# Login to the batch account

az batch account login `

--name $(BatchAccount) `

--resource-group $(ResourceGroup)

if ($LASTEXITCODE -ne 0) {

Write-Error "Failed to login to batch account"

exit 1

}

Write-Host "Successfully logged in to batch account"

# Pool configuration

$poolId = "$(PoolId)"

$imageTriple = "$(ImagePublisher):$(ImageOffer):$(ImageSku)"

# Set error handling

$ErrorActionPreference = "Continue"

# Check if pool exists

Write-Host "Checking if pool exists..."

az batch pool show --pool-id $poolId 2>&1 | Out-Null

if ($LASTEXITCODE -ne 0) {

Write-Host "Pool $poolId not found - creating..."

# Create pool WITHOUT specifying node-agent-sku-id

# Azure Batch will automatically select the appropriate SKU

az batch pool create `

--id $poolId `

--vm-size $(VmSize) `

--target-dedicated-nodes 1 `

--image $imageTriple `

--start-task-command-line "/bin/echo pool ready"

if ($LASTEXITCODE -ne 0) {

throw "Pool creation failed."

}

Write-Host "Pool $poolId created successfully"

}

else {

Write-Host "Pool $poolId already exists"

}

2—————-

# Login to the batch account

az batch account login `

--name $(BatchAccount) `

--resource-group $(ResourceGroup)

if ($LASTEXITCODE -ne 0) {

Write-Error "Failed to login to batch account"

exit 1

}

Write-Host "Successfully logged in to batch account"

# Find the correct node agent SKU for Ubuntu 18.04

Write-Host "Finding correct node agent SKU for Ubuntu 18.04..."

$skuId = az batch pool node-agent-skus list `

--query "[?imageReferences[?publisher=='canonical' && offer=='UbuntuServer' && sku=='18.04-LTS']].id" `

--output tsv | Select-Object -First 1

if ([string]::IsNullOrEmpty($skuId)) {

# Try with uppercase 'Canonical'

$skuId = az batch pool node-agent-skus list `

--query "[?imageReferences[?publisher=='Canonical' && offer=='UbuntuServer' && sku=='18.04-LTS']].id" `

--output tsv | Select-Object -First 1

}

Write-Host "Found node agent SKU: $skuId"

# Pool configuration

$poolId = "$(PoolId)"

$imageTriple = "$(ImagePublisher):$(ImageOffer):$(ImageSku)"

# Set error handling

$ErrorActionPreference = "Continue"

# Check if pool exists

Write-Host "Checking if pool exists..."

az batch pool show --pool-id $poolId 2>&1 | Out-Null

if ($LASTEXITCODE -ne 0) {

Write-Host "Pool $poolId not found - creating..."

# Create pool with discovered node agent SKU

az batch pool create `

--id $poolId `

--vm-size $(VmSize) `

--target-dedicated-nodes 1 `

--image $imageTriple `

--node-agent-sku-id $skuId `

--start-task-command-line "/bin/echo pool ready"

if ($LASTEXITCODE -ne 0) {

throw "Pool creation failed."

}

Write-Host "Pool $poolId created successfully"

}

else {

Write-Host "Pool $poolId already exists"

}