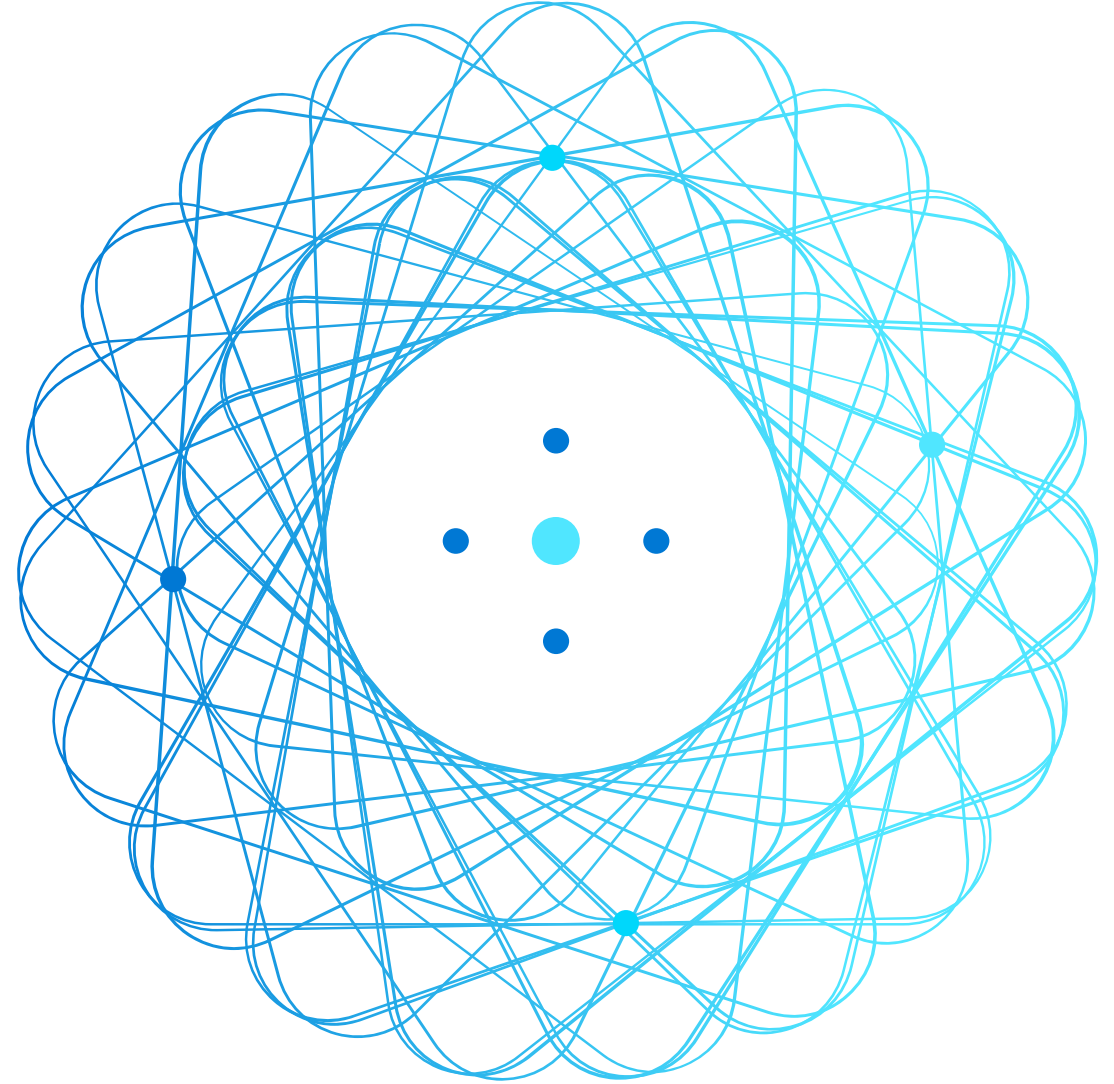


Explore the Azure Machine Learning workspace



Module Agenda

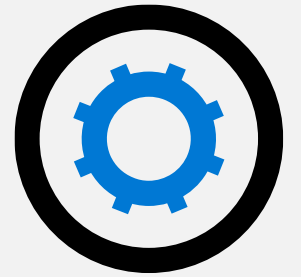


Explore Azure Machine Learning workspace resources and assets



Explore developer tools for workspace interaction

Explore Azure Machine Learning workspace resources and assets



Introducing Azure Machine Learning

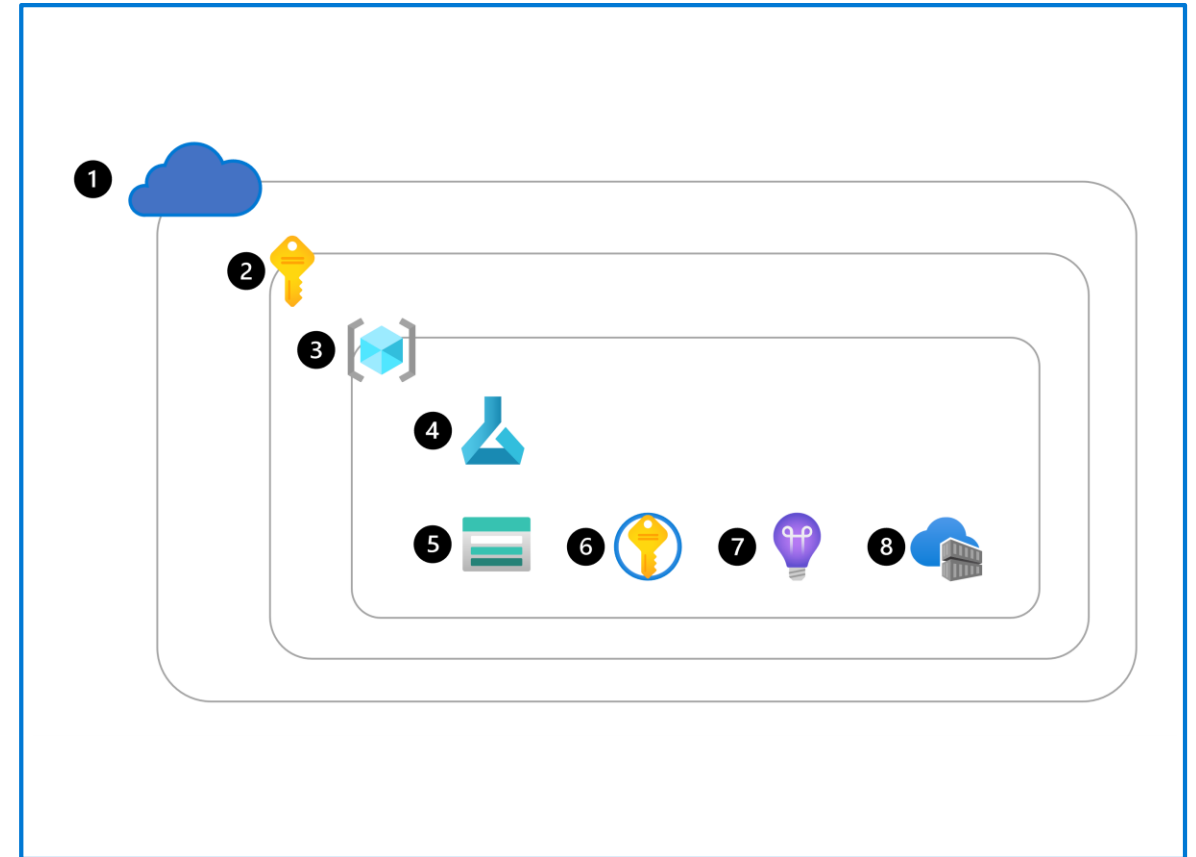
Azure Machine Learning provides a platform for data scientists to train, deploy, and manage their machine learning models on the Microsoft Azure platform.

It also provides a comprehensive set of resources and assets to train and deploy effective machine learning models.

Understand the Azure Machine Learning service

To create an Azure Machine Learning service, you'll have to:

1. Get access to **Azure**, for example through the Azure portal.
2. Sign in to get access to an **Azure subscription**.
3. Create a **resource group** within your subscription.
4. Create an **Azure Machine Learning service** to create a workspace.
5. **Azure Storage Account**: To store files and notebooks used in the workspace, and to store metadata of jobs and models.
6. **Azure Key Vault**: To securely manage secrets such as authentication keys and credentials used by the workspace.
7. **Application Insights**: To monitor predictive services in the workspace.
8. **Azure Container Registry**: Created when needed to store images for Azure Machine Learning environments.



Create the workspace

You can create an Azure Machine Learning workspace in any of the following ways:

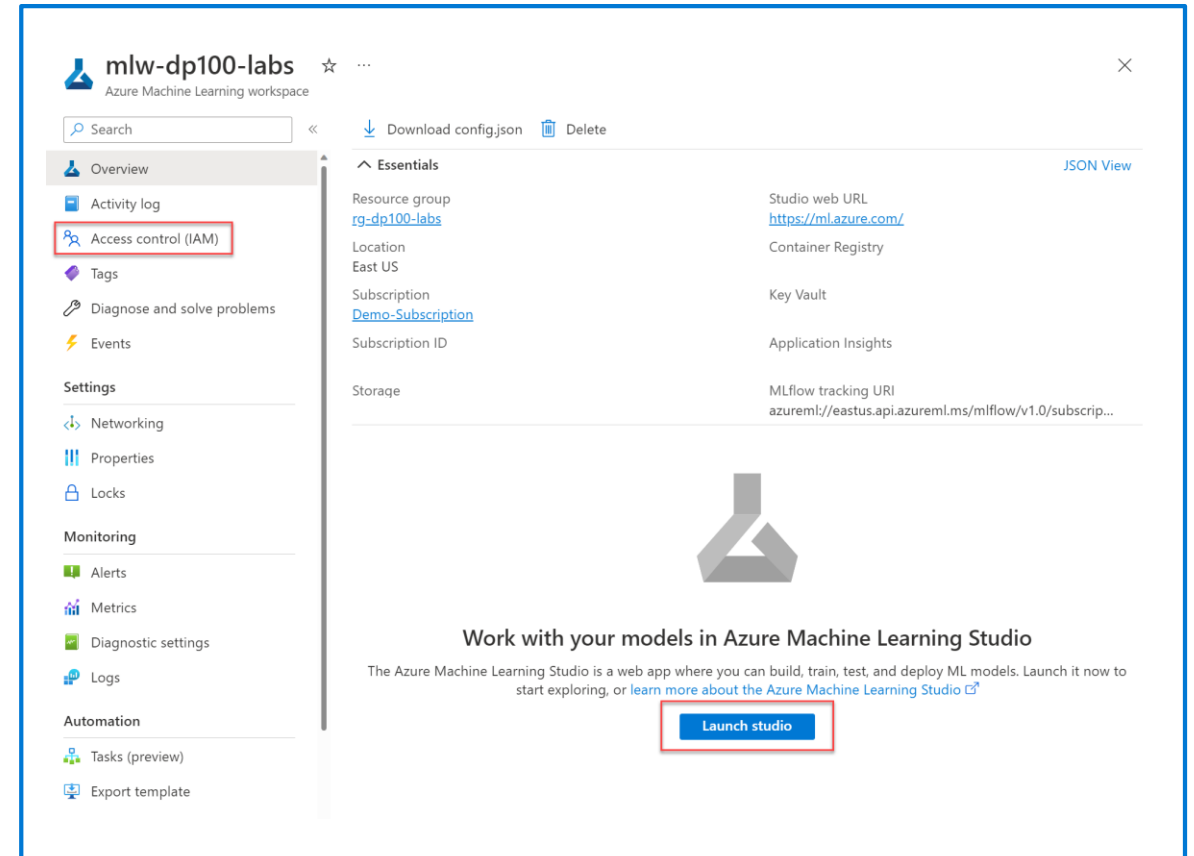
- Use the user interface in the Azure portal.
- Create an Azure Resource Manager (ARM) template.
- Use the Azure Command Line Interface (CLI) with the Azure Machine Learning CLI extension.
- Use the Azure Machine Learning Python SDK.

Python

```
from azure.ai.ml.entities import Workspace
workspace_name = "mlw-example"
ws_basic = Workspace(
    name=workspace_name,
    location="eastus",
)
ml_client.workspaces.begin_create(ws_basic)
```

Explore the workspace in the Azure portal

- Give others access to the Azure Machine Learning workspace, using the **Access control**.
- Launch the **Azure Machine Learning studio**, an easy-to-use interface to create, manage, and use resources and assets in the workspace.



Give access to the Azure Machine Learning workspace

Access is granted in Azure using role-based access control (RBAC)

There are three general built-in roles that you can use across resources and resource groups to assign permissions to other users:

- **Owner:** Gets full access to all resources, and can grant access to others using access control.
- **Contributor:** Gets full access to all resources, but can't grant access to others.
- **Reader:** Can only view the resource, but isn't allowed to make any changes.

Additionally, Azure Machine Learning has specific built-in roles you can use:

- AzureML Data Scientist
- AzureML Compute Operator

To fully customize permissions, create a custom role.

Identify Azure Machine Learning resources



The workspace - the top-level resource for Azure Machine Learning. The workspace stores all logs, metrics, outputs, models, and snapshots of your code.



Compute resources - There are four types of compute in the Azure Machine Learning workspace: Compute instance, Compute cluster, Inference cluster and Attached compute.



Datastores - all data is stored in datastores, which are references to Azure data services. When connected to the workspace, two datastores will be added to your workspace: `workspacefilestore` and `workspaceblobstore`

Identify Azure Machine Learning assets



Models - You can train machine learning models with various frameworks, like Scikit-learn or PyTorch. A common way to store such models is to package the model as a Python pickle file (.pkl extension).



Environments - specify software packages, environment variables, and software settings to run scripts. An environment is stored as an image in the Azure Container Registry created with the workspace when it's used for the first time.



Data - You can use data assets to easily access data every time, without having to provide authentication every time you want to access it. When you create a data asset in the workspace, you'll specify the path to point to the file or folder, and the name and version.

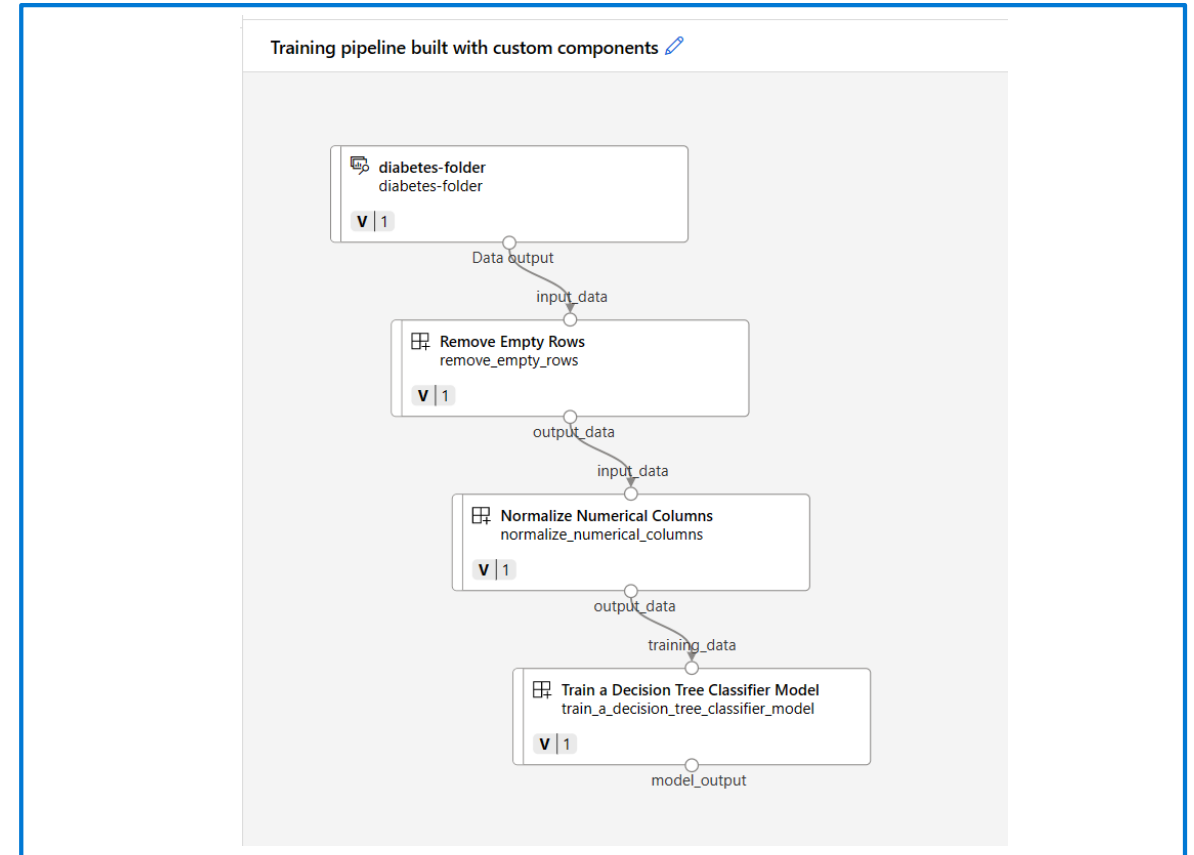


Components - make it easier to share code with component in a workspace.

Author and run a pipeline with the designer

Use the designer for:

- Easy and quick exploration during the initial phase of training machine learning models.
- Swiftly create pipelines using components you've created and registered in the workspace.



Explore algorithms and hyperparameter values with Automated Machine Learning

Create a new Automated ML job

- Select data asset
- Configure job
- Select task and settings
- Hyperparameter configuration (Computer Vision only)
- Validate and test

Classification
To predict one of several categories in the target column. yes/no, blue, red, green.

Regression
To predict continuous numeric values.

Time series forecasting ✓
To predict values based on time.

The time series forecasting method requires some additional information.

Time column * ⓘ
WeekStarting (Date) ▼

Time series identifier(s) ⓘ
☒ Autodetect

Frequency * ⓘ
☒ Autodetect

Forecast horizon * ⓘ
☒ Autodetect

☐ Enable deep learning ⓘ

Back Next Cancel

Run a Jupyter notebook

The screenshot displays the Azure Machine Learning Studio interface. On the left, the 'Notebooks' sidebar shows a file explorer with a tree structure: 'Users' > 'madiepev' > 'azure-ml-labs' > 'Instructions' > 'Labs' > '01' > 'src'. The selected notebook is 'Run training script.ipynb'. The main area shows the notebook content, which includes a title 'Run a training script with the Python SDK', an introduction paragraph, a 'Before you start' section with a note about installing the `azure-ai-ml` package, a code cell with the command `pip show azure-ai-ml`, and a 'Connect to your workspace' section. The top toolbar includes options to 'Edit in VS Code (pr...)' and 'Compute instance: ci11...'. The status bar indicates 'Last saved a few seconds ago'.

Notebooks

Files Samples

Users

madiepev

azure-ml-labs

Instructions

Labs

01

src

Run training script.ipynb

02

03

04

05

06

07

08

09

10

Run training script.ipynb X

Edit in VS Code (pr...

Compute instance: ci11...

Viewing

Last saved a few seconds ago

Run a training script with the Python SDK

You can use the Python SDK for Azure Machine Learning to submit scripts as jobs. By using jobs, you can easily keep track of the input parameters and outputs when training a machine learning model.

Before you start

You'll need the latest version of the **azureml-ai-ml** package to run the code in this notebook. Run the cell below to verify that it is installed.

Note: If the **azure-ai-ml** package is not installed, run `pip install azure-ai-ml` to install it.

+ Code + Markdown

```
1 pip show azure-ai-ml
```

Press shift + enter to run

Connect to your workspace

With the required SDK packages installed, now you're ready to connect to your workspace.

To connect to a workspace, we need identifier parameters - a subscription ID, resource group name, and workspace name. The resource group name and workspace name are already filled in for you. You only need the subscription ID to complete the command.

To find the necessary parameters, click on the subscription and workspace name at the top right of the Studio. A pane will open on the right.

Run a script as a job

You can run a script as a job in Azure Machine Learning.

When you submit a job to the workspace, all inputs and outputs will be stored in the workspace.

There are different types of jobs:

- **Command:** Execute a single script.
- **Sweep:** Perform hyperparameter tuning when executing a single script.
- **Pipeline:** Run a pipeline consisting of multiple scripts or components.

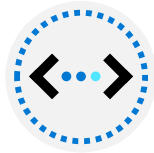
The screenshot displays the Azure Machine Learning workspace interface for a job named 'diabetes-train-mlflow'. The job is in a 'Completed' state, indicated by a green checkmark. The breadcrumb navigation shows the path: Microsoft Non-Production > mlw-dp100-labs > Jobs > diabetes-training > diabetes-train-mlflow. The job details are organized into several sections:

- Overview:** Shows the job status as 'Completed' with a green checkmark. Below this are tabs for Overview, Metrics, Images, Child jobs, Outputs + logs, and Code. Action buttons include Refresh, Connect to compute, Edit and submit, Register model, Cancel, and Delete.
- Properties:** A table-like section listing job metadata:
 - Status: Completed (with green checkmark)
 - Created on: Nov 4, 2022
 - Start time: Nov 4, 2022
 - Duration: 1m 36.99s
 - Compute duration: 1m 36.99s
 - Name: yellow_head
 - Command: `python train.py --training_data ${{(inputs.diabetes_data)}} --reg_rate ${{(inputs.reg_rate)}}`
 - Created by: (empty)
 - Job type: Command
 - Experiment: diabetes-training
- Inputs:** Shows the input 'reg_rate' with a value of '0.01'. The input name is 'diabetes_data' and the data source is 'diabetes-data:1'.
- Outputs:** Shows the output 'mlflow_log_model' with the model name 'azureml_yellow_head_..._output_mlflow_log_model_1729586584:1'.
- Tags:** A list of tags including 'estimator_class : sklearn.linear_model_logistic.LogisticRegression', 'estimator_name : LogisticRegression', and 'model_type : LogisticRegression'.
- Params:** A list of hyperparameters such as 'C : 100.0', 'class_weight : None', 'dual : False', 'fit_intercept : True', 'intercept_scaling : 1', 'l1_ratio : None', 'max_iter : 100', 'multi_class : auto', 'n_jobs : None', 'penalty : l2', 'random_state : None', 'Regularization rate : 0.01', 'solver : liblinear', and 'tol : 0.0001'.

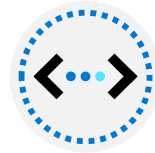
Exercise - Explore the workspace

In this exercise, you will:

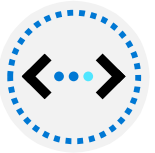
Task 1: Create an Azure Machine Learning workspace



Task 2: Explore the Azure Machine Learning studio



Task 3: Author and run a training pipeline with the designer



Instructions

Follow these instructions to complete the exercise:

1. View the exercise repo at <https://microsoftlearning.github.io/mslearn-azure-ml/>.
2. Complete the **Explore the Azure Machine Learning workspace** exercise.

Knowledge check



A data scientist needs access to the Azure Machine Learning workspace to run a script as a job. Which role should be used to give the data scientist the necessary access to the workspace?

- ☐ Reader
 - ☒ AzureML Data Scientist
 - ☐ AzureML Compute Operator
-



The data scientist wants to run a single script to train a model. What type of job is the best fit to run a single script?

- ☒ Azure Command
- ☐ Azure Pipeline
- ☐ Azure Sweep

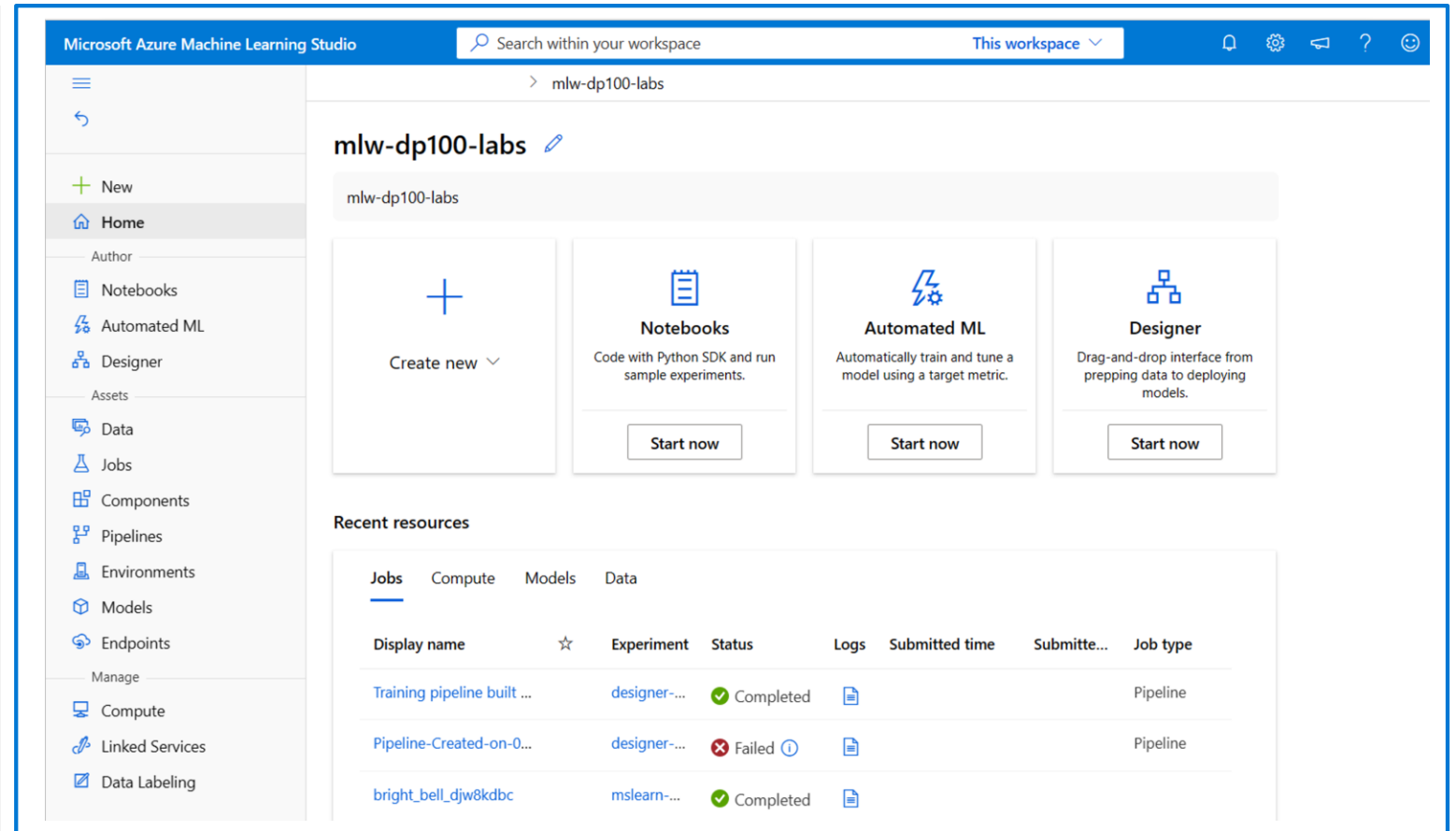
Explore developer tools for workspace interaction



Explore the studio

There are two common ways to access the Azure Machine Learning studio:

- Launch the studio from the **Overview** page in the Azure portal
- Navigate to the studio directly by signing in at **<https://ml.azure.com>**



Explore the Python SDK

1. Install the Python SDK

```
pip install azure-ai-ml
```

2. Connect to the workspace

```
from azure.ai.ml import MLClient
from azure.identity import DefaultAzureCredential

ml_client = MLClient(
    DefaultAzureCredential(), subscription_id,
    resource_group, workspace
)
```

Explore the CLI



The Azure CLI is commonly used by administrators and engineers to automate tasks in Azure.



There are many advantages to using the Azure CLI with Azure Machine Learning. The Azure CLI allows you to:

- Automate the creation and configuration of assets and resources to make it **repeatable**.
- Ensure **consistency** for assets and resources that must be replicated in multiple environments
- Incorporate machine learning asset configuration into developer operations **(DevOps) workflows**, such as **continuous integration** and **continuous deployment (CI/CD)** pipelines.

Install the Azure CLI



You can install the Azure CLI on a **Linux**, **Mac**, or **Windows** computer.



With the Azure CLI, you run commands or scripts to manage Azure resources.



You can also use the Azure CLI from a browser through the **Azure Cloud Shell**.

Install the Azure Machine Learning extension

1. Install on Windows

```
az extension add -n ml -y
```

2. Work with the Azure CLI

```
az ml compute create --file compute.yml --resource-group my-resource-group  
--workspace-name my-workspace
```

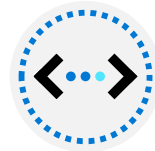
Exercise - Explore the developer tools

In this exercise, you will:

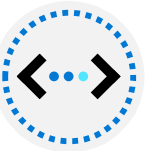
Task 1: Create resources with the Azure CLI.



Task 2: Explore the Azure Machine Learning workspace with the studio



Task 3: Use the Python SDK to train a model



Instructions

Follow these instructions to complete the exercise:

1. View the exercise repo at <https://microsoftlearning.github.io/mslearn-azure-ml/>.
2. Complete the **Explore developer tools for workspace interaction** exercise.

Knowledge check



A data scientist wants to experiment by training a machine learning model and tracking it with Azure Machine Learning. Which tool should be used to train the model by running a script from their preferred environment?



A machine learning model to predict the sales forecast has been developed. Every week, new sales data is ingested and the model needs to be retrained on the newest data before generating the new forecast. Which tool should be used to retrain the model every week?

Knowledge check



A data scientist wants to experiment by training a machine learning model and tracking it with Azure Machine Learning. Which tool should be used to train the model by running a script from their preferred environment?

- ☐ The Azure Machine Learning studio
 - ☒ The Python SDK
 - ☐ The Azure CLI
-



A machine learning model to predict the sales forecast has been developed. Every week, new sales data is ingested, and the model needs to be retrained on the newest data before generating the new forecast. Which tool should be used to retrain the model every week?

- ☐ The Azure Machine Learning studio
- ☐ The Python SDK
- ☒ The Azure CLI

