# ------------------------------------------------------------------

# Variables

# ------------------------------------------------------------------

$batchAccount = "$(BatchAccount)" # lvbatchdev

$resourceGrp = "$(ResourceGroup)" # lv-batch-dev

$region = "$(Region)" # eastus2

$vmSize = "$(VmSize)" # Standard\_DS1\_v2

# \*\*\*\* Fix 1 – NO spaces in the string \*\*\*\*

$endpoint = "https://$batchAccount.$region.batch.azure.com"

# ------------------------------------------------------------------

# Login (same as before, endpoint now valid)

# ------------------------------------------------------------------

az batch account login `

--name $batchAccount `

--resource-group $resourceGrp `

--account-endpoint $endpoint

if ($LASTEXITCODE -ne 0) { throw "Batch login failed" }

Write‑Host "✅ Logged in to Batch"

# ------------------------------------------------------------------

# Build the auto‑pool JSON (unchanged except SKU line)

# ------------------------------------------------------------------

$jobId = "$(ApplicationId)-$(Build.BuildId)"

$jobJson = @"

{

"id": "$jobId",

"poolInfo": {

"autoPoolSpecification": {

"autoPoolIdPrefix": "autopool",

"poolLifetimeOption": "job",

"keepAlive": false,

"pool": {

"vmSize": "$vmSize",

"virtualMachineConfiguration": {

"imageReference": {

"publisher": "canonical",

"offer": "UbuntuServer",

"sku": "20\_04-lts",

"version": "latest"

},

"nodeAgentSKUId": "batch.node.ubuntu 20.04"

},

"targetDedicatedNodes": 1

}

}

}

}

"@

# ------------------------------------------------------------------

# Write JSON to temp file

# ------------------------------------------------------------------

$tmp = [System.IO.Path]::GetTempFileName()

$jobJson | Out‑File -Encoding UTF8 $tmp

# ------------------------------------------------------------------

# \*\*\*\* Fix 2 – rely on the stored context, no bad endpoint argument

# ------------------------------------------------------------------

az batch job create --json-file $tmp

if ($LASTEXITCODE -ne 0) { throw "Job creation with auto‑pool failed" }

Remove-Item $tmp -Force

Write‑Host "🎉 Job $jobId created with auto‑pool"