Jupyter Notebook for Playing with Spark, Python, R, Scala

These instructions show how to build or connect to the our existing Jupyter Notebook playground for experimenting with Spark, Python, R, and Scala.

Connecting to Technology Nursery's Jupyter Notebook

You can use the Technology Nursery Jupyter Notebook to play around with Python 3, R, Scala, Spark, Tensorflow and Keras.

CPU Based Notebook

This Notebook has only CPU capability, along with Python 3, R, Scala, Spark, Tensorflow and Keras.

- In a browser, click on the link to navigate and login to jupyter.technologynursery.org

The link will login by automatically passing the appropriate token for this server. *Note that the service will create a new token whenever it gets restarted. A service restart can happen every time the server's packages are updated and the server is restarted.*

GPU Based Notebook

> **Availability Status**
>
> jupyter-gpu is currently **down**. The service is made available from my laptop (Alienware17r3L) within the Navarro Computing LLC subnet. I am willing to make the service available upon request; email ralph@navarrocomputing.com

This Notebook has GPU and CPU capability, along with Python2, tensorflow-gpu and keras. Not included are Spark, R, and Scala.

- In a browser, click on the link to navigate and login to jupyter-gpu.technologynursery.org

The link will login by automatically passing the appropriate token for this server. *Note that the service will create a new token whenever it gets restarted. A service restart can happen every time the server's packages are updated and the server is restarted.*

Building Your Own Playground using Docker

The following requires you to have an accessible Docker environment.

1. Pull and Run the jupyter/all-spark-notebook Docker image from Docker Hub

   ```
   docker run -it --rm -p 8888:8888 jupyter/all-spark-notebook
   ```

2. Once the system is running, the system will display a URL containing the security token. Copy this URL to the clipboard or file. You will need this URL to login to the system for the first time.
   For example only:
   http://localhost:8888/?token=5b5029ad3ee362907aaa703520a3dabc022fad62ef7b90ae

3. If you are using Docker Toolbox, forward port 8888 from the Guest to the Host.
   a. Open VirtualBox.
   b. Select the 'default' Virtual Machine (VM) and click Settings.
   c. In default-settings window, select Network tab.
   d. Select Adapter 1 tab
   e. Select Advanced. Additional options are presented.
   f. Click 'Port Forwarding' button. A Port Forwarding Rules window opens.
   g. On the right side of the window, click the green '+' sign to create a new rule.
   h. Name the new rule: Jupyter, Host Port set to 8888, Guest Port set to 8888, and leave Protocol set to TCP.
   i. Click OK to accept the new rule.
   j. Click OK to accept the new Network setting.
   k. Close the VirtualBox window. Any VMs that were running will continue to run.

Contact Information to Report Issues

If this system is down or unavailable, please call, text and/or email Ralph A. Navarro Jr. .

Related articles

- Jupyter Notebook for Playing with Spark, Python, R, Scala