

Design a machine learning solution

- Determine the appropriate compute specifications for a training workload
- Describe model deployment requirements
- Select which development approach to use to build or train a model

Manage an Azure Machine Learning workspace

- Create an Azure Machine Learning workspace
- Manage a workspace by using developer tools for workspace interaction

- Set up Git integration for source control
- Create and manage registries

Manage data in an Azure Machine Learning workspace

- Select Azure Storage resources
- Register and maintain datastores
- Create and manage data assets

Manage compute for experiments in Azure Machine Learning

- Create compute targets for experiments and training

- Select an environment for a machine learning use case
- Configure attached compute resources, including Azure Synapse Spark pools and serverless Spark compute
- Monitor compute utilization

Explore data by using data assets and data stores

- Access and wrangle data during interactive development
- Wrangle interactive data with attached Synapse Spark pools and serverless Spark compute

Use notebooks for custom model training

- Develop code by using a compute instance
- Track model training by using MLflow
- Evaluate a model
- Train a model by using Python SDK v2
- Use the terminal to configure a compute instance

Convert the functionalities in to function or class based

Tune hyperparameters with Azure

Machine Learning

- Select a sampling method
- Define the search space
- Define the primary metric
- Define early termination options

Here you can try random, grid or bayesian sampling

Run model training scripts

- Configure job run settings for a script
- Configure compute for a job run
- Consume data from a data asset in a

job

- Run a script as a job by using Azure Machine Learning
- Use MLflow to log metrics from a job run
- Use logs to troubleshoot job run errors

Implement training pipelines

- Create a pipeline
- Pass data between steps in a pipeline
- Run and schedule a pipeline
- Monitor pipeline runs
- Create custom components

- Use component-based pipelines

You can use yaml or kubeflow or Azure ml pipeline features here

Manage models in Azure Machine Learning

- Describe MLflow model output
- Identify an appropriate framework to package a model
- Assess a model by using responsible AI principles

Deploy a model

- Configure settings for online

deployment

- Configure compute for a batch deployment
- Deploy a model to an online endpoint
- Deploy a model to a batch endpoint
- Test an online deployed service
- Invoke the batch endpoint to start a batch scoring job
- Trigger an Azure Machine Learning job, including from Azure DevOps or GitHub
- Automate model retraining based on new data additions or data changes
- Define event-based retraining triggers

