Enable access to GitHub from your account

Before you can publish notebooks on GitHub, you must enable your Cloud Pak for Data as a Service account to access GitHub. You enable access by creating a personal access token with the required access scope in GitHub and linking the token to your Cloud Pak for Data as a Service account.

Follow these steps to create a personal access token:

Navigate to your Cloud Pak for Data as a Service account settings, and click the Integrations tab.

Click the GitHub personal access tokens link on the dialog and generate a new token.

On the New personal access token page, select repo scope and then click to generate a token.

Copy the generated access token and paste it in the GitHub integration dialog window in Cloud Pak for Data as a Service.

Link a project to a GitHub repository

After you have saved the access token, your project must be connected to an existing GitHub repository. You can only link to one existing GitHub repository from a project. Private repositories are supported.

To link a project to an existing GitHub repository, you must have administrator permission to the project. All project collaborators, who have adminstrator or editor permission, can publish files to this GitHub repository. However, these users must have permission to access the repository. Granting user permissions to repositories must be done in GitHub.

To connect a project to an existing GitHub repository:

Select the Manage tab and go to the Services and Integrations page.

Click the Third-party integrations tab.

Click Connect integration.

Enter your generated access token from Github.

Now you can begin publishing notebooks on GitHub.

Note:

For information on how to change your Git integration, refer to Managing your integrations.

Publish a notebook on GitHub

To publish a notebook on GitHub:

Open the notebook in edit mode.

Click the GitHub integration icon (Shows the upload icon) and select Publish on GitHub from the opened notebook's action bar.

When you enter the name of the file you want to publish on GitHub, you can specify a folder path in the GitHub repository. Note that notebook files are always pushed to the master branch.

Parent topic: Managing the lifecycle of notebooks and scripts

Certainly! To build a project by loading and preprocessing a dataset, you will need to use a programming language and libraries that are suitable for your project. Below is a general example using Python and common libraries such as Pandas and NumPy. In this example, I'll use a fictional dataset as a placeholder. You should replace it with your actual dataset.

python

Copy code

```
# Import necessary libraries

import pandas as pd

import numpy as np


# Load your dataset
# Replace 'your_dataset.csv' with the actual file path or URL of your dataset
data = pd.read_csv('your_dataset.csv')

# Explore the dataset to understand its structure
# Display the first few rows of the dataset
print(data.head())

# Perform basic preprocessing steps
# Here are some common preprocessing steps:
# 1. Handling missing values
data.dropna(inplace=True)  # Remove rows with missing values or use other methods to impute them

# 2. Encoding categorical variables (if applicable)
# You can use techniques like one-hot encoding or label encoding

# 3. Feature scaling/normalization (if applicable)
# Scale/normalize numerical features to have a similar scale, e.g., using Min-Max scaling or Standardization

# 4. Split the dataset into training and testing sets
from sklearn.model_selection import train_test_split
X = data.drop('target_column', axis=1)  # Replace 'target_column' with the name of your target variable
y = data['target_column']
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)

# Your dataset is now loaded, preprocessed, and split into training and testing sets.
# You can proceed with building your project, whether it's m
```