$batchAccount = "lvbatchdev"

$resourceGroup = "lv-batch-dev"

$accountEndpoint = "https://lvbatchdev.eastus.batch.azure.com"

Write-Host "=== Recycle Pool Nodes - Using Managed Identity (AAD) ==="

# 1. Login to Azure CLI using Managed Identity

Write-Host "Logging into Azure CLI using Managed Identity..."

az login --identity

if ($LASTEXITCODE -ne 0) {

Write-Error "❌ Failed to login using Managed Identity"

exit 1

}

# 2. Set Azure CLI context to the Batch account (do NOT use az batch account login)

Write-Host "Setting Azure CLI context to Batch account..."

az batch account set `

--name $batchAccount `

--resource-group $resourceGroup

if ($LASTEXITCODE -ne 0) {

Write-Error "❌ Failed to set Batch account context"

exit 1

}

# 3. Set environment variable for account endpoint if needed (optional)

$env:AZURE\_BATCH\_ACCOUNT = $batchAccount

$env:AZURE\_BATCH\_ENDPOINT = $accountEndpoint

# 4. Function to recycle a pool (no changes needed here)

function Recycle-Pool {

param ([string]$poolId)

Write-Host "`nRecycling pool: $poolId..."

Write-Host "Step 1: Scaling down to 0 nodes..."

az batch pool resize --pool-id $poolId --target-dedicated-nodes 0

if ($LASTEXITCODE -ne 0) {

Write-Error "❌ Failed to scale down pool $poolId"

exit 1

}

Write-Host "Waiting for nodes to deallocate..."

Start-Sleep -Seconds 30

Write-Host "Step 2: Scaling up to 1 node..."

az batch pool resize --pool-id $poolId --target-dedicated-nodes 1

if ($LASTEXITCODE -ne 0) {

Write-Error "❌ Failed to scale up pool $poolId"

exit 1

}

Write-Host "✅ Pool $poolId recycled successfully."

}

# 5. Recycle all your fixed pools

Recycle-Pool -poolId "unzip\_servicemac\_pool"

Recycle-Pool -poolId "audio\_conversion\_servicemac\_pool"

Write-Host "`n✅ All specified pools recycled successfully. Start task will re-execute on new nodes."