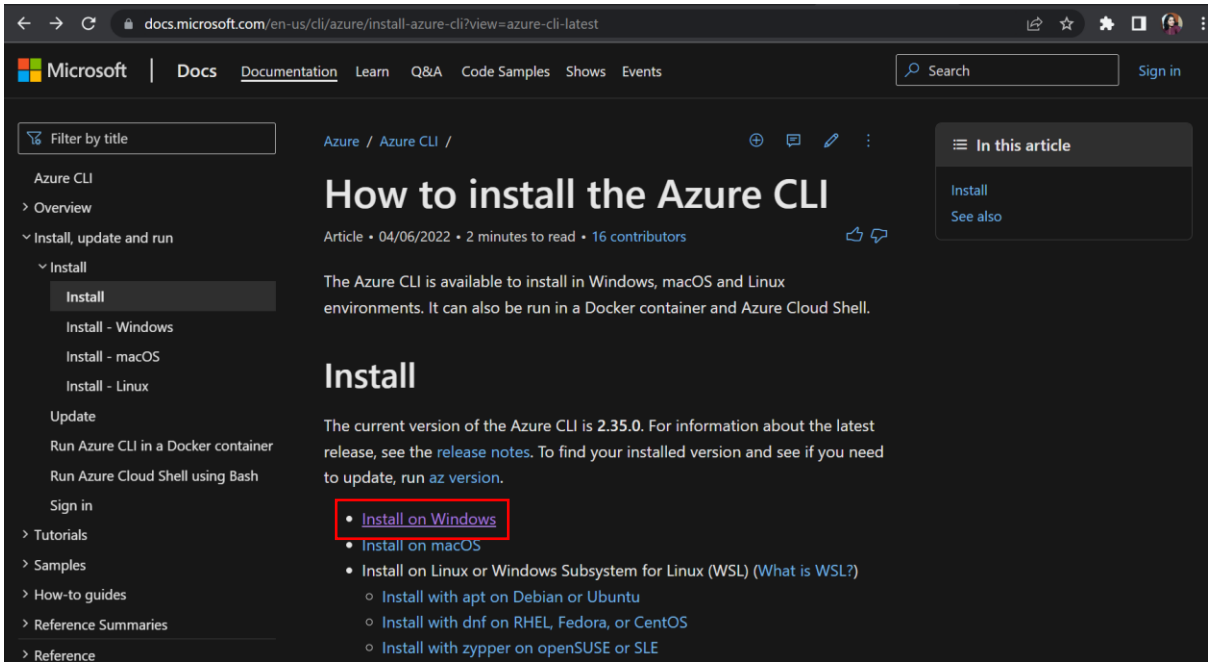
PRACTICAL NO - 04

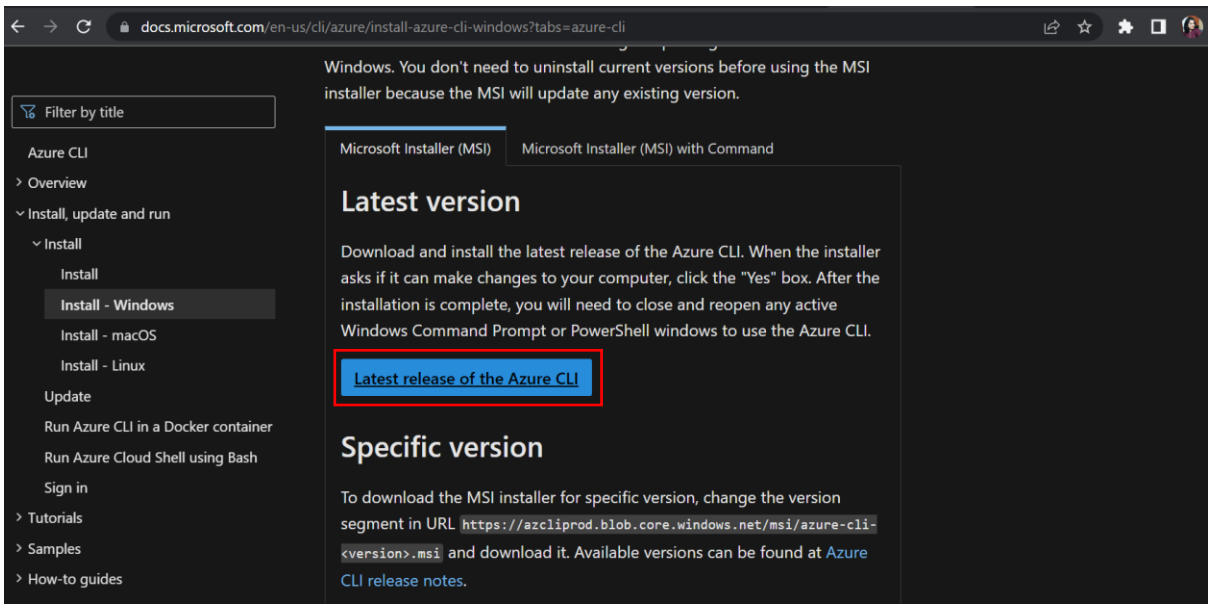
Create an AKS Cluster with Azure CLI

Step I – Install Azure CLI

- Go to https://docs.microsoft.com/en-us/cli/azure/install-azure-cli?view=azure-cli-latest ➤ Click on Install on Windows

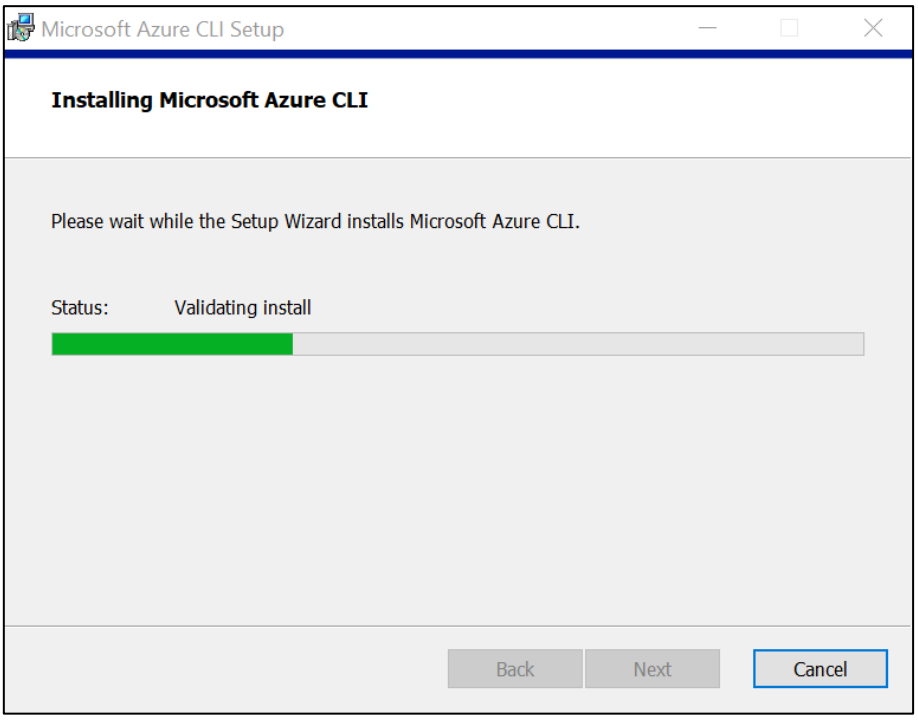


- You will navigate Install Azure CLI on Windows page. Scroll down ➤ Click on Latest version of the Azure CLI

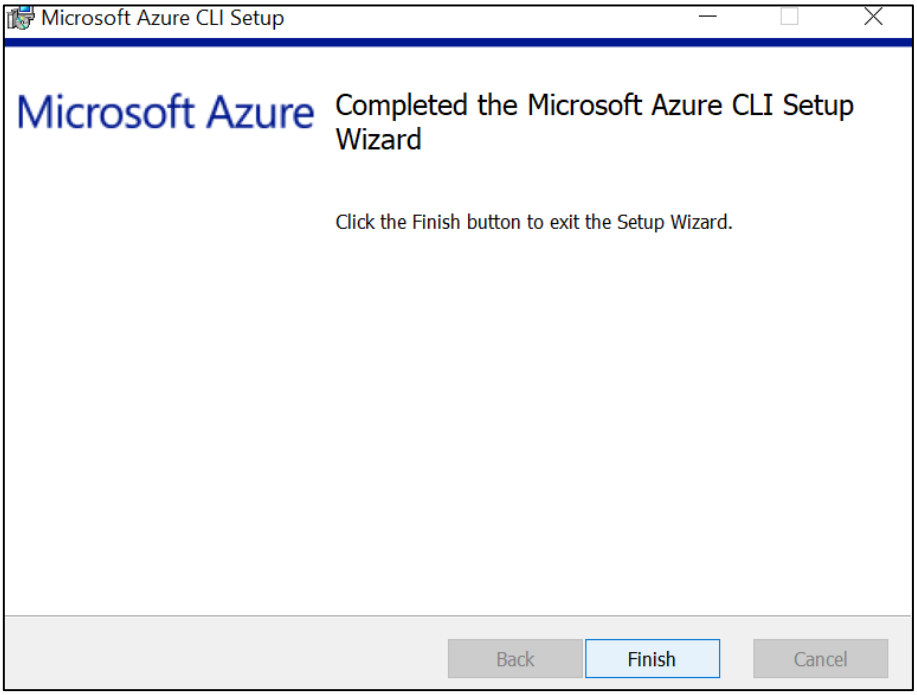


- It will download *'azure-cli-2.35.0.msi'* file. Double click and Run it.
- The Microsoft Azure CLI Setup window will appear. Accept the License Agreement and click on *Install*.

- The Azure CLI installation process will begin. Wait for a few minutes to complete the installation.

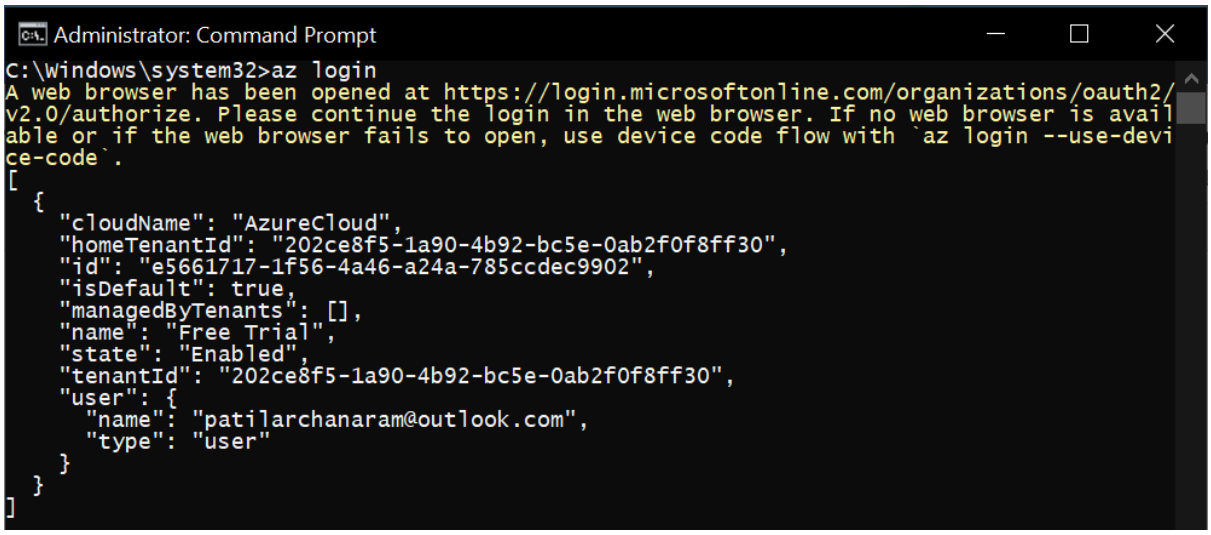


- Finally click on the Finish button.



Step II – Login to your Azure account.

- Open command prompt as an administrator.
- Sign in to the Azure CLI by using the *az login* command. To finish the authentication process, follow the steps displayed in your terminal.



- Run *az version* command to find the version and dependent libraries that are installed.

- To upgrade to the latest version, run *az upgrade* command.

```
C:\Windows\system32>az upgrade
This command is in preview and under development. Reference and support levels: https://
aka.ms/CLI_refstatus
You already have the latest azure-cli version: 2.35.0
Upgrade finished.You can enable auto-upgrade with 'az config set auto-upgrade.enable=yes
'. More details in https://docs.microsoft.com/cli/azure/update-azure-cli#automatic-updat
e
```

Step III – Create a Resource Group.

- An Azure resource group is a logical group in which Azure resources are deployed and managed.
- When you create a resource group, you are asked to specify a location. This location is where resource group metadata is stored, it is also where your resources run in Azure. If you don't want to specify another region during resource creation, create a resource group using the *az group create* command.

az group create --name myResourceGroup --location EastAsia

OR

az group create -n myResourceGroup -l EastAsia

```
C:\Windows\system32>az group create -n myResourceGroup -l EastAsia
{
  "id": "/subscriptions/e5661717-1f56-4a46-a24a-785ccdec9902/resourceGroups/myRe
sourceGroup",
  "location": "eastasia",
  "managedBy": null,
  "name": "myResourceGroup",
  "properties": {
    "provisioningState": "Succeeded"
  },
  "tags": null,
  "type": "Microsoft.Resources/resourceGroups"
}
```

Step III – Create AKS Cluster.

- Use the *az aks create* command to create an AKS cluster. The following example creates a cluster named myAKSCluster with one node. This will take several minutes to complete.

az aks create --resource-group myResourceGroup --name myAKSCluster --node-count 1 --enable-addons monitoring --generate-ssh-keys

OR

az aks create -g myResourceGroup -n myAKSCluster -c 1 --generate-ssh-keys

- After a few minutes, the command completes and returns JSON-formatted information about the cluster.

```
C:\Windows\system32>az aks create -g myResourceGroup -n myAKSCluster -c 1 --gene
rate-ssh-keys
{
  "aadProfile": null,
  "addonProfiles": null,
  "agentPoolProfiles": [
    {
      "availabilityZones": null,
      "count": 1,
      "creationData": null,
      "enableAutoScaling": false,
      "enableEncryptionAtHost": false,
      "enableFips": false,
      "enableNodePublicIp": false,
      "enableUltraSsd": false,
      "gpuInstanceProfile": null,
      "kubeletConfig": null,
      "kubeletDiskType": "OS",
      "linuxOsConfig": null,
      "maxCount": null,
      "maxPods": 110,
      "minCount": null,
      "mode": "System",
      "name": "nodepool1",
      "nodeImageVersion": "AKSUbuntu-1804gen2containerd-2022.03.29",
      "nodeLabels": null,
      "nodePublicIpPrefixId": null,
      "nodeTaints": null,
      "orchestratorVersion": "1.21.9",
      "osDiskSizeGb": 128,
      "osDiskType": "Managed",
      "osSku": "Ubuntu",
      "osType": "Linux",
      "podSubnetId": null,
      "powerState": {
        "code": "Running"
      },
      "provisioningState": "Succeeded",
      "proximityPlacementGroupId": null,
      "scaleDownMode": null,
      "scaleSetEvictionPolicy": null,
      "scaleSetPriority": null,
      "spotMaxPrice": null,
      "tags": null,
      "type": "VirtualMachineScaleSets",
      "upgradeSettings": null,
      "vmSize": "Standard_DS2_v2",
      "vnetSubnetId": null,
      "workloadRuntime": null
    }
  ],
  "apiServerAccessProfile": null,
```

Step IV - Connect to the cluster

- Configure *kubectl* to connect to your Kubernetes cluster using the *az aks get-credentials* command. The following command downloads credentials and configures the Kubernetes CLI to use them.

az aks get-credentials -g [Resource Group] -n [Name of Azure Kubernetes Service]

```
C:\Windows\system32>az aks get-credentials -g myResourceGroup -n myAKSCluster
Merged "myAKSCluster" as current context in C:\Users\patil\.kube\config
```

- Verify the connection to your cluster using *kubectl get* to return a list of the cluster nodes.

kubectl get nodes

```
C:\Windows\system32>kubectl get nodes
NAME                                STATUS   ROLES   AGE   VERSION
aks-nodepool1-16494706-vmss000000   Ready    agent   56m   v1.21.9
```