

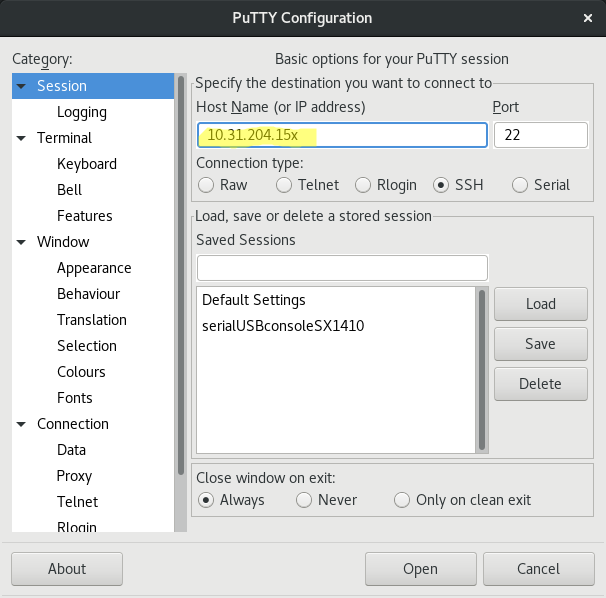
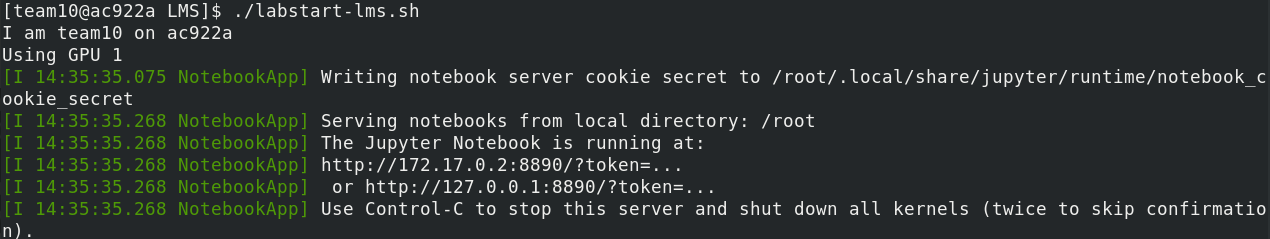
**UT Arlington Deep Learning Workshop**

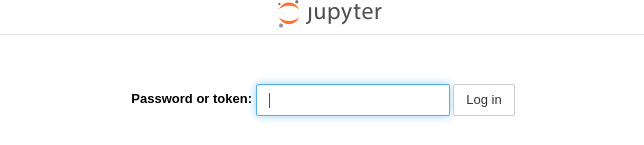
# Lab Exercise: Watson Machine Learning Community Edition: LMS and DDL

## Speaker Names: Glen Corneau, IBM

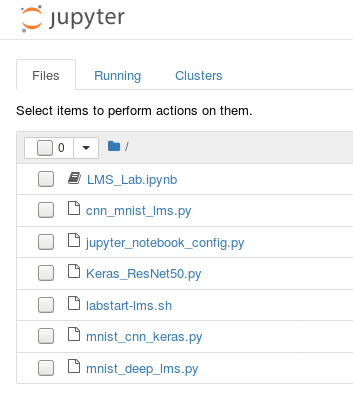


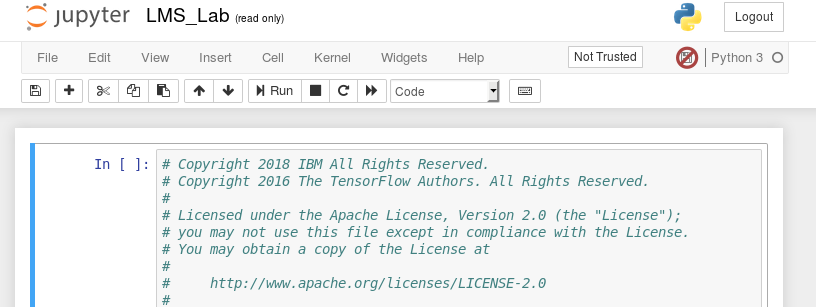
|  |  |
| --- | --- |
| **Teams 01, 02, 03, 04** | Server IP: 10.31.204.151 |
| **Teams 05, 06, 07, 08** | Server IP: 10.31.204.153 |
| **Teams 09, 10, 11, 12** | Server IP: 10.31.204.155 |
| **Teams 13, 14, 15, 16** | Server IP: 10.31.204.157 |
| **Teams 17, 18, 19, 20** | Server IP: 10.31.204.159 |
| **Jupyter port** | <Server IP>: **88xx** where **xx**  = team # (01-20) |

1.  **For this lab you will need both a web browser and a SSH connection to the IBM AC922 Server.  
     
   Click on the PuTTY icon on the Desktop and enter the assigned IP address for your server.**
2. **Your login id is teamXX and password is abcd1234**
3. **Change to the LMS directory:  
     
   cd LMS**
4. **Start the LMS Docker container and jupyter notebook that you'll be connecting to via the web browser:  
     
   ./labstart-lms.sh  
     
   DON'T click the link you see, it won't work!**
5. **In the laptop web browser go to "http://10.31.204.15x:88xx"  
      
    Be sure and use your assigned server's IP address  
    and your team port number!   
    Example, team10 = 10.31.204.155:8810  
     
   You should see a Jupyter notebook prompt as follows.  
     
     
     
     
     
     
     
     
   Jupyter Notebook password is: abcd1234**



1. **Click the link for “LMS\_Lab.ipynb” to launch the notebook.**



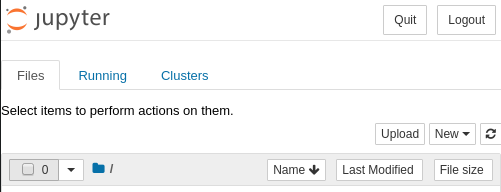
1. **Using Jupyter Notebooks inside a Docker container**This Docker container has text and code provided by the IBM Watson Machine Learning Community Edition development team and the github repositories for the 3D-Unet CNN and BraTS images. Within the notebook are commands that could be run from the command-line or from within a Python script. These notebook files are designed to be created, viewed, edited, manipulated, and run in the Jupyter client within a web browser. Notebook files contain various 'cells' - Heading cells, Markdown cells, Code cells, and Raw cells.

If you're interested in what ipynb files look like, you can open a console from within the main Jupyter tab and inspect the LMS\_lab.ipynb file. For additional information on these 'notebook' files, take a look at:

<http://jupyternotebook.readthedocs.io/en/latest/notebook.html>

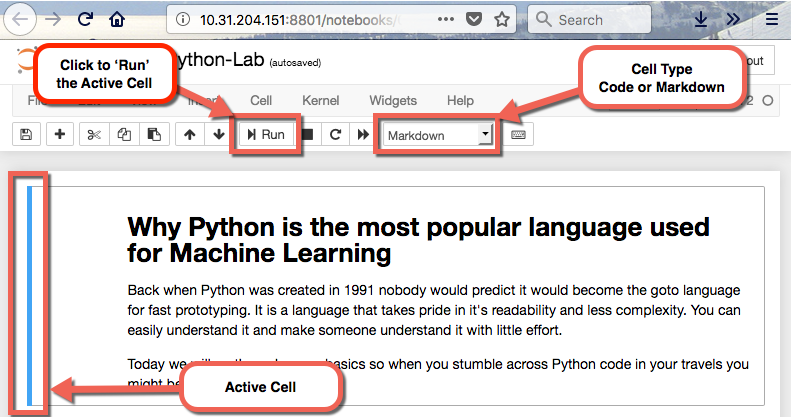
Note, however, that .ipynb files are not typically created by hand; they are created using the Jupyter Notebook.

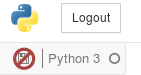
1. **Opening a Terminal using Jupyter**

From the Home Page Jupyter tab (not the one you just opened the notebook within, the  
 first tab) click the **New** dropdown and select **Terminal.**This should open a new tab with a command line terminal which you can use during the lab.  
  
Return to the LMS lab notebook tab you opened already.

1. **Using Jupyter**

In the generic screen shot below (it's not a copy of this lab), the blue bar on the left highlights the active cell you are in, and to the right of the control buttons along the top, you can see that this particular cell is a 'Markdown' cell, which is really just a text cell used to document things in the notebook. The ‘Run’ button will step through code, section by section.



In your environment, opening a Jupyter notebook file may have output from a previous invocation. (Jupyter by default automatically save Notebooks, so this is how output from a previous session might be present). TheDDL portion of the lab uses a read-only copy of the notebook (you can tell from the crossed out icon in the top right) so you shouldn't have any other output.

1. **Step through the Notebook**



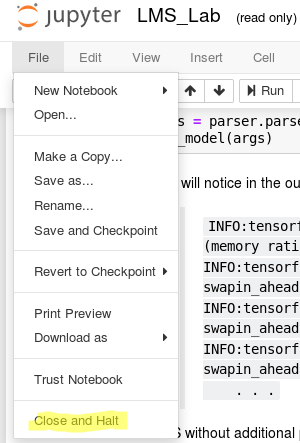
Go ahead and click the Run button once to see how it advances you to the next cell.

Ok, now that you understand the mechanics of the Jupyter Notebook, you're ready to step through the lab. As you do this, you'll see that Markdown cells simply get highlighted, and Code cells run. And some of the output of Code cells that generate output will appear inline in the notebook.

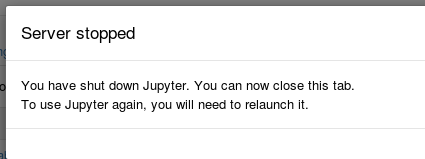
**However, don't simply click, click, click through the notebook!**

Take your time to read the Markdown text to understand what the notebook is attempting to demonstrate. If you don't do this, you'll get to the end and ask yourself "what did I just do, and why? what did I learn?".

When you click the button to then run Code cell yourself, the Input indicator will change to In [\*], sometimes for only a instant, and sometimes for several seconds or minutes. When the asterisk in the brackets is replaced by a number (e.g. In [12]), the code in that cell has finished running, and you can click to move to the next cell. (While you can click ahead without waiting, it's less confusing if you wait for Code cells to complete before stepping ahead.)

If you so desire, and if time permits, you can also look at the other python files in that directory for examples of utilizing LMS.

1. **Close out the LMS Lab  
     
   Click on the File → Close and Halt   
     
   If it prompts you to Leave the page, do so.**
2. **Return to the prompt on the SSH session.  
     
     
     
     
     
     
     
     
     
     
     
     
     
     
     
     
   DDL Lab**



1. **Change to the DDL directory  
     
   Back on the Putty SSH session window,   
   cd ../DDL  
   conda activate**
2. **This lab does NOT use a Docker container!  
    Start the the Jupyter notebook**

**./labstart-ddl.sh**

1. **In the web browser type "http://10.31.204.15x:88xx"  
      
    Be sure and use your assigned server's IP address  
    and your team port number!   
    Example, team10 = 10.31.204.155:8810**
2. **Click the link for “DDL\_Lab.ipynb” to launch the notebook.**
3. **When complete, just like before, close out the DDL Lab via File → Close and Halt. Then in the Jupyter tab, Quit**