**Lab Exercise-0.1 Create Bucket in Minio using Metaflow**

Here’s a simple lab exercise using Metaflow to create a bucket in MinIO. This example assumes you have a MinIO server running and accessible. The flow will check if a bucket exists and create it if it doesn't.

**Metaflow Script to Create a Bucket in MinIO**

from metaflow import FlowSpec, step

from minio import Minio

from minio.error import S3Error

class CreateBucketFlow(FlowSpec):

# Define bucket name as an instance variable

bucket\_name = "mybucket" # Specify your bucket name here

@step

def start(self):

"""Start the flow and initialize the MinIO client."""

print("Starting the bucket creation process...")

self.next(self.create\_bucket)

@step

def create\_bucket(self):

"""Create a bucket in MinIO."""

# Create the MinIO client as a local variable

minio\_client = Minio(

"localhost:9000", # Update with your MinIO endpoint

access\_key="minioadmin", # Your MinIO access key

secret\_key="minioadmin", # Your MinIO secret key

secure=False

)

try:

# Attempt to create the bucket

minio\_client.make\_bucket(self.bucket\_name)

print(f"Bucket '{self.bucket\_name}' created successfully.")

except S3Error as e:

if e.code == 'BucketAlreadyOwnedByYou':

print(f"Bucket '{self.bucket\_name}' already exists.")

else:

print(f"Error creating bucket '{self.bucket\_name}': {e}")

raise

self.next(self.end)

@step

def end(self):

"""End the flow."""

print("Bucket creation process completed.")

if \_\_name\_\_ == '\_\_main\_\_':

CreateBucketFlow()