



Red Hat OpenShift Data Science self-managed 1.32

Getting started with Red Hat OpenShift Data Science self-managed

Learn how to work in an OpenShift Data Science environment

Red Hat OpenShift Data Science self-managed 1.32 Getting started with Red Hat OpenShift Data Science self-managed

Learn how to work in an OpenShift Data Science environment

Legal Notice

Copyright © 2023 Red Hat, Inc.

The text of and illustrations in this document are licensed by Red Hat under a Creative Commons Attribution–Share Alike 3.0 Unported license ("CC-BY-SA"). An explanation of CC-BY-SA is available at

<http://creativecommons.org/licenses/by-sa/3.0/>

. In accordance with CC-BY-SA, if you distribute this document or an adaptation of it, you must provide the URL for the original version.

Red Hat, as the licensor of this document, waives the right to enforce, and agrees not to assert, Section 4d of CC-BY-SA to the fullest extent permitted by applicable law.

Red Hat, Red Hat Enterprise Linux, the Shadowman logo, the Red Hat logo, JBoss, OpenShift, Fedora, the Infinity logo, and RHCE are trademarks of Red Hat, Inc., registered in the United States and other countries.

Linux[®] is the registered trademark of Linus Torvalds in the United States and other countries.

Java[®] is a registered trademark of Oracle and/or its affiliates.

XFS[®] is a trademark of Silicon Graphics International Corp. or its subsidiaries in the United States and/or other countries.

MySQL[®] is a registered trademark of MySQL AB in the United States, the European Union and other countries.

Node.js[®] is an official trademark of Joyent. Red Hat is not formally related to or endorsed by the official Joyent Node.js open source or commercial project.

The OpenStack[®] Word Mark and OpenStack logo are either registered trademarks/service marks or trademarks/service marks of the OpenStack Foundation, in the United States and other countries and are used with the OpenStack Foundation's permission. We are not affiliated with, endorsed or sponsored by the OpenStack Foundation, or the OpenStack community.

All other trademarks are the property of their respective owners.

Abstract

Log in and start up your notebook server to get started working with your notebooks in Jupyter.


Table of Contents

CHAPTER 1. LOGGING IN TO OPENSIFT DATA SCIENCE	3
CHAPTER 2. THE OPENSIFT DATA SCIENCE USER INTERFACE	4
2.1. GLOBAL NAVIGATION	5
2.2. SIDE NAVIGATION	5
CHAPTER 3. NOTIFICATIONS IN OPENSIFT DATA SCIENCE	8
CHAPTER 4. CREATING A DATA SCIENCE PROJECT	9
CHAPTER 5. CREATING A PROJECT WORKBENCH	10
5.1. LAUNCHING JUPYTER AND STARTING A NOTEBOOK SERVER	11
5.2. OPTIONS FOR NOTEBOOK SERVER ENVIRONMENTS	13
CHAPTER 6. TUTORIALS FOR DATA SCIENTISTS	19
6.1. ACCESSING TUTORIALS	22
CHAPTER 7. ENABLING SERVICES CONNECTED TO OPENSIFT DATA SCIENCE	23
CHAPTER 8. DISABLING APPLICATIONS CONNECTED TO OPENSIFT DATA SCIENCE	25
8.1. REMOVING DISABLED APPLICATIONS FROM OPENSIFT DATA SCIENCE	26
CHAPTER 9. SUPPORT REQUIREMENTS AND LIMITATIONS	27
9.1. SUPPORTED BROWSERS	27
9.2. SUPPORTED SERVICES	27
9.3. SUPPORTED PACKAGES	28
CHAPTER 10. COMMON QUESTIONS	29
CHAPTER 11. TROUBLESHOOTING COMMON PROBLEMS IN JUPYTER FOR ADMINISTRATORS	30
11.1. A USER RECEIVES A 404: PAGE NOT FOUND ERROR WHEN LOGGING IN TO JUPYTER	30
11.2. A USER'S NOTEBOOK SERVER DOES NOT START	30
11.3. THE USER RECEIVES A DATABASE OR DISK IS FULL ERROR OR A NO SPACE LEFT ON DEVICE ERROR WHEN THEY RUN NOTEBOOK CELLS	31
CHAPTER 12. TROUBLESHOOTING COMMON PROBLEMS IN JUPYTER FOR USERS	33
12.1. I SEE A 403: FORBIDDEN ERROR WHEN I LOG IN TO JUPYTER	33
12.2. MY NOTEBOOK SERVER DOES NOT START	33
12.3. I SEE A DATABASE OR DISK IS FULL ERROR OR A NO SPACE LEFT ON DEVICE ERROR WHEN I RUN MY NOTEBOOK CELLS	33

CHAPTER 1. LOGGING IN TO OPENSIFT DATA SCIENCE

Log in to OpenShift Data Science from a browser for easy access to Jupyter and your data science projects.

Procedure

1. Browse to the OpenShift Data Science instance URL and click **Log in with OpenShift**
 - If you are a data scientist user, your administrator must provide you with the OpenShift Data Science instance URL, for example, **`https://rhods-dashboard-redhat-ods-applications.apps.example.abc1.p1.openshiftapps.com/`**
 - If you have access to OpenShift Container Platform, you can browse to the OpenShift Container Platform web console and click the **Application Launcher** () → **Red Hat OpenShift Data Science**.
2. Click the name of your identity provider, for example, **GitHub**.
3. Enter your credentials and click **Log in** (or equivalent for your identity provider).

Verification

- OpenShift Data Science opens on the **Enabled applications** page.

Troubleshooting

- If you see **An authentication error occurred** or **Could not create user** when you try to log in:
 - You might have entered your credentials incorrectly. Confirm that your credentials are correct.
 - You might have an account in more than one configured identity provider. If you have logged in with a different identity provider previously, try again with that identity provider.

Additional resources

- [Launching Jupyter and starting a notebook server](#)

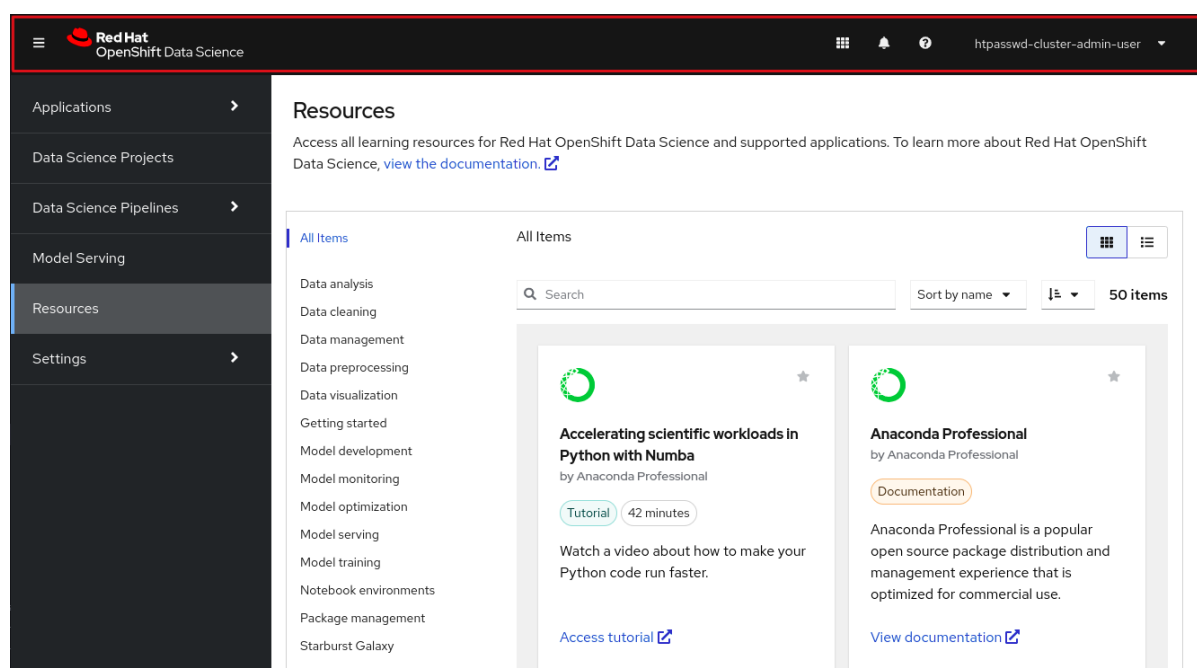
CHAPTER 2. THE OPENSIFT DATA SCIENCE USER INTERFACE

The Red Hat OpenShift Data Science interface is based on the OpenShift web console user interface.

The Red Hat OpenShift Data Science user interface is divided into several areas:

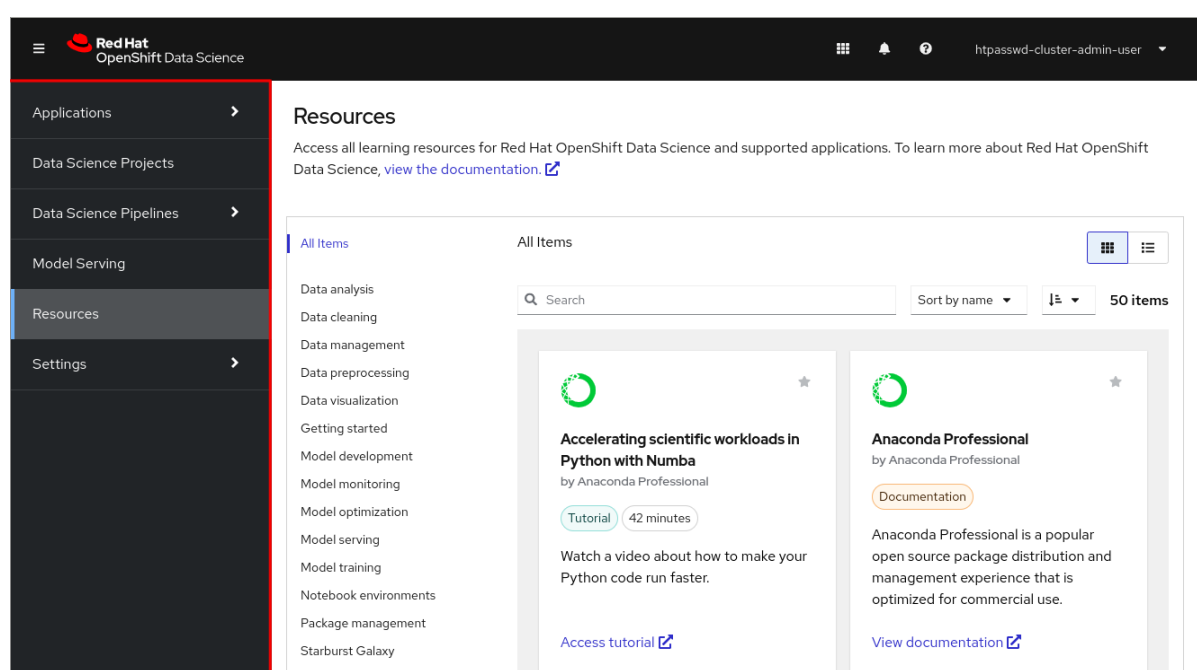
- The global navigation bar, which provides access to useful controls, such as **Help** and **Notifications**.

Figure 2.1. The global navigation bar



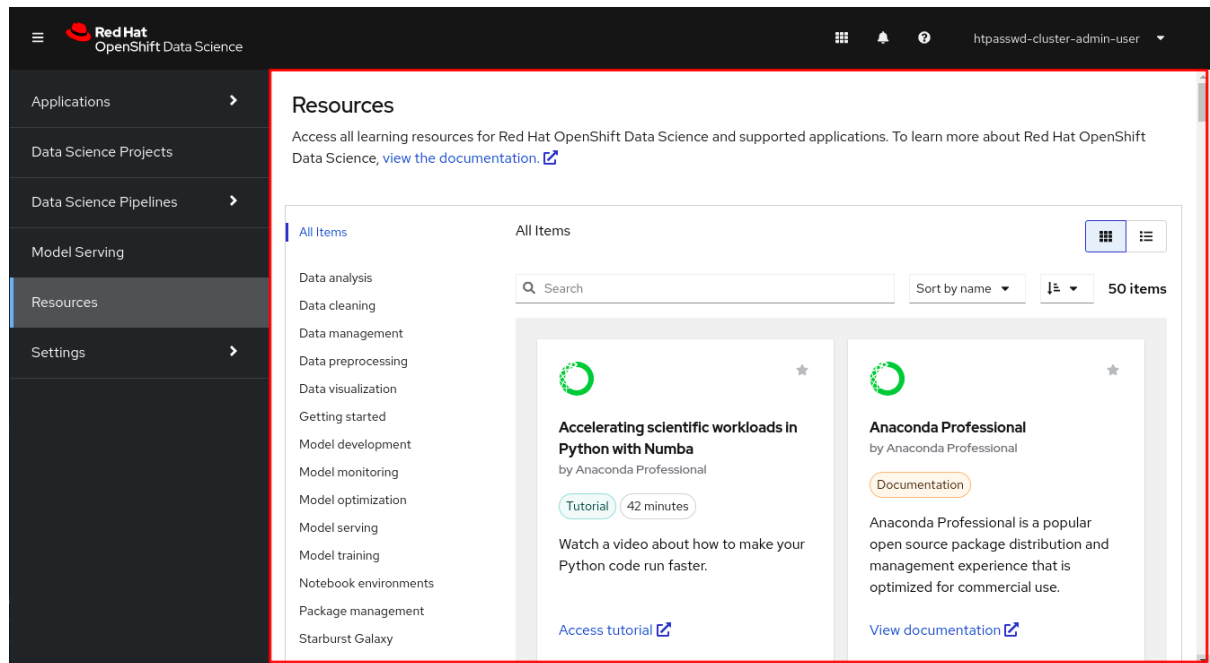
- The side navigation menu, which contains different categories of pages available in OpenShift Data Science.

Figure 2.2. The side navigation menu






- The main display area, which displays the current page and shares space with any drawers currently displaying information, such as notifications or quick start guides. The main display area also displays the **Notebook server control panel** where you can launch Jupyter by starting and configuring a notebook server. Administrators can also use the **Notebook server control panel** to manage other users' notebook servers.

Figure 2.3. The main display area



2.1. GLOBAL NAVIGATION

There are four items in the top navigation:

- The **Toggle side navigation menu** button () toggles whether or not the side navigation is displayed.
- The **Notifications** button () opens and closes the **Notifications** drawer, letting you read current and previous notifications in more detail.
- The **Help** menu () provides a link to create a ticket with Red Hat Support and access the OpenShift Data Science documentation.
- The **User** menu displays the name of the currently logged-in user and provides access to the **Log out** button.

2.2. SIDE NAVIGATION

There are three main sections in the side navigation:

Applications → Enabled

The **Enabled** page displays applications that are enabled and ready to use on OpenShift Data Science. This page is the default landing page for OpenShift Data Science.

Click the **Launch application** button on an application card to open the application interface in a new tab. If an application has an associated quick start tour, click the drop-down menu on the application's card and select **Open quick start** to access it. This page also displays applications and components

that have been disabled by your administrator. Disabled applications are denoted with **Disabled** on the application's card. Click **Disabled** on the application's card to access links allowing you to remove the card itself, and to re-validate its license, if the license had previously expired.

Applications → Explore

The **Explore** page displays applications that are available for use with OpenShift Data Science. Click on a card for more information about the application or to access the **Enable** button. The **Enable** button is visible only if an application does not require an OpenShift Operator installation.

Data science projects

The **Data science projects** page allows you to organize your data science work into a single project. From this page, you can create and manage data science projects. You can also enhance the capabilities of your data science project by adding workbenches, adding storage to your project's cluster, adding data connections, and adding model servers.

Data Science Pipelines → Pipelines

The **Pipelines** page allows you to import, manage, track, and view data science pipelines. Using Red Hat OpenShift Data Science pipelines, you can standardize and automate machine learning workflows to enable you to develop and deploy your data science models.

Data Science Pipelines → Runs

The **Runs** page allows you to define, manage, and track executions of a data science pipeline. A pipeline run is a single execution of a data science pipeline. You can also view a record of previously executed and scheduled runs for your data science project.

Model Serving

The **Model Serving** page allows you to manage and view the status of your deployed models. You can use this page to deploy data science models to serve intelligent applications, or to view existing deployed models. You can also determine the inference endpoint of a deployed model.

Resources

The **Resources** page displays learning resources such as documentation, how-to material, and quick start tours. You can filter visible resources using the options displayed on the left, or enter terms into the search bar.

Settings → Notebook images

The **Notebook image settings** page allows you to configure custom notebook images that cater to your project's specific requirements. After you have added custom notebook images to your deployment of OpenShift Data Science, they are available for selection when creating a notebook server.

Settings → Cluster settings

The **Cluster settings** page allows you perform the following administrative tasks on your cluster:

- Enable or disable Red Hat's ability to collect data about OpenShift Data Science usage on your cluster.
- Configure how resources are claimed within your cluster by changing the default size of the cluster's persistent volume claim (PVC).
- Reduce resource usage in your OpenShift Data Science deployment by stopping notebook servers that have been idle.
- Schedule notebook pods on tainted nodes by adding tolerations.

Settings → User management

The **User and group settings** page allows you to define OpenShift Data Science user group and admin group membership.

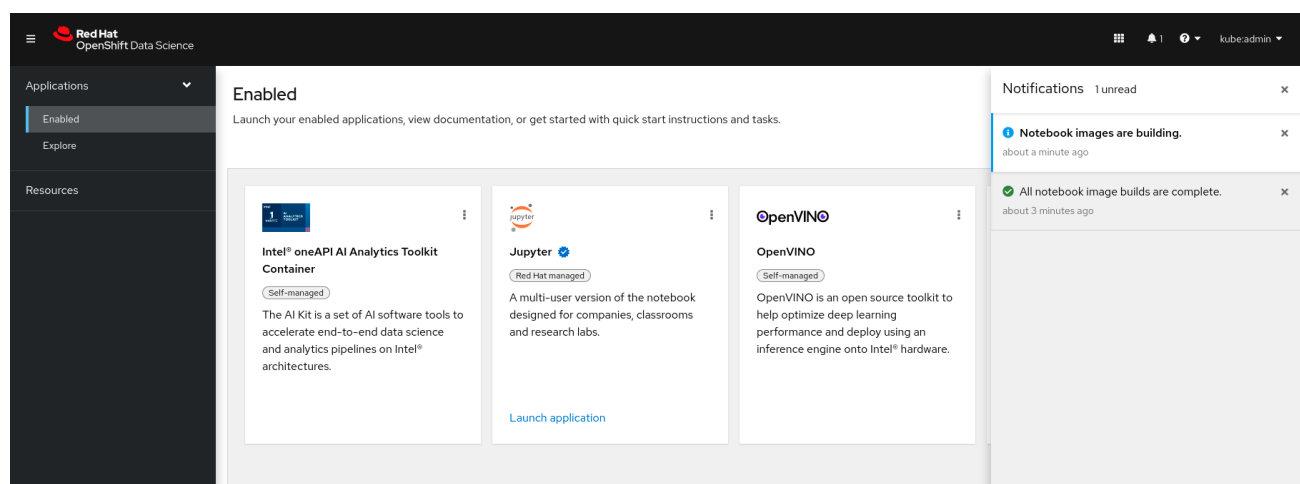
CHAPTER 3. NOTIFICATIONS IN OPENSIFT DATA SCIENCE

Red Hat OpenShift Data Science displays notifications when important events happen in the cluster.

Notification messages are displayed in the lower left corner of the Red Hat OpenShift Data Science interface when they are triggered.

If you miss a notification message, click the **Notifications** button () to open the **Notifications** drawer and view unread messages.

Figure 3.1. The Notifications drawer



CHAPTER 4. CREATING A DATA SCIENCE PROJECT

To start your data science work, create a data science project. Creating a project helps you organize your work in one place. You can also enhance the capabilities of your data science project by adding workbenches, adding storage to your project's cluster, adding data connections, and adding model servers.

Prerequisites

- You have logged in to Red Hat OpenShift Data Science.
- If you are using specialized OpenShift Data Science groups, you are part of the user group or admin group (for example, **rhods-users** or **rhods-admin**) in OpenShift.

Procedure

1. From the OpenShift Data Science dashboard, click **Data Science Projects**.
The **Data science projects** page opens.
2. Click **Create data science project**.
The **Create a data science project** dialog opens.
3. Enter a **name** for your data science project.
4. Optional: Edit the **resource name** for your data science project. The resource name must consist of lowercase alphanumeric characters, -, and must start and end with an alphanumeric character.
5. Enter a **description** for your data science project.
6. Click **Create**.
A project details page opens. From here, you can create workbenches, add cluster storage, and add data connections to your project.

Verification

- The data science project that you created is displayed on the **Data science projects** page.

CHAPTER 5. CREATING A PROJECT WORKBENCH

To examine and work with data models in an isolated area, you can create a workbench. This workbench enables you to create a new Jupyter notebook from an existing notebook container image to access its resources and properties. For data science projects that require data to be retained, you can add container storage to the workbench you are creating.

Prerequisites

- You have logged in to Red Hat OpenShift Data Science.
- If you are using specialized OpenShift Data Science groups, you are part of the user group or admin group (for example, **rhods-users** or **rhods-admin**) in OpenShift.
- You have created a data science project that you can add a workbench to.

Procedure

1. From the OpenShift Data Science dashboard, click **Data Science Projects**.
The **Data science projects** page opens.
2. Click the name of the project that you want to add the workbench to.
The **Details** page for the project opens.
3. Click **Create workbench** in the **Workbenches** section.
The **Create workbench** page opens.
4. Configure the properties of the workbench you are creating.
 - a. Enter a **name** for your workbench.
 - b. Enter a **description** for your workbench.
 - c. Select the **notebook image** to use for your workbench server.
 - d. Select the **container size** for your server.
 - e. Optional: Select and specify values for any new **environment variables**.



NOTE

To enable data science pipelines in JupyterLab in self-managed deployments, create the following environment variable:

PIPELINES_SSL_SA_CERTS=/var/run/secrets/kubernetes.io/serviceaccount/ca.crt

- f. Configure the storage for your OpenShift Data Science cluster.
 - i. Select **Create new persistent storage** to create storage that is retained after you log out of OpenShift Data Science. Fill in the relevant fields to define the storage.
 - ii. Select **Use existing persistent storage** to reuse existing storage then select the storage from the **Persistent storage** list.
5. Click **Create workbench**.

Verification

- The workbench that you created appears on the **Details** page for the project.
- Any cluster storage that you associated with the workbench during the creation process appears on the **Details** page for the project.
- The **Status** column, located in the **Workbenches** section of the **Details** page, displays a status of **Starting** when the workbench server is starting, and **Running** when the workbench has successfully started.

5.1. LAUNCHING JUPYTER AND STARTING A NOTEBOOK SERVER

Launch Jupyter and start a notebook server to start working with your notebooks.

Prerequisites

- You have logged in to Red Hat OpenShift Data Science.
- You know the names and values you want to use for any environment variables in your notebook server environment, for example, **AWS_SECRET_ACCESS_KEY**.
- If you want to work with a very large data set, work with your administrator to proactively increase the storage capacity of your notebook server.

Procedure

1. Locate the **Jupyter** card on the **Enabled applications** page.
2. Click **Launch application**.
If you see an **Access permission needed** message, you are not in the default user group or the default administrator group for OpenShift Data Science. Contact your administrator so that they can add you to the correct group using [Adding users for OpenShift Data Science](#) .

If you have not previously authorized the **jupyter-nb-`<username>`** service account to access your account, the **Authorize Access** page appears prompting you to provide authorization. Inspect the permissions selected by default, and click the **Allow selected permissions** button.

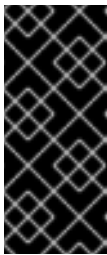
If your credentials are accepted, the **Notebook server control panel** opens displaying the **Start a notebook server** page.
3. Start a notebook server.
This is not required if you have previously opened Jupyter.
 - a. Select the **Notebook image** to use for your server.
 - b. If the notebook image contains multiple versions, select the version of the notebook image from the **Versions** section.



NOTE

When a new version of a notebook image is released, the previous version remains available and supported on the cluster. This gives you time to migrate your work to the latest version of the notebook image.

- c. Select the **Container size** for your server.
- d. Optional: Select the **Number of GPUs** (graphics processing units) for your server.



IMPORTANT

Using GPUs to accelerate workloads is only supported with the PyTorch, TensorFlow, and CUDA notebook server images. In addition, you can specify the number of GPUs required for your notebook server only if GPUs are enabled on your cluster. To learn how to enable GPU support, see [Enabling GPU support in OpenShift Data Science](#).

- e. Optional: Select and specify values for any new **Environment variables**.
The interface stores these variables so that you only need to enter them once. Example variable names for common environment variables are automatically provided for frequently integrated environments and frameworks, such as Amazon Web Services (AWS).



IMPORTANT

Ensure that you select the **Secret** checkbox for any variables with sensitive values that must be kept private, such as passwords.

- f. Optional: Select the **Start server in current tab** checkbox if necessary.
- g. Click **Start server**.
The **Starting server** progress indicator appears. Click **Expand event log** to view additional information about the server creation process. Depending on the deployment size and resources you requested, starting the server can take up to several minutes. Click **Cancel** to cancel the server creation.

After the server starts, you see one of the following behaviors:

- If you previously selected the **Start server in current tab** checkbox, the JupyterLab interface opens in the current tab of your web browser.
- If you did not previously select the **Start server in current tab** checkbox, the **Starting server** dialog box prompts you to open the server in a new browser tab or in the current tab.
The JupyterLab interface opens according to your selection.

Verification

- The JupyterLab interface opens.

Additional resources

[Options for notebook server environments](#).

Troubleshooting

- If you see the "Unable to load notebook server configuration options" error message, contact your administrator so that they can review the logs associated with your Jupyter pod and determine further details about the problem.

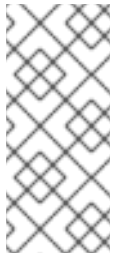
5.2. OPTIONS FOR NOTEBOOK SERVER ENVIRONMENTS

When you start Jupyter for the first time, or after stopping your notebook server, you must select server options in the **Start a notebook server** wizard so that the software and variables that you expect are available on your server. This section explains the options available in the **Start a notebook server** wizard in detail.

The **Start a notebook server** page is divided into several sections:

Notebook image

Specifies the container image that your notebook server is based on. Different notebook images have different packages installed by default. If the notebook image contains multiple versions, you can select the notebook image version to use from the **Versions** section.



NOTE

Notebook images are supported for a minimum of one year. Major updates to pre-configured notebook images occur approximately every six months. Therefore, two supported notebook images are typically available at any given time. To use the latest package versions, Red Hat recommends that you use the most recently added notebook image.

After you start a notebook image, you can check which Python packages are installed on your notebook server and which version of the package you have by running the **pip** tool in a notebook cell.

The following table shows the package versions used in the available notebook images:

Table 5.1. Notebook image options

Image name	Image version	Preinstalled packages
CUDA	2 (Recommended)	<ul style="list-style-type: none"> Python 3.9 CUDA 11.8 JupyterLab 3.5 Notebook 6.5
	1	<ul style="list-style-type: none"> Python 3.8 CUDA 11.4 JupyterLab 3.2 Notebook 6.4

Image name	Image version	Preinstalled packages
Minimal Python (default)	2 (Recommended)	<ul style="list-style-type: none"> ● Python 3.9 ● JupyterLab 3.5 ● Notebook 6.5
	1	<ul style="list-style-type: none"> ● Python 3.8 ● JupyterLab 3.2 ● Notebook 6.4
PyTorch	2 (Recommended)	<ul style="list-style-type: none"> ● Python 3.9 ● JupyterLab 3.5 ● Notebook 6.5 ● PyTorch 1.13 ● CUDA 11.7 ● TensorBoard 2.11 ● Boto3 1.26 ● Kafka-Python 2.0 ● Matplotlib 3.6 ● Numpy 1.24 ● Pandas 1.5 ● Scikit-learn 1.2 ● SciPy 1.10

Image name	Image version	Preinstalled packages
	1	<ul style="list-style-type: none"> ● Python 3.8 ● JupyterLab 3.2 ● Notebook 6.4 ● PyTorch 1.8 ● CUDA 10.2 ● TensorBoard 2.6 ● Boto3 1.17 ● Kafka-Python 2.0 ● Matplotlib 3.4 ● Numpy 1.19 ● Pandas 1.2 ● Scikit-learn 0.24 ● SciPy 1.6
Standard Data Science	2 (Recommended)	<ul style="list-style-type: none"> ● Python 3.9 ● JupyterLab 3.5 ● Notebook 6.5 ● Boto3 1.26 ● Kafka-Python 2.0 ● Matplotlib 3.6 ● Pandas 1.5 ● Numpy 1.24 ● Scikit-learn 1.2 ● SciPy 1.10

Image name	Image version	Preinstalled packages
	1	<ul style="list-style-type: none"> ● Python 3.8 ● JupyterLab 3.2 ● Notebook 6.4 ● Boto3 1.17 ● Kafka-Python 2.0 ● Matplotlib 3.4 ● Pandas 1.2 ● Numpy 1.19 ● Scikit-learn 0.24 ● SciPy 1.6
TensorFlow	2 (Recommended)	<ul style="list-style-type: none"> ● Python 3.9 ● JupyterLab 3.5 ● Notebook 6.5 ● TensorFlow 2.11 ● TensorBoard 2.11 ● CUDA 11.8 ● Boto3 1.26 ● Kafka-Python 2.0 ● Matplotlib 3.6 ● Numpy 1.24 ● Pandas 1.5 ● Scikit-learn 1.2 ● SciPy 1.10

Image name	Image version	Preinstalled packages
	1	<ul style="list-style-type: none"> ● Python 3.8 ● JupyterLab 3.2 ● Notebook 6.4 ● TensorFlow 2.7 ● TensorBoard 2.6 ● CUDA 11.4 ● Boto3 1.17 ● Kafka-Python 2.0 ● Matplotlib 3.4 ● Numpy 1.19 ● Pandas 1.2 ● Scikit-learn 0.24 ● SciPy 1.6
TrustyAI	1	<ul style="list-style-type: none"> ● Python 3.9 ● JupyterLab 3.5 ● Notebook 6.5 ● TrustyAI 0.2 ● Boto3 1.26 ● Kafka-Python 2.0 ● Matplotlib 3.6 ● Numpy 1.24 ● Pandas 1.5 ● Scikit-learn 1.2 ● SciPy 1.10

Deployment size

Specifies the compute resources available on your notebook server.

Container size controls the number of CPUs, the amount of memory, and the minimum and maximum request capacity of the container.

Number of GPUs specifies the number of graphics processing units attached to the container.



IMPORTANT

Using GPUs to accelerate workloads is only supported with the PyTorch, TensorFlow, and CUDA notebook server images. In addition, you can specify the number of GPUs required for your notebook server only if GPUs are enabled on your cluster. To learn how to enable GPU support, see [Enabling GPU support in OpenShift Data Science](#).

Environment variables

Specifies the name and value of variables to be set on the notebook server. Setting environment variables during server startup means that you do not need to define them in the body of your notebooks, or with the Jupyter command line interface. Some recommended environment variables are shown in the table.

Table 5.2. Recommended environment variables

Environment variable option	Recommended variable names
AWS	<ul style="list-style-type: none">• AWS_ACCESS_KEY_ID specifies your Access Key ID for Amazon Web Services.• AWS_SECRET_ACCESS_KEY specifies your Secret access key for the account specified in AWS_ACCESS_KEY_ID.

Additional resources

- [Launching Jupyter and starting a notebook server](#)

CHAPTER 6. TUTORIALS FOR DATA SCIENTISTS

To help you get started quickly, you can access learning resources for Red Hat OpenShift Data Science and its supported applications. These resources are available on the **Resources** tab of the Red Hat OpenShift Data Science user interface.

Table 6.1. Tutorials

Resource Name	Description
Accelerating scientific workloads in Python with Numba	Watch a video about how to make your Python code run faster.
Building interactive visualizations and dashboards in Python	Explore a variety of data across multiple notebooks and learn how to deploy full dashboards and applications.
Building machine learning models with scikit-learn	Learn how to build machine learning models with scikit-learn for supervised learning, unsupervised learning, and classification problems.
Building a binary classification model	Train a model to predict if a customer is likely to subscribe to a bank promotion.
Choosing Python tools for data visualization	Use the PyViz.org website to help you decide on the best open source Python data visualization tools for you.
Exploring Anaconda for data science	Learn about Anaconda, a freemium open source distribution of the Python and R programming languages.
Getting started with Pachyderm concepts	Learn Pachyderm's main concepts by creating pipelines that perform edge detection on a few images.
GPU Computing in Python with Numba	Learn how to create GPU accelerated functions using Numba.
Run a Python notebook to generate results in IBM Watson OpenScale	Run a Python notebook to create, train, and deploy a machine learning model.
Running an AutoAI experiment to build a model	Watch a video about building a binary classification model for a marketing campaign.
Training a regression model in Pachyderm	Learn how to create a sample housing data repository using a Pachyderm cluster to run experiments, analyze data, and set up regression.

Resource Name	Description
Using Dask for parallel data analysis	Analyze medium-sized datasets in parallel locally using Dask, a parallel computing library that scales the existing Python ecosystem.
Using Jupyter notebooks in Watson Studio	Watch a video about working with Jupyter notebooks in Watson Studio.
Using Pandas for data analysis in Python	Learn how to use pandas, a data analysis library for the Python programming language.

Table 6.2. Quick start guides

Resource Name	Description
Creating a Jupyter notebook	Create a Jupyter notebook in JupyterLab.
Creating a Machine Learning Model using the NVIDIA GPU Add-on	Creating a Machine Learning model on Jupyter that uses the GPUs that you have made available.
Creating an Anaconda-enabled Jupyter notebook	Create an Anaconda-enabled Jupyter notebook and access Anaconda packages that are curated for security and compatibility.
Deploying a model with Watson Studio	Import a notebook in Watson Studio and use AutoAI to build and deploy a model.
Deploying a sample Python application using Flask and OpenShift	Deploy your data science model out of a Jupyter notebook and into a Flask application to use as a development sandbox.
Importing Pachyderm Beginner Tutorial Notebook	Load Pachyderm's beginner tutorial notebook and learn about Pachyderm's main concepts such as data repositories, pipelines, and using the pachctl CLI from your cells.
Installing and verifying the NVIDIA GPU Add-on	Learn how to install and verify that Jupyter detects the GPUs available for use.
Opening and updating a SKLearn model with canary deployment	Open a SKLearn model and update it using canary deployment practices.
Querying data with Starburst Galaxy	Learn to query data using Starburst Galaxy from a Jupyter notebook.

Resource Name	Description
Securing a deployed model using Red Hat OpenShift API Management	Protect a model service API using Red Hat OpenShift API Management.
Using the Intel® oneAPI AI Analytics Toolkit (AI Kit) Notebook	Run a data science notebook sample with the Intel® oneAPI AI Analytics Toolkit.
Using the OpenVINO toolkit	Quantize an ONNX computer vision model using the OpenVINO model optimizer and use the result for inference from a notebook.

Table 6.3. How to guides

Resource Name	Description
How to choose between notebook runtime environment options	Explore available options for configuring your notebook runtime environment.
How to clean, shape, and visualize data	Learn how to clean and shape tabular data using IBM Watson Studio data refinery.
How to create a connection to access data	Learn how to create connections to various data sources across the platform.
How to create a deployment space	Learn how to create a deployment space for machine learning.
How to create a notebook in Watson Studio	Learn how to create a basic Jupyter notebook in Watson Studio.
How to create a project in Watson Studio	Learn how to create an analytics project in Watson Studio.
How to create a project that integrates with Git	Learn how to add assets from a Git repository into a project.
How to install Python packages on your notebook server	Learn how to install additional Python packages on your notebook server.
How to load data into a Jupyter notebook	Learn how to integrate data sources into a Jupyter notebook by loading data.
How to serve a model using OpenVINO Model Server	Learn how to deploy optimized models with the OpenVINO Model Server using OpenVINO custom resources.

Resource Name	Description
How to set up Watson OpenScale	Learn how to track and measure outcomes from models with OpenScale.
How to update notebook server settings	Learn how to update the settings or the notebook image on your notebook server.
How to use data from Amazon S3 buckets	Learn how to connect to data in S3 Storage using environment variables.
How to view installed packages on your notebook server	Learn how to see which packages are installed on your running notebook server.

6.1. ACCESSING TUTORIALS

You can access learning resources for Red Hat OpenShift Data Science and supported applications.

Prerequisites

- Ensure that you have logged in to Red Hat OpenShift Data Science.
- You have logged in to the OpenShift Container Platform web console.

Procedure

1. On the Red Hat OpenShift Data Science home page, click **Resources**. The **Resources** page opens.
2. Click **Access tutorial** on the relevant card.

Verification

- You can view and access the learning resources for Red Hat OpenShift Data Science and supported applications.

Additional resources

- [Common questions](#).

CHAPTER 7. ENABLING SERVICES CONNECTED TO OPENSIFT DATA SCIENCE

You must enable SaaS-based services, such as Anaconda Professional Edition, before using them with Red Hat OpenShift Data Science. On-cluster services are enabled automatically.

Typically, you can install services, or enable services connected to OpenShift Data Science using one of the following methods:

- Enabling the service from the **Explore** page on the OpenShift Data Science dashboard, as documented in the following procedure.
- Installing the Operator for the service from OperatorHub. OperatorHub is a web console for cluster administrators to discover and select Operators to install on their cluster. It is deployed by default in OpenShift Container Platform ([Installing from OperatorHub using the web console](#)).



NOTE

Deployments containing Operators installed from OperatorHub may not be fully supported by Red Hat.

- Installing the Operator for the service from Red Hat Marketplace ([Install Operators](#)).
- Installing the service as an Operator to your OpenShift Container Platform cluster ([Adding Operators to a cluster](#)).

For some services (such as Jupyter), the service endpoint is available on the tile for the service on the **Enabled** page of OpenShift Data Science. Certain services cannot be accessed directly from their tiles, for example, OpenVINO and Anaconda provide notebook images for use in Jupyter and do not provide an endpoint link from their tile. Additionally, it may be useful to store these endpoint URLs as environment variables for easy reference in a notebook environment.

Some independent software vendor (ISV) applications must be installed in specific OpenShift Data Science Operator namespaces. However, do not install ISV applications in namespaces associated with OpenShift Data Science Operators unless you are specifically directed to do so on the card for the application's card on the dashboard.

To help you get started quickly, you can access the service's learning resources and documentation on the **Resources** page, or by clicking the relevant link on the tile for the service on the **Enabled** page.

Prerequisites

- You have logged in to OpenShift Data Science.
- Your administrator has installed or configured the service on your OpenShift cluster.

Procedure

1. On the OpenShift Data Science home page, click **Explore**.
The **Explore** page opens.
2. Click the card of the service that you want to enable.
3. Click **Enable** on the drawer for the service.

4. If prompted, enter the service's key and click **Connect**.
5. Click **Enable** to confirm that you are enabling the service.

Verification

- The service that you enabled appears on the **Enabled** page.
- The service endpoint is displayed on the tile for the service on the **Enabled** page.

CHAPTER 8. DISABLING APPLICATIONS CONNECTED TO OPENSIFT DATA SCIENCE

You can disable applications and components so that they do not appear on the OpenShift Data Science dashboard when you no longer want to use them, for example, when data scientists no longer use an application or when the application's license expires.

Disabling unused applications allows your data scientists to manually remove these application cards from their OpenShift Data Science dashboard so that they can focus on the applications that they are most likely to use. See [Removing disabled applications from OpenShift Data Science](#) for more information about manually removing application cards.



IMPORTANT

Do not follow this procedure when disabling the following applications:

- Anaconda Professional Edition. You cannot manually disable Anaconda Professional Edition. It is automatically disabled only when its license expires.
- You have logged in to the OpenShift Container Platform web console.
- You are assigned the **cluster-admin** role in OpenShift Container Platform.
- You have installed or configured the service on your OpenShift Container Platform cluster.
- The application or component that you want to disable is enabled and appears on the **Enabled** page.

Procedure

1. In the OpenShift Container Platform web console, change into the **Administrator** perspective.
2. Change into the **redhat-ods-applications** project.
3. Click **Operators → Installed Operators**.
4. Click on the Operator that you want to uninstall. You can enter a keyword into the **Filter by name** field to help you find the Operator faster.
5. Delete any Operator resources or instances by using the tabs in the Operator interface. During installation, some Operators require the administrator to create resources or start process instances using tabs in the Operator interface. These must be deleted before the Operator can uninstall correctly.
6. On the **Operator Details** page, click the **Actions** drop-down menu and select **Uninstall Operator**.
An **Uninstall Operator?** dialog box is displayed.
7. Select **Uninstall** to uninstall the Operator, Operator deployments, and pods. After this is complete, the Operator stops running and no longer receives updates.



IMPORTANT

Removing an Operator does not remove any custom resource definitions or managed resources for the Operator. Custom resource definitions and managed resources still exist and must be cleaned up manually. Any applications deployed by your Operator and any configured off-cluster resources continue to run and must be cleaned up manually.

Verification

- The Operator is uninstalled from its target clusters.
- The Operator no longer appears on the **Installed Operators** page.
- The disabled application is no longer available for your data scientists to use, and is marked as **Disabled** on the **Enabled** page of the OpenShift Data Science dashboard. This action may take a few minutes to occur following the removal of the Operator.

8.1. REMOVING DISABLED APPLICATIONS FROM OPENSIFT DATA SCIENCE

After your administrator has disabled your unused applications, you can manually remove them from the Red Hat OpenShift Data Science dashboard. Disabling and removing unused applications allows you to focus on the applications that you are most likely to use.

Prerequisites

- Ensure that you have logged in to Red Hat OpenShift Data Science.
- You have logged in to the OpenShift Container Platform web console.
- Your administrator has previously disabled the application that you want to remove.

Procedure

1. In the OpenShift Data Science interface, click **Enabled**.
The **Enabled** page opens. Disabled applications are denoted with **Disabled** on the card for the application.
2. Click **Disabled** on the card of the application that you want to remove.
3. Click the link to remove the application card.

Verification

- The card for the disabled application no longer appears on the **Enabled** page.

CHAPTER 9. SUPPORT REQUIREMENTS AND LIMITATIONS

Review this section to understand the requirements for Red Hat support and any limitations to Red Hat support of Red Hat OpenShift Data Science.

9.1. SUPPORTED BROWSERS


Red Hat OpenShift Data Science supports the latest version of the following browsers:


- Google Chrome
- Mozilla Firefox
- Safari

9.2. SUPPORTED SERVICES

Red Hat OpenShift Data Science supports the following services:

Table 9.1. Supported services

Service Name	Description
Anaconda Professional Edition	Anaconda Professional Edition is a popular open source package distribution and management experience that is optimized for commercial use.
IBM Watson Studio	IBM Watson Studio is a platform for embedding AI and machine learning into your business and creating custom models with your own data.
Intel® oneAPI AI Analytics Toolkits	The AI Kit is a set of AI software tools to accelerate end-to-end data science and analytics pipelines on Intel® architectures.
Jupyter	<p>Jupyter is a multi-user version of the notebook designed for companies, classrooms, and research labs.</p> <div>  <p>IMPORTANT</p> <p>While every effort is made to make Red Hat OpenShift Data Science resilient to OpenShift node failure, upgrades, and similarly disruptive operations, individual users' notebook environments can be interrupted during these events. If an OpenShift node restarts or becomes unavailable, any user notebook environment on that node is restarted on a different node. When this occurs, any ongoing process executing in the user's notebook environment is interrupted, and the user needs to re-execute it when their environment becomes available again.</p> <p>Due to this limitation, Red Hat recommends that processes for which interruption is unacceptable are not executed in the Jupyter notebook server environment on OpenShift Data Science.</p> </div>

Service Name	Description
Pachyderm	<p>Use Pachyderm's data versioning, pipeline and lineage capabilities to automate the machine learning life cycle and optimize machine learning operations.</p> <div>  <div> <p>NOTE</p> <p>The pachd pod that Pachyderm creates intentionally does not provide terminal access from the OpenShift web console. If you try to access the terminal view for the pachd pod, you see an error. This behavior is expected.</p> </div> </div>
Red Hat OpenShift API Management	OpenShift API Management is a service that accelerates time-to-value and reduces the cost of delivering API-first, microservices-based applications.
OpenVINO	OpenVINO is an open-source toolkit to help optimize deep learning performance and deploy using an inference engine onto Intel hardware.
Starburst Galaxy	Starburst Galaxy is a fully managed service to run high-performance queries across your various data sources using SQL.

9.3. SUPPORTED PACKAGES

The latest supported notebook server images in Red Hat OpenShift Data Science are installed with Python by default. See the table in [Options for notebook server environments](#) for a complete list of packages and versions included in these images.

You can install packages that are compatible with the supported version of Python on any notebook server that has the binaries required by that package. If the required binaries are not included on the notebook server image you want to use, contact Red Hat Support to request that the binary be considered for inclusion.

You can install packages on a temporary basis by using the **pip install** command. You can also provide a list of packages to the **pip install** command using a **requirements.txt** file. See [Installing Python packages on your notebook server](#) for more information.

You must re-install these packages each time you start your notebook server.

You can remove packages by using the **pip uninstall** command.

Additional resources

- [Installing Python packages on your notebook server](#)
- [Options for notebook server environments](#)

CHAPTER 10. COMMON QUESTIONS

In addition to documentation, Red Hat provides several "how-to" documents that answer common questions a data scientist might have as they work.

The currently available "how to" documents are linked here:

- [How to update notebook server settings](#)
- [How to install Python packages on your notebook server](#)
- [How to view installed packages on your notebook server](#)
- [How to use data from Amazon S3 buckets](#)

CHAPTER 11. TROUBLESHOOTING COMMON PROBLEMS IN JUPYTER FOR ADMINISTRATORS

If your users are experiencing errors in Red Hat OpenShift Data Science relating to Jupyter, their notebooks, or their notebook server, read this section to understand what could be causing the problem, and how to resolve the problem.

If you cannot see the problem here or in the release notes, contact Red Hat Support.

11.1. A USER RECEIVES A 404: PAGE NOT FOUND ERROR WHEN LOGGING IN TO JUPYTER

Problem

If you have configured specialized OpenShift Data Science user groups, the user name might not be added to the default user group for OpenShift Data Science.

Diagnosis

Check whether the user is part of the default user group.

1. Find the names of groups allowed access to Jupyter.
 - a. Log in to OpenShift Container Platform web console.
 - a. Click **User Management** → **Groups**.
 - b. Click the name of your user group, for example, **rhods-users**.
The **Group details** page for that group appears.
 1. Click the **Details** tab for the group and confirm that the **Users** section for the relevant group, contains the users who have permission to access Jupyter.

Resolution

- If the user is not added to any of the groups allowed access to Jupyter, follow [Adding users for OpenShift Data Science](#) to add them.
- If the user is already added to a group that is allowed to access Jupyter, contact Red Hat Support.

11.2. A USER'S NOTEBOOK SERVER DOES NOT START

The OpenShift Container Platform cluster that hosts the user's notebook server might not have access to enough resources, or the Jupyter pod may have failed.

1. Log in to OpenShift Container Platform web console.
1. Delete and restart the notebook server pod for this user.
 - a. Click **Workloads** → **Pods** and set the **Project** to **rhods-notebooks**.
 - b. Search for the notebook server pod that belongs to this user, for example, **jupyter-nb-
<username>-***.

If the notebook server pod exists, an intermittent failure may have occurred in the notebook server pod.

If the notebook server pod for the user does not exist, continue with diagnosis.

2. Check the resources currently available in the OpenShift cluster against the resources required by the selected notebook server image.
If worker nodes with sufficient CPU and RAM are available for scheduling in the cluster, continue with diagnosis.
3. Check the state of the Jupyter pod.

Resolution

- If there was an intermittent failure of the notebook server pod:
 - a. Delete the notebook server pod that belongs to the user.
 - b. Ask the user to start their notebook server again.
- If the notebook server does not have sufficient resources to run the selected notebook server image, either add more resources to the OpenShift cluster, or choose a smaller image size.
- If the Jupyter pod is in a **FAILED** state:
 - a. Retrieve the logs for the **jupyter-nb-*** pod and send them to Red Hat Support for further evaluation.
 - b. Delete the **jupyter-nb-*** pod.
- If none of the previous resolutions apply, contact Red Hat Support.

11.3. THE USER RECEIVES A DATABASE OR DISK IS FULL ERROR OR A NO SPACE LEFT ON DEVICE ERROR WHEN THEY RUN NOTEBOOK CELLS

Problem

The user might have run out of storage space on their notebook server.

Diagnosis

1. Log in to Jupyter and start the notebook server that belongs to the user having problems. If the notebook server does not start, follow these steps to check whether the user has run out of storage space:
 - a. Log in to OpenShift Container Platform web console.
 - a. Click **Workloads** → **Pods** and set the **Project** to **rhods-notebooks**.
 - b. Click the notebook server pod that belongs to this user, for example, **jupyter-nb-<idp>-<username>-***.
 - c. Click **Logs**. The user has exceeded their available capacity if you see lines similar to the following:

Unexpected error while saving file: XXXX database or disk is full

Resolution

- Increase the user's available storage by expanding their persistent volume: [Expanding persistent volumes](#)
- Work with the user to identify files that can be deleted from the **/opt/app-root/src** directory on their notebook server to free up their existing storage space.

CHAPTER 12. TROUBLESHOOTING COMMON PROBLEMS IN JUPYTER FOR USERS

If you are seeing errors in Red Hat OpenShift Data Science related to Jupyter, your notebooks, or your notebook server, read this section to understand what could be causing the problem.

If you cannot see your problem here or in the release notes, contact Red Hat Support.

12.1. I SEE A 403: FORBIDDEN ERROR WHEN I LOG IN TO JUPYTER

Problem

If your administrator has configured specialized OpenShift Data Science user groups, your user name might not be added to the default user group or the default administrator group for OpenShift Data Science.

Resolution

Contact your administrator so that they can add you to the correct group/s.

12.2. MY NOTEBOOK SERVER DOES NOT START

The OpenShift Platform Container cluster that hosts your notebook server might not have access to enough resources, or the Jupyter pod may have failed.

Resolution

Check the logs in the **Events** section in OpenShift for error messages associated with the problem. For example:

```
Server requested
2021-10-28T13:31:29.830991Z [Warning] 0/7 nodes are available: 2 Insufficient memory,
2 node(s) had taint {node-role.kubernetes.io/infra: }, that the pod didn't tolerate, 3 node(s) had taint
{node-role.kubernetes.io/master: },
that the pod didn't tolerate.
```

Contact your administrator with details of any relevant error messages so that they can perform further checks.

12.3. I SEE A DATABASE OR DISK IS FULL ERROR OR A NO SPACE LEFT ON DEVICE ERROR WHEN I RUN MY NOTEBOOK CELLS

Problem

You might have run out of storage space on your notebook server.

Resolution

Contact your administrator so that they can perform further checks.

