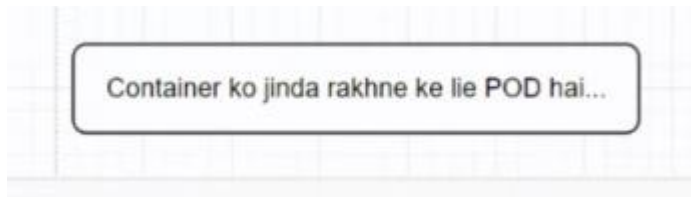


02 November 2024

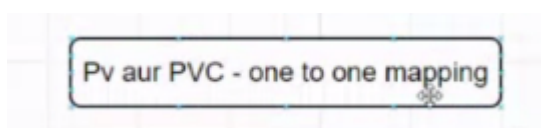
1)



2) Storage class is of 2 types – static and dynamic

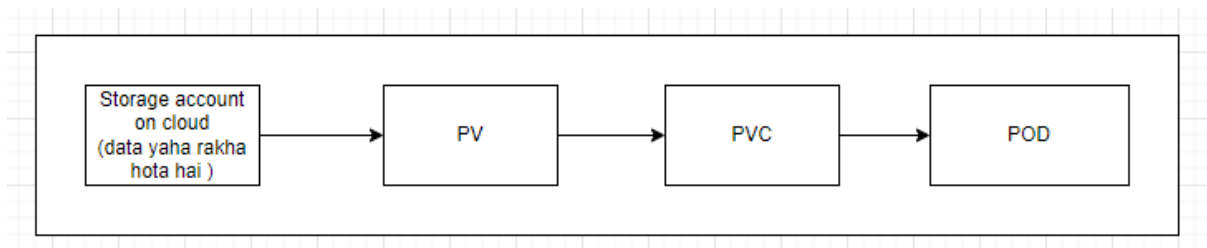
+++++

AGENDA = DYNAMIC STORAGE CLASS

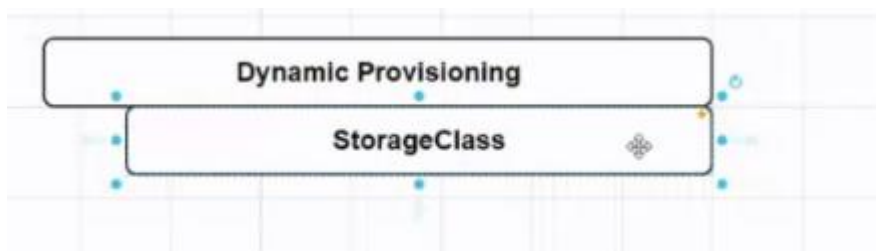


21) PV and pvc me 1 to 1 mapping hoti hai mtlb 1 pv se 1 hi pvc connect hoga. But 1 pvc can be mounted on multiple pods. Basically pvc ek hi rhega.

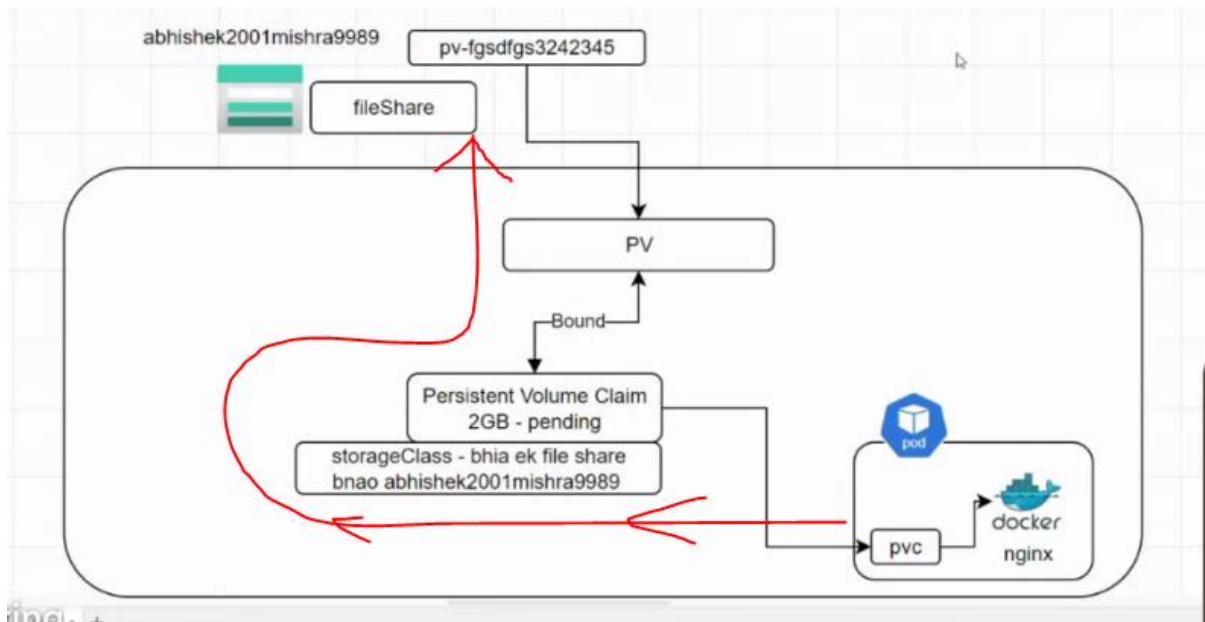
22) Dynamic provisioning = Ek aisa tarika jisme pvc banne ke baad pv aur storage account khud se ban jaata hai



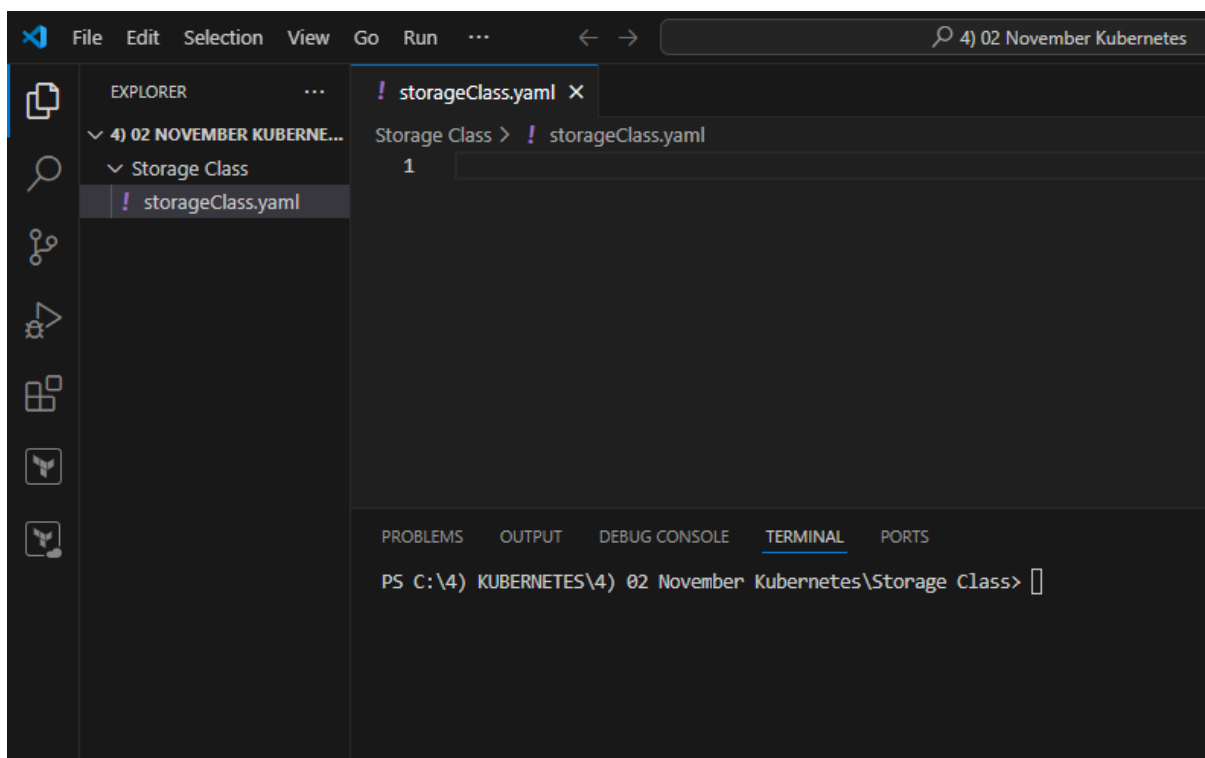
23) Dynamic provisioning of volume in k8s is done by "Storage class".



24) Storage class actually automates the process of making file share and persistent volume.



25) Now making storage class so create folder “4) 02 November Kubernetes” and in that create Storage class folder and file in vs code

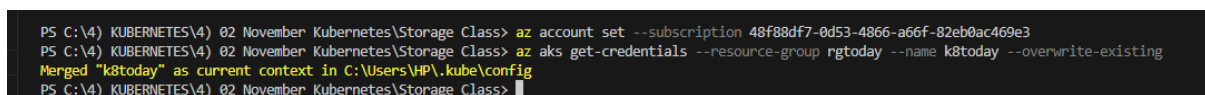


26) Connect k8s cluster in it

az login

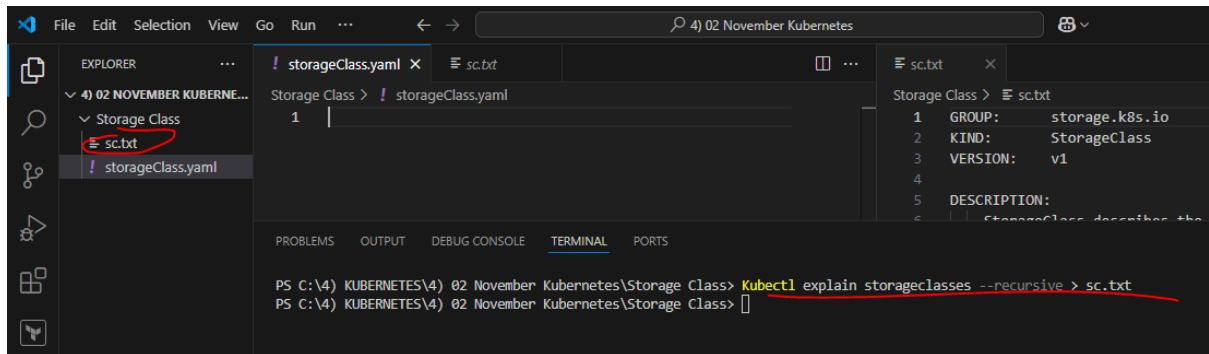
az account set --subscription 48f88df7-0d53-4866-a66f-82eb0ac469e3

az aks get-credentials --resource-group rgtoday --name k8today --overwrite-existing

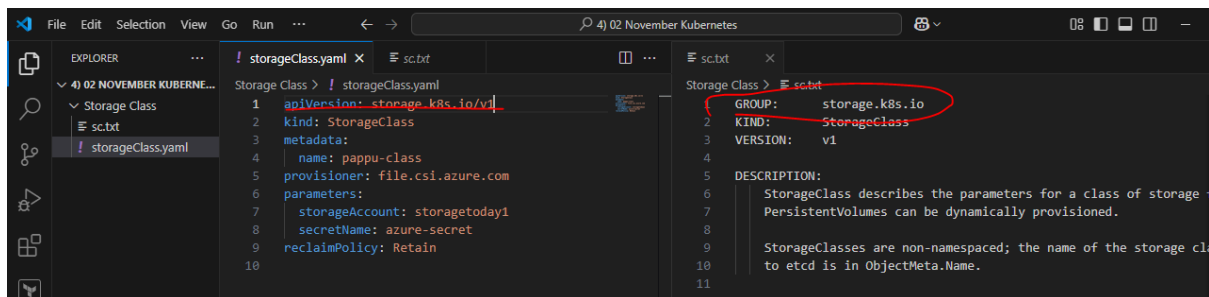


27) Bring doc of storage class

Kubectl explain storageclasses --recursive > sc.txt

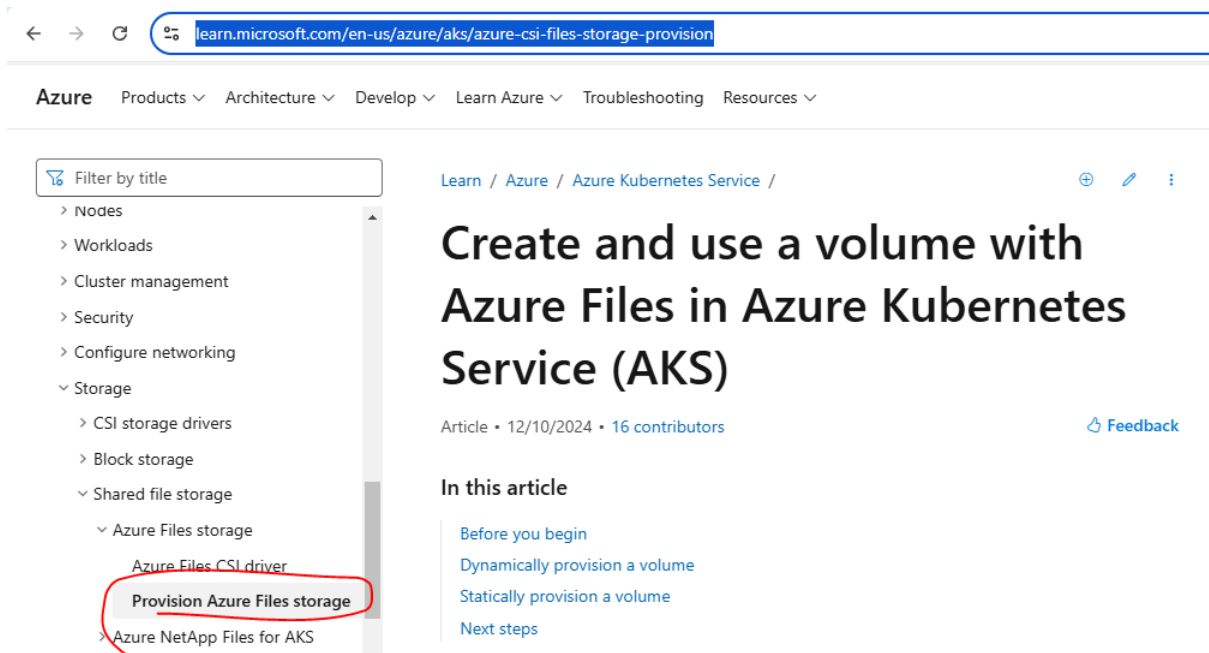


NOTE : 1) Whenever group comes in our doc then it will be apiVersion not v1



28) **provisioner**: = means kaha se provision krna hai, azure storage acc ke file share se, ya aws ki s3 bucket se ya gcp ki bucket se

29) SEARCH = <https://learn.microsoft.com/en-us/azure/aks/azure-csi-files-storage-provision>



```

kind: StorageClass
apiVersion: storage.k8s.io/v1
metadata:
  name: my-azurefile
  provisioner: file.csi.azure.com # replace with "kubern
allowVolumeExpansion: true
mountOptions:
  - dir_mode=0777
    file_mode=0777

```

30) Create storage account

Create secret also

kubectl create secret generic azure-secret --from-literal=azurestorageaccountname=<...> --from-literal=azurestorageaccountkey=<...>

kubectl create secret generic azure-secret --from-literal=azurestorageaccountname=storagetoday1 --from-literal=azurestorageaccountkey=8B4Sq5RVc+pfihTkc+EKe2JBkYChHbRj0Z5KM9d+gR5NpIkFKVq43+KGM2gpHKksgd9IMRZ4iurT+AST+HiJXw==

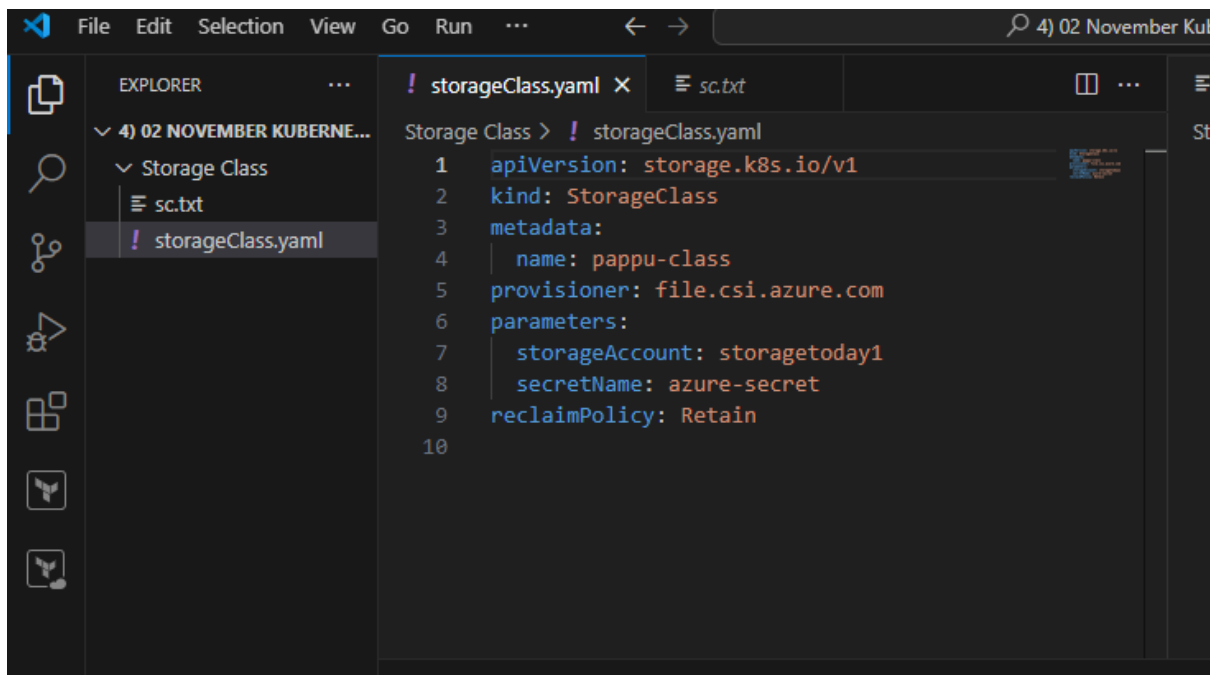
```

PS C:\4) KUBERNETES\4) 02 November Kubernetes\Storage Class> kubectl create secret generic azure-secret --from-literal=azurestorageaccountname=storagetoday1 --from-literal=azurestorageaccountkey=8B4Sq5RVc+pfihTkc+EKe2JBkYChHbRj0Z5KM9d+gR5NpIkFKVq43+KGM2gpHKksgd9IMRZ4iurT+AST+HiJXw==
secret/azure-secret created
PS C:\4) KUBERNETES\4) 02 November Kubernetes\Storage Class>

```

31) allowVolumeExpansion: true = to increase volume in it

32) Now storageClass.yaml is ready to create pv, storage account automatically.



33) **kubectl apply -f storageClass.yaml** = Create storage class

```
PS C:\4) KUBERNETES\4) 02 November Kubernetes\Storage Class> kubectl apply -f storageClass.yaml
storageclass.storage.k8s.io/pappu-class created
PS C:\4) KUBERNETES\4) 02 November Kubernetes\Storage Class> 
```

34) kubectl get sc

```
PS C:\4) KUBERNETES\4) 02 November Kubernetes\Storage Class> kubectl get sc
NAME                PROVISIONER          RECLAIMPOLICY  VOLUMEBINDINGMODE  ALLOWVOLUMEEXPANSION  AGE
azurefile            file.csi.azure.com   Delete         Immediate          true                  148m
azurefile-csi        file.csi.azure.com   Delete         Immediate          true                  148m
azurefile-csi-premium file.csi.azure.com   Delete         Immediate          true                  148m
azurefile-premium    file.csi.azure.com   Delete         Immediate          true                  148m
default (default)    disk.csi.azure.com   Delete         WaitForFirstConsumer true                  148m
managed              disk.csi.azure.com   Delete         WaitForFirstConsumer true                  148m
managed-csi          disk.csi.azure.com   Delete         WaitForFirstConsumer true                  148m
managed-csi-premium  disk.csi.azure.com   Delete         WaitForFirstConsumer true                  148m
managed-premium      disk.csi.azure.com   Delete         WaitForFirstConsumer true                  148m
pappu-class          file.csi.azure.com   Retain         Immediate          false                 3m30s
PS C:\4) KUBERNETES\4) 02 November Kubernetes\Storage Class> 
```

35) Storage classes cannot be made in any specific name space its actually made for whole cluster. We can verify below seeing namespace colum for storageclasses is false

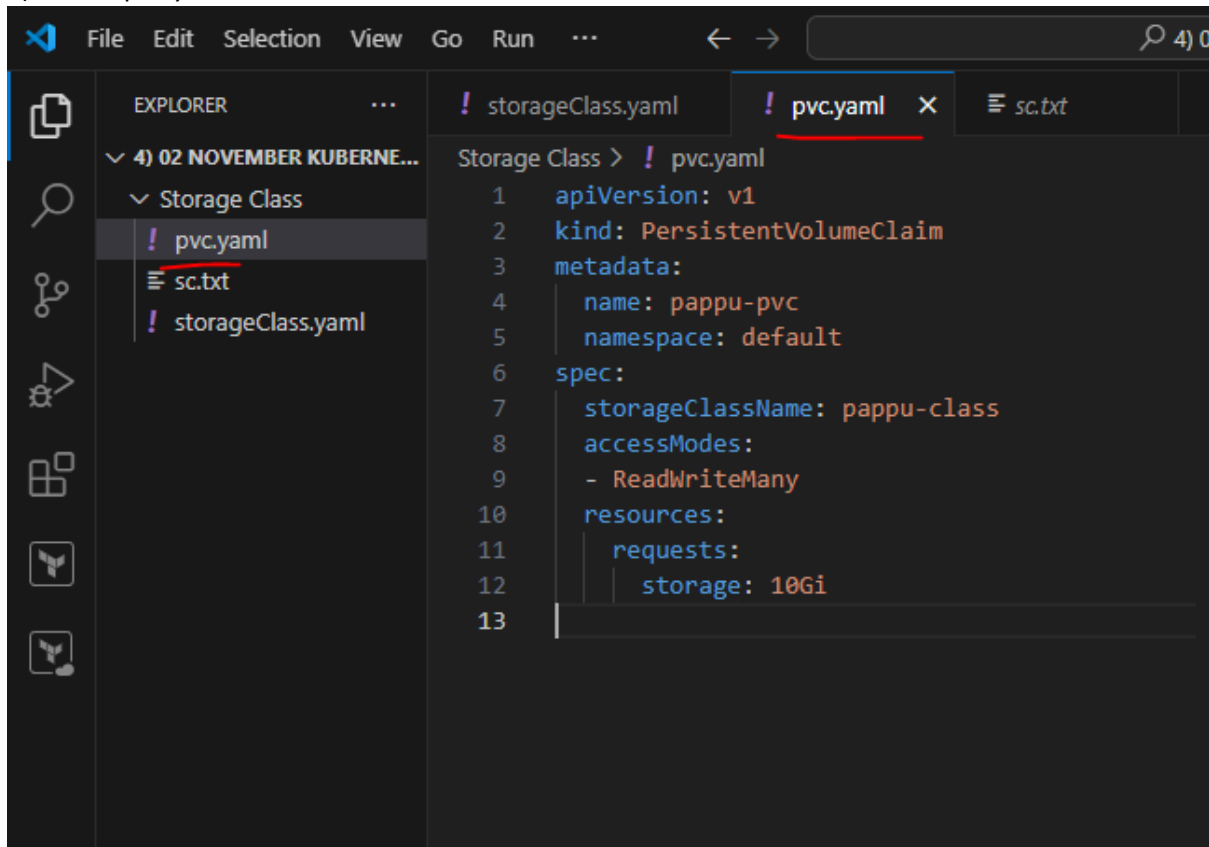
```
PS C:\4) KUBERNETES\4) 02 November Kubernetes\Storage Class> kubectl api-resources
NAME                SHORTNAMES  APIVERSION  NAMESPACE  KIND
bindings            sc          storage.k8s.io/v1  false      Binding
storageclasses      sc          storage.k8s.io/v1  false      StorageClass
volumeattachments   sc          storage.k8s.io/v1  false      VolumeAttachment
PS C:\4) KUBERNETES\4) 02 November Kubernetes\Storage Class> 
```

+++++

AGENDA – CREATE PVC.YAML

NOTE: 1) PV are at cluster level and PVC are at namespace level

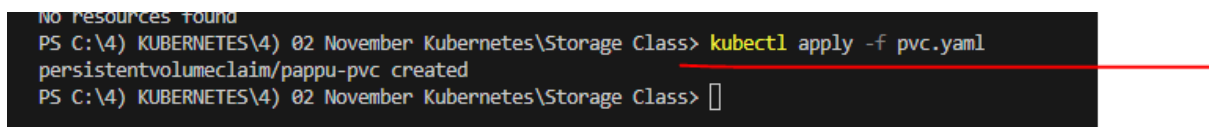
1) Create pvc.yaml file



The screenshot shows the Visual Studio Code interface. The Explorer sidebar on the left shows a folder named '4) 02 NOVEMBER KUBERNE...' containing three files: 'pvc.yaml' (highlighted with a red underline), 'sc.txt', and 'storageClass.yaml'. The main editor area shows the 'pvc.yaml' file with the following content:

```
1  apiVersion: v1
2  kind: PersistentVolumeClaim
3  metadata:
4    name: pappu-pvc
5    namespace: default
6  spec:
7    storageClassName: pappu-class
8    accessModes:
9      - ReadWriteMany
10   resources:
11     requests:
12       storage: 10Gi
13
```

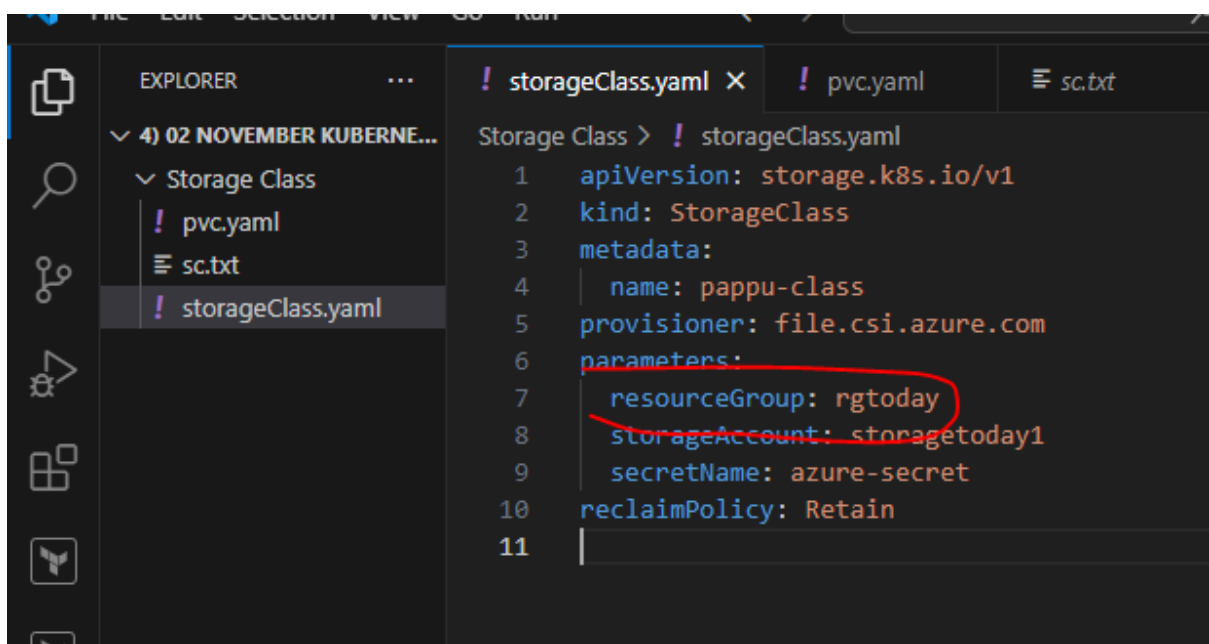
2) **kubectl apply -f pvc.yaml** = creating pvc



The screenshot shows a terminal window with the following output:

```
No resources found
PS C:\4) KUBERNETES\4) 02 November Kubernetes\Storage Class> kubectl apply -f pvc.yaml
persistentvolumeclaim/pappu-pvc created
PS C:\4) KUBERNETES\4) 02 November Kubernetes\Storage Class> 
```

3) We had got error because we didn't mention rg



The screenshot shows the Visual Studio Code interface. The Explorer sidebar on the left shows the same folder, but now 'storageClass.yaml' is highlighted. The main editor area shows the 'storageClass.yaml' file with the following content:

```
1  apiVersion: storage.k8s.io/v1
2  kind: StorageClass
3  metadata:
4    name: pappu-class
5  provisioner: file.csi.azure.com
6  parameters:
7    resourceGroup: rgtoday
8    storageAccount: storagetoday1
9    secretName: azure-secret
10 reclaimPolicy: Retain
11
```

4) Deleting

kubectl get pvc

kubectl get sc

```
PS C:\4) KUBERNETES\4) 02 November Kubernetes\Storage Class> kubectl get pvc
NAME          STATUS    VOLUME          CAPACITY   ACCESS MODES   STORAGECLASS   VOLUMEATTRIBUTESCLASS   AGE
pappu-pvc     Pending  pappu-class     1Gi        RWX             pappu-class    <unset>                 9m11s
PS C:\4) KUBERNETES\4) 02 November Kubernetes\Storage Class> kubectl get sc
NAME          PROVISIONER      RECLAIMPOLICY   VOLUMEBINDINGMODE   ALLOWVOLUMEEXPANSION   AGE
azurefile     file.csi.azure.com Delete          Immediate            true                   179m
azurefile-csi file.csi.azure.com Delete          Immediate            true                   179m
azurefile-csi-premium file.csi.azure.com Delete          Immediate            true                   179m
azurefile-premium file.csi.azure.com Delete          Immediate            true                   179m
default (default) disk.csi.azure.com Delete          WaitForFirstConsumer true                   179m
managed       disk.csi.azure.com Delete          WaitForFirstConsumer true                   179m
managed-csi   disk.csi.azure.com Delete          WaitForFirstConsumer true                   179m
managed-csi-premium disk.csi.azure.com Delete          WaitForFirstConsumer true                   179m
managed-premium disk.csi.azure.com Delete          WaitForFirstConsumer true                   179m
pappu-class   file.csi.azure.com Retain          Immediate            false                  34m
PS C:\4) KUBERNETES\4) 02 November Kubernetes\Storage Class>
```

Kubectl delete pvc pappu-pvc

Kubectl delete sc pappu-class

```
PS C:\4) KUBERNETES\4) 02 November Kubernetes\Storage Class> Kubectl delete pvc pappu-pvc
persistentvolumeclaim "pappu-pvc" deleted
PS C:\4) KUBERNETES\4) 02 November Kubernetes\Storage Class> Kubectl delete sc pappu-class
storageclass.storage.k8s.io "pappu-class" deleted
PS C:\4) KUBERNETES\4) 02 November Kubernetes\Storage Class>
```

5) Added rg now again create sc and pvc

6) Not able to do above agenda as assignment is create a user managed identity. Then attach it to cluster and then everything will be done for above agenda i.e. automatically create pv and file share in storage account.

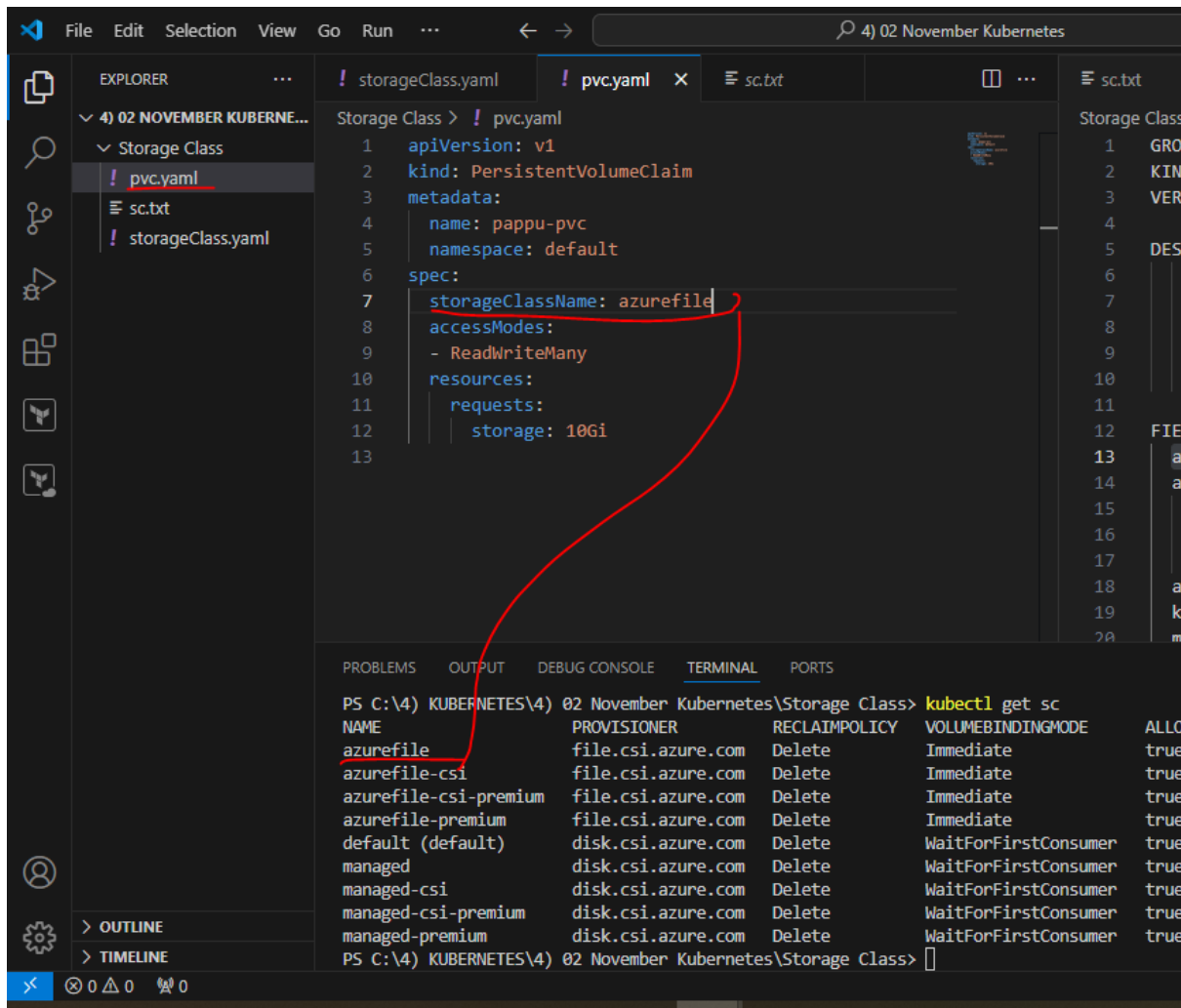
+++++

AGENDA – USING AUTOMATICALLY MADE STORAGE CLASS

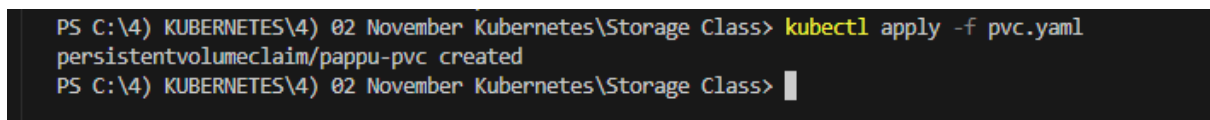
1) **kubectl get sc** = already created sc

```
PS C:\4) KUBERNETES\4) 02 November Kubernetes\Storage Class> kubectl get sc
NAME          PROVISIONER      RECLAIMPOLICY   VOLUMEBINDINGMODE   ALLOWVOLUMEEXPANSION   AGE
azurefile     file.csi.azure.com Delete          Immediate            true                   4h17m
azurefile-csi file.csi.azure.com Delete          Immediate            true                   4h17m
azurefile-csi-premium file.csi.azure.com Delete          Immediate            true                   4h17m
azurefile-premium file.csi.azure.com Delete          Immediate            true                   4h17m
default (default) disk.csi.azure.com Delete          WaitForFirstConsumer true                   4h17m
managed       disk.csi.azure.com Delete          WaitForFirstConsumer true                   4h17m
managed-csi   disk.csi.azure.com Delete          WaitForFirstConsumer true                   4h17m
managed-csi-premium disk.csi.azure.com Delete          WaitForFirstConsumer true                   4h17m
managed-premium disk.csi.azure.com Delete          WaitForFirstConsumer true                   4h17m
PS C:\4) KUBERNETES\4) 02 November Kubernetes\Storage Class>
```

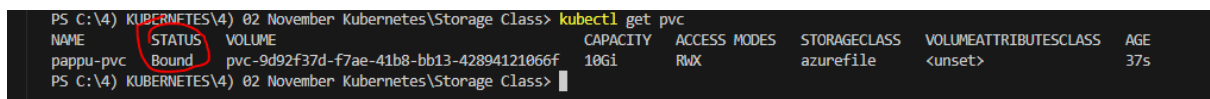
2) Use already created storage class



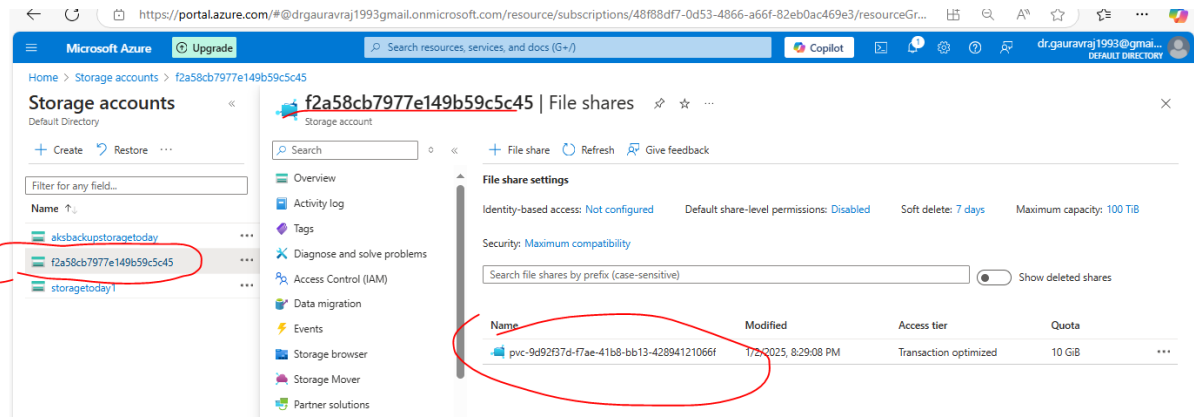
3) **kubectl apply -f pvc.yaml** = create pvc



4) **kubectl get pvc**



5) Now pvc got bounded and a storage account and file share is created automatically.

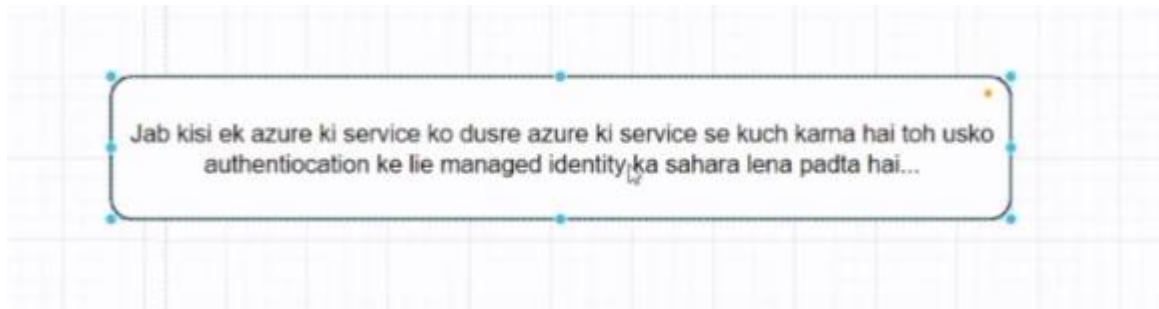


6)

+++++

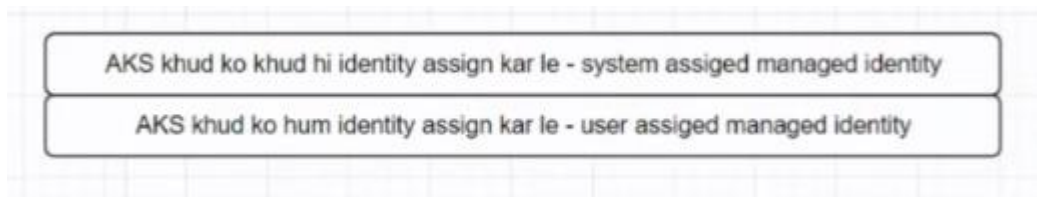
AGENDA – Assign managed identity to our aks cluster

1) Why managed identity is used?



2) So aks ko azure ke storage account se baat krna hai to managed identity aks pr chipka denge

3) Managed identity are of two types – System assigned and user assigned



4) SEARCH = <https://learn.microsoft.com/en-us/azure/aks/use-managed-identity#enable-a-user-assigned-managed-identity>

learn.microsoft.com/en-us/azure/aks/use-managed-identity#enable-a-user-assigned-managed-identity

Learn Discover Product documentation Development languages Topics

Azure Products Architecture Develop Learn Azure Troubleshooting Resources

Filter by title

- Use managed identities
 - Create a service principal
 - Enable access to AKS clusters using Trusted Access
 - Limit access to cluster configuration file
 - Define API server authorized IP ranges
 - Manage cluster certificates and rotation
 - Use custom certificate authorities
 - Use Azure Policy
 - Control deployments with Azure Policy
 - Create an OIDC Issuer for your cluster
 - Use KMS etcd encryption
 - Secure container access to resources
- Node security
- Authentication and authorization
- Application security
- Configure networking
- Storage

Download PDF

Learn / Azure / Azure Kubernetes Service /

Use a managed identity in Azure Kubernetes Service (AKS)

Article • 08/02/2024 • 39 contributors

Feedback

In this article

- Overview
- Before you begin
- Enable a system-assigned managed identity
- Enable a user-assigned managed identity
- Determine which type of managed identity a cluster is using
- Use a pre-created kubelet managed identity
- Summary of managed identities used by AKS
- Limitations
- Next steps
- Show less

Azure Kubernetes Service (AKS) clusters require a Microsoft Entra identity to access Azure resources like load balancers and managed disks. Managed identities for Azure resources are the recommended way to authorize access from an AKS cluster to other Azure services.

To verify that a system-assigned managed identity is enabled for the cluster after it has been created, see [Determine which type of managed identity a cluster is using](#).

Update an existing AKS cluster to use a system-assigned managed identity

To update an existing AKS cluster that is using a service principal to use a system-assigned managed identity instead, run the [az aks update](#) command with the `--enable-managed-identity` parameter.

```
Azure CLI Copy Open Cloud Shell
az aks update \
  --resource-group myResourceGroup \
  --name myManagedCluster \
  --enable-managed-identity
```

After you update the cluster to use a system-assigned managed identity instead of a service principal, the

5) **az aks update --resource-group rgtoday --name k8today --enable-managed-identity** = run to Update an existing AKS cluster to use a system-assigned managed identity

```
pappa-pvc: ~$ az aks update --resource-group rgtoday --name k8today --enable-managed-identity
PS C:\(4) KUBERNETES\4) 02 November Kubernetes\Storage Class> az aks update --resource-group rgtoday --name k8today --enable-managed-identity
Running ..
```

6) So now automatically managed identity created and it got attached to our k8s cluster

Home > Managed Identities

Default Directory

+ Create Manage view Refresh Export to CSV Open query Assign tags Delete

Filter for any field... Subscription equals all Resource group equals all Location equals all Add filter

Showing 1 to 4 of 4 records. No grouping

Name	Type	Resource group	Location
azurekeyvaultsecretsprovider-primary-aks-cluster	Managed Identity	MC_rghappy_primary-aks-cluster_cana...	Canada Central
ingressapplicationgateway-primary-aks-cluster	Managed Identity	MC_rghappy_primary-aks-cluster_cana...	Canada Central
k8today-agentpool	Managed Identity	MC_rgtoday_k8today_polandcentral	Poland Central
primary-aks-cluster-agentpool	Managed Identity	MC_rghappy_primary-aks-cluster_cana...	Canada Central

6) Go to storage account – access control – give contributor role – select below details – review and assign

Microsoft Azure Upgrade Search resources, services, and docs (G+/) Copilot dr.gauravraj1993@gmail... DEFAULT DIRECTORY

Home > Storage accounts > storagetoday1 | Access Control (IAM) > Add role assignment

Role Members Conditions Review + assign

Selected role Contributor

Assign access to ☐ User, group, or service principal ☒ Managed identity

Members + Select members

Name	Object ID	Type
No members selected		

Description Optional

Review + assign Previous Next

Select managed identities

Some results might be hidden due to your ABAC condition.

Subscription * Free Trial

Managed identity Kubernetes service (2)

Select Search by name

- k8today /subscriptions/48f88df7-0d53-4866-a66f-82eb0ac469e3/resourceGroups/rgtoday/...
- primary-aks-cluster /subscriptions/48f88df7-0d53-4866-a66f-82eb0ac469e3/resourceGroups/rghappy/...

Selected members: No members selected. Search for and add one or more members you want to assign to the role for this resource.

Learn more about RBAC

Select Close Feedback

7) Create storageClass1.yaml

File Edit Selection View Go Run ... 4) 02 November Ku

EXPLORER

- 4) 02 NOVEMBER KUBERNE...
- Storage Class
 - pvc.yaml
 - sc.txt
 - storageClass.yaml
 - storageclass1.yaml

Storage Class > ! storageclass1.yaml

```

1  apiVersion: storage.k8s.io/v1
2  kind: StorageClass
3  metadata:
4    name: storage1-class
5  provisioner: file.csi.azure.com
6  parameters:
7    resourceGroup: rgtoday
8    storageAccount: storagetoday1
9  reclaimPolicy: Retain

```

8) `kubectl apply -f storageclass1.yaml`

```
PS C:\4) KUBERNETES\4) 02 November Kubernetes\Storage Class> kubectl apply -f storageclass1.yaml
storageclass.storage.k8s.io/storage1-class created
```

9) `kubectl get sc`

```
PS C:\4) KUBERNETES\4) 02 November Kubernetes\Storage Class> kubectl get sc
NAME                PROVISIONER          RECLAIMPOLICY   VOLUMEBINDINGMODE   ALLOWVOLUMEEXPANSION   AGE
azurefile            file.csi.azure.com   Delete          Immediate            true                   5h55m
azurefile-csi        file.csi.azure.com   Delete          Immediate            true                   5h55m
azurefile-csi-premium file.csi.azure.com   Delete          Immediate            true                   5h55m
azurefile-premium    file.csi.azure.com   Delete          Immediate            true                   5h55m
default (default)    disk.csi.azure.com   Delete          WaitForFirstConsumer true                   5h55m
managed              disk.csi.azure.com   Delete          WaitForFirstConsumer true                   5h55m
managed-csi          disk.csi.azure.com   Delete          WaitForFirstConsumer true                   5h55m
managed-csi-premium  disk.csi.azure.com   Delete          WaitForFirstConsumer true                   5h55m
managed-premium      disk.csi.azure.com   Delete          WaitForFirstConsumer true                   5h55m
storage1-class       file.csi.azure.com   Retain          Immediate            false                  49s
```

10) Go to subscription and give contributor access to aks cluster

Microsoft Azure | Upgrade | Search resources, services, and docs (G+)

Home > Subscriptions > Free Trial | Access control (IAM) >

Add role assignment

Role: **Contributor** | Members: **Members** | Conditions: **Conditions** | Review + assign

Selected role: Contributor

Assign access to: ☐ User, group, or service principal ☒ Managed identity

Members: [+ Select members](#)

Name	Object ID	Type
No members selected		

Description: Optional

[Review + assign](#) [Previous](#) [Next](#)

Select managed identities

Some results might be hidden due to your ABAC condition.

Subscription: Free Trial

Managed identity: Kubernetes service (2)

Select:

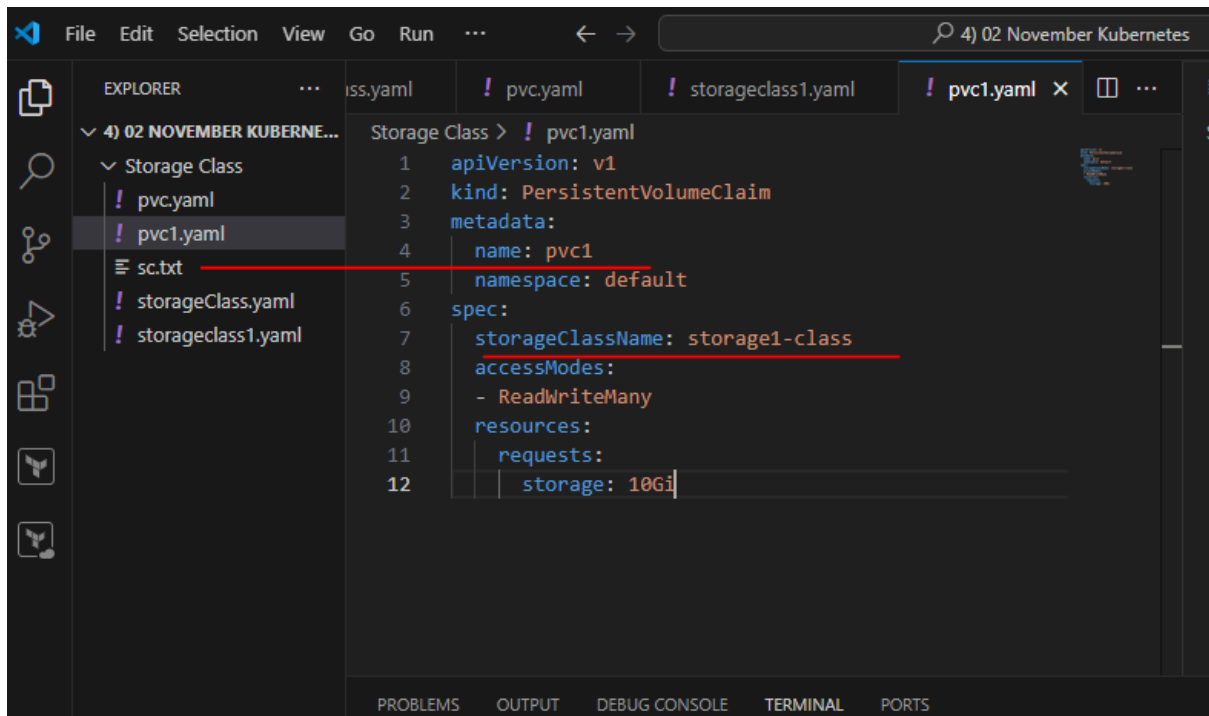
primary-aks-cluster
/subscriptions/48188df7-0d53-4866-a66f-82eb0ac469e3/resourceGroups/rghappy...

Selected members:

k8today
/subscriptions/48188df7-0d53-4866-a66f-82eb0ac469e3/resourceGroups/r... [Remove](#)

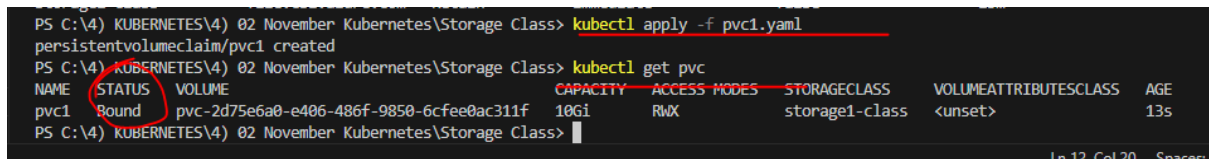
[Select](#) [Close](#) [Feedback](#)

11) Create new pvc.yaml

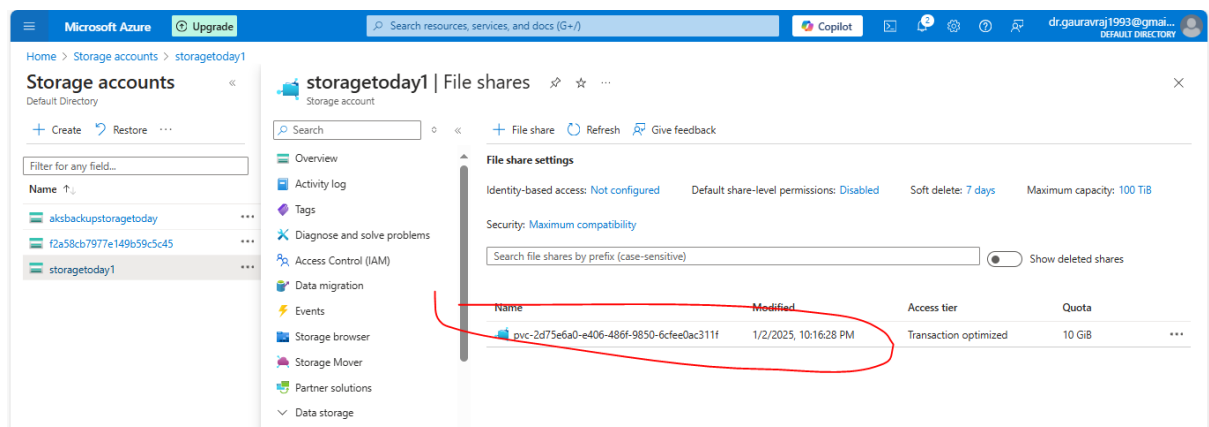


12) **kubectl apply -f pvc1.yaml**

kubectl get pvc



This time its bounded means new file share created in storage account



AGENDA – CONFIG MAPS

- 1) Config map is like space where we keep less sensitive things. For high sensitive things we use secrets.
- 2) 1) Configmap = It maps configurations which are not sensitive like that don't have password, key.

