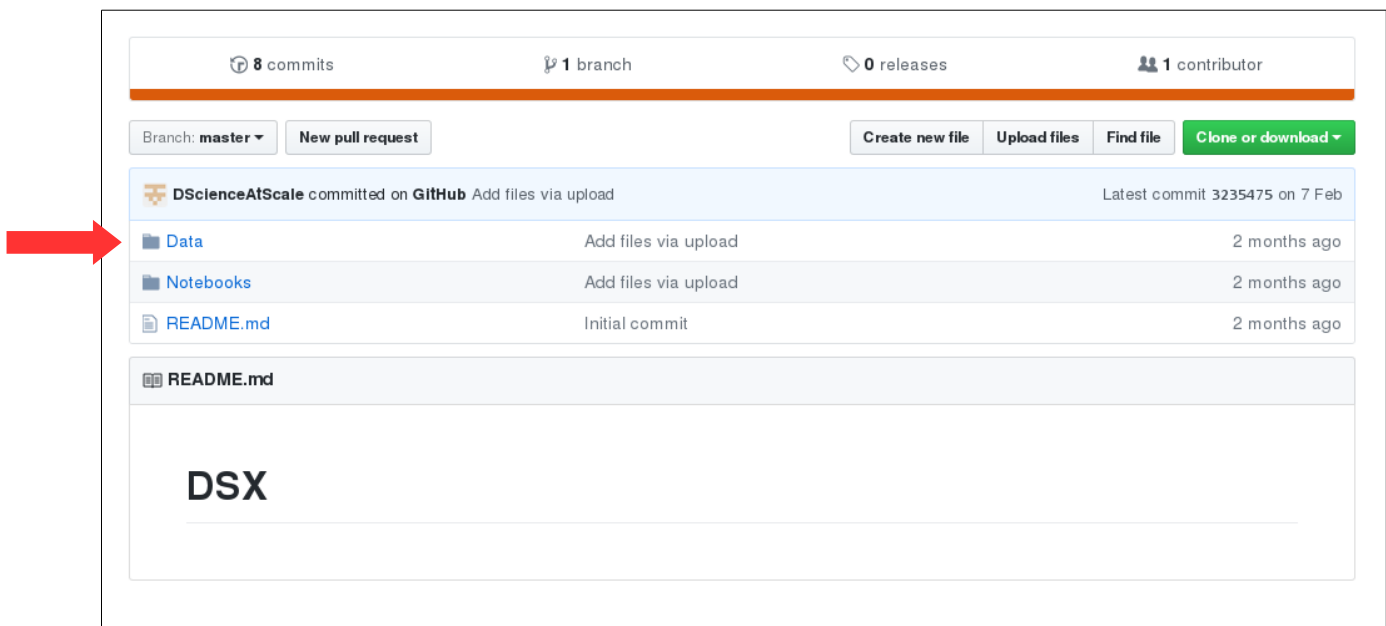


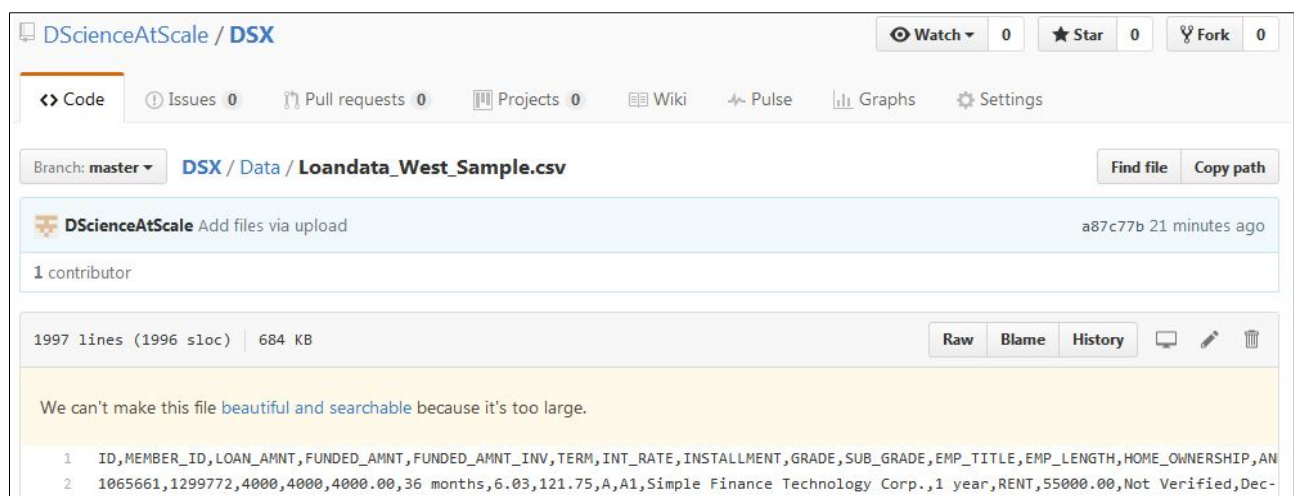
PRACTICAL INTRODUCTION TO THE IBM DATA SCIENCE EXPERIENCE PLATFORM

Prerequisites: Getting a data sample

1. Browse to: github.com/DScienceAtScale/DSX
2. Click on Data

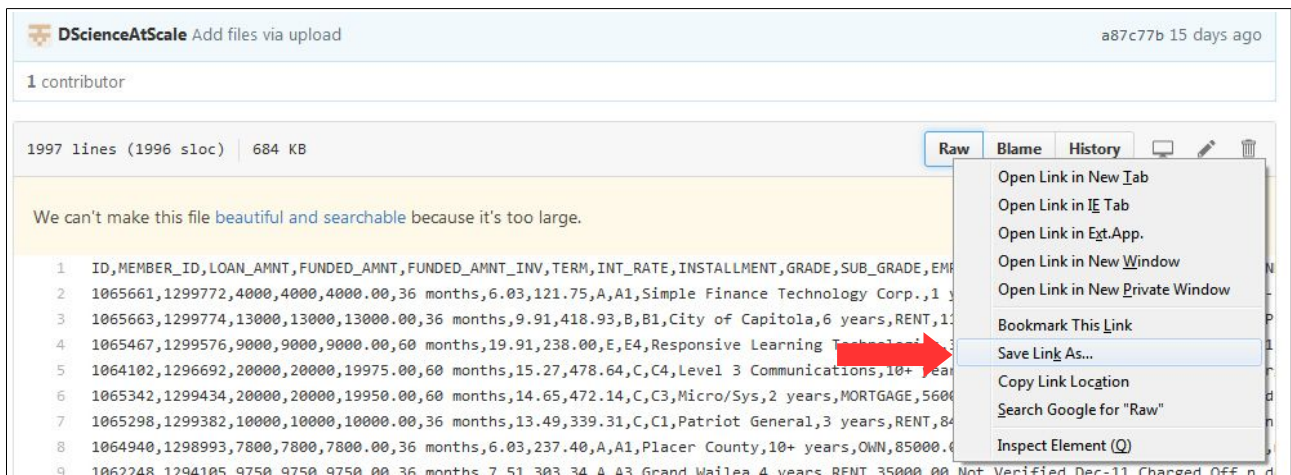
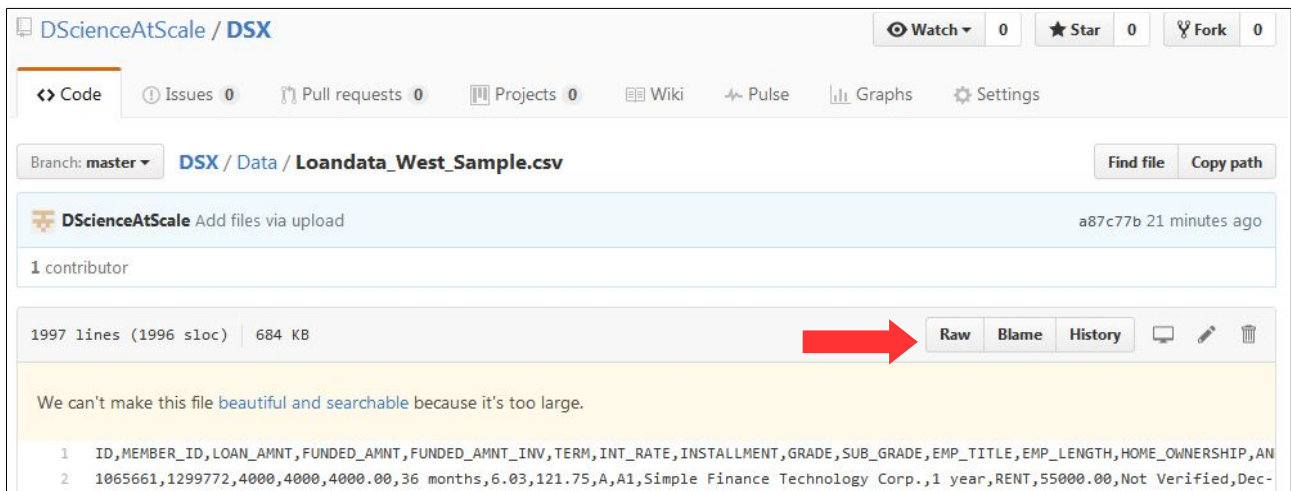


3. Click on “Loandata_West_Sample.csv”. You should see a screen like this

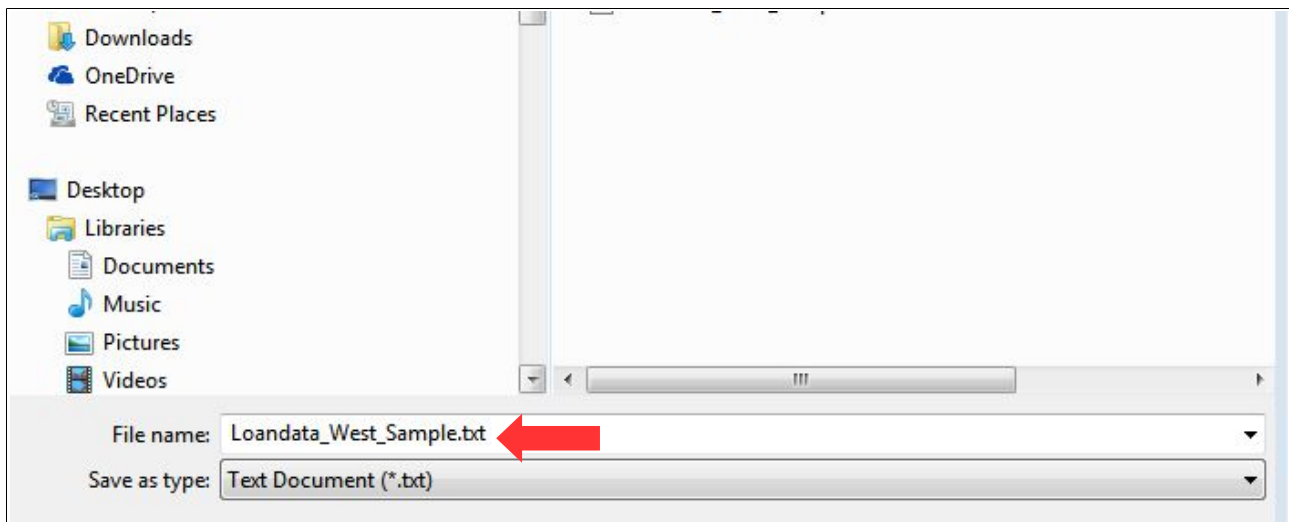


4. Right click on the “Raw” button (do not left click. If you decide to left click, then you will need

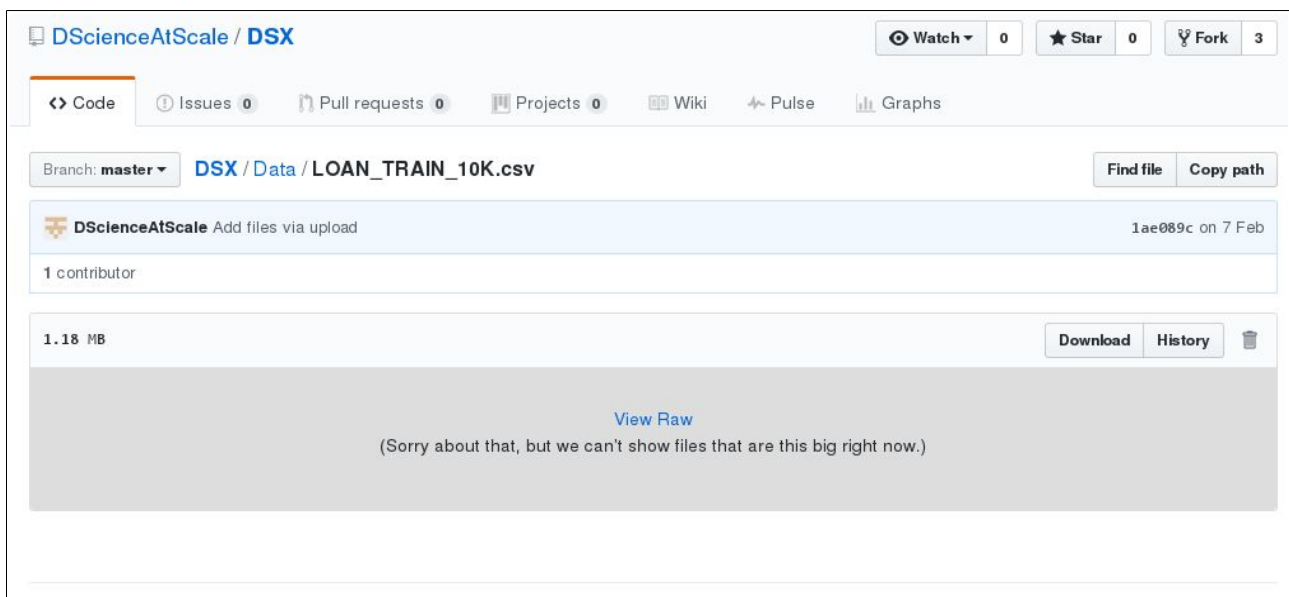
to use the File>Save Page As –or equivalent– menu from your browser) and select “Save Link As” or “Save Target As”, (depending on your browser) as per the two snapshots below.



5. The target file type should normally show a “csv” extension. However, there may be cases, especially noticed with Internet Explorer, where the target file type appears as “txt”, as per the snapshot below:



6. If this happens, cancelling out of this screen and repeating the right-clicking on “Raw” usually brings up the target file as CSV. If not the case, please try a different browser.



7. Save the “Loandata_West_Sample.csv” file to your hard drive.

8. Repeat the process to download the "LOAN_TRAIN_10K.csv" file as well.

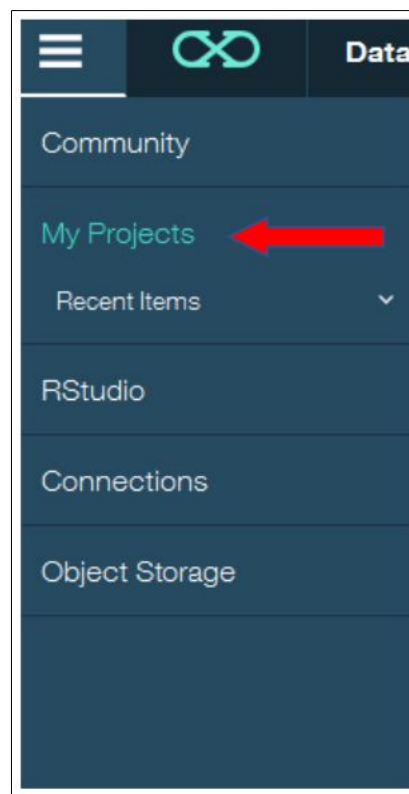
SECTION 1: Creating and working with a Project

Projects allow data scientists to collaborate by sharing data assets and code and more generally enable a team of data scientists to easily implement a full lifecycle of analytics assets, including versioning of assets, recording of comments for formal reviews and other typical activities which govern most enterprise grade work.

You will also encounter the notion of Owners and Collaborators while manipulating Projects. It is important to note that owners and collaborators of Catalogs and Projects are distinct (you may be added as a collaborator to a Catalog, but not to a project or vice-versa). When a project is created, it can be associated with a catalog, but privileges associated with the project do not translate into catalog access privileges, which must be set by the catalog administrator separately.

Projects serve as an overall umbrella where several assets may be added and managed: Data Assets (files or connections), Analytic Assets (Notebooks), bookmarks, etc....

1. From the contextual menu, go to My Projects



2. Create a new Project using either “+” sign (see snapshot below). You can call your project DSXProj1.

New Project

Name
DSXProj1

92

Description

Project description

3000

Spark Service
DSX-Spark

Target Object Storage Instance
DSX-ObjectStorage

Target Container
DSXProj1

256

Create **Cancel**

9. When your project is created, click on it (if not already open) and go to the “Data Assets” section in the project action bar, then click on “browse” in the right-hand side of the screen.

My Projects > DSXProj1

Overview **Data Assets** Bookmarks Collaborators Settings

Find in Data Assets

Data Assets [+ add data assets](#)

NAME	TYPE	SERVICE	LAST MODIFIED	ACTIONS
you currently have no data assets				

Files **Connections**

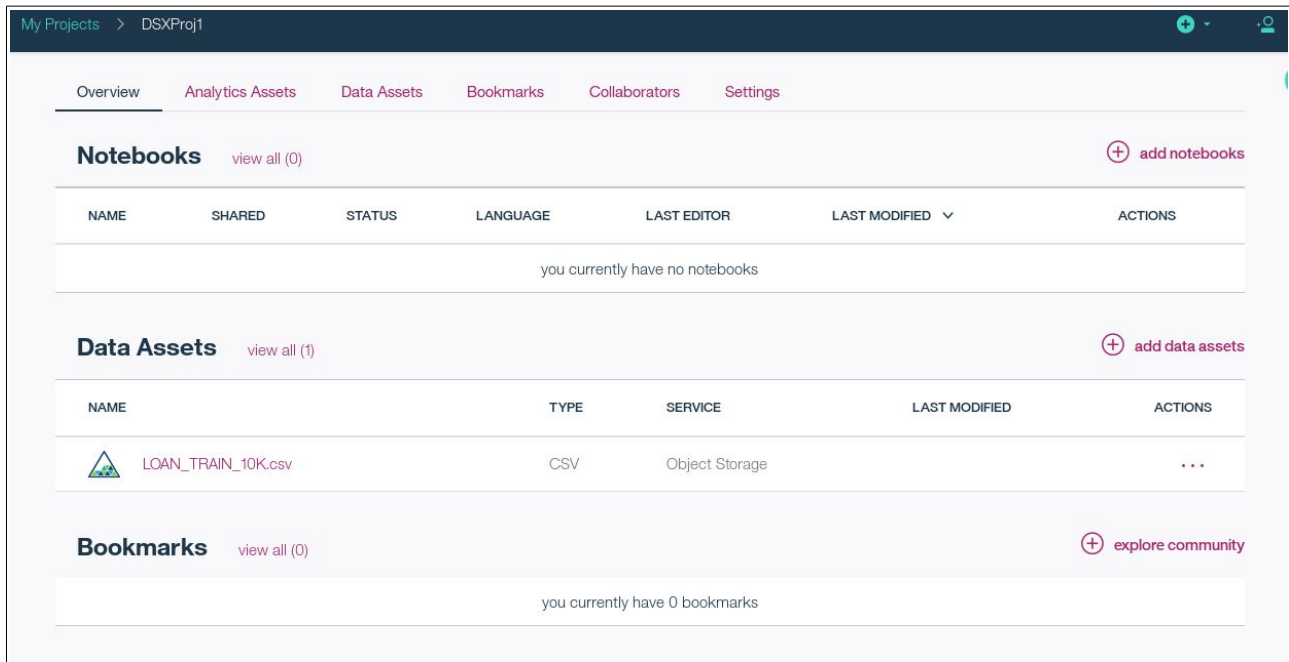
or browse
your files to add a new file

No files found.

Apply


10. Select the locally stored LOAN_TRAIN_10K.CSV file and click OK. Wait for the file to get uploaded to your Object Storage instance. Make sure it is selected and click Apply.

The LOAN_TRAIN_10K.csv file should now be listed under Data Assets in your project.



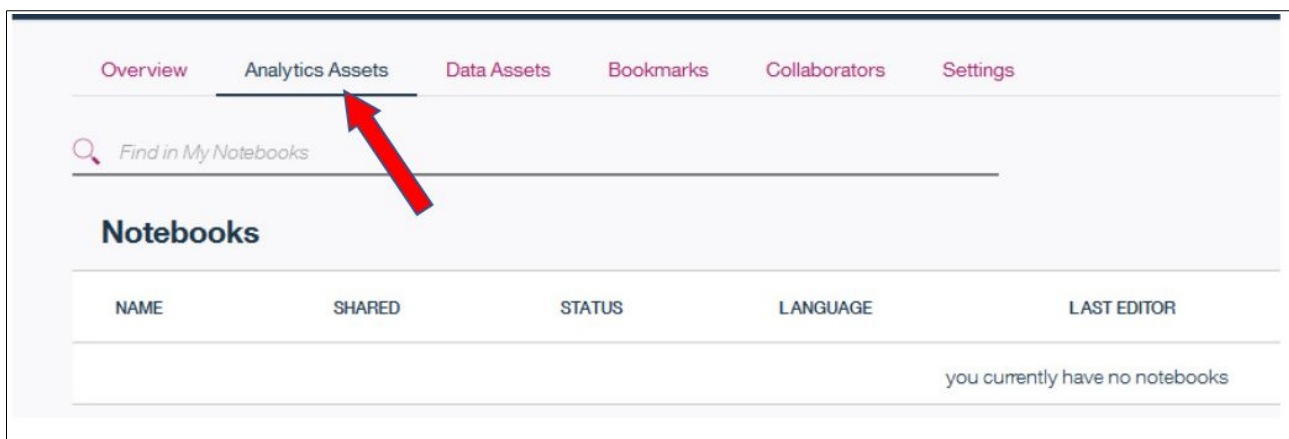
The screenshot shows the 'My Projects' interface for a project named 'DSXProj1'. The top navigation bar includes 'Overview', 'Analytics Assets', 'Data Assets', 'Bookmarks', 'Collaborators', and 'Settings'. The 'Data Assets' tab is active, displaying a table with one entry: 'LOAN_TRAIN_10K.csv'. The table has columns for NAME, SHARED, STATUS, LANGUAGE, LAST EDITOR, LAST MODIFIED, and ACTIONS. The entry shows a CSV file of type 'Object Storage'. There are also sections for 'Notebooks' and 'Bookmarks', both showing 'you currently have no notebooks' and 'you currently have 0 bookmarks' respectively.

NAME	SHARED	STATUS	LANGUAGE	LAST EDITOR	LAST MODIFIED	ACTIONS
you currently have no notebooks						

NAME	TYPE	SERVICE	LAST MODIFIED	ACTIONS
 LOAN_TRAIN_10K.csv	CSV	Object Storage		...

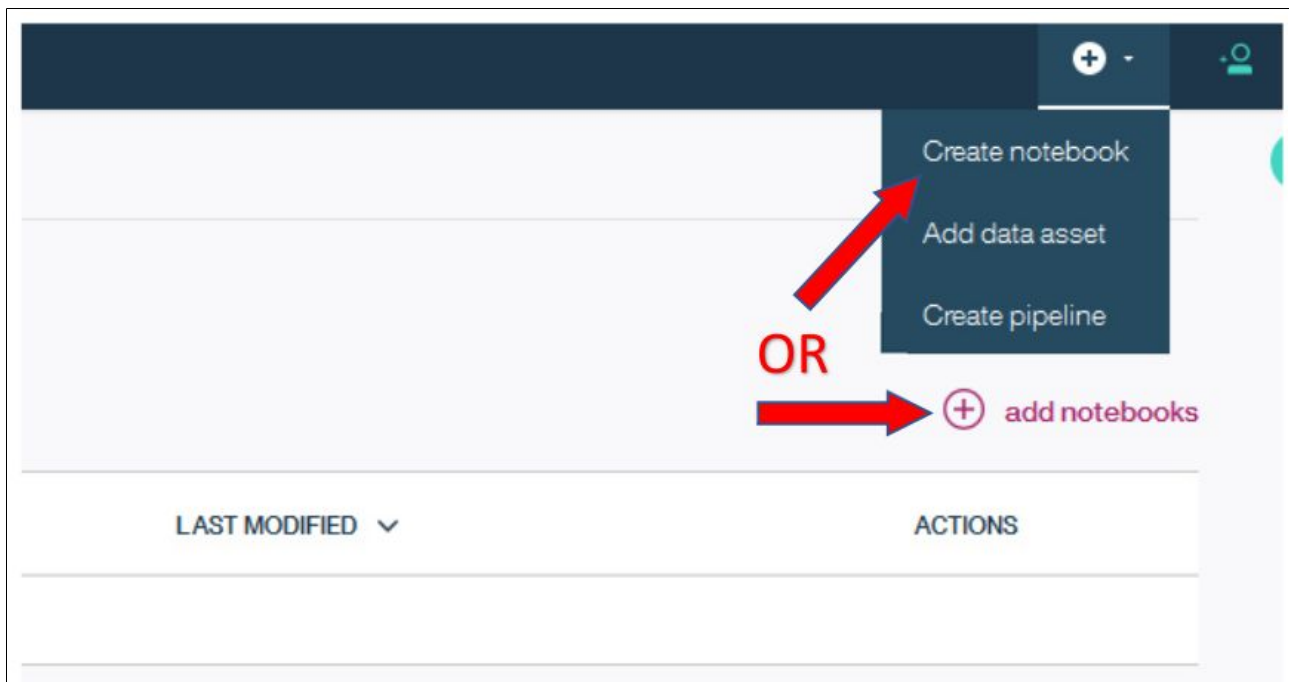
NAME	SHARED	STATUS	LANGUAGE	LAST EDITOR
you currently have 0 bookmarks				

11. In the Analytic Assets section of the project, create a notebook.



The screenshot shows the 'My Projects' interface for a project named 'DSXProj1'. The top navigation bar includes 'Overview', 'Analytics Assets', 'Data Assets', 'Bookmarks', 'Collaborators', and 'Settings'. The 'Analytics Assets' tab is highlighted with a red arrow. Below the navigation bar, there is a search bar labeled 'Find in My Notebooks'. The 'Notebooks' section is displayed, showing a table with columns for NAME, SHARED, STATUS, LANGUAGE, and LAST EDITOR. The table is currently empty, and a message at the bottom states 'you currently have no notebooks'.

NAME	SHARED	STATUS	LANGUAGE	LAST EDITOR
you currently have no notebooks				



12. Open the notebook creation page. Type in a name for the notebook, select the notebook creation method as “From URL”, and select the Spark service instance to be associated with this notebook. For the actual URL from which to create the notebook, follow the instructions in the subsequent steps.

Important note: This notebook will be exported to a common GitHub repository in a few steps below. When creating this notebook, please select a name that is unique enough so that it does not conflict with / overwrite other students’ notebooks.

Create Notebook

Blank From File **From URL**

Name*
Notebook1 **Pick a unique name for your notebook** 41 Characters Remaining

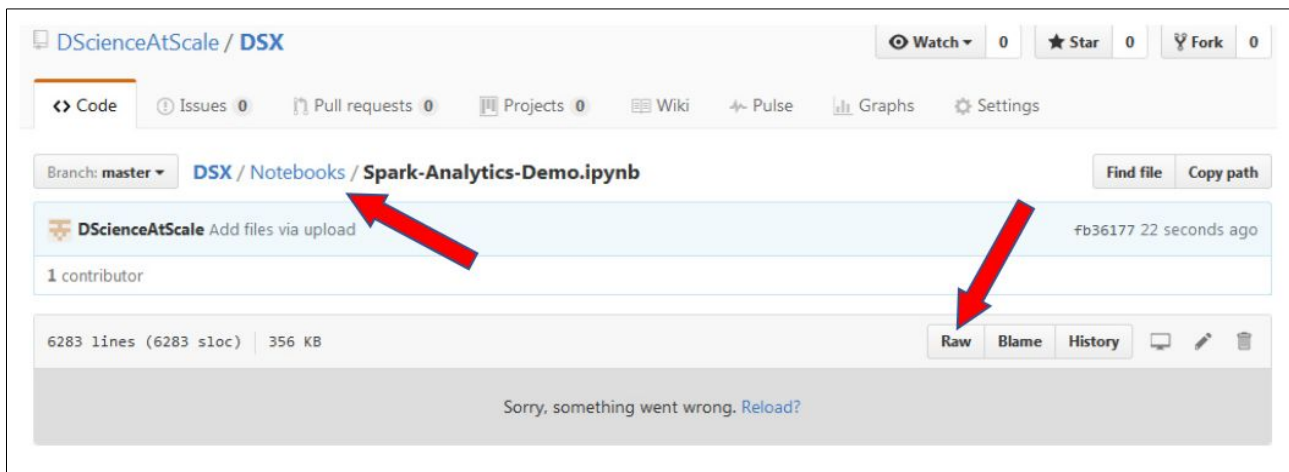
Description
Type your Description here

Notebook URL*
Remote notebook **See subsequent steps for this URL**

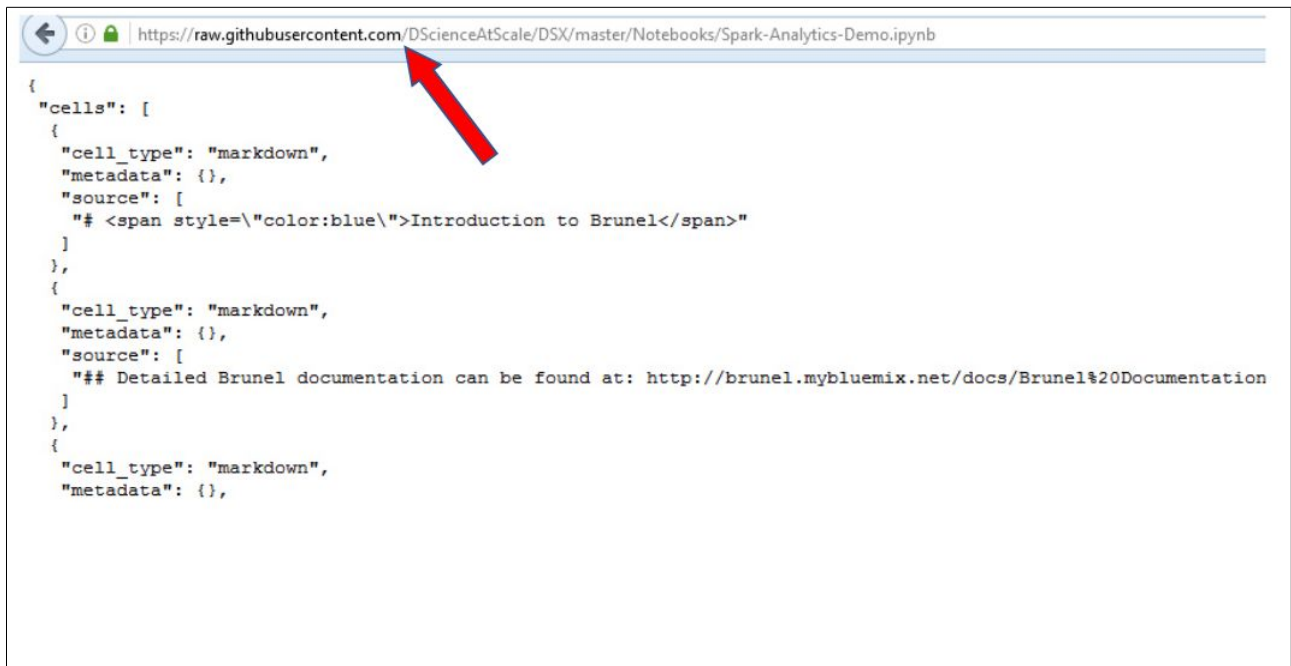
Spark Service*
dsx-instance

Associate this notebook with the IBM Analytics for Apache Spark Service of your choice.

13. Open a new tab in your browser.
14. Go to github.com/DScienceAtScale/DSX/
15. Click on “Notebooks”
16. Click on the notebook named “Spark-Analytics-Demo.ipynb”
17. Click on “Raw”, as per the screenshot below:



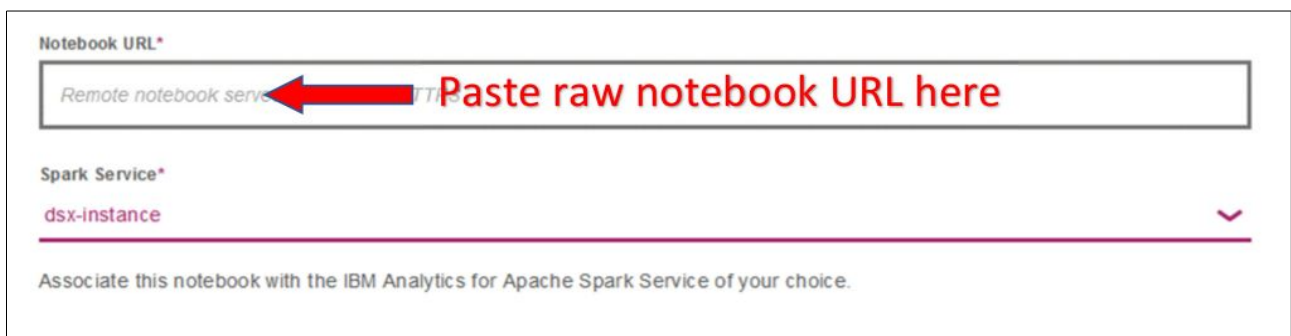
18. Open the raw content of the notebook and select / copy the URL.



The screenshot shows a web browser displaying the raw content of a Jupyter notebook from GitHub. The URL in the address bar is `https://raw.githubusercontent.com/DScienceAtScale/DSX/master/Notebooks/Spark-Analytics-Demo.ipynb`. A red arrow points to this URL. The notebook content is a JSON array of cells. The first cell is a markdown cell with the text `Introduction to Brunel`. The second cell is a markdown cell with the text `## Detailed Brunel documentation can be found at: http://brunel.mybluemix.net/docs/Brunel%20Documentation`. The third cell is a markdown cell with no visible content.

```
{
  "cells": [
    {
      "cell_type": "markdown",
      "metadata": {},
      "source": [
        "# <span style='color:blue'>Introduction to Brunel</span>"
      ]
    },
    {
      "cell_type": "markdown",
      "metadata": {},
      "source": [
        "## Detailed Brunel documentation can be found at: http://brunel.mybluemix.net/docs/Brunel%20Documentation"
      ]
    },
    {
      "cell_type": "markdown",
      "metadata": {},
      "source": []
    }
  ]
}
```

19. Take the copied URL and paste it back in the notebook creation page in the URL box as shown below.



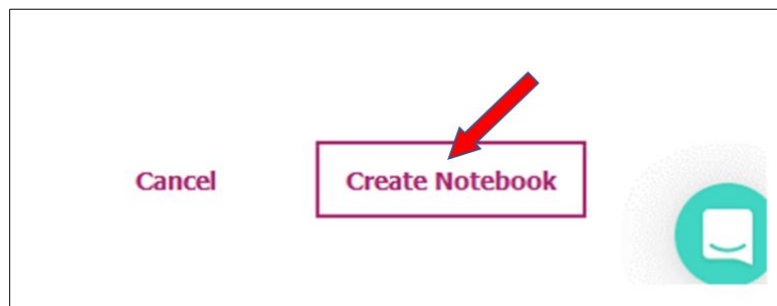
The screenshot shows the notebook creation page. The 'Notebook URL*' field is highlighted with a red box and a red arrow pointing to it. The text 'Paste raw notebook URL here' is written in red next to the field. The 'Spark Service*' dropdown menu is set to 'dsx-instance'. Below the dropdown, there is a note: 'Associate this notebook with the IBM Analytics for Apache Spark Service of your choice.'

Notebook URL*

Spark Service* dsx-instance

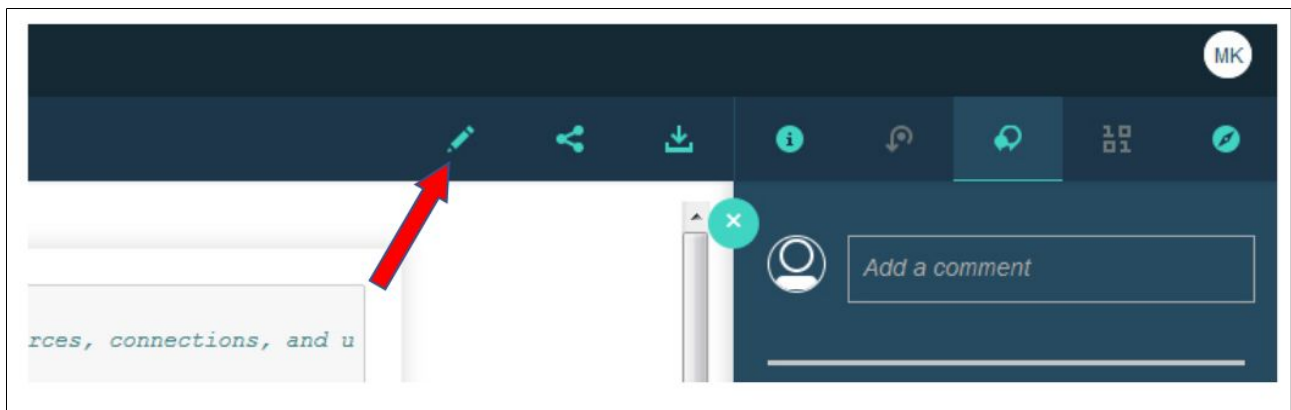
Associate this notebook with the IBM Analytics for Apache Spark Service of your choice.

20. Click on “Create Notebook” in the bottom right corner of the same page.



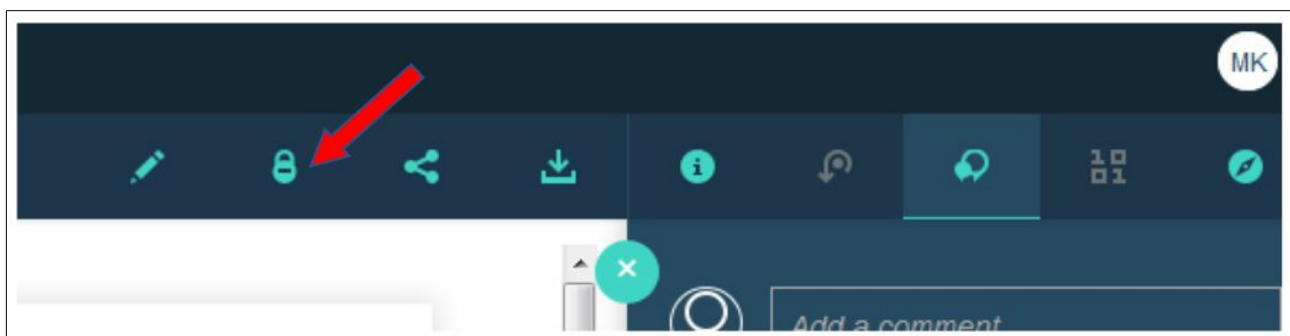
Important remarks

Each time you open a notebook (not including the creation step), it is usually in “Viewing” only mode and not editable. To be able to edit it, remember to click on the “pencil” shaped icon.



In the “Analytics Assets” view of your Project, it is possible for a notebook to show as locked (padlock icon next to notebook), as shown in the snapshot below. Locked notebooks cannot be edited. You can unlock the notebook by clicking on the padlock icon and selecting “unlock”. The same padlock icon will appear right next to the pencil icon mentioned above when a locked notebook is opened.

Overview Analytics Assets Data Assets Bookmarks Collaborators Settings						
Notebooks view all (8) add notebooks						
NAME	SHARED	STATUS	LANGUAGE	LAST EDITOR	LAST MODIFIED	ACTIONS
notebook			Python 2.7	Mokhtar Kandil	21 Feb 2017	
TensorFlow			Python 2.7	Mokhtar Kandil	13 Feb 2017	

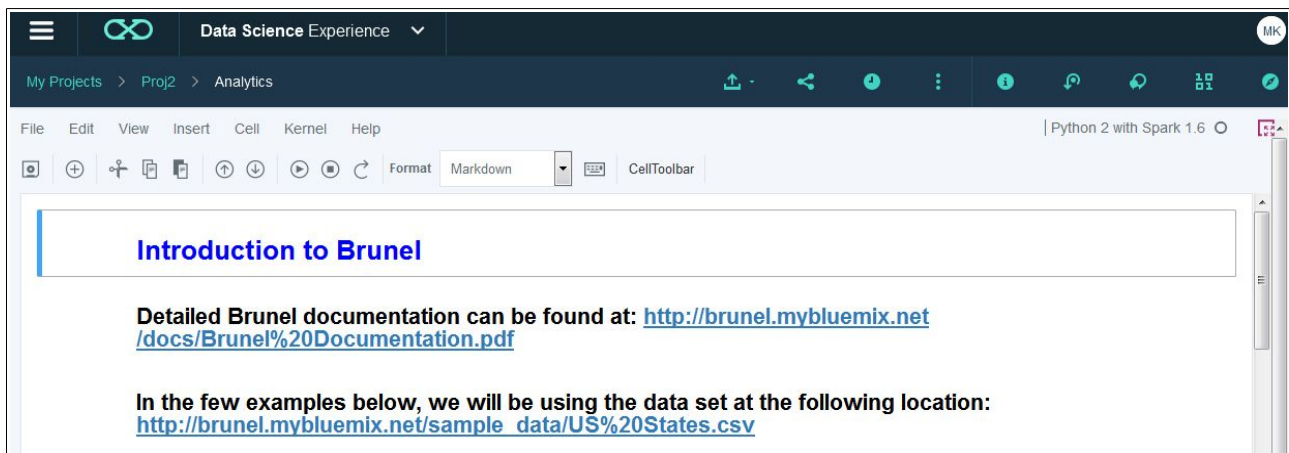


21. The proposed Spark-Analytics notebook introduces a few concepts

- From the IBM developed “Brunel” visualization library. (Note that the matplotlib visualizations from the famous Python “Pandas” library is another very popular way of visualizing data in Python notebooks. You can find examples from matplotlib in the “BlocPower” notebook suggested further down in this lab).
- An example of connecting from Python to a dashDB instance and loading the content of a table as a Spark data frame.
- A few examples for storing the data frame as a sparkSQL temporary table and executing a

few SQL statements on it.

22. The notebook will open as shown in the snapshot below. Take a few minutes to run the cells in the notebook.



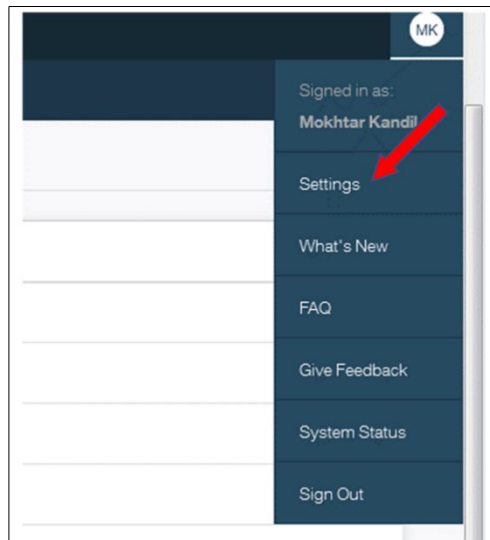
23. It is also recommended to analyze / study the “BlocPower” notebook available in GitHub for familiarity with many advanced concepts and visualizations using Python and Spark technology. This notebook requires uploading three data sets to your local project and adding them to the data assets section. A shiny R application application is also created subsequently to developing a machine learning model in the notebook. For details, please visit the following URL <https://github.com/IBMDDataScience/SparkSummitDemo>

24. You can create the BlocPower notebook from URL using the same steps as described just above for the Spark-Analytics-Demo notebook. The BlocPower notebook also requires three data sets to run. You can find those under the data directory in the GitHub repository mentioned above.

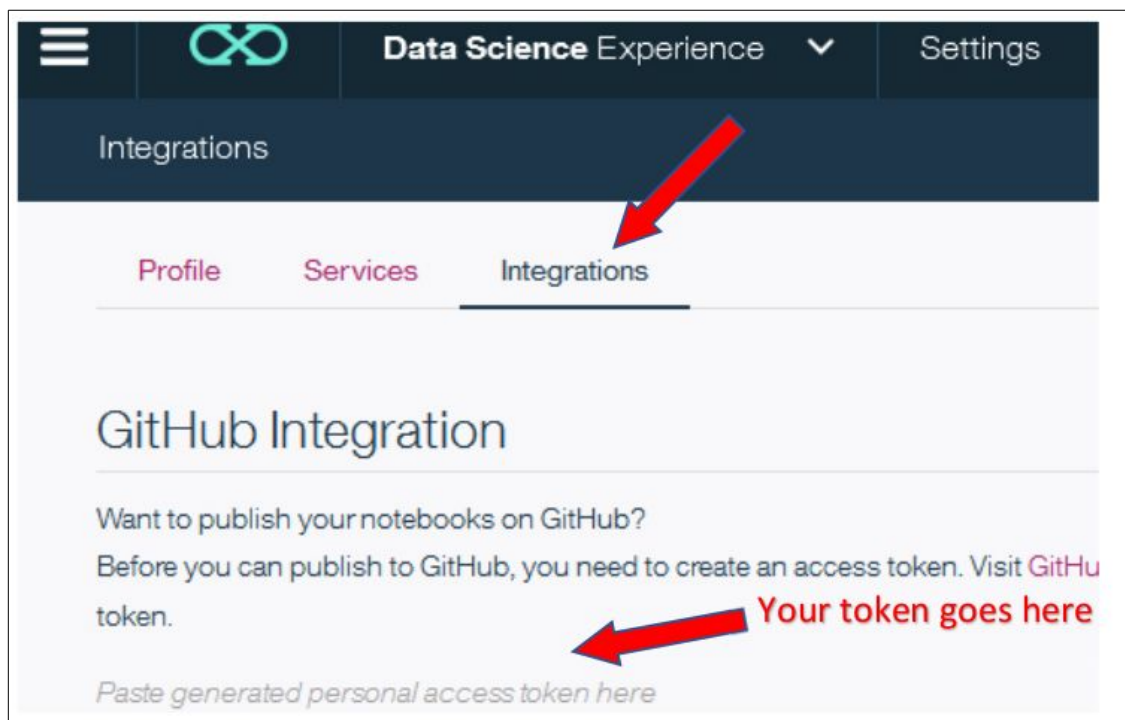
Download the following three files to your local drive and add them to the data assets of your project as described previously in this lab: **BlocPower_T.csv**, **CDD-HDD-Features.csv**, **HDD-Features.csv**.

25. To wrap up this lab, you will export your Spark-Analytics-Demo notebook created with a unique name to the repository at github.com/DScienceAtScale/Publish

26. In the top right corner of your screen, click on your initials and select “Settings” from the drop down menu.



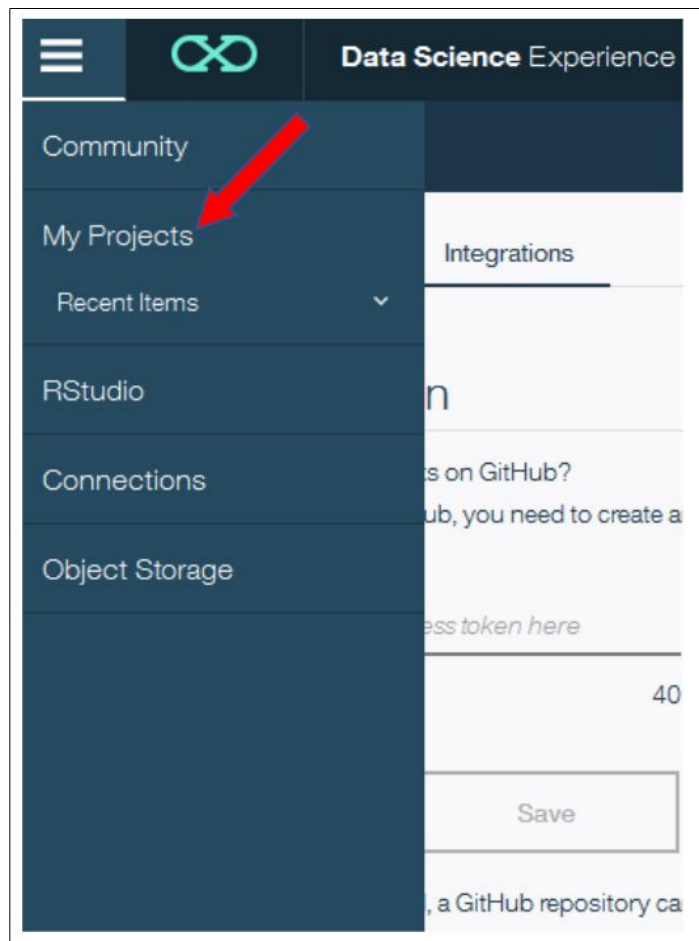
27. Select “Integrations” and copy paste the GitHub access token from the step below into the target section as shown in this screenshot.



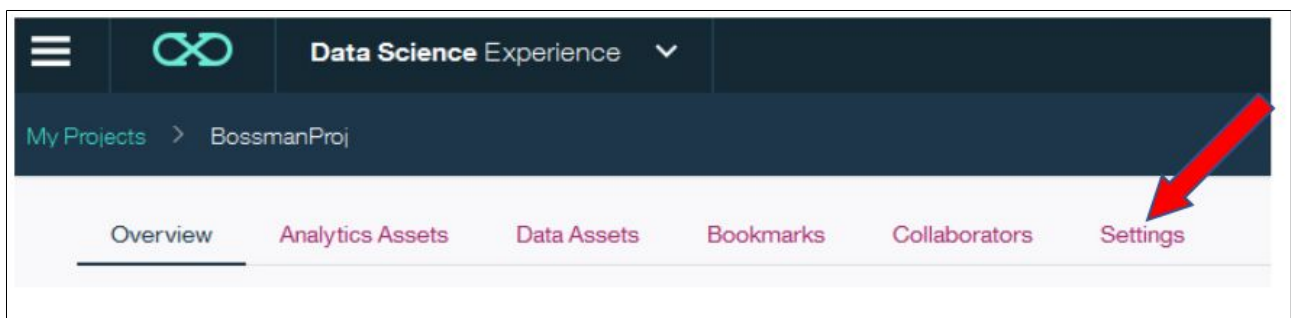
28. For the GitHub access token to the repository, use (copy/paste) the following string (and click on Save):

8838f67f7a81f27f9d5e8ca768909dfe2e50b2da

29. Going back to the Project page:



30. Click on your project and select the Settings tab.



31. Scroll to the bottom of the page, and connect your project with the target GitHub repository. Type the same string you see in the snapshot below into the Repository URL field (<https://github.com/DScienceAtScale/Publish>)

Connect to a GitHub Repository

Repository URL

`https://github.com/DScienceAtScale/Publish`

Add

32. You can now open your notebook and publish it to GitHub



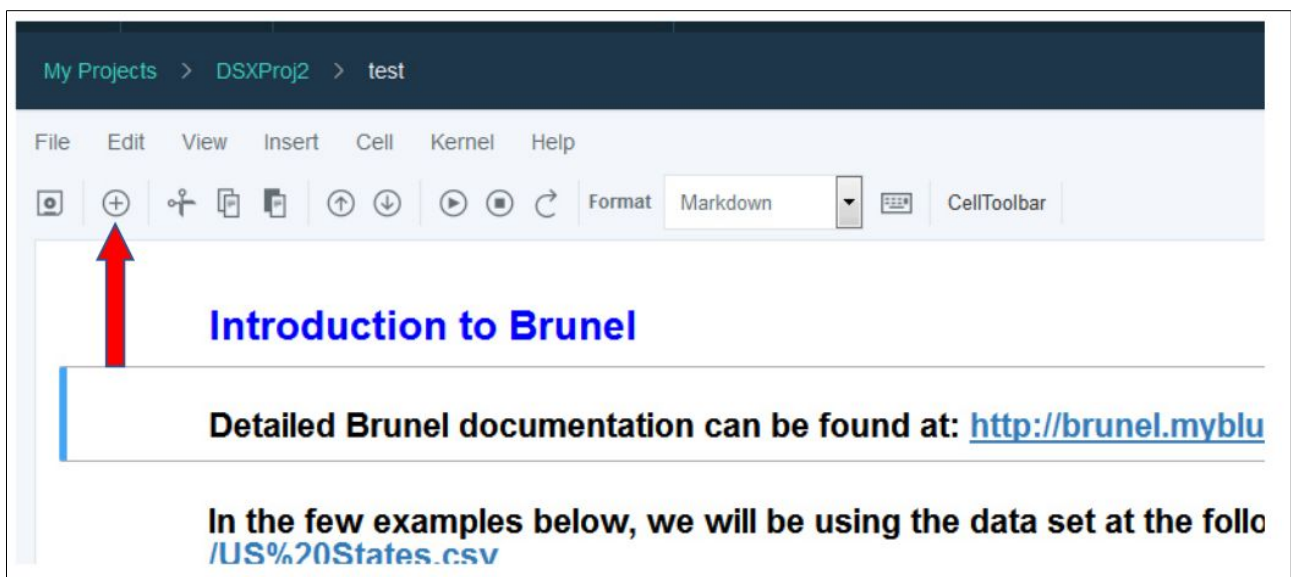
33. Save a second version of your notebook and export it again to GitHub

34. Verify the presence of your notebooks within the GitHub repository

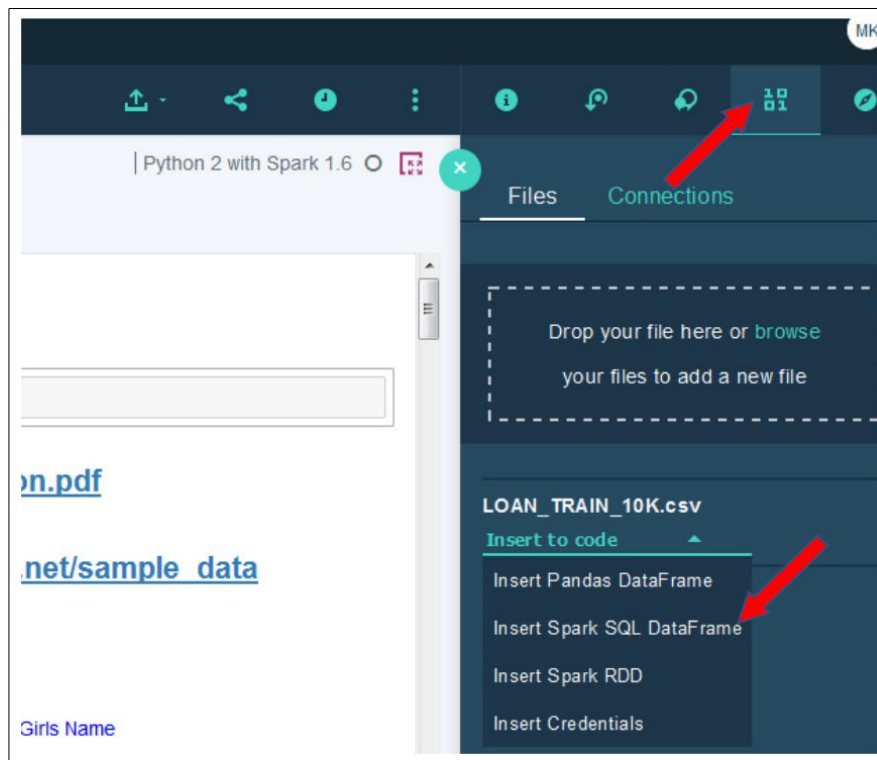
35. Give the GitHub URL to a colleague and have them open your latest notebook directly from GitHub

SECTION 2: Additional Notebook work (Optional)

1. Open your notebook created from Spark-Analytics-Demo.ipynb
2. Click on the “pencil” icon to make the notebook editable
3. Use the “+” icon in the notebook menu to insert new empty cells, preferably at the end of the notebook.



4. Navigate to the right side of the screen and show the data assets from the project available to this notebook. You should be able to see the LOAN_TRAIN_10K.csv file which was previously added as a data asset in this project (see screenshot below).
5. Use the “insert Spark SQL DataFrame menu” to inject code in the current cell to read the data file as a Spark data frame (see screenshot below).



6. Use the count method on the Spark data frame to count the number of entries in the data frame
7. Can you reuse some of the Brunel examples to the dataset you just added to the project above, or any other data set that is relevant to one of your projects or customers?
8. Can you modify the code that connects to a dashDB instance to reuse the credentials used at the beginning of this lab to load the LOANDATA_WEST_SAMPLE set?
9. Can you figure out how many entries were in the LOANDATA_WEST_SAMPLE dataset?