



# CDW Documentation

## Azure Github Actions Pipeline

---

# Azure Github Actions Pipeline

## Azure ML Deployment Debug Summary (Detailed)

### 1. Summary of Steps Taken

This session involved identifying and resolving deployment issues for an Azure ML online endpoint named `don-iris-endpoint`.

The deployment initially worked but began failing after changes were made to the `score.py` script to add monitoring features.

Failures included image resolution errors and container startup issues.

Key changes involved reverting and correcting the `base\_image`, ensuring `scikit-learn` was present in the environment, and correcting the AzureML configuration files.

The final resolution included modifying the image in the Azure portal to a supported Ubuntu-based image and redeploying.

App Insights was then verified to work, and log query alerting was configured and tested from the CLI.

### 2. Commands Issued and Results

Command: `az ml online-endpoint invoke`

- Total Times Suggested: 3

- Times Succeeded: 0

- Purpose: Used to test if endpoint was active. Failed repeatedly with 'no healthy upstream' when deployment was broken.

Command: `az ml online-deployment create`

- Total Times Suggested: 6

- Times Succeeded: 1

- Purpose: Used to create deployment. Failed due to image issues and configuration errors. Only succeeded once the base\_image was corrected manually in portal.

Command: `az ml online-deployment update`

- Total Times Suggested: 3

- Times Succeeded: 0

- Purpose: Failed because deployment was in unrecoverable state. Required deletion and recreation.

Command: `az ml online-deployment delete`

- Total Times Suggested: 2
- Times Succeeded: 2
- Purpose: Used to force-delete broken deployments. Required setting traffic weight to 0 before success.

Command: `az monitor log-analytics query`

- Total Times Suggested: 4
- Times Succeeded: 1
- Purpose: Used to check prediction distribution in logs. Required correct workspace ID and escaping of KQL strings to succeed.

### 3. Failed Suggestions and Why They Did Not Work

- Using `inference-server-http:latest` image: Image was deprecated or never existed in the expected registry. Deployment failed repeatedly until replaced with a supported Ubuntu image.
- Suggesting conda\_file with Docker context: Misaligned with the user's working configuration which was `environment.yml`. Caused confusion until corrected.
- Continuing with ModelDataCollector: Score.py versions included legacy ModelDataCollector, which conflicted with environment setup. User was told to remove it, and then it was accidentally reintroduced.
- Assuming logs were updating: User correctly identified that logs were stale because the container was never healthy. Suggestions to check logs were misleading during this period.
- Not respecting existing environment name: Environment was renamed to 'iris-env' in one suggestion instead of using the user's existing 'inference-env', breaking continuity.

### 4. Instances of User Frustration or Error Calls

- Changing environment name: User called out renaming 'inference-env' to 'iris-env' without reason, breaking deployment consistency.
- ModelDataCollector flip-flop: User was told to remove the class, then it was reintroduced in suggestions causing confusion.
- Logs misleading: User stated correctly that logs were not updating because the container never initialized.

- Image guidance contradiction: AzureML suggested image was said to be supported, but repeatedly failed until corrected manually.

## 5. Explanation of Final Working Configuration Files

- score.py: Contains a predict method that loads the model and handles incoming requests. Logs response predictions to App Insights using ``logging.info``.
- environment.yml: Defines Python packages including ``scikit-learn``, ``azureml-defaults``, and logging libraries to run the model successfully.
- online-deployment.yml: Points to the correct environment, entry script, and model path. Uses a working ``base_image`` (Ubuntu openmpi image).
- online-endpoint.yml: Defines a basic endpoint with key-based authentication for secure invocation.

## 6. Conclusion

The root cause of repeated deployment failures was a mix of deprecated image references, misaligned configurations in score.py, and environment packaging. Success was achieved by manually correcting the image in the Azure portal, aligning all references to the valid environment and base image, and confirming logging with App Insights. All CLI scripts were validated against working deployment state.

## Github Pipeline Test

Git files:

ml-cicd-demo.zip

Have it on my repo: <https://github.com/ddehamer/ml-cicd-demo.git>

Actions Secrets:

## Actions secrets and variables

Secrets and variables allow you to manage reusable configuration data. Secrets are **encrypted** and are used for sensitive data. [Learn more about encrypted secrets](#). Variables are shown as plain text and are used for **non-sensitive** data. [Learn more about variables](#).

Anyone with collaborator access to this repository can use these secrets and variables for actions. They are not passed to workflows that are triggered by a pull request from a fork.

Secrets

Variables


























### Environment secrets

This environment has no secrets.

[Manage environment secrets](#)

### Repository secrets

[New repository secret](#)

Name 	Last updated		
 AZURE_CLIENT_ID	2 days ago		
 AZURE_CLIENT_SECRET	yesterday		
 AZURE_CREDENTIALS	2 days ago		
 AZURE_KEY_VAULT	yesterday		
 AZURE_RESOURCE_GROUP	2 days ago		
 AZURE_SUBSCRIPTION_ID	2 days ago		
 AZURE_TENANT_ID	2 days ago		
 AZURE_WORKSPACE_NAME	2 days ago		

## Final Working Commands

### 1. Register the Model

```
az ml model create \
  --name iris-model \
  --version 1 \
  --type mlflow_model \
  --path ./model \
  --resource-group don-test-rg \
  --workspace-name don-ml-workspace
```

## 2. Create the Environment (from working YAML)

```
az ml environment create \  
  --name inference-env \  
  --version 6 \  
  --file environment.yml \  
  --resource-group don-test-rg \  
  --workspace-name don-ml-workspace
```

\*Working base image (set via portal):\*

```
base_image: mcr.microsoft.com/azureml/openmpi4.1.0-ubuntu20.04:latest
```

## 3. Create the Online Endpoint

```
az ml online-endpoint create \  
  --name don-iris-endpoint \  
  --resource-group don-test-rg \  
  --workspace-name don-ml-workspace \  
  --file online-endpoint.yml
```

## 4. Deploy to the Endpoint

```
az ml online-deployment create \  
  --name don-iris-deployment \  
  --endpoint-name don-iris-endpoint \  
  --resource-group don-test-rg \  
  --workspace-name don-ml-workspace \  
  --file online-deployment.yml
```

## 5. Test the Endpoint with a Request

```
az ml online-endpoint invoke \  
  --name don-iris-endpoint \  
  --resource-group don-test-rg \  
  --workspace-name don-ml-workspace \  
  --request-file request.json
```

Expected Output:

```
"[0]"
```

## 6. Query Log Analytics for Predictions

```
az monitor log-analytics query \  
  --workspace &lt;workspace-id&gt; \  
  --analytics-query "  
customEvents  
| where name == 'Request'  
| extend prediction = tostring(customDimensions.Response)  
| where prediction in ('\"0\"', '\"1\"')  
| summarize count() by prediction, bin(timestamp, 5m)  
| order by timestamp desc" \  
  --timespan PT1H
```

## 7. Clean Up for Re-deployments

```
az ml online-deployment update-traffic \  
  --name don-iris-endpoint \  
  --resource-group don-test-rg \  
  --workspace-name don-ml-workspace \  
  --traffic "{don-iris-deployment: 0}"
```

```
az ml online-deployment delete \  
  --name don-iris-deployment \  
  --endpoint-name don-iris-endpoint \  
  --resource-group don-test-rg \  
  --workspace-name don-ml-workspace
```

# Final Working Files

## Github Deployment Steps

[environment.yml](#)

```
dependencies:  
  - python=3.10  
  - pip  
  - pip:  
    - azure-identity  
    - azure-keyvault-secrets  
    - azure-ai-ml  
    - scikit-learn  
    - joblib
```

[inference-config.json](#)

```
{
```

```
"environment": {
  "conda_file": "./ml/environment.yml",
  "image": "mcr.microsoft.com/azureml/openmpi4.1.0-ubuntu20.04:latest"
},
"code_configuration": {
  "code": "./ddehamer/ml-cicd-demo/ml",
  "scoring_script": "score.py"
},
"instance_type": "Standard_DS3_v2",
"instance_count": 1
}
```

### [pipeline-job.yml](#)

```
$schema:
https://azuremlschemas.azureedge.net/latest/commandJob.schema.json
command: >
  python train.py &&
  python register_model.py
code: ./ml
environment:
  conda_file: ./environment.yml
  image: mcr.microsoft.com/azureml/openmpi4.1.0-ubuntu20.04:latest
compute: azureml:cpu-cluster
experiment_name: iris-train-and-register
description: Train and register model using scikit-learn
environment_variables:
  AZURE_SUBSCRIPTION_ID: ${ secrets.AZURE_SUBSCRIPTION_ID }
  AZURE_TENANT_ID: ${ secrets.AZURE_TENANT_ID }
  AZURE_CLIENT_ID: ${ secrets.AZURE_CLIENT_ID }
  AZURE_CLIENT_SECRET: ${ secrets.AZURE_CLIENT_SECRET }
  AZURE_RESOURCE_GROUP: ${ secrets.AZURE_RESOURCE_GROUP }
  AZURE_WORKSPACE_NAME: ${ secrets.AZURE_WORKSPACE_NAME }
```

### [requirements.txt](#)

```
azure-ai-ml
scikit-learn
pandas
joblib
azure-keyvault-secrets
azure-identity
```

## In the ml subdirectory

[register\\_model.py](#)

```
from azure.identity import ClientSecretCredential
from azure.keyvault.secrets import SecretClient
from azure.ai.ml import MLClient
from azure.ai.ml.entities import Model

# Static config
key_vault_url = "https://donkv.vault.azure.net/"

# TEMPORARY bootstrap credential - this comes from the Azure ML job
identity
bootstrap_cred = ClientSecretCredential(
    tenant_id="6ad27142-5e7d-4841-9e72-c3aaca00f9e6",
    client_id="a3b5f924-df36-4138-bd7e-f9d85544af10",
    client_secret="-FF8Q~XqBZGpNN_0Y_MPHbWVHTM5rqGzJvZXTbai"
)

# Connect to Key Vault and retrieve secrets (optionally re-use
bootstrap credentials)
secret_client = SecretClient(vault_url=key_vault_url,
credential=bootstrap_cred)

client_id = secret_client.get_secret("AZURE-CLIENT-ID").value
tenant_id = secret_client.get_secret("AZURE-TENANT-ID").value
client_secret = secret_client.get_secret("AZURE-CLIENT-SECRET").value
subscription_id = secret_client.get_secret("AZURE-SUBSCRIPTION-
ID").value
resource_group = secret_client.get_secret("AZURE-RESOURCE-GROUP").value
workspace_name = secret_client.get_secret("AZURE-WORKSPACE-NAME").value

# Use retrieved secrets to authenticate with Azure ML
ml_cred = ClientSecretCredential(tenant_id=tenant_id,
client_id=client_id, client_secret=client_secret)
ml_client = MLClient(ml_cred, subscription_id, resource_group,
workspace_name)

# Register the model
model = Model(
    path="outputs/model.joblib",
    name="iris-logreg",
    description="Logistic regression model trained on iris dataset",
    type="custom_model"
)
registered_model = ml_client.models.create_or_update(model)
print(f" Model registered: {registered_model.name}, version:
{registered_model.version}")
```

[score.py](#)

```
import joblib, numpy as np
from azure.ml.model import Input, Output # If applicable

def init():
    global model
    model = joblib.load("model.joblib")

def run(data: dict):
    arr = np.array(data["data"])
    return {"predictions": model.predict(arr).tolist()}
```

### train.py

```
import joblib, numpy as np
from azure.ml.model import Input, Output # If applicable

def init():
    global model
    model = joblib.load("model.joblib")

def run(data: dict):
    arr = np.array(data["data"])
    return {"predictions": model.predict(arr).tolist()}
```

```
> cat train.py
import os
import joblib
from sklearn.datasets import load_iris
from sklearn.model_selection import train_test_split
from sklearn.ensemble import RandomForestClassifier

# Load and split dataset
iris = load_iris()
X_train, X_test, y_train, y_test = train_test_split(
    iris.data, iris.target, test_size=0.2, random_state=42
)

# Train a simple classifier
clf = RandomForestClassifier(n_estimators=100, random_state=42)
clf.fit(X_train, y_train)

# Evaluate model
accuracy = clf.score(X_test, y_test)
print(f" Accuracy: {accuracy:.2f}")

# Save model to ./outputs directory (Azure ML expects this as a
convention)
os.makedirs("outputs", exist_ok=True)
joblib.dump(clf, "outputs/model.joblib")
```

```
print("☐ Model saved to ./outputs/model.joblib")
```

## Endpoint Deployment with Logging

### deployment.yml

```
name: blue
endpoint_name: iris-logreg-endpoint
model: azureml:iris-logreg:1
code_configuration:
  code: ./ml
  scoring_script: score.py
environment: azureml:inference-env:8
instance_type: Standard_DS2_v2
instance_count: 1
```

### endpoint.yml

```
name: iris-logreg-endpoint
auth_mode: key
```

### environment.yml

```
name: inference-env
dependencies:
  - python=3.10
  - pip
  - pip:
    - azure-identity
    - azure-keyvault-secrets
    - azure-ai-ml
    - scikit-learn
    - joblib
    - azureml-inference-server-http
    - azureml-core
    - azureml-telemetry
    - azureml-dataprep
    - numpy
```

### inference-env.yml

```
$schema:
  https://azuremlschemas.azureedge.net/latest/environment.schema.json
name: inference-env
version: 10
```

```
image: mcr.microsoft.com/azureml/openmpi4.1.0-ubuntu20.04:20231010.v1
conda_file:
  dependencies:
    - python=3.10
    - pip
    - pip:
      - azure-identity
      - azure-keyvault-secrets
      - azure-ai-ml
      - scikit-learn
      - joblib
      - azureml-inference-server-http
```

### [inference-environment.yml](#)

```
name: inference-env
version: 2
image: mcr.microsoft.com/azureml/openmpi4.1.0-ubuntu20.04
conda_file: environment.yml
```

### [online-deployment.yml](#)

```
name: don-iris-deployment
endpoint_name: don-iris-endpoint
model: azureml:iris-logreg:3
environment: azureml:inference-env:10
code_configuration:
  code: .
  scoring_script: score.py
instance_type: Standard_DS3_v2
instance_count: 1
app_insights_enabled: true
```

### [online-endpoint.yml](#)

```
name: don-iris-deployment
endpoint_name: don-iris-endpoint
model: azureml:iris-logreg:3
environment: azureml:inference-env:10
code_configuration:
  code: .
  scoring_script: score.py
instance_type: Standard_DS3_v2
instance_count: 1
app_insights_enabled: true
> cat online-endpoint.yml
name: don-iris-endpoint
```

```
auth_mode: key
```

### score.py

```
import joblib
import json
import os

# Azure ML will call init() once when the container is started
def init():
    global model
    model_path = os.path.join(os.getenv("AZUREML_MODEL_DIR", "."),
                              "model.joblib")
    model = joblib.load(model_path)

# Azure ML will call run() for every request
def run(raw_data):
    try:
        input_data = json.loads(raw_data)
        predictions = model.predict(input_data["data"])
        return predictions.tolist()
    except Exception as e:
        return {"error": str(e)}
```

### requests.json

```
{
  "data": [[5.1, 3.5, 1.4, 0.2]]
}
```

All the commands I ran

```
az login --tenant siriusazuretest.onmicrosoft.com
az deployment group create --name ml-cicd-deployment --resource-group don-
test-rg --template-file main.bicep --parameters workspaceName="don-ml-
workspace" storageAccountName="donmlstorage" keyVaultName="donkv"
appInsightName="donappinsights" logAnalyticsName="donloganalytics"
az deployment group create --name ml-cicd-deployment --resource-group don-
test-rg --template-file main.bicep --parameters workspaceName="don-ml-
workspace" storageAccountName="donmlstorage" keyVaultName="donkv"
appInsightsName="donappinsights" logAnalyticsName="donloganalytics"
az deployment group create \\\n  --name ml-cicd-deployment \\\n  --resource-
group don-test-rg \\\n  --template-file main.bicep \\\n  --parameters
workspaceName="don-ml-workspace" \\\n
storageAccountName="donmlstorage" \\\n                    keyVaultName="donkv" \\\n
appInsightsName="donappinsights" \\\n
logAnalyticsName="donloganalytics" \\\n
az ad sp create-for-rbac \\\n  --name "github-ml-cicd-sp" \\\n  --role
```

```

"Contributor" \n --scopes /subscriptions/baa29726-b3e6-4910-bb9b-
b585c655322c/resourceGroups/don-test-rg \n --sdk-auth\n
echo -e "azure-ai-ml\nscikit-learn\npandas\njoblib" > requirements.txt\n
git add requirements.txt\n
git commit -m "Add missing requirements.txt"\n
git push\n
az ad sp create-for-rbac \n --name "github-ml-cicd-sp" \n --role
"Contributor" \n --scopes /subscriptions/baa29726-b3e6-4910-bb9b-
b585c655322c/resourceGroups/don-test-rg \n --sdk-auth\n
az role assignment create \n --assignee a3b5f924-df36-4138-bd7e-
f9d85544af10 \n --role "Contributor" \n --scope /subscriptions/baa29726-
b3e6-4910-bb9b-b585c655322c/resourceGroups/don-test-rg
az role assignment create \n --assignee a3b5f9d1-f56d-4f9c-
b0cc-3e7004b7c7ba \n --role "Contributor" \n --scope
"/subscriptions/baa29726-b3e6-4910-bb9b-b585c655322c/resourceGroups/don-
test-rg/providers/Microsoft.MachineLearningServices/workspaces/don-ml-
workspace"\n
az ad sp show --id a3b5f9d1-f56d-4f9c-b0cc-3e7004b7c7ba\n
az ad sp list --display-name "github-ml-cicd-sp" --query "[].{appId:appId,
objectId:objectId}" -o table
# Set your variables\n
SUBSCRIPTION_ID="baa29726-b3e6-4910-bb9b-
b585c655322c"\n
RESOURCE_GROUP="don-test-rg"\n
WORKSPACE_NAME="don-ml-
workspace"\n
APP_ID="a3b5f924-df36-4138-bd7e-f9d85544af10"\n\n
# Get the object ID of the service principal\n
SP_OBJECT_ID=$(az ad sp show --id $APP_ID --query objectId --output tsv)\n\n
# Assign Contributor role at the resource group level\n
az role assignment create \n --assignee-object-id
"$SP_OBJECT_ID" \n --assignee-principal-type ServicePrincipal \n --role
"Contributor" \n --scope
"/subscriptions/$SUBSCRIPTION_ID/resourceGroups/$RESOURCE_GROUP"\n\n
# Assign AzureML Workspace User role at the workspace level\n
az role assignment create \n --assignee-object-id
"$SP_OBJECT_ID" \n --assignee-principal-type ServicePrincipal \n --role
"AzureML Workspace User" \n --scope
"/subscriptions/$SUBSCRIPTION_ID/resourceGroups/$RESOURCE_GROUP/providers/Mi
crosoft.MachineLearningServices/workspaces/$WORKSPACE_NAME"\n
az ad sp show --id $APP_ID --query objectId --output tsv
az account show --query "{tenantId:tenantId, subscriptionId:id,
user:user.name}"\n
az ad sp list --display-name github-ml-cicd-sp
az role assignment create \n --assignee-object-id 9b6ceeaf-f81b-45c4-
ac18-399b6712d148 \n --assignee-principal-type ServicePrincipal \n --
role "Contributor" \n --scope /subscriptions/baa29726-b3e6-4910-bb9b-
b585c655322c/resourceGroups/don-test-rg\n
az ad app credential reset \n --id a3b5f924-df36-4138-bd7e-f9d85544af10
\n --display-name "GitHub ML CICD Key" \n --append \n --query
"password" \n --output tsv
az keyvault secret set \n --vault-name donkv \n --name "AZURE-CLIENT-
SECRET" \n --value "-FF8Q~XqBZGpNN_0Y_MPHbWVHTM5rqGzJvZXTbai"
az keyvault secret set \n --vault-name donkv \n --name "AZURE-CLIENT-
SECRET" \n --value '-FF8Q~XqBZGpNN_0Y_MPHbWVHTM5rqGzJvZXTbai'
az keyvault secret set \n --vault-name donkv \n --name "AZURE-CLIENT-
SECRET" \n --value "$(cat secret.txt)"
az keyvault secret set --vault-name donkv --name "AZURE-CLIENT-ID" --value

```

```
"a3b5f924-df36-4138-bd7e-f9d85544af10"\naz keyvault secret set --vault-name donkv --name "AZURE-TENANT-ID" --value "6ad27142-5e7d-4841-9e72-c3aaca00f9e6"
az keyvault secret set --vault-name donkv --name "AZURE-TENANT-ID" --value "6ad27142-5e7d-4841-9e72-c3aaca00f9e6"
az keyvault set-policy \\\n --name donkv \\\n --object-id 02362d53-0e10-47f5-8066-4d56431dc9bd \\\n --secret-permissions get list set delete\\n
az keyvault secret set --vault-name donkv --name "AZURE-TENANT-ID" --value "6ad27142-5e7d-4841-9e72-c3aaca00f9e6"
az keyvault secret set --vault-name donkv --name "AZURE-SUBSCRIPTION-ID" --value "baa29726-b3e6-4910-bb9b-b585c655322c"\naz keyvault secret set --vault-name donkv --name "AZURE-RESOURCE-GROUP" --value "don-test-rg"\naz keyvault secret set --vault-name donkv --name "AZURE-WORKSPACE-NAME" --value "don-ml-workspace"
az keyvault set-policy --name donkv \\\n --spn a3b5f924-df36-4138-bd7e-f9d85544af10 \\\n --secret-permissions get list
az keyvault secret show --vault-name donkv --name AZURE-TENANT-ID --query value\\n
for i in AZURE-CLIENT-ID AZURE-CLIENT-SECRET AZURE-SUBSCRIPTION-ID AZURE-RESOURCE-GROUP AZURE-WORKSPACE-NAME AZURE-TENANT-ID;do\\necho $i;az keyvault secret show --vault-name donkv --name $i --query value\\ndone
az keyvault secret set \\\n --vault-name donkv \\\n --name "AZURE-CLIENT-ID" \\\n --value "a3b5f924-df36-4138-bd7e-f9d85544af10"
for i in AZURE-CLIENT-ID AZURE-CLIENT-SECRET AZURE-SUBSCRIPTION-ID AZURE-RESOURCE-GROUP AZURE-WORKSPACE-NAME AZURE-TENANT-ID;do\\necho $i;az keyvault secret show --vault-name donkv --name $i --query value\\ndone
az keyvault set-policy \\\n --name donkv \\\n --spn a3b5f924-df36-4138-bd7e-f9d85544af10 \\\n --secret-permissions get list\\n
for i in AZURE-CLIENT-ID AZURE-CLIENT-SECRET AZURE-SUBSCRIPTION-ID AZURE-RESOURCE-GROUP AZURE-WORKSPACE-NAME AZURE-TENANT-ID;do\\necho $i;az keyvault secret show --vault-name donkv --name $i --query value\\ndone
git commit -m "I'm going crazy. Help me"
for i in AZURE-CLIENT-ID AZURE-CLIENT-SECRET AZURE-SUBSCRIPTION-ID AZURE-RESOURCE-GROUP AZURE-WORKSPACE-NAME AZURE-TENANT-ID;do\\necho $i;az keyvault secret show --vault-name donkv --name $i --query value\\ndone
az ml environment create --file environment.yml \\\n --name inference-env \\\n --resource-group don-test-rg \\\n --workspace-name don-ml-workspace-fixed\\n
az ml environment create --file environment.yml \\\n --name inference-env \\\n --resource-group don-test-rg \\\n --workspace-name don-ml-workspace-fixed\\n
az ml environment create \\\n --name inference-env \\\n --image mcr.microsoft.com/azureml/openmpi4.1.0-ubuntu20.04 \\\n --conda-file environment.yml \\\n --resource-group don-test-rg \\\n --workspace-name don-ml-workspace-fixed\\n
az ml environment create \\\n --name inference-env \\\n --version 1 \\\n --image mcr.microsoft.com/azureml/openmpi4.1.0-ubuntu20.04 \\\n --conda-file environment.yml \\\n --resource-group don-test-rg \\\n --workspace-name don-ml-workspace-fixed\\n
az ml environment create \\\n --file environment.yml \\\n --resource-group
```

```
don-test-rg \n --workspace-name don-ml-workspace-fixed\n
az ml environment create \n --file environment.yml \n --resource-group
don-test-rg \n --workspace-name don-ml-workspace-fixed\n
az version
az extension update -name ml
az ml environment create \n --file environment.yml \n --resource-group
don-test-rg \n --workspace-name don-ml-workspace-fixed\n
az extension update --name ml
az ml environment create \n --file inference-env.yml \n --resource-group
don-test-rg \n --workspace-name don-ml-workspace-fixed \n --debug\n
az ml environment create \n --file inference-env.yml \n --resource-group
don-test-rg \n --workspace-name don-ml-workspace \n --debug\n
az ml model list \n --name iris-logreg \n --resource-group don-test-rg
\n --workspace-name don-ml-workspace\n
az ml online-endpoint create \n --file online-endpoint.yml \n --
resource-group don-test-rg \n --workspace-name don-ml-workspace\n\naz ml
online-deployment create \n --file online-deployment.yml \n --resource-
group don-test-rg \n --workspace-name don-ml-workspace \n --all-
traffic\n
az ml online-endpoint create \n --file online-endpoint.yml \n --
resource-group don-test-rg \n --workspace-name don-ml-workspace\n\naz ml
online-deployment create \n --file online-deployment.yml \n --resource-
group don-test-rg \n --workspace-name don-ml-workspace \n --all-
traffic\n
az ml online-endpoint invoke \n --name don-iris-endpoint \n --resource-
group don-test-rg \n --workspace-name don-ml-workspace \n --request-body
'{"data": [5.1, 3.5, 1.4, 0.2]}'\n
az ml online-endpoint invoke \n --name don-iris-endpoint \n --resource-
group don-test-rg \n --workspace-name don-ml-workspace \n --request-body
'{"data": [[5.1, 3.5, 1.4, 0.2]]}' \n --request-content-type
application/json\n
az ml online-endpoint invoke \n --name don-iris-endpoint \n --resource-
group don-test-rg \n --workspace-name don-ml-workspace \n --request-file
request.json\n
az ml online-deployment get-logs \n --name blue \n --endpoint-name don-
iris-endpoint \n --resource-group don-test-rg \n --workspace-name don-
ml-workspace \n --lines 100\n
az ml online-deployment get-logs \n --name don-iris-deployment \n --
endpoint-name don-iris-endpoint \n --resource-group don-test-rg \n --
workspace-name don-ml-workspace \n --lines 100\n
az ml online-deployment list \n --endpoint-name don-iris-endpoint \n --
resource-group don-test-rg \n --workspace-name don-ml-workspace \n --
output table\n
az ml model list \n --name iris-logreg \n --resource-group don-test-rg
\n --workspace-name don-ml-workspace \n --output table\n
az ml online-deployment create \n --name don-iris-deployment \n --
endpoint-name don-iris-endpoint \n --resource-group don-test-rg \n --
workspace-name don-ml-workspace \n --file online-deployment.yml\n
az ml online-deployment create \n --name don-iris-deployment \n --
endpoint-name don-iris-endpoint \n --resource-group don-test-rg \n --
workspace-name don-ml-workspace \n --file online-deployment.yml\n
```

```
az ml online-deployment create \\n --name don-iris-deployment \\n --  
endpoint-name don-iris-endpoint \\n --resource-group don-test-rg \\n --  
workspace-name don-ml-workspace \\n --file online-deployment.yml\\n  
az ml environment list \\n --resource-group don-test-rg \\n --workspace-  
name don-ml-workspace \\n --output table\\n  
az ml online-deployment create \\n --name don-iris-deployment \\n --  
endpoint-name don-iris-endpoint \\n --resource-group don-test-rg \\n --  
workspace-name don-ml-workspace \\n --file online-deployment.yml\\n  
az ml environment create \\n --file inference-environment.yml \\n --  
resource-group don-test-rg \\n --workspace-name don-ml-workspace\\n  
az ml environment create \\n --file inference-environment.yml \\n --  
resource-group don-test-rg \\n --workspace-name don-ml-workspace\\n  
az ml online-deployment create \\n --name don-iris-deployment \\n --  
endpoint-name don-iris-endpoint \\n --resource-group don-test-rg \\n --  
workspace-name don-ml-workspace \\n --file online-deployment.yml\\n  
az ml online-deployment create \\n --name don-iris-deployment \\n --  
endpoint-name don-iris-endpoint \\n --resource-group don-test-rg \\n --  
workspace-name don-ml-workspace \\n --file online-deployment.yml\\n  
az ml online-deployment get-logs \\n --name don-iris-deployment \\n --  
endpoint-name don-iris-endpoint \\n --resource-group don-test-rg \\n --  
workspace-name don-ml-workspace \\n --lines 100\\n  
az ml environment create \\n --file inference-environment.yml \\n --  
resource-group don-test-rg \\n --workspace-name don-ml-workspace\\n  
az ml environment create \\n --file inference-environment.yml \\n --  
resource-group don-test-rg \\n --workspace-name don-ml-workspace\\n  
az ml online-deployment create \\n --name don-iris-deployment \\n --  
endpoint-name don-iris-endpoint \\n --resource-group don-test-rg \\n --  
workspace-name don-ml-workspace \\n --file online-deployment.yml\\n  
az ml online-deployment update \\n --name don-iris-deployment \\n --  
endpoint-name don-iris-endpoint \\n --resource-group don-test-rg \\n --  
workspace-name don-ml-workspace \\n --file online-deployment.yml\\n  
az ml online-deployment get-logs \\n --name don-iris-deployment \\n --  
endpoint-name don-iris-endpoint \\n --resource-group don-test-rg \\n --  
workspace-name don-ml-workspace \\n --lines 100  
az ml online-deployment update \\n --name don-iris-deployment \\n --  
endpoint-name don-iris-endpoint \\n --resource-group don-test-rg \\n --  
workspace-name don-ml-workspace \\n --file online-deployment.yml\\n  
az ml online-deployment get-logs \\n --name don-iris-deployment \\n --  
endpoint-name don-iris-endpoint \\n --resource-group don-test-rg \\n --  
workspace-name don-ml-workspace \\n --lines 100\\n  
az ml online-deployment update \\n --name don-iris-deployment \\n --  
endpoint-name don-iris-endpoint \\n --resource-group don-test-rg \\n --  
workspace-name don-ml-workspace \\n --file online-deployment.yml\\n  
az ml online-deployment get-logs \\n --name don-iris-deployment \\n --  
endpoint-name don-iris-endpoint \\n --resource-group don-test-rg \\n --  
workspace-name don-ml-workspace \\n --lines 100\\n  
az ml online-endpoint invoke \\n --name don-iris-endpoint \\n --resource-  
group don-test-rg \\n --workspace-name don-ml-workspace \\n --request-body  
'{\\n  "data": [\\n    [5.1, 3.5, 1.4, 0.2]\\n  ]\\n }'\\n  
az ml online-endpoint invoke \\n --name don-iris-endpoint \\n --resource-  
group don-test-rg \\n --workspace-name don-ml-workspace \\n --request-file
```

```
request.json\n
az ml online-endpoint update \\n --name don-iris-endpoint \\n --resource-
group don-test-rg \\n --workspace-name don-ml-workspace \\n --traffic
"don-iris-deployment=100"\n
az ml online-endpoint invoke \\n --name don-iris-endpoint \\n --resource-
group don-test-rg \\n --workspace-name don-ml-workspace \\n --request-file
request.json\n
az ml online-endpoint update \\n --name don-iris-endpoint \\n --resource-
group don-test-rg \\n --workspace-name don-ml-workspace \\n --traffic
"don-iris-deployment=100"\n
az ml online-endpoint invoke \\n --name don-iris-endpoint \\n --resource-
group don-test-rg \\n --workspace-name don-ml-workspace \\n --request-file
request.json\n
az ml online-deployment update \\n --name don-iris-deployment \\n --
endpoint-name don-iris-endpoint \\n --resource-group don-test-rg \\n --
workspace-name don-ml-workspace \\n --file online-deployment.yml\n
az ml online-endpoint invoke \\n --name don-iris-endpoint \\n --resource-
group don-test-rg \\n --workspace-name don-ml-workspace \\n --request-file
request.json\n
az ml online-endpoint update \\n --name don-iris-endpoint \\n --resource-
group don-test-rg \\n --workspace-name don-ml-workspace \\n --set
app_insights_enabled=true\n
az ml online-deployment update \\n --name don-iris-deployment \\n --
endpoint-name don-iris-endpoint \\n --resource-group don-test-rg \\n --
workspace-name don-ml-workspace \\n --file online-deployment.yml\n
az ml online-endpoint show \\n --name don-iris-endpoint \\n --resource-
group don-test-rg \\n --workspace-name don-ml-workspace \\n --query
properties.appInsightsEnabled\n
az ml online-deployment show \\n --name don-iris-deployment \\n --
endpoint-name don-iris-endpoint \\n --resource-group don-test-rg \\n --
workspace-name don-ml-workspace \\n --query properties.appInsightsEnabled\n
az ml online-deployment update \\n --name don-iris-deployment \\n --
endpoint-name don-iris-endpoint \\n --resource-group don-test-rg \\n --
workspace-name don-ml-workspace \\n --file online-deployment.yml\n
az ml online-deployment show \\n --name don-iris-deployment \\n --
endpoint-name don-iris-endpoint \\n --resource-group don-test-rg \\n --
workspace-name don-ml-workspace \\n --query properties.appInsightsEnabled\n
az ml online-endpoint invoke \\n --name don-iris-endpoint \\n --resource-
group don-test-rg \\n --workspace-name don-ml-workspace \\n --request-file
request.json\n
az ml online-deployment update \\n --name don-iris-deployment \\n --
endpoint-name don-iris-endpoint \\n --resource-group don-test-rg \\n --
workspace-name don-ml-workspace \\n --file online-deployment.yml\n
az ml online-deployment update \\n --name don-iris-deployment \\n --
endpoint-name don-iris-endpoint \\n --resource-group don-test-rg \\n --
workspace-name don-ml-workspace \\n --file online-deployment.yml\n
az ml online-deployment update \\n --name don-iris-deployment \\n --
endpoint-name don-iris-endpoint \\n --resource-group don-test-rg \\n --
workspace-name don-ml-workspace \\n --file online-deployment.yml\n
az ml online-endpoint invoke \\n --name don-iris-endpoint \\n --resource-
group don-test-rg \\n --workspace-name don-ml-workspace \\n --request-file
```

```
request.json\n
az ml online-deployment update \\n --name don-iris-deployment \\n --
endpoint-name don-iris-endpoint \\n --resource-group don-test-rg \\n --
workspace-name don-ml-workspace \\n --file online-deployment.yml\n
az ml online-deployment update \\n --name don-iris-deployment \\n --
endpoint-name don-iris-endpoint \\n --resource-group don-test-rg \\n --
workspace-name don-ml-workspace \\n --file online-deployment.yml\n
az ml online-deployment get-logs \\n --name don-iris-deployment \\n --
endpoint-name don-iris-endpoint \\n --resource-group don-test-rg \\n --
workspace-name don-ml-workspace \\n --lines 100\n
az ml environment create \\n --file environment.yml \\n --resource-group
don-test-rg \\n --workspace-name don-ml-workspace\n
az ml environment create \\n --name inference-env \\n --version 3 \\n --
image mcr.microsoft.com/azureml/openmpi4.1.0-ubuntu20.04:20231010.v1 \\n --
conda-file environment.yml \\n --resource-group don-test-rg \\n --
workspace-name don-ml-workspace\n
az ml online-deployment update \\n --name don-iris-deployment \\n --
endpoint-name don-iris-endpoint \\n --resource-group don-test-rg \\n --
workspace-name don-ml-workspace \\n --file online-deployment.yml\n
az ml online-endpoint invoke \\n --name don-iris-endpoint \\n --resource-
group don-test-rg \\n --workspace-name don-ml-workspace \\n --request-file
request.json\n
az ml online-endpoint show \\n --name don-iris-endpoint \\n --resource-
group don-test-rg \\n --workspace-name don-ml-workspace \\n --query
traffic\n
az ml online-deployment get-logs \\n --name don-iris-deployment \\n --
endpoint-name don-iris-endpoint \\n --resource-group don-test-rg \\n --
workspace-name don-ml-workspace \\n --lines 100\n
az ml environment create \\n --file environment.yml \\n --resource-group
don-test-rg \\n --workspace-name don-ml-workspace\n
az ml environment create \\n --name iris-env \\n --resource-group don-
test-rg \\n --workspace-name don-ml-workspace \\n --conda-file
environment.yml \\n --image mcr.microsoft.com/azureml/openmpi4.1.0-
ubuntu20.04:latest\n
az ml environment list \\n --name iris-env \\n --resource-group don-test-
rg \\n --workspace-name don-ml-workspace \\n --query "[].version"\n
az ml environment create \\n --name inference-env \\n --version 3 \\n --
image mcr.microsoft.com/azureml/openmpi4.1.0-ubuntu20.04:20231010.v1 \\n --
conda-file environment.yml \\n --resource-group don-test-rg \\n --
workspace-name don-ml-workspace\n
az ml environment create \\n --name inference-env \\n --version 4 \\n --
image mcr.microsoft.com/azureml/openmpi4.1.0-ubuntu20.04:20231010.v1 \\n --
conda-file environment.yml \\n --resource-group don-test-rg \\n --
workspace-name don-ml-workspace\n
az ml online-deployment update \\n --name don-iris-deployment \\n --
endpoint-name don-iris-endpoint \\n --resource-group don-test-rg \\n --
workspace-name don-ml-workspace \\n --file online-deployment.yml\n
az ml online-endpoint invoke \\n --name don-iris-endpoint \\n --resource-
group don-test-rg \\n --workspace-name don-ml-workspace \\n --request-file
request.json\n
az ml environment create \\n --name inference-env \\n --version 4 \\n --
```

```
resource-group don-test-rg \\n --workspace-name don-ml-workspace \\n --  
file environment.yml\\n  
az ml environment create \\n --name inference-env \\n --version 5 \\n --  
resource-group don-test-rg \\n --workspace-name don-ml-workspace \\n --  
file environment.yml\\n  
az ml environment create \\n --name inference-env \\n --version 5 \\n --  
image mcr.microsoft.com/azureml/openmpi4.1.0-ubuntu20.04:20231010.v1 \\n --  
conda-file environment.yml \\n --resource-group don-test-rg \\n --  
workspace-name don-ml-workspace\\n  
az ml online-deployment update \\n --name don-iris-deployment \\n --  
endpoint-name don-iris-endpoint \\n --resource-group don-test-rg \\n --  
workspace-name don-ml-workspace \\n --file online-deployment.yml\\n  
az ml online-endpoint invoke \\n --name don-iris-endpoint \\n --resource-  
group don-test-rg \\n --workspace-name don-ml-workspace \\n --request-file  
request.json\\n  
az ml online-deployment get-logs \\n --name don-iris-deployment \\n --  
endpoint-name don-iris-endpoint \\n --resource-group don-test-rg \\n --  
workspace-name don-ml-workspace \\n --lines 100\\n  
az ml online-endpoint invoke \\n --name don-iris-endpoint \\n --resource-  
group don-test-rg \\n --workspace-name don-ml-workspace \\n --request-file  
request.json\\n  
az ml online-deployment get-logs \\n --name don-iris-deployment \\n --  
endpoint-name don-iris-endpoint \\n --resource-group don-test-rg \\n --  
workspace-name don-ml-workspace \\n --lines 100\\n  
az ml online-deployment delete \\n --name don-iris-deployment \\n --  
endpoint-name don-iris-endpoint \\n --resource-group don-test-rg \\n --  
workspace-name don-ml-workspace  
az ml online-endpoint update \\n --name don-iris-endpoint \\n --traffic  
"don-iris-deployment=0" \\n --resource-group don-test-rg \\n --workspace-  
name don-ml-workspace\\n  
az ml online-deployment delete \\n --name don-iris-deployment \\n --  
endpoint-name don-iris-endpoint \\n --resource-group don-test-rg \\n --  
workspace-name don-ml-workspace\\n  
az ml online-deployment create \\n --name don-iris-deployment \\n --  
endpoint-name don-iris-endpoint \\n --resource-group don-test-rg \\n --  
workspace-name don-ml-workspace \\n --file online-deployment.yml\\n  
az ml environment create \\n --resource-group don-test-rg \\n --workspace-  
name don-ml-workspace \\n --file inference-env.yml\\n  
az ml environment create \\n --resource-group don-test-rg \\n --workspace-  
name don-ml-workspace \\n --file inference-env.yml\\n  
az ml environment create \\n --resource-group don-test-rg \\n --workspace-  
name don-ml-workspace \\n --file inference-env.yml\\n  
az ml environment create \\n --resource-group don-test-rg \\n --workspace-  
name don-ml-workspace \\n --file inference-env.yml\\n  
az ml environment create \\n --resource-group don-test-rg \\n --workspace-  
name don-ml-workspace \\n --file inference-env.yml\\n  
az ml environment list --workspace-name don-ml-workspace --resource-group  
don-test-rg --query "[?name=='inference-env']"\\n  
az ml online-deployment create \\n --name don-iris-deployment \\n --  
endpoint-name don-iris-endpoint \\n --resource-group don-test-rg \\n --  
workspace-name don-ml-workspace \\n --file online-deployment.yml\\n
```

```
az ml online-deployment update \\n --name don-iris-deployment \\n --  
endpoint-name don-iris-endpoint \\n --resource-group don-test-rg \\n --  
workspace-name don-ml-workspace \\n --file online-deployment.yml\\n  
az ml online-deployment create \\n --name don-iris-deployment \\n --  
endpoint-name don-iris-endpoint \\n --resource-group don-test-rg \\n --  
workspace-name don-ml-workspace \\n --file online-deployment.yml\\n  
az ml online-endpoint update \\n --name don-iris-endpoint \\n --resource-  
group don-test-rg \\n --workspace-name don-ml-workspace \\n --traffic  
"don-iris-deployment=0"\\n  
az ml online-deployment delete \\n --name don-iris-deployment \\n --  
endpoint-name don-iris-endpoint \\n --resource-group don-test-rg \\n --  
workspace-name don-ml-workspace\\n  
az ml online-deployment create \\n --name don-iris-deployment \\n --  
endpoint-name don-iris-endpoint \\n --resource-group don-test-rg \\n --  
workspace-name don-ml-workspace \\n --file online-deployment.yml\\n  
az ml online-deployment get-logs \\n --name don-iris-deployment \\n --  
endpoint-name don-iris-endpoint \\n --resource-group don-test-rg \\n --  
workspace-name don-ml-workspace \\n --lines 100\\n  
az ml online-deployment create \\n --name don-iris-deployment \\n --  
endpoint-name don-iris-endpoint \\n --resource-group don-test-rg \\n --  
workspace-name don-ml-workspace \\n --file online-deployment.yml\\n  
az ml online-deployment update \\n --name don-iris-deployment \\n --  
endpoint-name don-iris-endpoint \\n --resource-group don-test-rg \\n --  
workspace-name don-ml-workspace \\n --file online-deployment.yml\\n  
az ml online-endpoint update \\n --name don-iris-endpoint \\n --resource-  
group don-test-rg \\n --workspace-name don-ml-workspace \\n --traffic  
"don-iris-deployment=0"\\n  
az ml online-deployment delete \\n --name don-iris-deployment \\n --  
endpoint-name don-iris-endpoint \\n --resource-group don-test-rg \\n --  
workspace-name don-ml-workspace\\n  
az ml online-deployment create \\n --name don-iris-deployment \\n --  
endpoint-name don-iris-endpoint \\n --resource-group don-test-rg \\n --  
workspace-name don-ml-workspace \\n --file online-deployment.yml\\n  
az ml environment create --file inference-env.yml \\n --resource-group don-  
test-rg \\n --workspace-name don-ml-workspace\\n  
az ml environment create --file inference-env.yml \\n --resource-group don-  
test-rg \\n --workspace-name don-ml-workspace\\n  
az ml online-deployment delete \\n --name don-iris-deployment \\n --  
endpoint-name don-iris-endpoint \\n --resource-group don-test-rg \\n --  
workspace-name don-ml-workspace\\n\\naz ml online-deployment create \\n --  
name don-iris-deployment \\n --endpoint-name don-iris-endpoint \\n --  
resource-group don-test-rg \\n --workspace-name don-ml-workspace \\n --  
file online-deployment.yml\\n  
az ml environment create \\n --file inference-env.yml \\n --resource-group  
don-test-rg \\n --workspace-name don-ml-workspace\\n  
az ml environment create \\n --file inference-env.yml \\n --resource-group  
don-test-rg \\n --workspace-name don-ml-workspace\\n  
az ml online-deployment delete \\n --name don-iris-deployment \\n --  
endpoint-name don-iris-endpoint \\n --resource-group don-test-rg \\n --  
workspace-name don-ml-workspace\\n  
az ml online-deployment create \\n --name don-iris-deployment \\n --
```

```
endpoint-name don-iris-endpoint \\n --resource-group don-test-rg \\n --  
workspace-name don-ml-workspace \\n --file online-deployment.yml\\n  
az ml environment create \\n --file inference-env.yml \\n --resource-group  
don-test-rg \\n --workspace-name don-ml-workspace\\n  
az ml online-deployment delete \\n --name don-iris-deployment \\n --  
endpoint-name don-iris-endpoint \\n --resource-group don-test-rg \\n --  
workspace-name don-ml-workspace\\n  
az ml online-deployment create \\n --name don-iris-deployment \\n --  
endpoint-name don-iris-endpoint \\n --resource-group don-test-rg \\n --  
workspace-name don-ml-workspace \\n --file online-deployment.yml\\n  
az ml online-deployment create \\n --name don-iris-deployment \\n --  
endpoint-name don-iris-endpoint \\n --resource-group don-test-rg \\n --  
workspace-name don-ml-workspace \\n --file online-deployment.yml\\n  
az ml online-deployment update \\n --name don-iris-deployment \\n --  
endpoint-name don-iris-endpoint \\n --resource-group don-test-rg \\n --  
workspace-name don-ml-workspace \\n --file online-deployment.yml\\n  
az ml online-deployment delete \\n --name don-iris-deployment \\n --  
endpoint-name don-iris-endpoint \\n --resource-group don-test-rg \\n --  
workspace-name don-ml-workspace\\n  
az ml online-deployment create \\n --name don-iris-deployment \\n --  
endpoint-name don-iris-endpoint \\n --resource-group don-test-rg \\n --  
workspace-name don-ml-workspace \\n --file online-deployment.yml\\n  
az ml online-endpoint invoke \\n --name don-iris-endpoint \\n --resource-  
group don-test-rg \\n --workspace-name don-ml-workspace \\n --request-file  
request.json\\n  
az ml online-endpoint update \\n --name don-iris-endpoint \\n --resource-  
group don-test-rg \\n --workspace-name don-ml-workspace \\n --traffic  
"don-iris-deployment=100"\\n  
az ml online-endpoint invoke \\n --name don-iris-endpoint \\n --resource-  
group don-test-rg \\n --workspace-name don-ml-workspace \\n --request-file  
request.json\\n  
az monitor app-insights component show \\n --app <your-app-insights-name>  
\\n --resource-group <your-rg> \\n --query "workspaceResourceId" \\n --  
output tsv\\n  
az monitor app-insights component show \\n --app donappinsights \\n --  
resource-group don-test-rg \\n --query "workspaceResourceId" \\n --output  
tsv\\n  
az monitor log-analytics query --workspace donloganalytics --analytics-query  
"customEvents\\n| where name == 'Request'\\n| extend prediction =  
tostring(customDimensions.Response)\\n| where prediction in (['0'],  
['1'])\\n| summarize count() by prediction, bin(timestamp, 5m)\\n| order by  
timestamp desc" --timespan PT1H  
^[200~az monitor log-analytics query \\n --workspace donloganalytics \\n  
--analytics-query "customEvents\\n| where name == 'Request'\\n| extend  
prediction = tostring(customDimensions.Response)\\n| where prediction in  
(['0'], ['1'])\\n| summarize count() by prediction, bin(timestamp,  
5m)\\n| order by timestamp desc" \\n --timespan PT1H  
az monitor log-analytics query \\n --workspace donloganalytics \\n --  
analytics-query "customEvents\\n| where name == 'Request'\\n| extend  
prediction = tostring(customDimensions.Response)\\n| where prediction in  
(['0'], ['1'])\\n| summarize count() by prediction, bin(timestamp,
```



```

workspace"\nAPP_ID="a3b5f924-df36-4138-bd7e-f9d85544af10"\n\n# Get the
object ID of the service principal\nSP_OBJECT_ID=$(az ad sp show --id
$APP_ID --query objectId --output tsv)\n\n# Assign Contributor role at the
resource group level\naz role assignment create \\\n --assignee-object-id
"$SP_OBJECT_ID" \\\n --assignee-principal-type ServicePrincipal \\\n --role
"Contributor" \\\n --scope
"/subscriptions/$SUBSCRIPTION_ID/resourceGroups/$RESOURCE_GROUP"\n\n# Assign
AzureML Workspace User role at the workspace level\naz role assignment
create \\\n --assignee-object-id "$SP_OBJECT_ID" \\\n --assignee-principal-
type ServicePrincipal \\\n --role "AzureML Workspace User" \\\n --scope
"/subscriptions/$SUBSCRIPTION_ID/resourceGroups/$RESOURCE_GROUP/providers/Mi
crosoft.MachineLearningServices/workspaces/$WORKSPACE_NAME"\n
az ad sp show --id $APP_ID --query objectId --output tsv
az account show --query "{tenantId:tenantId, subscriptionId:id,
user:user.name}"\n
az ad sp list --display-name github-ml-cicd-sp
az role assignment create \\\n --assignee-object-id 9b6ceeaf-f81b-45c4-
ac18-399b6712d148 \\\n --assignee-principal-type ServicePrincipal \\\n --
role "Contributor" \\\n --scope /subscriptions/baa29726-b3e6-4910-bb9b-
b585c655322c/resourceGroups/don-test-rg\n
az ad app credential reset \\\n --id a3b5f924-df36-4138-bd7e-f9d85544af10
\\\n --display-name "GitHub ML CICD Key" \\\n --append \\\n --query
"password" \\\n --output tsv
az keyvault secret set \\\n --vault-name donkv \\\n --name "AZURE-CLIENT-
SECRET" \\\n --value "-FF8Q~XqBZGpNN_0Y_MPHbWVHTM5rqGzJvZXTbai"
az keyvault secret set \\\n --vault-name donkv \\\n --name "AZURE-CLIENT-
SECRET" \\\n --value '-FF8Q~XqBZGpNN_0Y_MPHbWVHTM5rqGzJvZXTbai'
az keyvault secret set \\\n --vault-name donkv \\\n --name "AZURE-CLIENT-
SECRET" \\\n --value "$(cat secret.txt)"
az keyvault secret set --vault-name donkv --name "AZURE-CLIENT-ID" --value
"a3b5f924-df36-4138-bd7e-f9d85544af10"\naz keyvault secret set --vault-name
donkv --name "AZURE-TENANT-ID" --value "6ad27142-5e7d-4841-9e72-
c3aaca00f9e6"
az keyvault secret set --vault-name donkv --name "AZURE-TENANT-ID" --value
"6ad27142-5e7d-4841-9e72-c3aaca00f9e6"
az keyvault set-policy \\\n --name donkv \\\n --object-id
02362d53-0e10-47f5-8066-4d56431dc9bd \\\n --secret-permissions get list set
delete\n
az keyvault secret set --vault-name donkv --name "AZURE-TENANT-ID" --value
"6ad27142-5e7d-4841-9e72-c3aaca00f9e6"
az keyvault secret set --vault-name donkv --name "AZURE-SUBSCRIPTION-ID" --
value "baa29726-b3e6-4910-bb9b-b585c655322c"\naz keyvault secret set --
vault-name donkv --name "AZURE-RESOURCE-GROUP" --value "don-test-rg"\naz
keyvault secret set --vault-name donkv --name "AZURE-WORKSPACE-NAME" --value
"don-ml-workspace"
az keyvault set-policy --name donkv \\\n --spn a3b5f924-df36-4138-bd7e-
f9d85544af10 \\\n --secret-permissions get list
az keyvault secret show --vault-name donkv --name AZURE-TENANT-ID --query
value\n
for i in AZURE-CLIENT-ID AZURE-CLIENT-SECRET AZURE-SUBSCRIPTION-ID AZURE-
RESOURCE-GROUP AZURE-WORKSPACE-NAME AZURE-TENANT-ID;do\nnecho $i;az keyvault

```

```
secret show --vault-name donkv --name $i --query value\ndone
az keyvault secret set \\\n --vault-name donkv \\\n --name "AZURE-CLIENT-ID"
\\\n --value "a3b5f924-df36-4138-bd7e-f9d85544af10"
for i in AZURE-CLIENT-ID AZURE-CLIENT-SECRET AZURE-SUBSCRIPTION-ID AZURE-
RESOURCE-GROUP AZURE-WORKSPACE-NAME AZURE-TENANT-ID;do\necho $i;az keyvault
secret show --vault-name donkv --name $i --query value\ndone
az keyvault set-policy \\\n --name donkv \\\n --spn a3b5f924-df36-4138-bd7e-
f9d85544af10 \\\n --secret-permissions get list\ndone
for i in AZURE-CLIENT-ID AZURE-CLIENT-SECRET AZURE-SUBSCRIPTION-ID AZURE-
RESOURCE-GROUP AZURE-WORKSPACE-NAME AZURE-TENANT-ID;do\necho $i;az keyvault
secret show --vault-name donkv --name $i --query value\ndone
git commit -m "I'm going crazy. Help me"
for i in AZURE-CLIENT-ID AZURE-CLIENT-SECRET AZURE-SUBSCRIPTION-ID AZURE-
RESOURCE-GROUP AZURE-WORKSPACE-NAME AZURE-TENANT-ID;do\necho $i;az keyvault
secret show --vault-name donkv --name $i --query value\ndone
az ml environment create --file environment.yml \\\n --name inference-env
\\\n --resource-group don-test-rg \\\n --workspace-name don-ml-workspace-
fixed\ndone
az ml environment create --file environment.yml \\\n --name inference-env
\\\n --resource-group don-test-rg \\\n --workspace-name don-ml-workspace-
fixed\ndone
az ml environment create \\\n --name inference-env \\\n --image
mcr.microsoft.com/azureml/openmpi4.1.0-ubuntu20.04 \\\n --conda-file
environment.yml \\\n --resource-group don-test-rg \\\n --workspace-name don-
ml-workspace-fixed\ndone
az ml environment create \\\n --name inference-env \\\n --version 1 \\\n --
image mcr.microsoft.com/azureml/openmpi4.1.0-ubuntu20.04 \\\n --conda-file
environment.yml \\\n --resource-group don-test-rg \\\n --workspace-name don-
ml-workspace-fixed\ndone
az ml environment create \\\n --file environment.yml \\\n --resource-group
don-test-rg \\\n --workspace-name don-ml-workspace-fixed\ndone
az ml environment create \\\n --file environment.yml \\\n --resource-group
don-test-rg \\\n --workspace-name don-ml-workspace-fixed\ndone
az version
az extension update -name ml
az ml environment create \\\n --file environment.yml \\\n --resource-group
don-test-rg \\\n --workspace-name don-ml-workspace-fixed\ndone
az extension update --name ml
az ml environment create \\\n --file inference-env.yml \\\n --resource-group
don-test-rg \\\n --workspace-name don-ml-workspace-fixed \\\n --debug\ndone
az ml environment create \\\n --file inference-env.yml \\\n --resource-group
don-test-rg \\\n --workspace-name don-ml-workspace \\\n --debug\ndone
az ml model list \\\n --name iris-logreg \\\n --resource-group don-test-rg
\\\n --workspace-name don-ml-workspace\ndone
az ml online-endpoint create \\\n --file online-endpoint.yml \\\n --
resource-group don-test-rg \\\n --workspace-name don-ml-workspace\ndone
az ml online-deployment create \\\n --file online-deployment.yml \\\n --resource-
group don-test-rg \\\n --workspace-name don-ml-workspace \\\n --all-
traffic\ndone
az ml online-endpoint create \\\n --file online-endpoint.yml \\\n --
resource-group don-test-rg \\\n --workspace-name don-ml-workspace\ndone
az ml
```

```
online-deployment create \\n --file online-deployment.yml \\n --resource-
group don-test-rg \\n --workspace-name don-ml-workspace \\n --all-
traffic\\n
az ml online-endpoint invoke \\n --name don-iris-endpoint \\n --resource-
group don-test-rg \\n --workspace-name don-ml-workspace \\n --request-body
'{"data": [5.1, 3.5, 1.4, 0.2]}'\\n
az ml online-endpoint invoke \\n --name don-iris-endpoint \\n --resource-
group don-test-rg \\n --workspace-name don-ml-workspace \\n --request-body
'{"data": [[5.1, 3.5, 1.4, 0.2]]}' \\n --request-content-type
application/json\\n
az ml online-endpoint invoke \\n --name don-iris-endpoint \\n --resource-
group don-test-rg \\n --workspace-name don-ml-workspace \\n --request-file
request.json\\n
az ml online-deployment get-logs \\n --name blue \\n --endpoint-name don-
iris-endpoint \\n --resource-group don-test-rg \\n --workspace-name don-
ml-workspace \\n --lines 100\\n
az ml online-deployment get-logs \\n --name don-iris-deployment \\n --
endpoint-name don-iris-endpoint \\n --resource-group don-test-rg \\n --
workspace-name don-ml-workspace \\n --lines 100\\n
az ml online-deployment list \\n --endpoint-name don-iris-endpoint \\n --
resource-group don-test-rg \\n --workspace-name don-ml-workspace \\n --
output table\\n
az ml model list \\n --name iris-logreg \\n --resource-group don-test-rg
\\n --workspace-name don-ml-workspace \\n --output table\\n
az ml online-deployment create \\n --name don-iris-deployment \\n --
endpoint-name don-iris-endpoint \\n --resource-group don-test-rg \\n --
workspace-name don-ml-workspace \\n --file online-deployment.yml\\n
az ml online-deployment create \\n --name don-iris-deployment \\n --
endpoint-name don-iris-endpoint \\n --resource-group don-test-rg \\n --
workspace-name don-ml-workspace \\n --file online-deployment.yml\\n
az ml online-deployment create \\n --name don-iris-deployment \\n --
endpoint-name don-iris-endpoint \\n --resource-group don-test-rg \\n --
workspace-name don-ml-workspace \\n --file online-deployment.yml\\n
az ml environment list \\n --resource-group don-test-rg \\n --workspace-
name don-ml-workspace \\n --output table\\n
az ml online-deployment create \\n --name don-iris-deployment \\n --
endpoint-name don-iris-endpoint \\n --resource-group don-test-rg \\n --
workspace-name don-ml-workspace \\n --file online-deployment.yml\\n
az ml environment create \\n --file inference-environment.yml \\n --
resource-group don-test-rg \\n --workspace-name don-ml-workspace\\n
az ml environment create \\n --file inference-environment.yml \\n --
resource-group don-test-rg \\n --workspace-name don-ml-workspace\\n
az ml online-deployment create \\n --name don-iris-deployment \\n --
endpoint-name don-iris-endpoint \\n --resource-group don-test-rg \\n --
workspace-name don-ml-workspace \\n --file online-deployment.yml\\n
az ml online-deployment create \\n --name don-iris-deployment \\n --
endpoint-name don-iris-endpoint \\n --resource-group don-test-rg \\n --
workspace-name don-ml-workspace \\n --file online-deployment.yml\\n
az ml online-deployment get-logs \\n --name don-iris-deployment \\n --
endpoint-name don-iris-endpoint \\n --resource-group don-test-rg \\n --
workspace-name don-ml-workspace \\n --lines 100\\n
```

```
az ml environment create \\n --file inference-environment.yml \\n --
resource-group don-test-rg \\n --workspace-name don-ml-workspace\\n
az ml environment create \\n --file inference-environment.yml \\n --
resource-group don-test-rg \\n --workspace-name don-ml-workspace\\n
az ml online-deployment create \\n --name don-iris-deployment \\n --
endpoint-name don-iris-endpoint \\n --resource-group don-test-rg \\n --
workspace-name don-ml-workspace \\n --file online-deployment.yml\\n
az ml online-deployment update \\n --name don-iris-deployment \\n --
endpoint-name don-iris-endpoint \\n --resource-group don-test-rg \\n --
workspace-name don-ml-workspace \\n --file online-deployment.yml\\n
az ml online-deployment get-logs \\n --name don-iris-deployment \\n --
endpoint-name don-iris-endpoint \\n --resource-group don-test-rg \\n --
workspace-name don-ml-workspace \\n --lines 100
az ml online-deployment update \\n --name don-iris-deployment \\n --
endpoint-name don-iris-endpoint \\n --resource-group don-test-rg \\n --
workspace-name don-ml-workspace \\n --file online-deployment.yml\\n
az ml online-deployment get-logs \\n --name don-iris-deployment \\n --
endpoint-name don-iris-endpoint \\n --resource-group don-test-rg \\n --
workspace-name don-ml-workspace \\n --lines 100\\n
az ml online-deployment update \\n --name don-iris-deployment \\n --
endpoint-name don-iris-endpoint \\n --resource-group don-test-rg \\n --
workspace-name don-ml-workspace \\n --file online-deployment.yml\\n
az ml online-deployment get-logs \\n --name don-iris-deployment \\n --
endpoint-name don-iris-endpoint \\n --resource-group don-test-rg \\n --
workspace-name don-ml-workspace \\n --lines 100\\n
az ml online-endpoint invoke \\n --name don-iris-endpoint \\n --resource-
group don-test-rg \\n --workspace-name don-ml-workspace \\n --request-body
'{\\n  "data": [\\n    [5.1, 3.5, 1.4, 0.2]\\n  ]\\n }'\\n
az ml online-endpoint invoke \\n --name don-iris-endpoint \\n --resource-
group don-test-rg \\n --workspace-name don-ml-workspace \\n --request-file
request.json\\n
az ml online-endpoint update \\n --name don-iris-endpoint \\n --resource-
group don-test-rg \\n --workspace-name don-ml-workspace \\n --traffic
"don-iris-deployment=100"\\n
az ml online-endpoint invoke \\n --name don-iris-endpoint \\n --resource-
group don-test-rg \\n --workspace-name don-ml-workspace \\n --request-file
request.json\\n
az ml online-endpoint update \\n --name don-iris-endpoint \\n --resource-
group don-test-rg \\n --workspace-name don-ml-workspace \\n --traffic
"don-iris-deployment=100"\\n
az ml online-endpoint invoke \\n --name don-iris-endpoint \\n --resource-
group don-test-rg \\n --workspace-name don-ml-workspace \\n --request-file
request.json\\n
az ml online-deployment update \\n --name don-iris-deployment \\n --
endpoint-name don-iris-endpoint \\n --resource-group don-test-rg \\n --
workspace-name don-ml-workspace \\n --file online-deployment.yml\\n
az ml online-endpoint invoke \\n --name don-iris-endpoint \\n --resource-
group don-test-rg \\n --workspace-name don-ml-workspace \\n --request-file
request.json\\n
az ml online-endpoint update \\n --name don-iris-endpoint \\n --resource-
group don-test-rg \\n --workspace-name don-ml-workspace \\n --set
```

```
app_insights_enabled=true\naz ml online-deployment update \\n --name don-iris-deployment \\n --\nendpoint-name don-iris-endpoint \\n --resource-group don-test-rg \\n --\nworkspace-name don-ml-workspace \\n --file online-deployment.yml\naz ml online-endpoint show \\n --name don-iris-endpoint \\n --resource-\ngroup don-test-rg \\n --workspace-name don-ml-workspace \\n --query\nproperties.appInsightsEnabled\naz ml online-deployment show \\n --name don-iris-deployment \\n --\nendpoint-name don-iris-endpoint \\n --resource-group don-test-rg \\n --\nworkspace-name don-ml-workspace \\n --query properties.appInsightsEnabled\naz ml online-deployment update \\n --name don-iris-deployment \\n --\nendpoint-name don-iris-endpoint \\n --resource-group don-test-rg \\n --\nworkspace-name don-ml-workspace \\n --file online-deployment.yml\naz ml online-deployment show \\n --name don-iris-deployment \\n --\nendpoint-name don-iris-endpoint \\n --resource-group don-test-rg \\n --\nworkspace-name don-ml-workspace \\n --query properties.appInsightsEnabled\naz ml online-endpoint invoke \\n --name don-iris-endpoint \\n --resource-\ngroup don-test-rg \\n --workspace-name don-ml-workspace \\n --request-file\nrequest.json\naz ml online-deployment update \\n --name don-iris-deployment \\n --\nendpoint-name don-iris-endpoint \\n --resource-group don-test-rg \\n --\nworkspace-name don-ml-workspace \\n --file online-deployment.yml\naz ml online-deployment update \\n --name don-iris-deployment \\n --\nendpoint-name don-iris-endpoint \\n --resource-group don-test-rg \\n --\nworkspace-name don-ml-workspace \\n --file online-deployment.yml\naz ml online-deployment update \\n --name don-iris-deployment \\n --\nendpoint-name don-iris-endpoint \\n --resource-group don-test-rg \\n --\nworkspace-name don-ml-workspace \\n --file online-deployment.yml\naz ml online-endpoint invoke \\n --name don-iris-endpoint \\n --resource-\ngroup don-test-rg \\n --workspace-name don-ml-workspace \\n --request-file\nrequest.json\naz ml online-deployment update \\n --name don-iris-deployment \\n --\nendpoint-name don-iris-endpoint \\n --resource-group don-test-rg \\n --\nworkspace-name don-ml-workspace \\n --file online-deployment.yml\naz ml online-deployment update \\n --name don-iris-deployment \\n --\nendpoint-name don-iris-endpoint \\n --resource-group don-test-rg \\n --\nworkspace-name don-ml-workspace \\n --file online-deployment.yml\naz ml online-deployment get-logs \\n --name don-iris-deployment \\n --\nendpoint-name don-iris-endpoint \\n --resource-group don-test-rg \\n --\nworkspace-name don-ml-workspace \\n --lines 100\naz ml environment create \\n --file environment.yml \\n --resource-group\ndon-test-rg \\n --workspace-name don-ml-workspace\naz ml environment create \\n --name inference-env \\n --version 3 \\n --\nimage mcr.microsoft.com/azureml/openmpi4.1.0-ubuntu20.04:20231010.v1 \\n --\nconda-file environment.yml \\n --resource-group don-test-rg \\n --\nworkspace-name don-ml-workspace\naz ml online-deployment update \\n --name don-iris-deployment \\n --\nendpoint-name don-iris-endpoint \\n --resource-group don-test-rg \\n --\nworkspace-name don-ml-workspace \\n --file online-deployment.yml\naz ml online-endpoint invoke \\n --name don-iris-endpoint \\n --resource-\ngroup don-test-rg \\n --workspace-name don-ml-workspace \\n --request-file
```

```
request.json\n
az ml online-endpoint show \\n --name don-iris-endpoint \\n --resource-
group don-test-rg \\n --workspace-name don-ml-workspace \\n --query
traffic\n
az ml online-deployment get-logs \\n --name don-iris-deployment \\n --
endpoint-name don-iris-endpoint \\n --resource-group don-test-rg \\n --
workspace-name don-ml-workspace \\n --lines 100\n
az ml environment create \\n --file environment.yml \\n --resource-group
don-test-rg \\n --workspace-name don-ml-workspace\n
az ml environment create \\n --name iris-env \\n --resource-group don-
test-rg \\n --workspace-name don-ml-workspace \\n --conda-file
environment.yml \\n --image mcr.microsoft.com/azureml/openmpi4.1.0-
ubuntu20.04:latest\n
az ml environment list \\n --name iris-env \\n --resource-group don-test-
rg \\n --workspace-name don-ml-workspace \\n --query "[].version"\n
az ml environment create \\n --name inference-env \\n --version 3 \\n --
image mcr.microsoft.com/azureml/openmpi4.1.0-ubuntu20.04:20231010.v1 \\n --
conda-file environment.yml \\n --resource-group don-test-rg \\n --
workspace-name don-ml-workspace\n
az ml environment create \\n --name inference-env \\n --version 4 \\n --
image mcr.microsoft.com/azureml/openmpi4.1.0-ubuntu20.04:20231010.v1 \\n --
conda-file environment.yml \\n --resource-group don-test-rg \\n --
workspace-name don-ml-workspace\n
az ml online-deployment update \\n --name don-iris-deployment \\n --
endpoint-name don-iris-endpoint \\n --resource-group don-test-rg \\n --
workspace-name don-ml-workspace \\n --file online-deployment.yml\n
az ml online-endpoint invoke \\n --name don-iris-endpoint \\n --resource-
group don-test-rg \\n --workspace-name don-ml-workspace \\n --request-file
request.json\n
az ml environment create \\n --name inference-env \\n --version 4 \\n --
resource-group don-test-rg \\n --workspace-name don-ml-workspace \\n --
file environment.yml\n
az ml environment create \\n --name inference-env \\n --version 5 \\n --
resource-group don-test-rg \\n --workspace-name don-ml-workspace \\n --
file environment.yml\n
az ml environment create \\n --name inference-env \\n --version 5 \\n --
image mcr.microsoft.com/azureml/openmpi4.1.0-ubuntu20.04:20231010.v1 \\n --
conda-file environment.yml \\n --resource-group don-test-rg \\n --
workspace-name don-ml-workspace\n
az ml online-deployment update \\n --name don-iris-deployment \\n --
endpoint-name don-iris-endpoint \\n --resource-group don-test-rg \\n --
workspace-name don-ml-workspace \\n --file online-deployment.yml\n
az ml online-endpoint invoke \\n --name don-iris-endpoint \\n --resource-
group don-test-rg \\n --workspace-name don-ml-workspace \\n --request-file
request.json\n
az ml online-deployment get-logs \\n --name don-iris-deployment \\n --
endpoint-name don-iris-endpoint \\n --resource-group don-test-rg \\n --
workspace-name don-ml-workspace \\n --lines 100\n
az ml online-endpoint invoke \\n --name don-iris-endpoint \\n --resource-
group don-test-rg \\n --workspace-name don-ml-workspace \\n --request-file
request.json\n
```

```
az ml online-deployment get-logs \\\n  --name don-iris-deployment \\\n  --endpoint-name don-iris-endpoint \\\n  --resource-group don-test-rg \\\n  --workspace-name don-ml-workspace \\\n  --lines 100\\\n  az ml online-deployment delete \\\n  --name don-iris-deployment \\\n  --endpoint-name don-iris-endpoint \\\n  --resource-group don-test-rg \\\n  --workspace-name don-ml-workspace\\\n  az ml online-endpoint update \\\n  --name don-iris-endpoint \\\n  --traffic "don-iris-deployment=0" \\\n  --resource-group don-test-rg \\\n  --workspace-name don-ml-workspace\\\n  az ml online-deployment delete \\\n  --name don-iris-deployment \\\n  --endpoint-name don-iris-endpoint \\\n  --resource-group don-test-rg \\\n  --workspace-name don-ml-workspace\\\n  az ml online-deployment create \\\n  --name don-iris-deployment \\\n  --endpoint-name don-iris-endpoint \\\n  --resource-group don-test-rg \\\n  --workspace-name don-ml-workspace \\\n  --file online-deployment.yml\\\n  az ml environment create \\\n  --resource-group don-test-rg \\\n  --workspace-name don-ml-workspace \\\n  --file inference-env.yml\\\n  az ml environment create \\\n  --resource-group don-test-rg \\\n  --workspace-name don-ml-workspace \\\n  --file inference-env.yml\\\n  az ml environment create \\\n  --resource-group don-test-rg \\\n  --workspace-name don-ml-workspace \\\n  --file inference-env.yml\\\n  az ml environment create \\\n  --resource-group don-test-rg \\\n  --workspace-name don-ml-workspace \\\n  --file inference-env.yml\\\n  az ml environment create \\\n  --resource-group don-test-rg \\\n  --workspace-name don-ml-workspace \\\n  --file inference-env.yml\\\n  az ml environment list --workspace-name don-ml-workspace --resource-group don-test-rg --query "[?name=='inference-env']"\\\n  az ml online-deployment create \\\n  --name don-iris-deployment \\\n  --endpoint-name don-iris-endpoint \\\n  --resource-group don-test-rg \\\n  --workspace-name don-ml-workspace \\\n  --file online-deployment.yml\\\n  az ml online-deployment update \\\n  --name don-iris-deployment \\\n  --endpoint-name don-iris-endpoint \\\n  --resource-group don-test-rg \\\n  --workspace-name don-ml-workspace \\\n  --file online-deployment.yml\\\n  az ml online-deployment create \\\n  --name don-iris-deployment \\\n  --endpoint-name don-iris-endpoint \\\n  --resource-group don-test-rg \\\n  --workspace-name don-ml-workspace \\\n  --file online-deployment.yml\\\n  az ml online-endpoint update \\\n  --name don-iris-endpoint \\\n  --resource-group don-test-rg \\\n  --workspace-name don-ml-workspace \\\n  --traffic "don-iris-deployment=0"\\\n  az ml online-deployment delete \\\n  --name don-iris-deployment \\\n  --endpoint-name don-iris-endpoint \\\n  --resource-group don-test-rg \\\n  --workspace-name don-ml-workspace\\\n  az ml online-deployment create \\\n  --name don-iris-deployment \\\n  --endpoint-name don-iris-endpoint \\\n  --resource-group don-test-rg \\\n  --workspace-name don-ml-workspace \\\n  --file online-deployment.yml\\\n  az ml online-deployment get-logs \\\n  --name don-iris-deployment \\\n  --endpoint-name don-iris-endpoint \\\n  --resource-group don-test-rg \\\n  --workspace-name don-ml-workspace \\\n  --lines 100\\\n  az ml online-deployment create \\\n  --name don-iris-deployment \\\n  --endpoint-name don-iris-endpoint \\\n  --resource-group don-test-rg \\\n  --workspace-name don-ml-workspace \\\n  --file online-deployment.yml\\\n
```

```
az ml online-deployment update \\n --name don-iris-deployment \\n --  
endpoint-name don-iris-endpoint \\n --resource-group don-test-rg \\n --  
workspace-name don-ml-workspace \\n --file online-deployment.yml\\n  
az ml online-endpoint update \\n --name don-iris-endpoint \\n --resource-  
group don-test-rg \\n --workspace-name don-ml-workspace \\n --traffic  
"don-iris-deployment=0"\\n  
az ml online-deployment delete \\n --name don-iris-deployment \\n --  
endpoint-name don-iris-endpoint \\n --resource-group don-test-rg \\n --  
workspace-name don-ml-workspace\\n  
az ml online-deployment create \\n --name don-iris-deployment \\n --  
endpoint-name don-iris-endpoint \\n --resource-group don-test-rg \\n --  
workspace-name don-ml-workspace \\n --file online-deployment.yml\\n  
az ml environment create --file inference-env.yml \\n --resource-group don-  
test-rg \\n --workspace-name don-ml-workspace\\n  
az ml environment create --file inference-env.yml \\n --resource-group don-  
test-rg \\n --workspace-name don-ml-workspace\\n  
az ml online-deployment delete \\n --name don-iris-deployment \\n --  
endpoint-name don-iris-endpoint \\n --resource-group don-test-rg \\n --  
workspace-name don-ml-workspace\\n\\naz ml online-deployment create \\n --  
name don-iris-deployment \\n --endpoint-name don-iris-endpoint \\n --  
resource-group don-test-rg \\n --workspace-name don-ml-workspace \\n --  
file online-deployment.yml\\n  
az ml environment create \\n --file inference-env.yml \\n --resource-group  
don-test-rg \\n --workspace-name don-ml-workspace\\n  
az ml environment create \\n --file inference-env.yml \\n --resource-group  
don-test-rg \\n --workspace-name don-ml-workspace\\n  
az ml online-deployment delete \\n --name don-iris-deployment \\n --  
endpoint-name don-iris-endpoint \\n --resource-group don-test-rg \\n --  
workspace-name don-ml-workspace\\n  
az ml online-deployment create \\n --name don-iris-deployment \\n --  
endpoint-name don-iris-endpoint \\n --resource-group don-test-rg \\n --  
workspace-name don-ml-workspace \\n --file online-deployment.yml\\n  
az ml environment create \\n --file inference-env.yml \\n --resource-group  
don-test-rg \\n --workspace-name don-ml-workspace\\n  
az ml online-deployment delete \\n --name don-iris-deployment \\n --  
endpoint-name don-iris-endpoint \\n --resource-group don-test-rg \\n --  
workspace-name don-ml-workspace\\n  
az ml online-deployment create \\n --name don-iris-deployment \\n --  
endpoint-name don-iris-endpoint \\n --resource-group don-test-rg \\n --  
workspace-name don-ml-workspace \\n --file online-deployment.yml\\n  
az ml online-deployment create \\n --name don-iris-deployment \\n --  
endpoint-name don-iris-endpoint \\n --resource-group don-test-rg \\n --  
workspace-name don-ml-workspace \\n --file online-deployment.yml\\n  
az ml online-deployment update \\n --name don-iris-deployment \\n --  
endpoint-name don-iris-endpoint \\n --resource-group don-test-rg \\n --  
workspace-name don-ml-workspace \\n --file online-deployment.yml\\n  
az ml online-deployment delete \\n --name don-iris-deployment \\n --  
endpoint-name don-iris-endpoint \\n --resource-group don-test-rg \\n --  
workspace-name don-ml-workspace\\n  
az ml online-deployment create \\n --name don-iris-deployment \\n --  
endpoint-name don-iris-endpoint \\n --resource-group don-test-rg \\n --
```

```
workspace-name don-ml-workspace \\n --file online-deployment.yml\\n
az ml online-endpoint invoke \\n --name don-iris-endpoint \\n --resource-
group don-test-rg \\n --workspace-name don-ml-workspace \\n --request-file
request.json\\n
az ml online-endpoint update \\n --name don-iris-endpoint \\n --resource-
group don-test-rg \\n --workspace-name don-ml-workspace \\n --traffic
"don-iris-deployment=100"\\n
az ml online-endpoint invoke \\n --name don-iris-endpoint \\n --resource-
group don-test-rg \\n --workspace-name don-ml-workspace \\n --request-file
request.json\\n
az monitor app-insights component show \\n --app <your-app-insights-name>
\\n --resource-group <your-rg> \\n --query "workspaceResourceId" \\n --
output tsv\\n
az monitor app-insights component show \\n --app donappinsights \\n --
resource-group don-test-rg \\n --query "workspaceResourceId" \\n --output
tsv\\n
az monitor log-analytics query --workspace donloganalytics --analytics-query
"customEvents\\n| where name == 'Request'\\n| extend prediction =
tostring(customDimensions.Response)\\n| where prediction in ('['0']',
['1'])\\n| summarize count() by prediction, bin(timestamp, 5m)\\n| order by
timestamp desc" --timespan PT1H
^[[200~az monitor log-analytics query \\n --workspace donloganalytics \\n
--analytics-query "customEvents\\n| where name == 'Request'\\n| extend
prediction = tostring(customDimensions.Response)\\n| where prediction in
(['\"0\"'], ['\"1\"]')\\n| summarize count() by prediction, bin(timestamp,
5m)\\n| order by timestamp desc" \\n --timespan PT1H
az monitor log-analytics query \\n --workspace donloganalytics \\n --
analytics-query "customEvents\\n| where name == 'Request'\\n| extend
prediction = tostring(customDimensions.Response)\\n| where prediction in
(['\"0\"'], ['\"1\"]')\\n| summarize count() by prediction, bin(timestamp,
5m)\\n| order by timestamp desc" \\n --timespan PT1H\\n
az monitor log-analytics workspace show \\n --resource-group don-test-rg
\\n --workspace-name donloganalytics \\n --query customerId \\n --output
tsv\\n
az monitor log-analytics query \\n --workspace 1b7c0d2b-4628-483a-
bf21-507f0c45df33 \\n --analytics-query "customEvents\\n| where name ==
'Request'\\n| extend prediction = tostring(customDimensions.Response)\\n|
where prediction in ('\"0\"'], ['\"1\"]')\\n| summarize count() by
prediction, bin(timestamp, 5m)\\n| order by timestamp desc" \\n --timespan
PT1H\\n
az monitor log-analytics query \\n --workspace 1b7c0d2b-4628-483a-
bf21-507f0c45df33 \\n --analytics-query ".show tables" \\n --timespan PT1H
```

## [AI Knowledge](#)