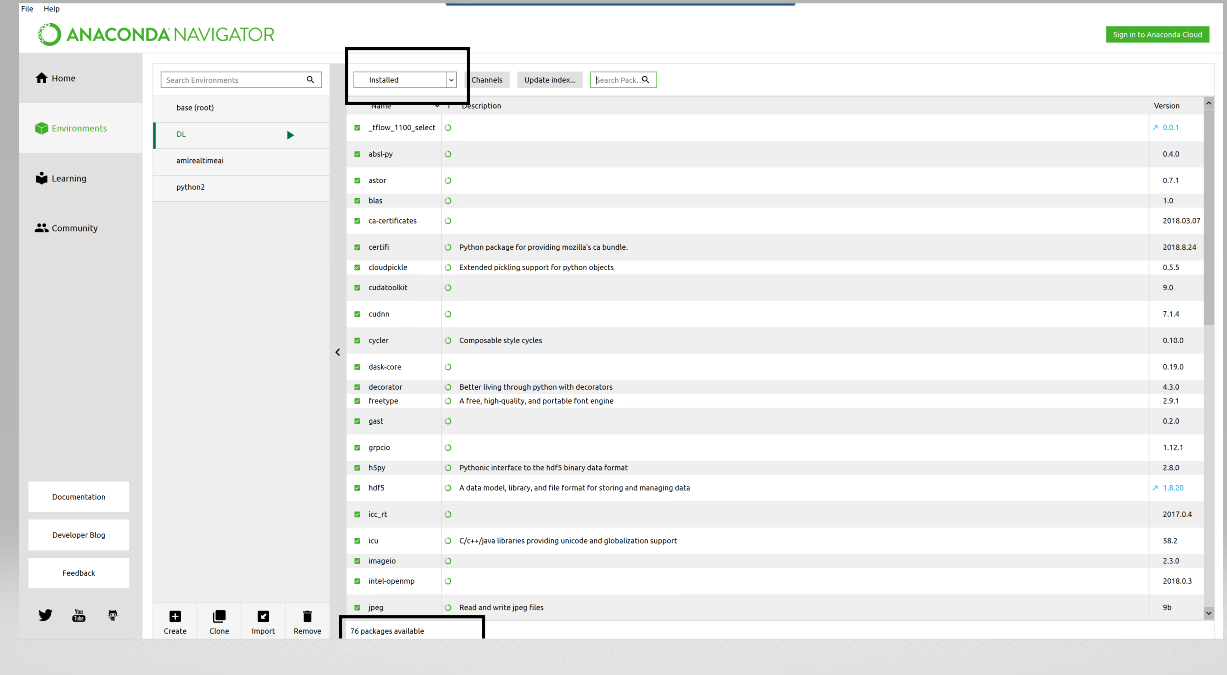
1. Add additional packages to the environment. Select the **Not Installed** dropdown and search for and select these additional packages: *keras-gpu, opencv, scikit-image, and scikit-learn*. Click **Apply**.



1. A message box listing the packages and their dependencies will display. Click **Apply**.



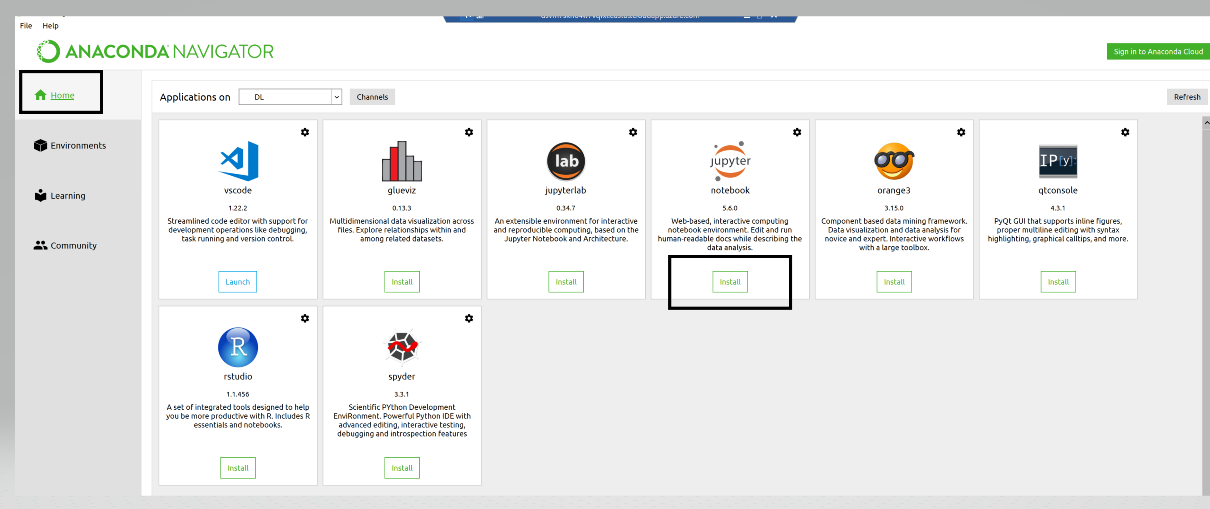
1. It will take a few minutes for all the packages and their dependencies to install. When complete, select **Installed** from the dropdown box and scroll through to see all 76 packages:



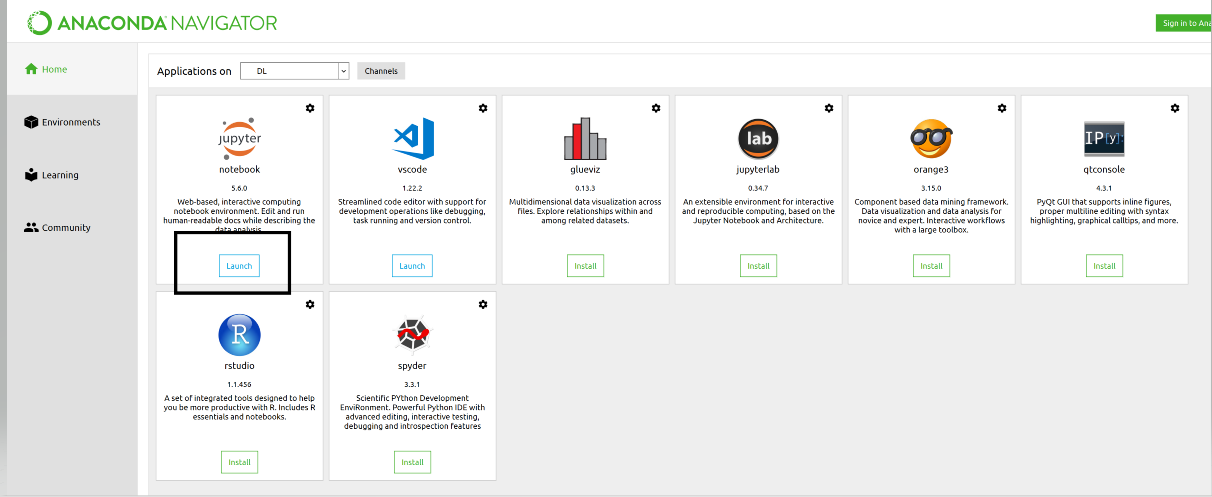
**Exercise 5: Install jupyter notebook and connect to the labs**

In this exercise, you will load a lab notebook into [JupyterHub](https://jupyterhub.readthedocs.io/en/stable/). Jupyter notebooks are widely used in the data-science community to explore, transform, and visualize data. Notebooks are highly interactive, and since they can include executable code, they provide the perfect platform for manipulating data and building predictive models from it.

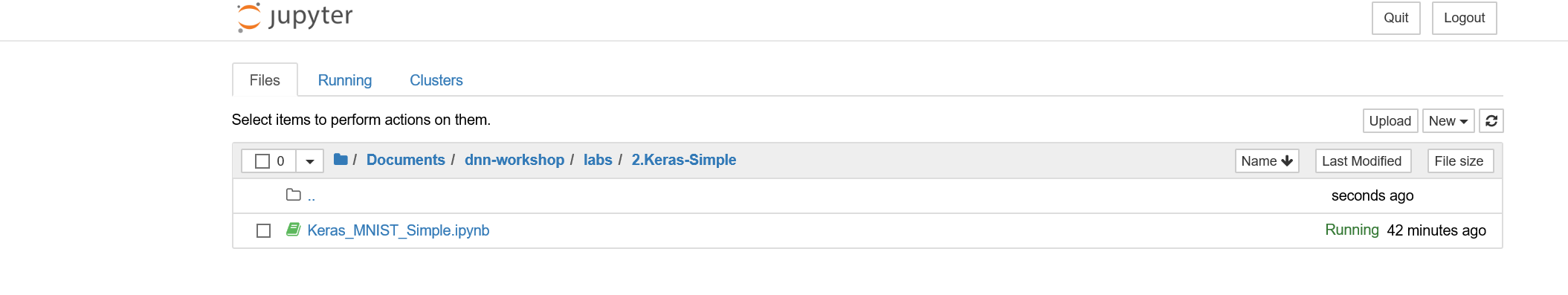
1. Navigate back to the **Home** page in *Anaconda Navigator* and click on **Install** for *jupyter notebook*:



1. Once the install is complete you can launch *jupyter notebook*:



1. Confirm you have access to the workshop notebooks. In jupyter, find the *Documents* folder, *dnn-workshop* folder and browse to the labs folder. Click into the labs to bring up the notebooks:



1. Confirm you can launch jupyter within the **DL** environment you created. Navigate to Environments and activate the DL environment by clicking the green arrow and selecting Open with Jupyter Notebook:

