

Fabric Workshop

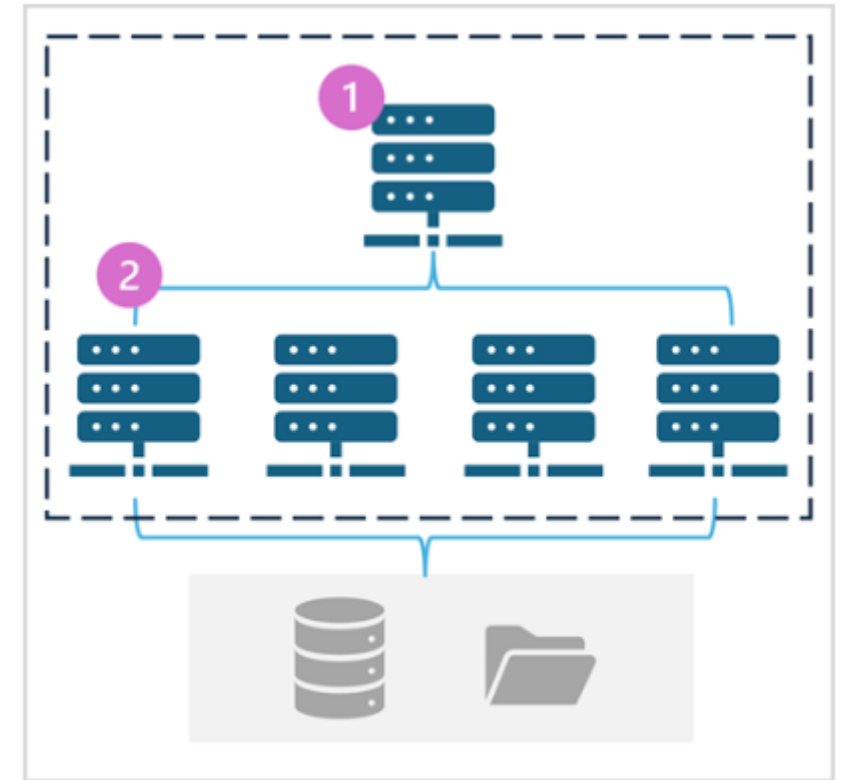
2025-02-27 and 2025-02-28

Spark and Fabric Notebooks

Walk-through

What is Spark?

- Unified analytics engine for large-scale data processing.
- Used for batch and stream processing.
- Able to be run on single machine or multiple nodes.
 - Coordinates work across multiple processing nodes in a cluster, known in Microsoft Fabric as a **Spark pool**.



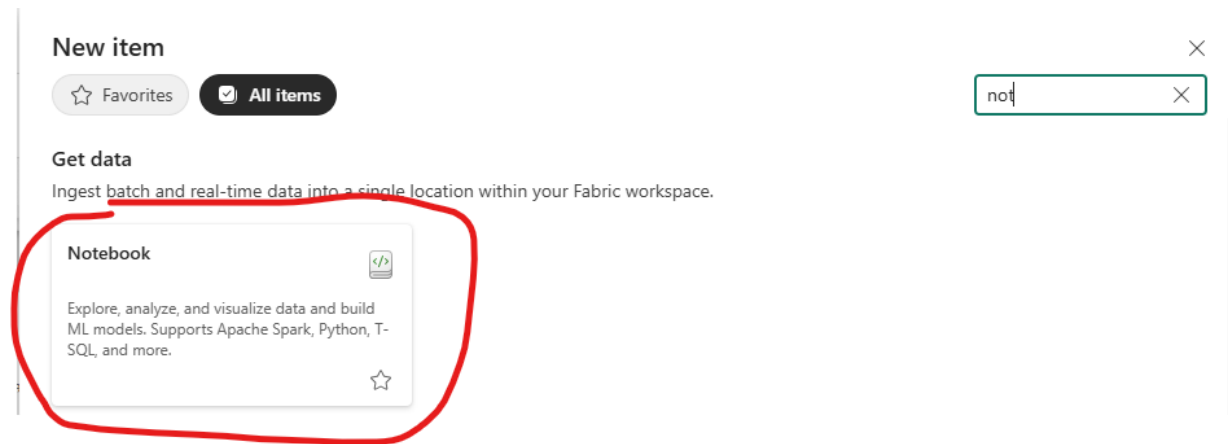
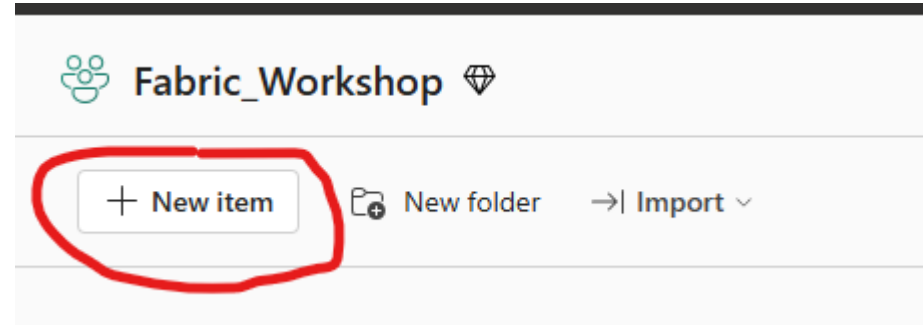
1. **Head node** coordinates via **driver**.
2. **Worker nodes** complete tasks via **executor**.



What are Fabric Notebooks?

- Primary programming experience in Microsoft Fabric.
- Multiple programming languages supported.
 - Spark – Scala, PySpark, SparkR, Spark SQL
 - Python (Preview)
 - T-SQL
- Applications
 - ETL
 - EDA
 - Machine Learning

Getting started in 2 steps...



Notebooks Demo and Tutorial

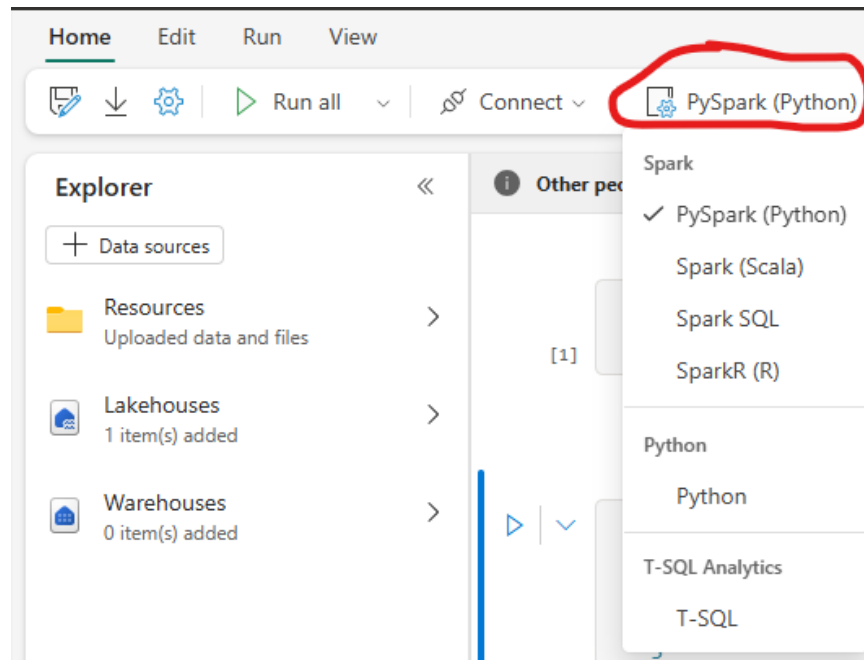


Fabric Notebooks Advanced Features

Walk-through

Different Programming Languages

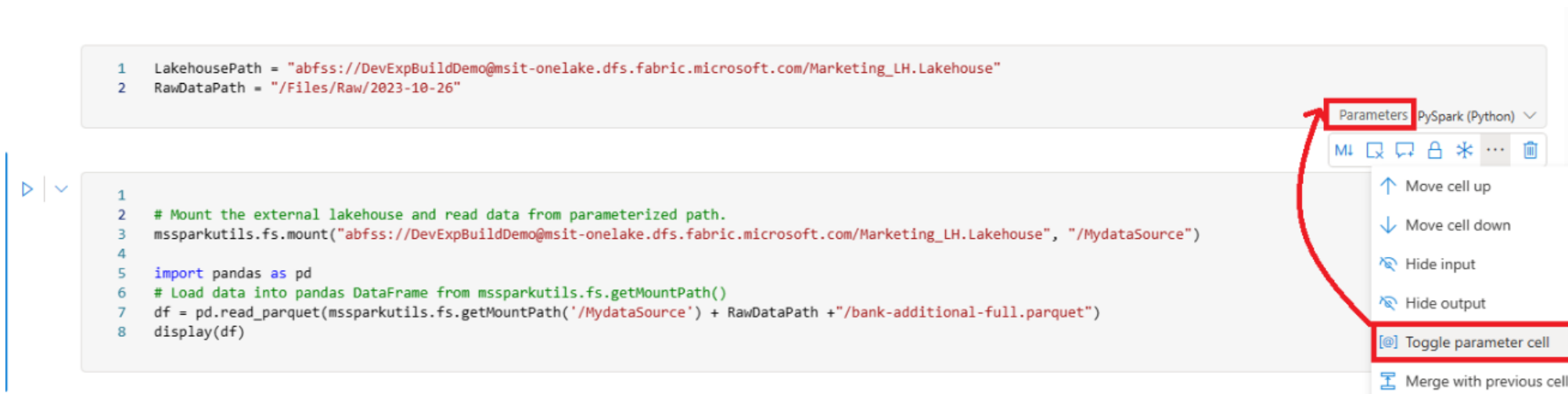
- Set for entire notebook for single cell.



Magic command	Language	Description
%%pyspark	Python	Execute a Python query against Apache Spark Context.
%%spark	Scala	Execute a Scala query against Apache Spark Context.
%%sql	SparkSQL	Execute a SparkSQL query against Apache Spark Context.
%%html	Html	Execute a HTML query against Apache Spark Context.
%%sparkr	R	Execute a R query against Apache Spark Context.

Parameter Cells

- Pass arguments via runMultiple or Pipeline.



The screenshot displays a Jupyter Notebook interface with two code cells. The first cell contains two lines of Python code defining paths. The second cell contains a multi-line Python script that mounts a lakehouse, imports pandas, and reads a parquet file. A context menu is open for the second cell, with a red arrow pointing to the 'Parameters' tab and another red box highlighting the '@ Toggle parameter cell' option.

```
1 LakehousePath = "abfss://DevExpBuildDemo@msit-onelake.dfs.fabric.microsoft.com/Marketing_LH.Lakehouse"
2 RawDataPath = "/Files/Raw/2023-10-26"
```

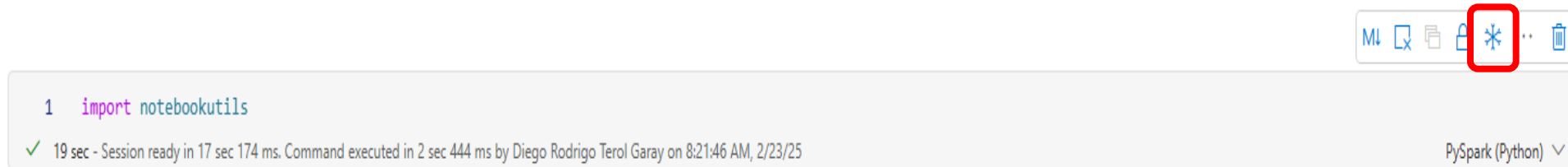
```
1
2 # Mount the external lakehouse and read data from parameterized path.
3 mssparkutils.fs.mount("abfss://DevExpBuildDemo@msit-onelake.dfs.fabric.microsoft.com/Marketing_LH.Lakehouse", "/MydataSource")
4
5 import pandas as pd
6 # Load data into pandas DataFrame from mssparkutils.fs.getMountPath()
7 df = pd.read_parquet(mssparkutils.fs.getMountPath('/MydataSource') + RawDataPath + "/bank-additional-full.parquet")
8 display(df)
```

Parameters PySpark (Python) ▾

- Move cell up
- Move cell down
- Hide input
- Hide output
- @ Toggle parameter cell**
- Merge with previous cell

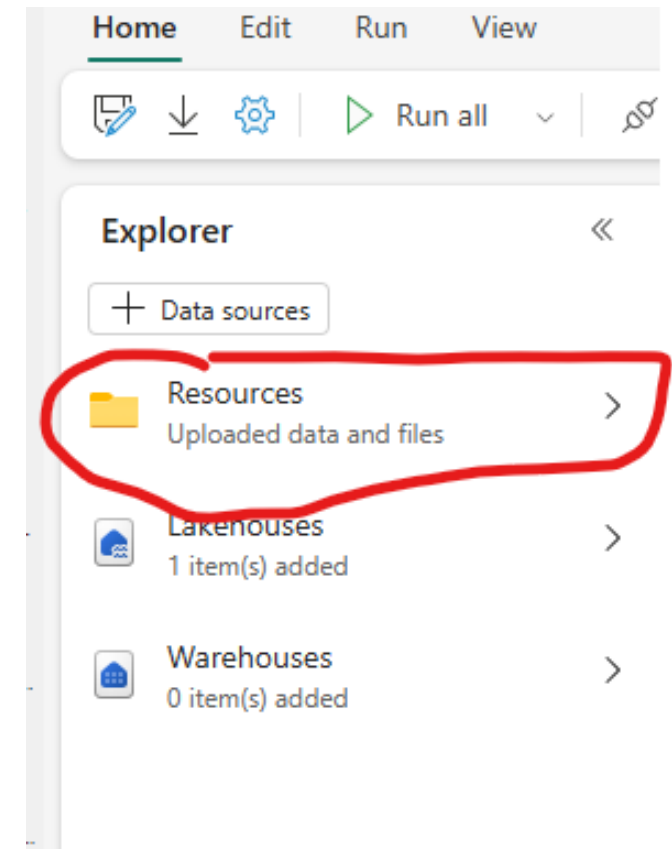
Frozen Cells

- Make cells read-only.



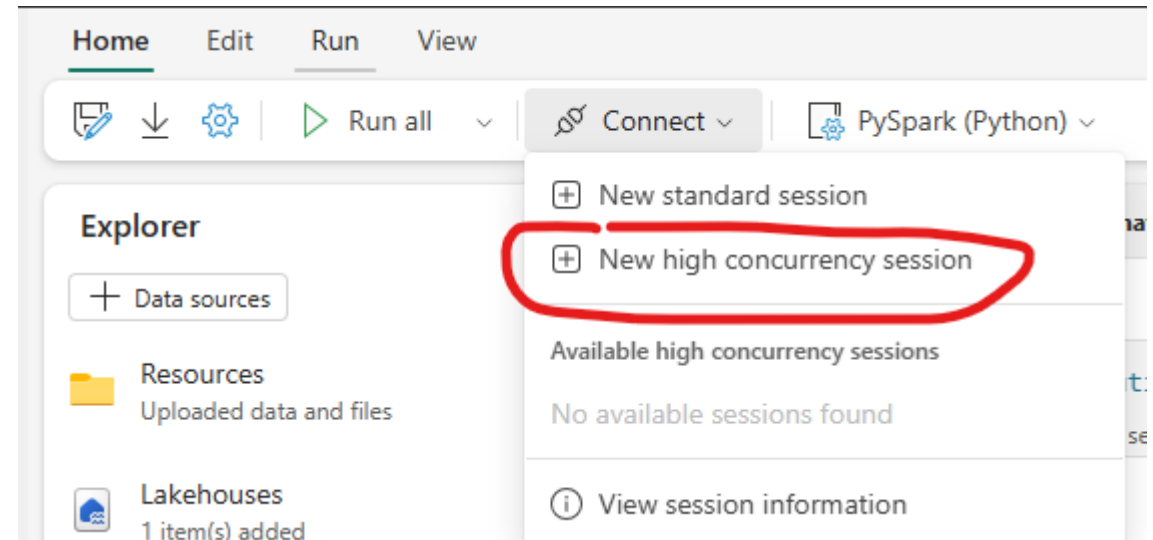
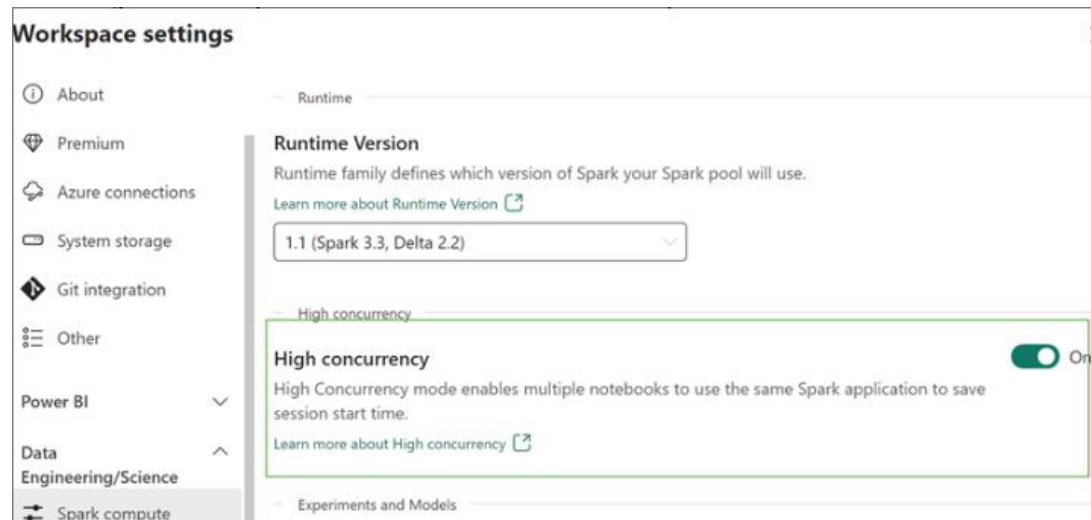
Notebook Resources

- File system space to store small-sized files (e.g., code modules, semantic models, images...).



High Concurrency

- Share Spark Sessions across multiple Notebooks.



Configure Spark Starter Pools

- Configure Starter Pool based on Capacity size and required resources.

Edit pool

Spark pool name *

StarterPool

Node family

Memory optimized

Node size

Medium

Autoscale

If enabled, your Apache Spark pool will automatically scale up and down based on the amount of activity.

☒ Enable autoscale

1

16

Dynamically allocate executors

☒ Enable allocate

1

15

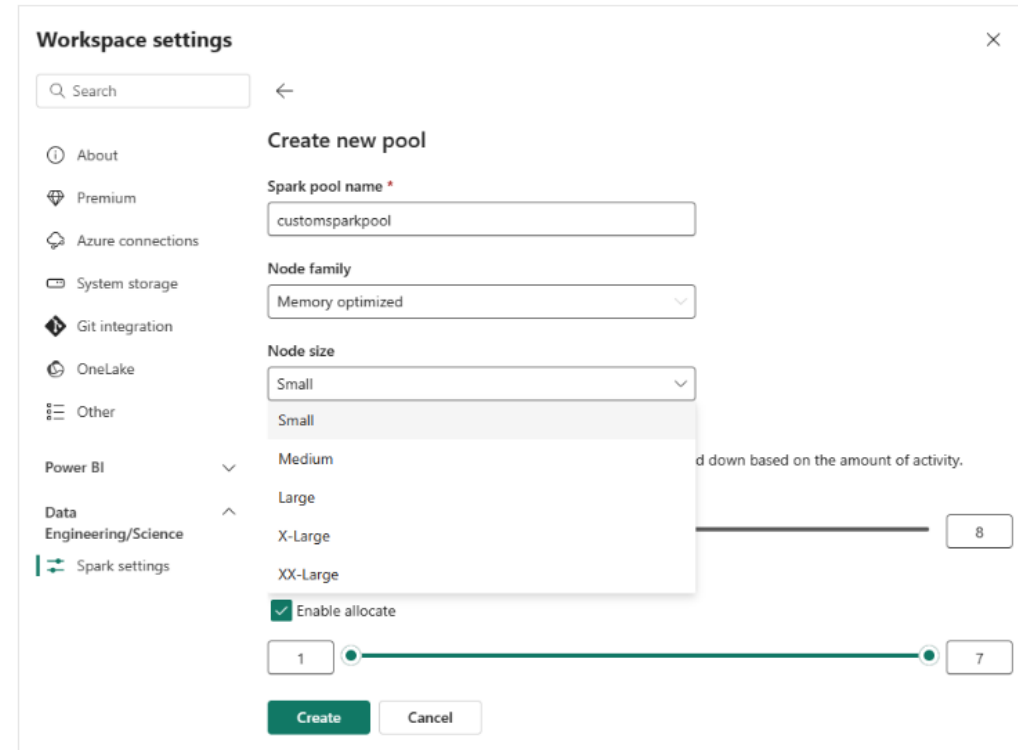
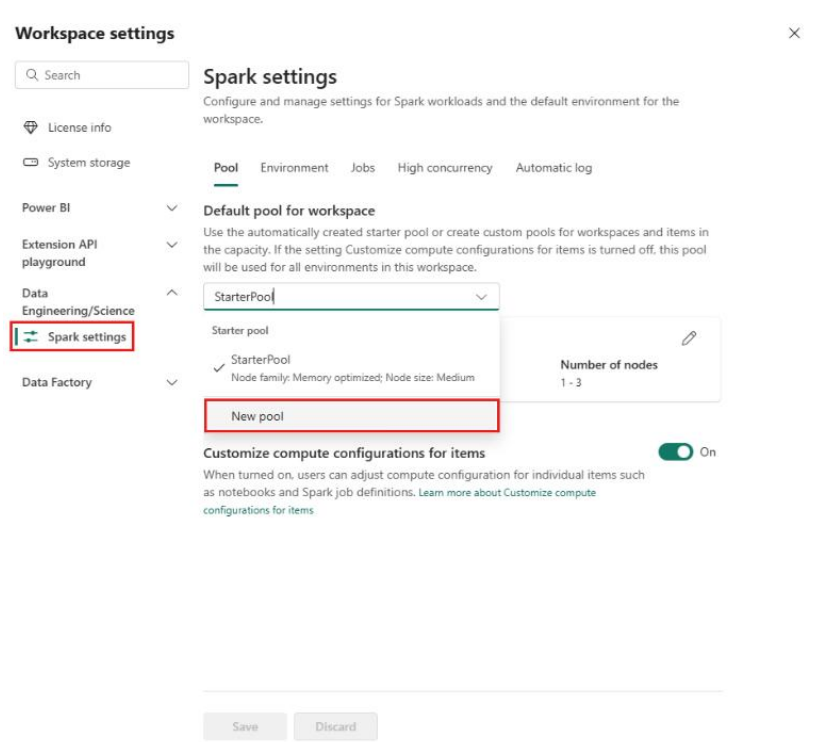
Save

Discard

SKU name	Capacity units	Spark VCores	Node size	Default max nodes	Max number of nodes
F2	2	4	Medium	1	1
F4	4	8	Medium	1	1
F8	8	16	Medium	2	2
F16	16	32	Medium	3	4
F32	32	64	Medium	8	8
F64	64	128	Medium	10	16
(Trial Capacity)	64	128	Medium	10	16
F128	128	256	Medium	10	32
F256	256	512	Medium	10	64
F512	512	1024	Medium	10	128
F1024	1024	2048	Medium	10	200
F2048	2048	4096	Medium	10	200


Work with Custom Spark Pools

- Create and configure based on required workloads.



Managing Libraries

- Available via Inline installation and Environment setup.

Python  Copy

```
%conda install altair          # install latest version through conda command
%conda install vega_datasets    # install latest version through conda command
```

Library type	Environment library management	Inline installation
Python Public (PyPI & Conda)	Supported	Supported
Python Custom (.whl)	Supported	Supported
R Public (CRAN)	Not supported	Supported
R custom (.tar.gz)	Supported as custom library	Supported
Jar	Supported as custom library	Supported

Environments



- Configure Spark compute.
- Manage public and custom Libraries.
- Manage resources.

The screenshot shows the 'Public Libraries' tab in the Databricks workspace. The interface includes a top navigation bar with 'Home' and 'Public Libraries' tabs. Below the navigation bar, there are buttons for '+ Add from PyPI', '+ Add from .yaml', 'Delete', and 'Export to .yaml'. A warning message states: 'Environment is in Preview. Other users in your organization may have access to this workspace. Do not use this item unless you trust all other users who may have access to the workspace.' The left sidebar contains a 'Libraries' section with 'Public Libraries' and 'Custom Libraries' options, and a 'Spark Compute' section with 'Compute' and 'Spark properties' options. The main content area is titled 'Public Libraries' and contains a search bar 'Filter by name'. Below the search bar is a table with columns: 'Library', 'Version', 'Source', 'Status', and 'Last updated'. The table has one row for the library 'fuzzywuzzy' with version '0.18.0', source 'PyPI', status 'Success', and last updated '10/24/23, 04:22:02 PM'. A 'Dependencies' button is located next to the library name, and a black arrow points to it.

Home Public Libraries

+ Add from PyPI + Add from .yaml Delete Export to .yaml

Environment is in Preview. Other users in your organization may have access to this workspace. Do not use this item unless you trust all other users who may have access to the workspace.

Libraries

Public Libraries

Custom Libraries

Spark Compute

Compute

Spark properties

Public Libraries

Search and add libraries from public repositories or via a .yaml file. They'll be available if you run your notebook or Spark job definition in this environment.

Filter by name

Library	Version	Source	Status	Last updated
fuzzywuzzy	0.18.0	PyPI	Success	10/24/23, 04:22:02 PM

Manage Notebooks with APIs

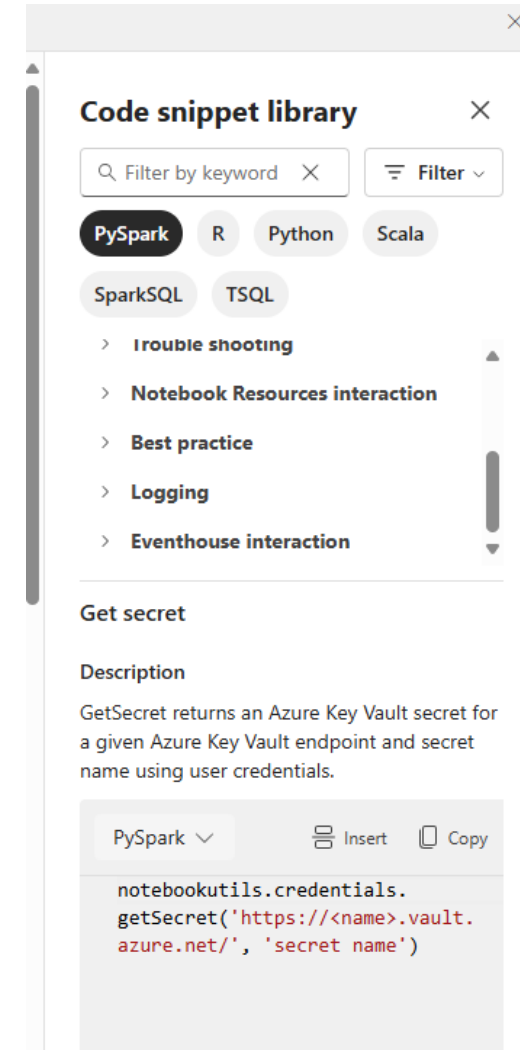
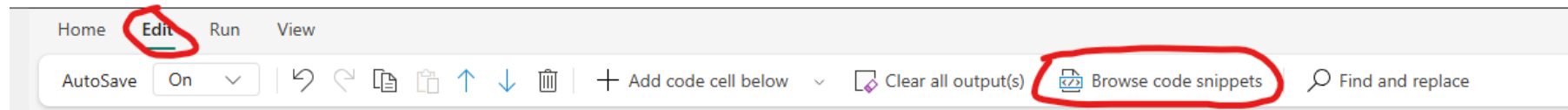
- Manage CRUD operations and Notebook Run instances.

Action	Description
Create item	Creates a notebook inside a workspace.
Update item	Updates the metadata of a notebook.
Update item definition	Updates the content of a notebook.
Delete item	Deletes a notebook.
Get item	Gets the metadata of a notebook.
Get item definition	Gets the content of a notebook.
List item	List all items in a workspace.

Action	Description
Run on demand Item Job	Run notebook with parameterization.
Cancel Item Job Instance	Cancel notebook job run.
Get Item Job Instance	Get notebook run status.

Code Snippets

- Get commonly used code patterns.



Questions?

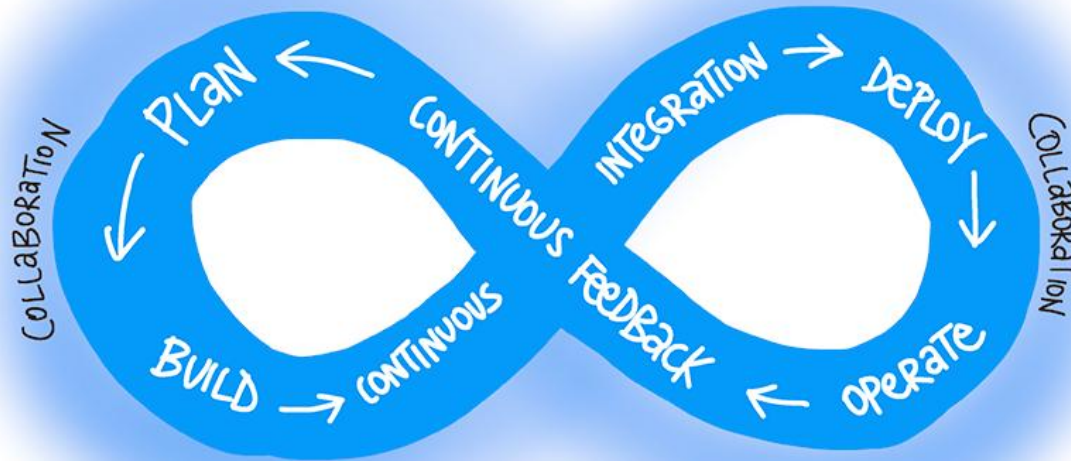


Azure Devops and Git Integration

Walk-through

What is DevOps?

- [What is DevOps? - Training | Microsoft Learn](#)



DevOps benefits for Fabric



Enhance collaboration (CI)

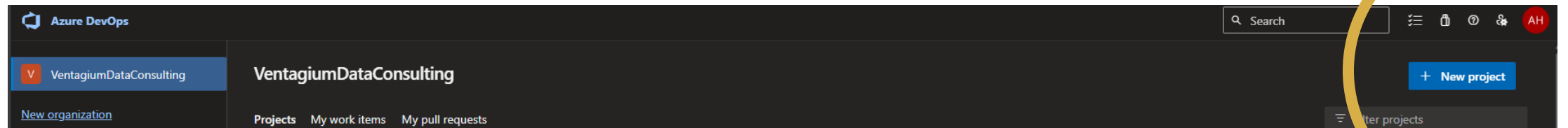


Leverage your development (CD)



Version control

Create a DevOps Project



Create a DevOps Project

Create new project

Project name *

Fabric Workshop

Description

Visibility

Public

Anyone on the internet can view the project. Certain features like TFVC are not supported.

Private

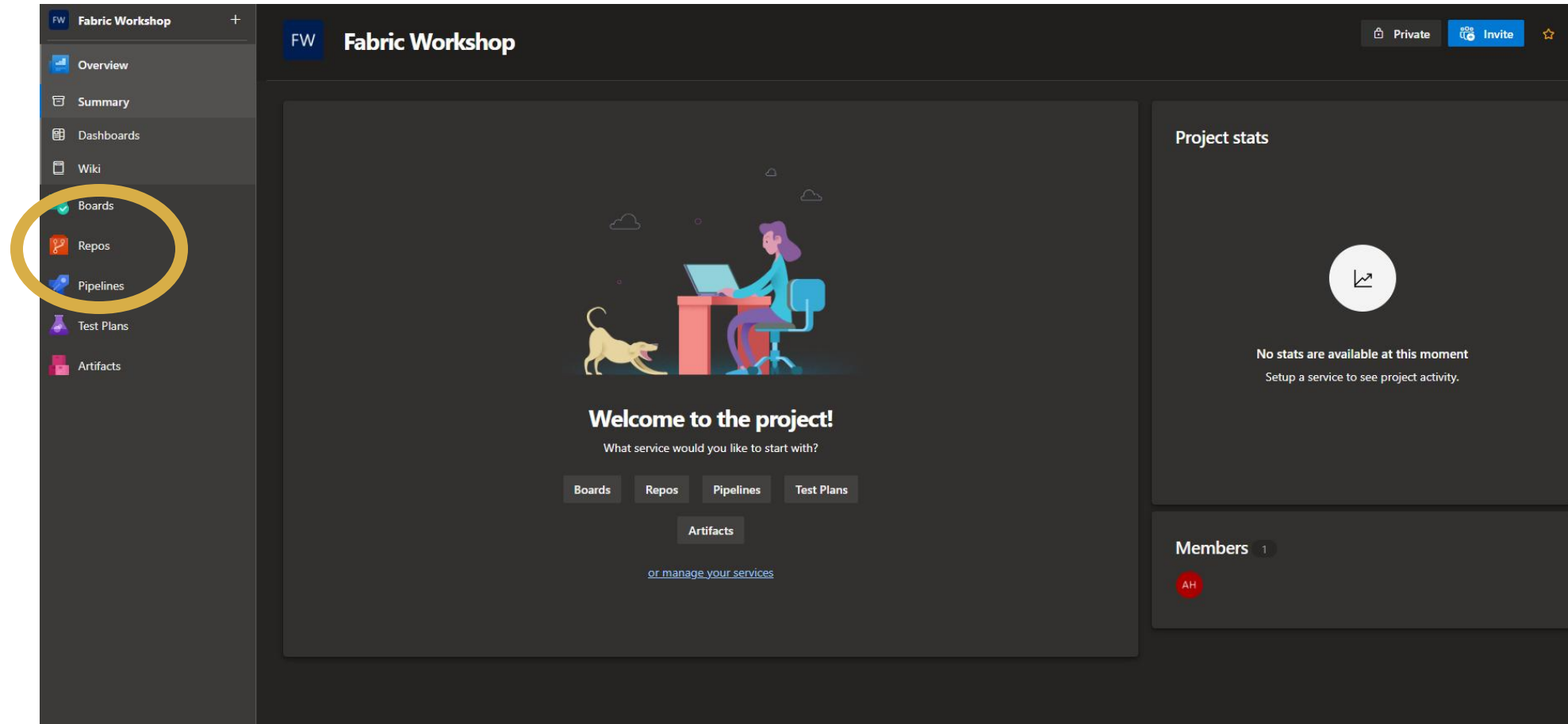
Only people you give access to will be able to view this project.

Advanced

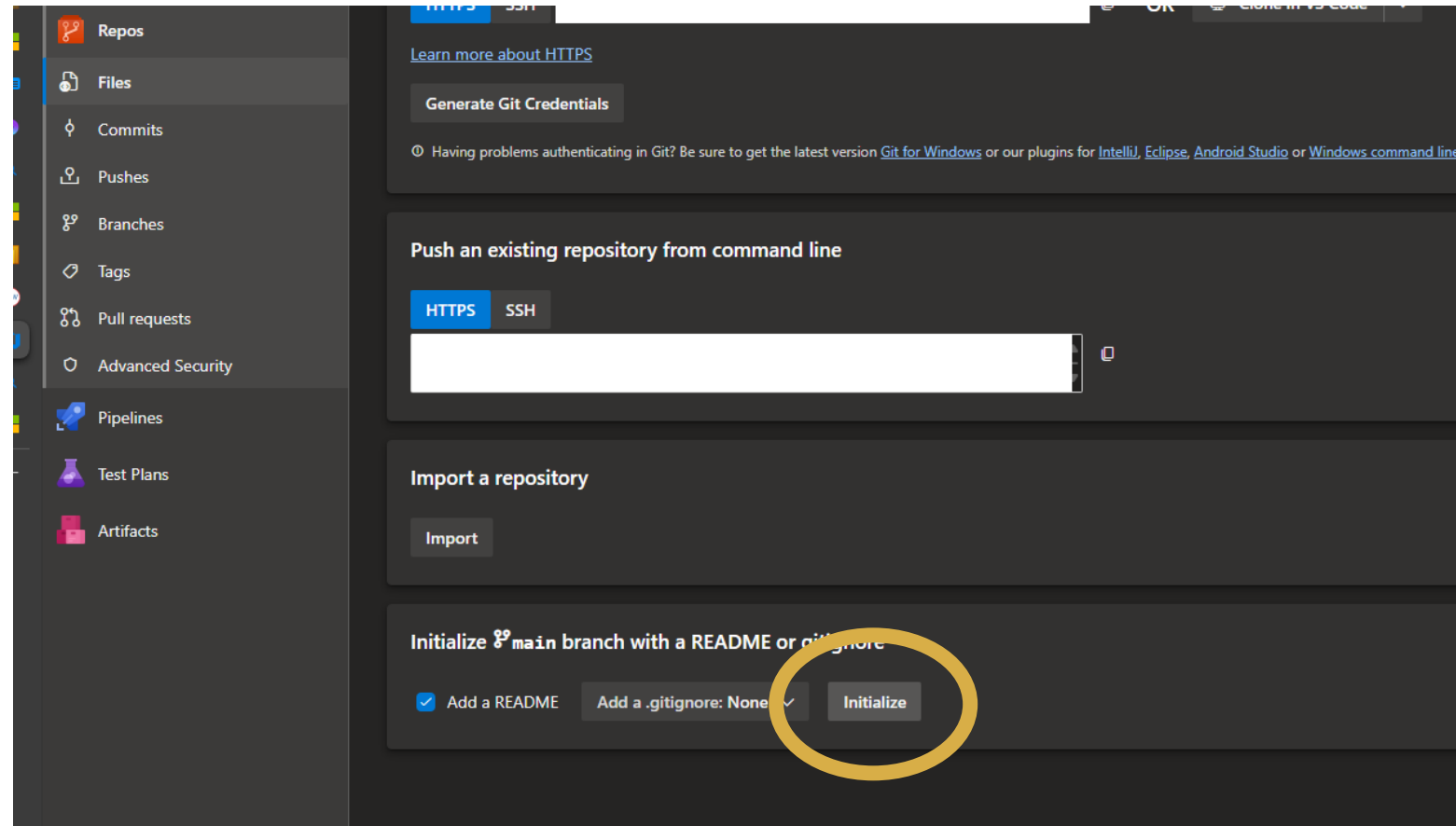
Cancel

Create

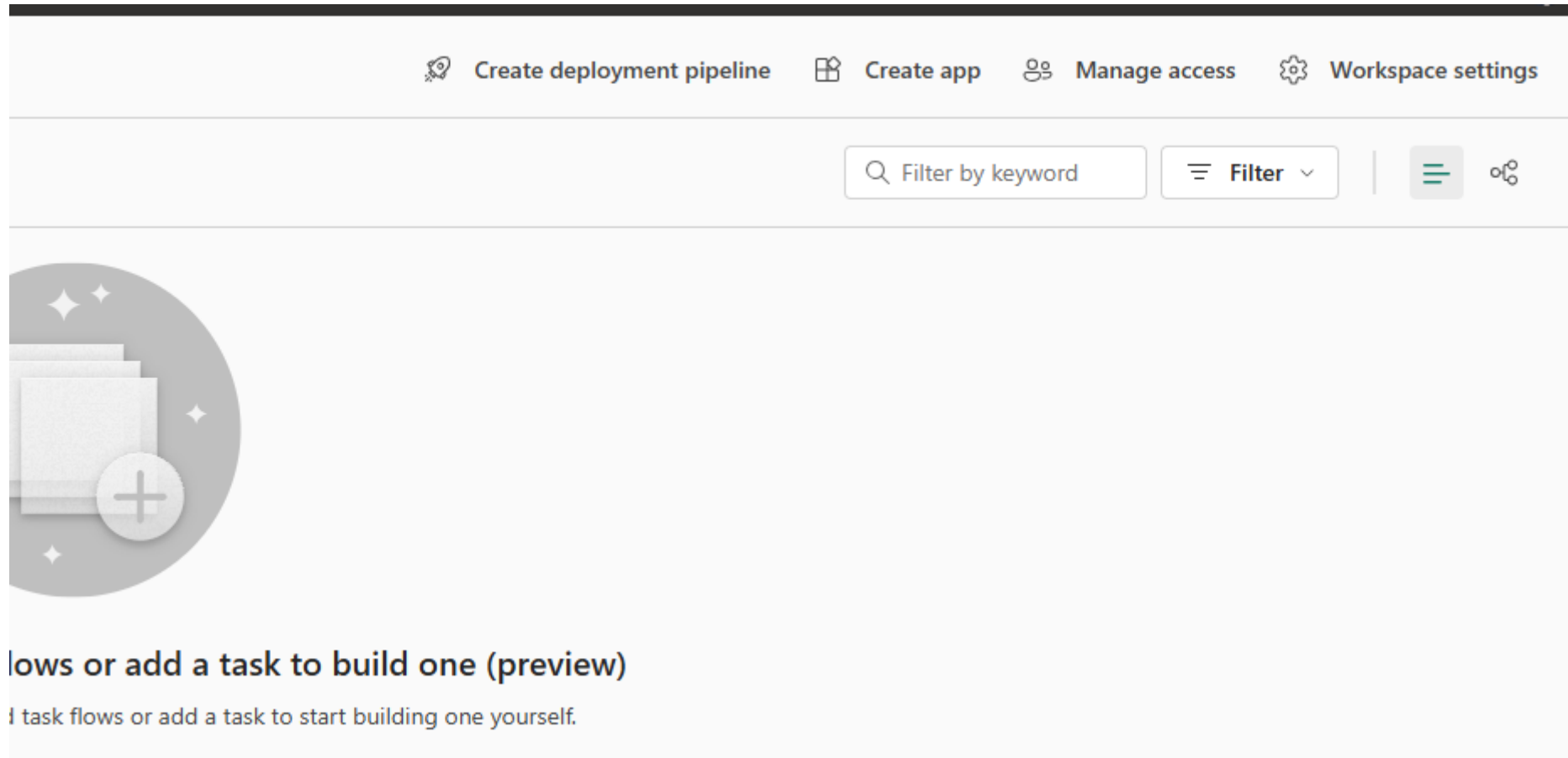
Initialize a Repo



Initialize a Repo



Enable Git Integration



Enable Git Integration

Workspace settings

General

- License info
- Azure connections
- System storage
- Git integration**
- OneLake
- Workspace identity
- Network security
- Monitoring

Power BI

General

About

Workspace image



Name *

Fabric_Workshop

Description

Describe this workspace (O

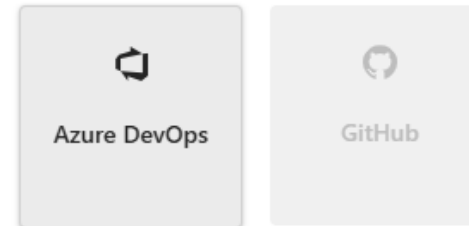
Git integration

Connect to Git to manage your code and back up your work. [Learn more](#)

Preview items Some item types are only available in preview when using Git. [Learn more](#)

Connect Git provider and account

Git provider














Accounts

☒ AAD account
alberto. com

Connect

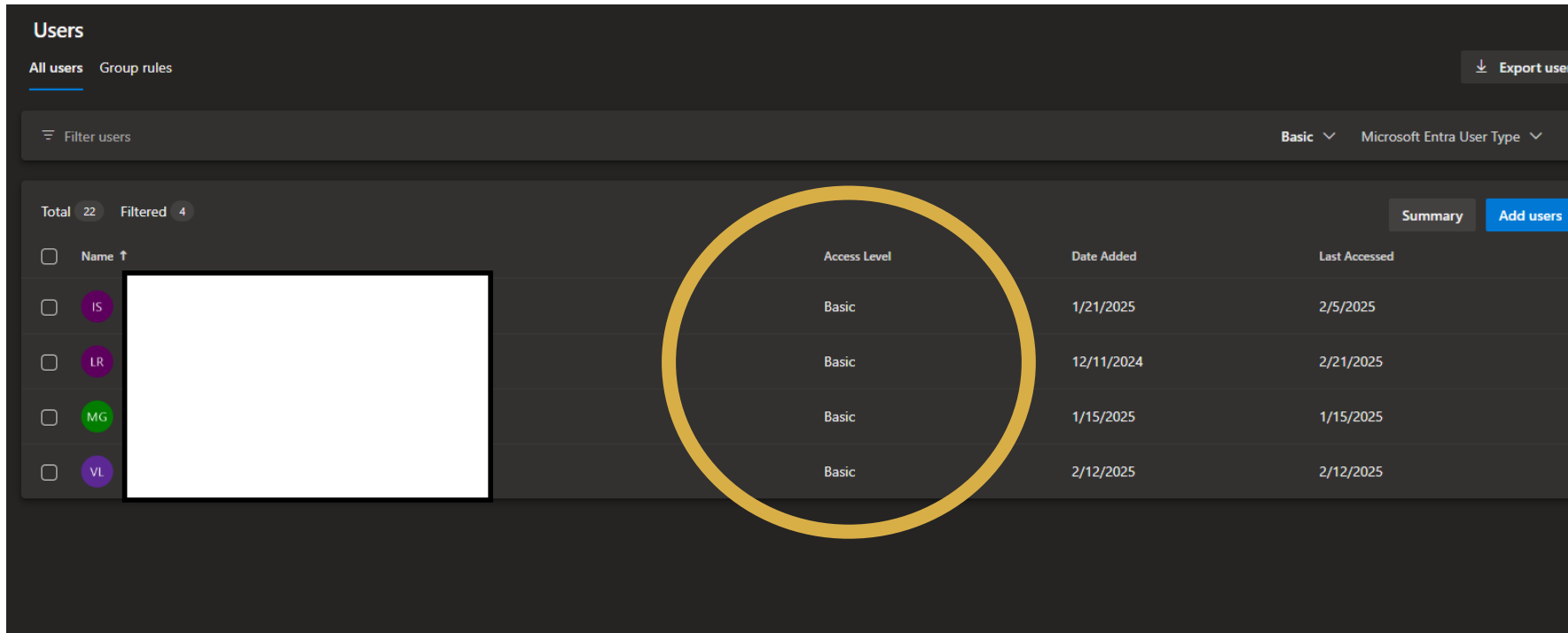
Connect Git repository and branch

Enable Git Integration

 	Name 	Git status	Type	Task
	OpenAI Demo		Folder	—
	Dataflow 1	 Unsupported	Dataflow Gen2	—
	Fabric Workshop 01	 Synced	SQL database	—
	Fabric Workshop 01	—	Semantic model...	—
	Fabric Workshop 01	—	SQL analytics en...	—
	Fabric Workshop 01	 Synced	Data pipeline	—

Create a Contributors Team

(Users require Basic access at least)



The screenshot displays the 'Users' management interface. At the top, there are tabs for 'All users' and 'Group rules', with 'All users' selected. An 'Export users' button is located in the top right corner. Below the tabs is a 'Filter users' section with a search bar. The main content area shows a table of users. The table has columns for 'Name', 'Access Level', 'Date Added', and 'Last Accessed'. A yellow circle highlights the 'Access Level' column, which contains the value 'Basic' for all four users listed. To the left of the table, there is a sidebar with a list of users (IS, LR, MG, VL) and a large white rectangular area. The table also includes a 'Summary' button and an 'Add users' button.

Name	Access Level	Date Added	Last Accessed
IS	Basic	1/21/2025	2/5/2025
LR	Basic	12/11/2024	2/21/2025
MG	Basic	1/15/2025	1/15/2025
VL	Basic	2/12/2025	2/12/2025

Create Contributors Team

Project Settings

Fabric Workshop

General

Overview

Teams

Permissions

Notifications

Service hooks

Dashboards

Boards

Project configuration

Teams

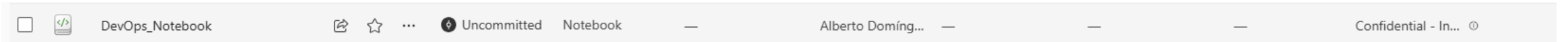
Total 2

New Team

Name ↓	Description	Members
<div>FC</div> Fabric Workshop Contributors		6
<div>FT</div> Fabric Workshop Team	<div>Default</div> The default project team.	1

Commit and Sync your Items

When someone edits an item, it will appear as uncommitted:








Next refresh	Endorsement
—	—
N/A	—
—	—
—	—
N/A	—
—	—
—	—



Source control ... ↺ ✕

Current branch: main

Changes 2 Updates

feat: idle change to track it on devOps

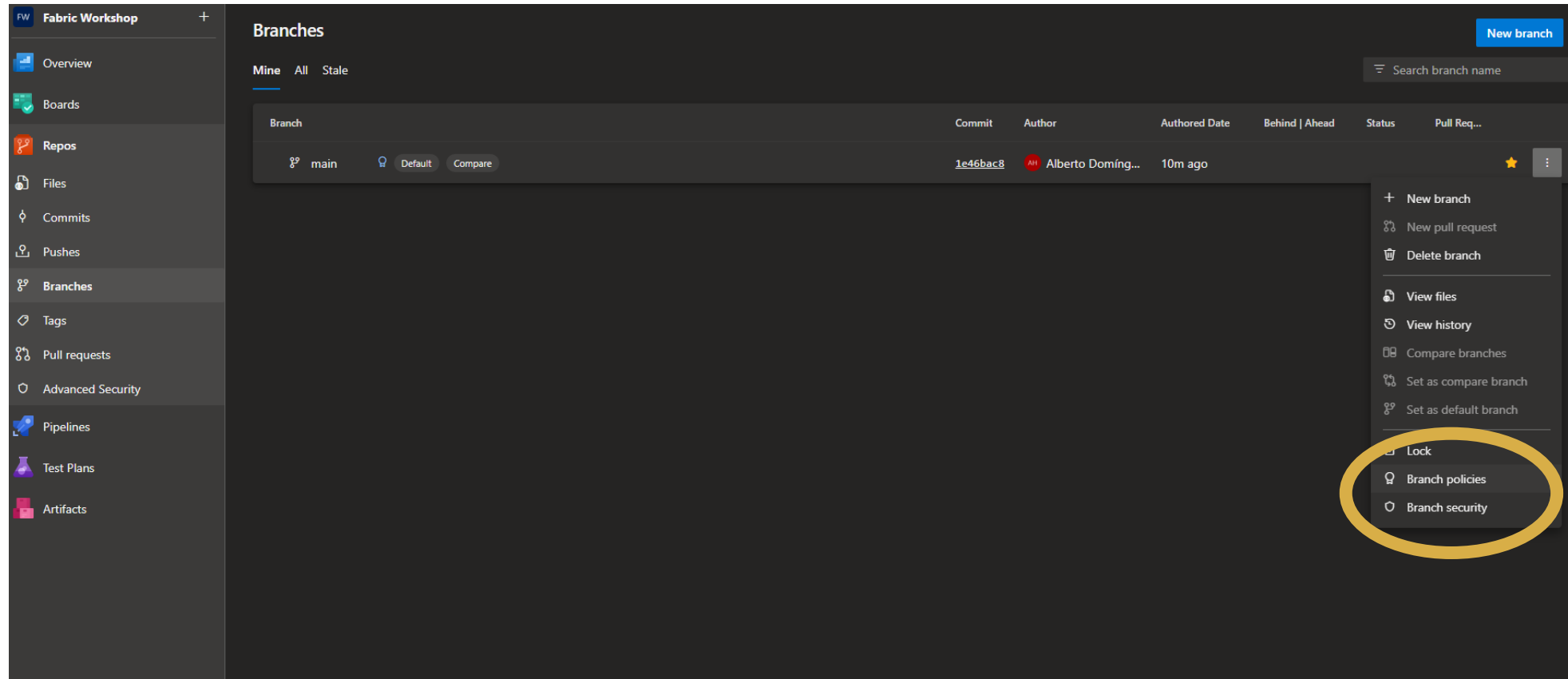
<input type="checkbox"/>	 Item	Status
<input type="checkbox"/>	 OpenAI_API_Notebook	
<input checked="" type="checkbox"/>	 DevOps_Notebook	

 Commit  Undo

Commit and Sync your Items

Click to edit Master title style

Create Branch policies



Branch Policies

Fabric_Workshop

Create deployment pipeline>Create app

+ New item

New folder

→ Import

Source control 2

Enter by key

	Name	Git status	Type	Task	Owner	Refreshed	Next refresh	Endorsement	Se
	OpenAI Demo		Folder	—	—	—	—	—	—
	Dataflow 1	⊘ Unsupported	Dataflow Gen2	—	Roberto Hernán...	—	N/A	—	—
	DevOps_Notebook	⦿ Uncommitted	Notebook	—	Alberto Domíng...	—	—	—	Co
	Fabric Workshop 01	✓ Synced	SQL database	—	Roberto Hernán...	—	—	—	Co
	Fabric Workshop 01	—	Semantic model...	—	Fabric_Workshop	2/18/2025, 10:45:1...	N/A	—	Co
	Fabric Workshop 01	—	SQL application	—	Roberto Hernán...	—	—	—	Co

Notifications

Clear all

Earlier

✗ Unable to complete commit request

View details below to learn more.

View details

6 minutes ago

✓ Your selected changes were committed.

14 minutes ago

ⓘ Session stopped successfully.

19 minutes ago

Branch Policies

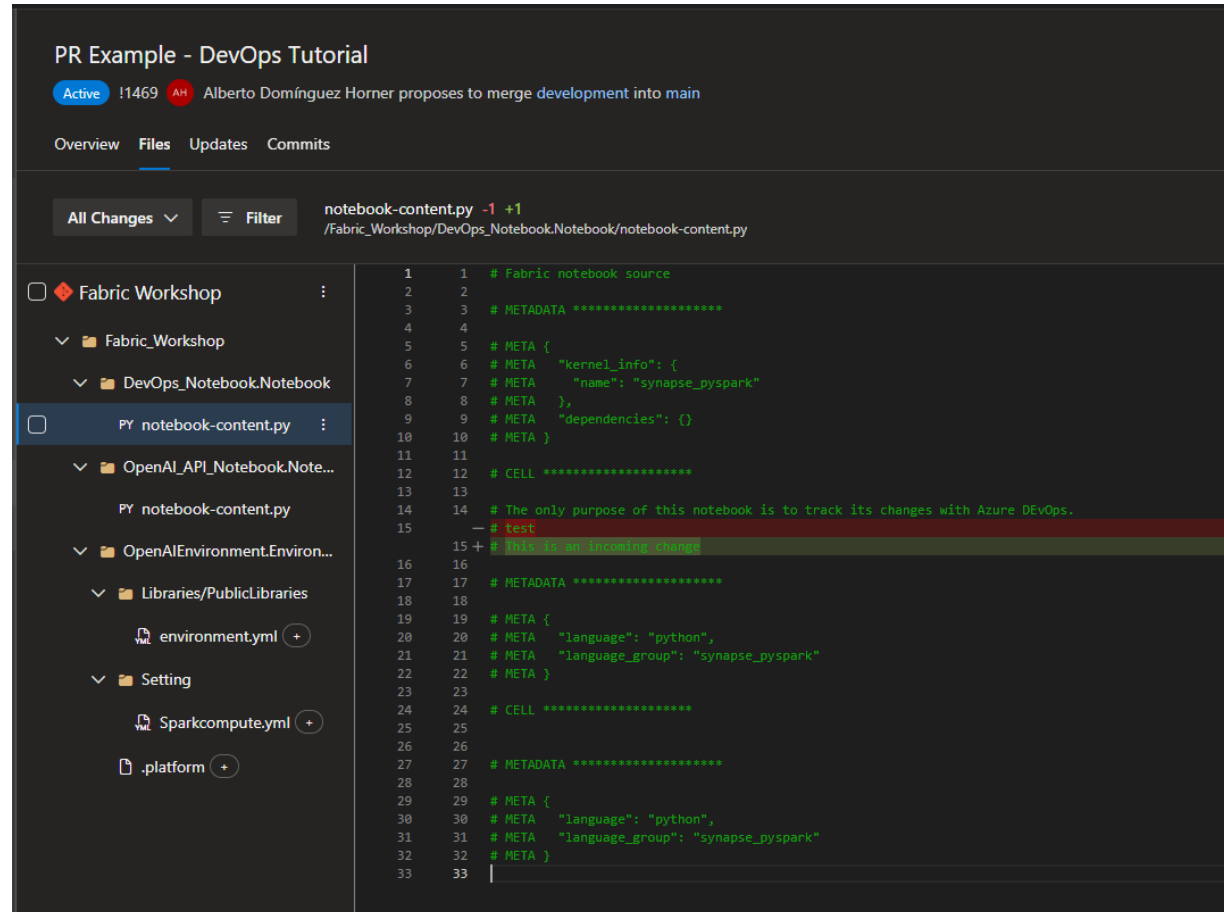
The screenshot displays a GitHub Pull Request (PR) titled "PR Example - DevOps Tutorial". The PR is in the "Active" state, with the number "11469" and the author "Alberto Domínguez Horner" (AH). The PR description is "Alberto Domínguez Horner proposes to merge development into main".

The interface includes several sections:

- Overview:** Shows the PR status and tabs for Files, Updates, and Commits.
- Checks:** A blue checkmark indicates "No merge conflicts" (Last checked: Just now). A blue bell icon indicates "Alberto Domínguez Horner must approve".
- Description:** A text area for the PR description.
- Comments:** A section for adding comments, with a "Show everything (3)" dropdown.
- Commit History:** A list of commits, including "Alberto Domínguez Horner pushed 1 commit" (Just now) and "test: incoming changes" (Just now).
- Branch Policies:** A section showing the PR's compliance with branch policies. It includes a "Required" section with a "Set auto-complete" button and an "Optional" section with "No optional reviewers".
- Tags:** A section with "No tags".
- Work items:** A section with "No work items".

A yellow oval highlights the "Branch Policies" section, which shows the PR is not compliant with the required reviewers policy. The policy is "Required" and lists "Alberto Domínguez Horner" as a required reviewer. The PR is currently "Not approved" (indicated by a red dot).

Continuous Integration by Pull Requests



The screenshot displays a GitHub Pull Request (PR) titled "PR Example - DevOps Tutorial". The PR is active and proposed by Alberto Domínguez Horner, aiming to merge the 'development' branch into 'main'. The interface shows the 'Files' tab, highlighting a change in the file 'notebook-content.py' within the 'Fabric_Workshop/DevOps_Notebook.Notebook' directory. The diff view shows a single line added (line 15):

```
15 + # This is an interesting change
```

The file's content is a Jupyter Notebook file (notebook-content.py) with the following structure:

```
1 1 # Fabric notebook source
2 2
3 3 # METADATA *****
4 4
5 5 # META {
6 6 # META   "kernel_info": {
7 7 # META     "name": "synapse_pyspark"
8 8 # META   },
9 9 # META   "dependencies": {}
10 10 # META }
11 11
12 12 # CELL *****
13 13
14 14 # The only purpose of this notebook is to track its changes with Azure DEVOps.
15 15 + # This is an interesting change
16 16
17 17 # METADATA *****
18 18
19 19 # META {
20 20 # META   "language": "python",
21 21 # META   "language_group": "synapse_pyspark"
22 22 # META }
23 23
24 24 # CELL *****
25 25
26 26
27 27 # METADATA *****
28 28
29 29 # META {
30 30 # META   "language": "python",
31 31 # META   "language_group": "synapse_pyspark"
32 32 # META }
33 33 |
```



Thanks!

Now you are ready to leverage Azure DevOps for Microsoft Fabric.