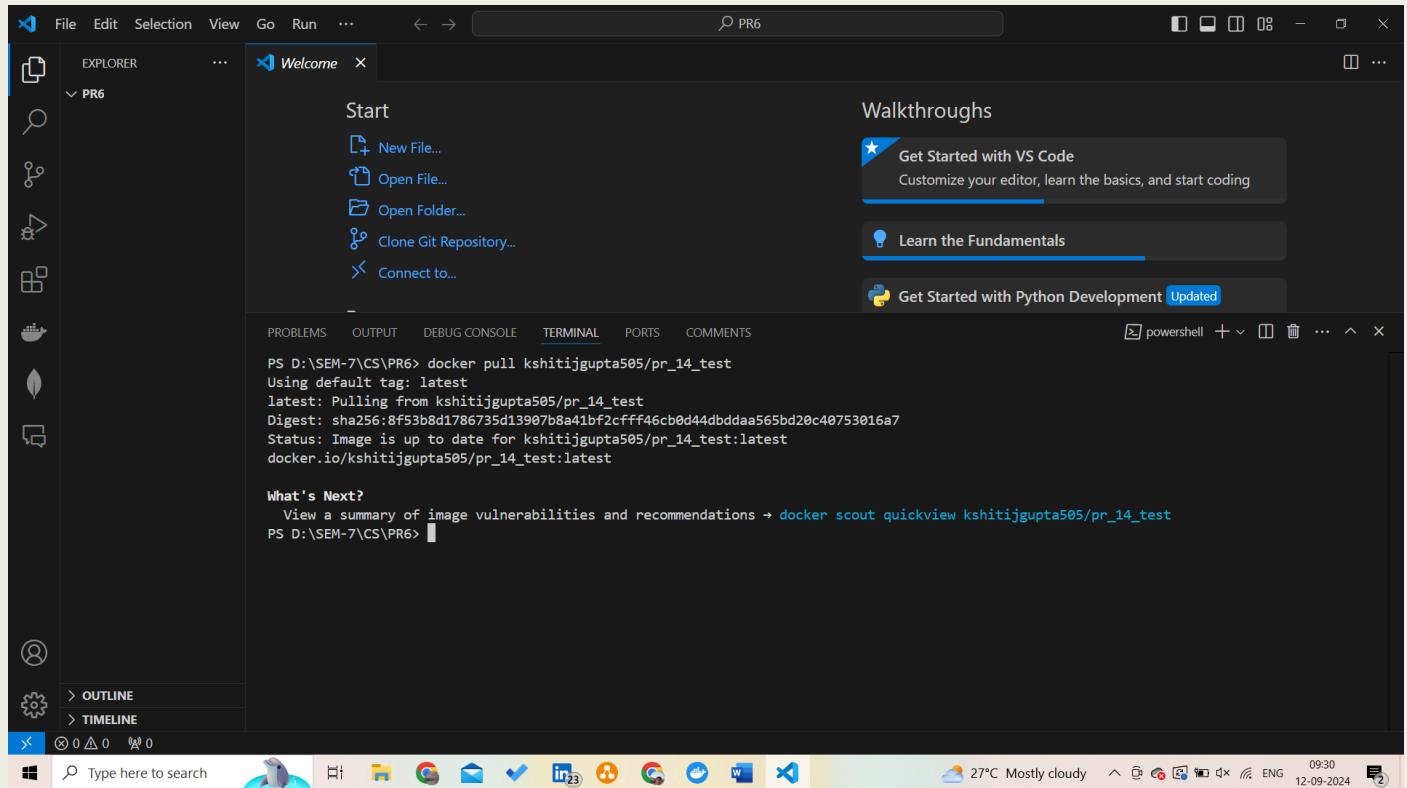


NAME: KSHITIJ GUPTA
Enrolment Number: 21162101007
Sub: CS

Practical – 6[Batch-71]

Go to practical 6 folder



A screenshot of the Visual Studio Code interface. The title bar says "PR6". The left sidebar shows icons for Explorer, Search, Open, and others. The main area has a "Welcome" tab open. On the left, under "EXPLORER", there is a folder named "PR6". In the center, there is a "Start" section with options like "New File...", "Open File...", "Open Folder...", "Clone Git Repository...", and "Connect to...". To the right, there is a "Walkthroughs" section with "Get Started with VS Code" (which is highlighted with a blue bar) and "Learn the Fundamentals". Below these, there is a "Get Started with Python Development" section. At the bottom of the screen, there is a taskbar with various icons and a system tray showing the date and time.

```
PS D:\SEM-7\CS\PR6> docker pull kshitijgupta505/pr_14_test
Using default tag: latest
latest: Pulling from kshitijgupta505/pr_14_test
Digest: sha256:8f53b8d1786735d13907b8a41bf2cff46cb0d44dbddaa565bd20c40753016a7
Status: Image is up to date for kshitijgupta505/pr_14_test:latest
docker.io/kshitijgupta505/pr_14_test:latest

What's Next?
View a summary of image vulnerabilities and recommendations → docker scout quickview kshitijgupta505/pr_14_test
PS D:\SEM-7\CS\PR6>
```

Pull the one random image from your repo

A screenshot of the Visual Studio Code interface. The left sidebar shows a project named 'PR6'. The main area has a 'Welcome' tab open. A terminal window is active, displaying the following command and output:

```
PS D:\SEM-7\CS\PR6> docker pull kshitijgupta505/pr_14_test
Using default tag: latest
latest: Pulling from kshitijgupta505/pr_14_test
Digest: sha256:8f53b8d1786735d13907b8a41bf2cff46cb0d44dbddaa565bd20c40753016a7
Status: Image is up to date for kshitijgupta505/pr_14_test:latest
docker.io/kshitijgupta505/pr_14_test:latest
```

Below the terminal, a 'What's Next?' section suggests running `docker scout quickview kshitijgupta505/pr_14_test`. The status bar at the bottom shows the date as 12-09-2024 and the time as 09:32.

A screenshot of a Windows terminal window. The command `docker pull kshitijgupta505/codeengine-prac9-eadc.git-49:latest` is run, followed by the output:

```
Status: Downloaded newer image for kshitijgupta505/codeengine-prac9-eadc.git-49:latest
docker.io/kshitijgupta505/codeengine-prac9-eadc.git-49:latest
```

Below the terminal, a 'What's Next?' section suggests running `docker scout quickview kshitijgupta505/codeengine-prac9-eadc.git-49`. The status bar at the bottom shows the date as 12-09-2024 and the time as 09:44.

A screenshot of the Visual Studio Code interface. The left sidebar shows a project named 'PR6'. The main area has a 'Welcome' tab open. A terminal window is active, displaying the following command and output:

```
C:\Users\Kshitij> docker pull kshitijgupta505/pr_14_test
Using default tag: latest
Error: remote trust data does not exist for docker.io/kshitijgupta505/pr_14_test: notary.docker.io does not have trust data for docker.io/kshitijgupta505/pr_14_test
C:\Users\Kshitij>
```

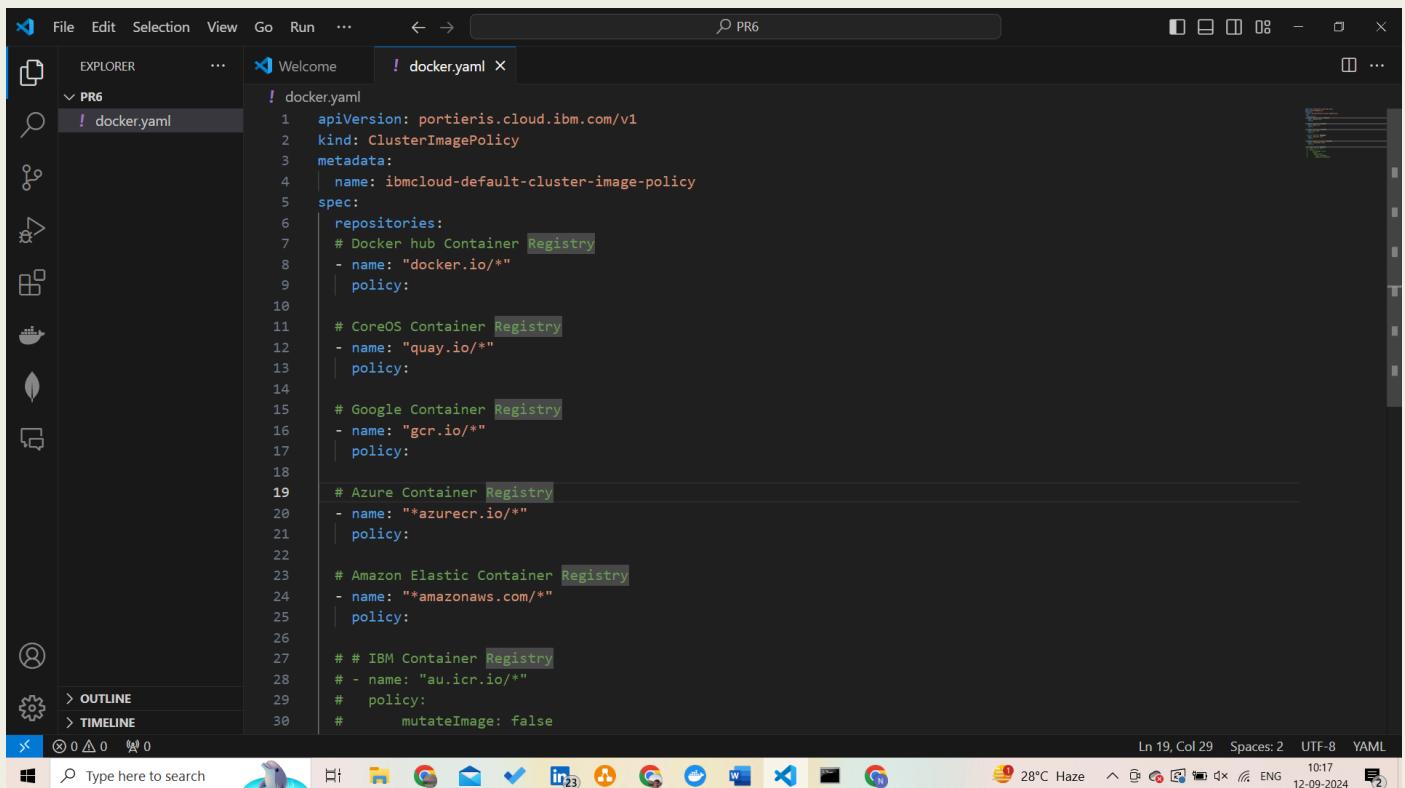
The status bar at the bottom shows the date as 12-09-2024 and the time as 09:54.

try to pull any library from docker hub

```
C:\Users\Kshitij>docker pull open-liberty
Using default tag: latest
Pull (1 of 1): open-liberty:latest@sha256:7b4b1c7943e71d919f27341ed125048cb7ee400d2adb7ef7f18c4ab1caf6d99a
docker.io/library/open-liberty@sha256:7b4b1c7943e71d919f27341ed125048cb7ee400d2adb7ef7f18c4ab1caf6d99a: Pulling from library/open-liberty
857cc8cb19c0: Downloading [=====] 10.26MB/29.54MB
b11acb1712e8: Downloading [=====] 10.33MB/12.16MB
8f3fe26e4ee6: Downloading [=====] 13.74MB/50.32MB
fcb9b1a29421: Waiting
69cd34ad566f: Waiting
011b7a49a438: Waiting
14d640cb051: Waiting
b5db331bf832: Waiting
c821542ebbee: Waiting
6bd77bbdad52f: Waiting
63cc890ac3e1: Waiting
c42008691933: Waiting
```



Now we Set the rule for images:



```
File Edit Selection View Go Run ... ⏴ ⏵ 🔍 PR6 docker.yaml

! docker.yaml
1  apiVersion: portieris.cloud.ibm.com/v1
2  kind: ClusterImagePolicy
3  metadata:
4    name: ibmcloud-default-cluster-image-policy
5  spec:
6    repositories:
7      # Docker hub Container Registry
8      - name: "docker.io/*"
9      | policy:
10
11      # CoreOS Container Registry
12      - name: "quay.io/*"
13      | policy:
14
15      # Google Container Registry
16      - name: "gcr.io/*"
17      | policy:
18
19      # Azure Container Registry
20      - name: "*azuredcr.io/*"
21      | policy:
22
23      # Amazon Elastic Container Registry
24      - name: "*amazonaws.com/*"
25      | policy:
26
27      # # IBM Container Registry
28      # - name: "au.icr.io/*"
29      #   policy:
30      #     mutateImage: false
```



Copy the image rule and go to custom recourse definition -> image police -> create image police(+ icon)

The screenshot shows the Kubernetes 'Create' dialog for a new resource. On the left, a sidebar lists various workload types like Cron Jobs, Daemon Sets, Deployments, etc. The main area has tabs for 'Create from input', 'Create from file', and 'Create from form'. A large text area contains the following YAML code:

```
4  name: portieris.imagepolicy
5  spec:
6    repositories:
7      # Docker hub Container Registry
8      - name: "docker.io/**"
9        policy:
10
11     # CoreOS Container Registry
12     - name: "quay.io/**"
13       policy:
14
15     # Google Container Registry
16     - name: "gcr.io/**"
17       policy:
18
19     # Azure Container Registry
20     - name: "azurecr.io/**"
21       policy:
22
23     # Amazon Elastic Container Registry
24     - name: "amazonaws.com/**"
25       policy:
26
```

Below the code are 'Upload' and 'Cancel' buttons. The status bar at the bottom shows system information: 28°C Haze, ENG, 10:22, 12-09-2024.

The screenshot shows the 'Custom Resource Definitions' page. The left sidebar includes 'Secrets', 'Storage Classes', 'Cluster' (with sub-options like Cluster Role Bindings, Cluster Roles, Events, Namespaces, Network Policies, Nodes, Persistent Volumes, Role Bindings, Roles, Service Accounts), 'Custom Resource Definitions' (which is selected and highlighted in blue), 'Settings', and 'About'. The main content area is titled 'Custom Resource Definitions' and displays a table of existing CRDs:

Name	Group	Full Name	Namespaced	Created	⋮
Image Policy	portieris.cloud.ibm.com	imagepolicies.portieris.cloud.ibm.com	True	3.minutes.ago	⋮
Cluster Image Policy	portieris.cloud.ibm.com	clusterimagepolicies.portieris.cloud.ibm.com	False	12.minutes.ago	⋮
Tigera Status	operator.tigera.io	tigerastatuses.operator.tigera.io	False	20.days.ago	⋮
Installation	operator.tigera.io	installations.operator.tigera.io	False	20.days.ago	⋮
Image Set	operator.tigera.io	imagesets.operator.tigera.io	False	20.days.ago	⋮
API Server	operator.tigera.io	apiservers.operator.tigera.io	False	20.days.ago	⋮
Network Set	crd.projectcalico.org	networksets.crd.projectcalico.org	True	20.days.ago	⋮
Network Policy	crd.projectcalico.org	networkpolicies.crd.projectcalico.org	True	20.days.ago	⋮
Kube Controllers Configuration	crd.projectcalico.org	kubecontrollersconfigurations.crd.projectcalico.org	False	20.days.ago	⋮
IP Reservation	crd.projectcalico.org	ipreservations.crd.projectcalico.org	False	20.days.ago	⋮

The status bar at the bottom shows system information: 29°C Haze, ENG, 10:43, 12-09-2024.

Inbox (922) - kshitijpgupta21@... | mycluster-syd01-u3c.2x4-group | mycluster-syd01-u3c.2x4-group | New Tab

au-syd.containers.cloud.ibm.com/kubeproxy/clusters/cr3cp7vs0jfn8bg998g/service/#/create?namespace=default

kubernetes default Search

Create

Workloads N

Cron Jobs

Daemon Sets

Deployments

Jobs

Pods

Replica Sets

Replication Controllers

Stateful Sets

Service

Ingresses N

Ingress Classes

Services N

Config and Storage

Config Maps N

Persistent Volume Claims N

Secrets N

Create from input Create from file Create from form

Enter YAML or JSON content specifying the resources to create to the currently selected namespace. [Learn more](#)

```
1 apiVersion: portieris.cloud.ibm.com/v1
2 kind: ClusterImagePolicy
3 metadata:
4   name: KSHITIJ-default-image-policy
5 spec:
6   repositories:
7     # Docker hub Container Registry
8     - name: "docker.io/*"
9     policy:
10    # CoreOS Container Registry
11    - name: "quay.io/*"
12    policy:
13    # Google Container Registry
14    - name: "gcr.io/*"
15    policy:
16    # Azure Container Registry
17    - name: "azurecr.io/*"
18    policy:
19    # Amazon Elastic Container Registry
20    - name: "amazoncr.io/*"
21    policy:
22    # Amazon ECR Container Registry
23
```

Upload Cancel

Type here to search 27°C Mostly sunny 09:45 23-09-2024

Inbox (922) - kshitijpgupta21@... | mycluster-syd01-u3c.2x4-group | mycluster-syd01-u3c.2x4-group | New Tab

au-syd.containers.cloud.ibm.com/kubeproxy/clusters/cr3cp7vs0jfn8bg998g/service/#/customresourcedefinitions/clusterimagepolicies.portieris.cloud.ibm.com?namespace=default

kubernetes default Search

Custom Resource Definitions > clusterimagepolicies.portieris.cloud.ibm.com

Workloads N

Cron Jobs

Daemon Sets

Deployments

Jobs

Pods

Replica Sets

Replication Controllers

Stateful Sets

Service

Ingresses N

Ingress Classes

Services N

Config and Storage

Config Maps N

Persistent Volume Claims N

Secrets N

Plural clusterimagepolicies Singular clusterimagepolicy Kind ClusterImagePolicy List Kind ClusterImagePolicyList

Objects

Name	Namespace	Created
kshitij-default-image-policy		45 seconds ago
default		10 days ago
ibmcloud-default-cluster-image-policy		10 days ago

Versions

Name	Served	Storage
v1	True	True

Conditions

Type here to search 27°C Mostly sunny 09:47 23-09-2024

Login into IBM account

The screenshot shows the Visual Studio Code interface. On the left is the Explorer sidebar with a project named 'PR4' containing files like 'app.js', 'Dockerfile', 'deployment.yaml', and 'Docker.docx'. The main editor area displays a 'deployment.yaml' file with the following content:

```
apiVersion: apps/v1
kind: Deployment
metadata:
  name: kshitij-deployment
spec:
  selector:
    matchLabels:
      app: node1-007
  replicas: 1
  template:
    metadata:
      labels:
        app: node1-007
    spec:
      containers:
        - name: nodecontainer
```

Below the editor is a terminal window showing the command 'ibmcloud login' being run in PowerShell. The output indicates successful authentication:

```
PS D:\SEM-7\CS\PR4> ibmcloud login -a https://cloud.ibm.com -u passcode -p EIPGGGF7K
API endpoint: https://cloud.ibm.com
Authenticating...
OK
```

The status bar at the bottom right shows the date and time as 23-09-2024 10:17.

Its give the error because I'm not in that particular folder

The screenshot shows the Visual Studio Code interface. The setup is identical to the previous one, with the 'deployment.yaml' file open in the editor. However, the terminal window now displays an error message:

```
PS D:\SEM-7\CS\PR4> ibmcloud ks cluster config --cluster cr3cp7vs0jfnd8bg998g
'congif' is not a registered command. Check your list of installed plug-ins. See 'C:\Program Files\IBM\Cloud\bin\ibmcloud.exe ks cluster help'.
```

This error occurs because the command 'congif' is misspelled as 'congif' in the terminal. The rest of the terminal output is correct, showing the configuration being updated successfully.

Command: `ibmcloud ks cluster config --cluster <C_ID>` for used to configure your IBM Cloud Kubernetes Service

This command give the error because we don't set the permission to access it or config it.

```

File Edit Selection View Go Run ...
File Explorer ... deployment.YAML
C:\Windows\System32\cmd.exe
Microsoft Windows [Version 10.0.19045.4780]
(c) Microsoft Corporation. All rights reserved.

D:\SEM-7\CS\PR4\Practical_4\project>ibmcloud ks cluster config --cluster cr3cp7vs0jfnd8bg998g
OK
The configuration for cr3cp7vs0jfnd8bg998 was downloaded successfully.

Added context for cr3cp7vs0jfnd8bg998g to the current kubeconfig file.
You can now execute 'kubectl' commands against your cluster. For example, run 'kubectl get nodes'.

D:\SEM-7\CS\PR4\Practical_4\project>kubectl apply -f deployment.YAML
Error from server: error when creating "deployment.YAML": admission webhook "trust.hooks.securityenforcement.admission.cloud.ibm.com" denied the request:
Deny "au.irc.io/kshitijname/new_img_007:1.0", no matching repositories in the ImagePolicies

D:\SEM-7\CS\PR4\Practical_4\project>

```

now in this permission set we set that we

```
apiVersion: portieris.cloud.ibm.com/v1
kind: ClusterImagePolicy
metadata:
  name: kshitij_pr6-default-image-policy
spec:
  repositories:
    # Docker hub Container Registry
    - name: "docker.io/*"
      policy:
        ...
    # CoreOS Container Registry
    - name: "quay.io/*"
      policy:
        ...
    # Google Container Registry
    - name: "gcr.io/*"
      policy:
        ...
    # Azure Container Registry
    - name: "*azuredcr.io/*"
      policy:
        ...
    # Amazon Elastic Container Registry
    - name: "au.icr.io/*"
      policy:
```

```
apiVersion: portieris.cloud.ibm.com/v1
kind: ClusterImagePolicy
metadata:
  name: kshitij_pr6-default-image-policy
spec:
  repositories:
    # Docker hub Container Registry
    - name: "docker.io/*"
      policy:
        ...
    # CoreOS Container Registry
    - name: "quay.io/*"
      policy:
        ...
    # Google Container Registry
    - name: "gcr.io/*"
      policy:
        ...
    # Azure Container Registry
    - name: "*azuredcr.io/*"
      policy:
        ...
    # Amazon Elastic Container Registry
    - name: "au.icr.io/*" highlighted
      policy:
```

Screenshot of a web browser showing the Kubernetes UI for Custom Resource Definitions (CRDs). The URL is au-syd.containers.cloud.ibm.com/kubeproxy/clusters/cr3cp7vs0jfnd8bg998g/service/#/customresourcedefinition/clusterimagepolicies.portieris.cloud.ibm.com?namespace=Experiment%206.

The left sidebar shows navigation links for Secrets, Storage Classes, Cluster (Cluster Role Bindings, Cluster Roles), Events, Namespaces, Network Policies, Nodes, Persistent Volumes, Role Bindings, Roles, Service Accounts, and Custom Resource Definitions.

The main content area displays three sections:

- Objects**: A table showing two entries under the 'Cluster' section:

Name	Namespace	Created
kshitijpr6-default-image-policy		2.minutes.ago
default		21.minutes.ago
kshitij-default-image-policy		40.minutes.ago
- Versions**: A table showing one entry:

Name	Served	Storage
v1	True	True
- Conditions**: A table showing two entries:

Type	Status	Last transition time	Reason	Message
NamesAccepted	True	10.days.ago	NoConflicts	no conflicts found
Established	True	10.days.ago	InitialNamesAccepted	the initial names have been accepted

now we again try deployment.yaml to deploy and its done can be done because we set that permission

Screenshot of a Windows Taskbar showing a command prompt window and the Kubernetes UI.

The command prompt window (C:\Windows\System32\cmd.exe) shows the output of running `ibmcloud ks cluster config --cluster cr3cp7vs0jfnd8bg998g`. It indicates that the configuration for `cr3cp7vs0jfnd8bg998g` was downloaded successfully.

The right side of the screen shows the Kubernetes UI for Custom Resource Definitions (CRDs). The URL is au-syd.containers.cloud.ibm.com/kubeproxy/clusters/cr3cp7vs0jfnd8bg998g/service/#/customresourcedefinition/clusterimagepolicies.portieris.cloud.ibm.com?namespace=Experiment%206.

The left sidebar shows navigation links for Secrets, Storage Classes, Cluster (Cluster Role Bindings, Cluster Roles), Events, Namespaces, Network Policies, Nodes, Persistent Volumes, Role Bindings, Roles, Service Accounts, and Custom Resource Definitions.

The main content area displays three sections:

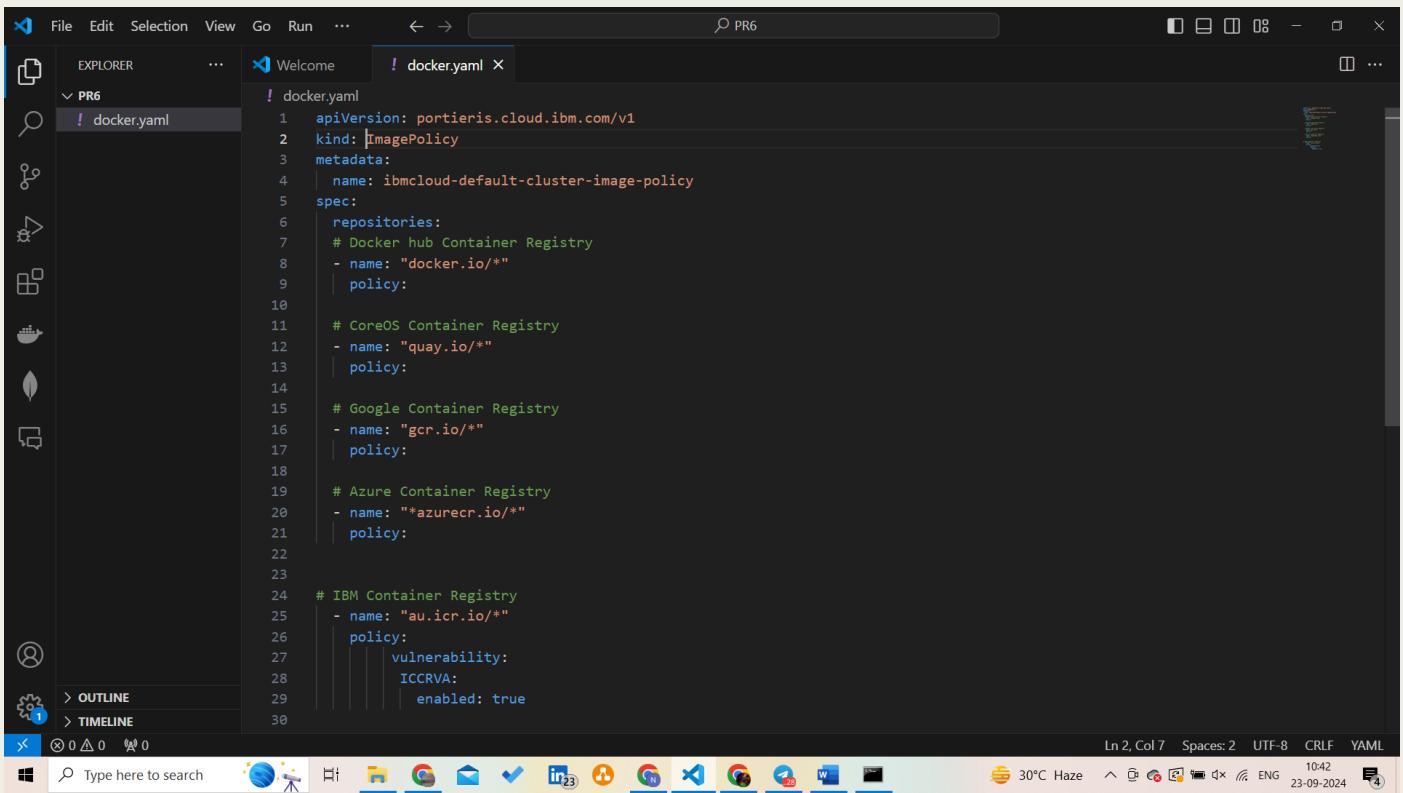
- Objects**: A table showing two entries under the 'Cluster' section:

Name	Namespace	Created
kshitijpr6-default-image-policy		3.minutes.ago
default		22.minutes.ago
kshitij-default-image-policy		41.minutes.ago
- Versions**: A table showing one entry:

Name	Served	Storage
v1	True	True
- Conditions**: A table showing two entries:

Type	Status	Last transition time	Reason	Message
NamesAccepted	True	10.days.ago	NoConflicts	no conflicts found
Established	True	10.days.ago	InitialNamesAccepted	the initial names have been accepted

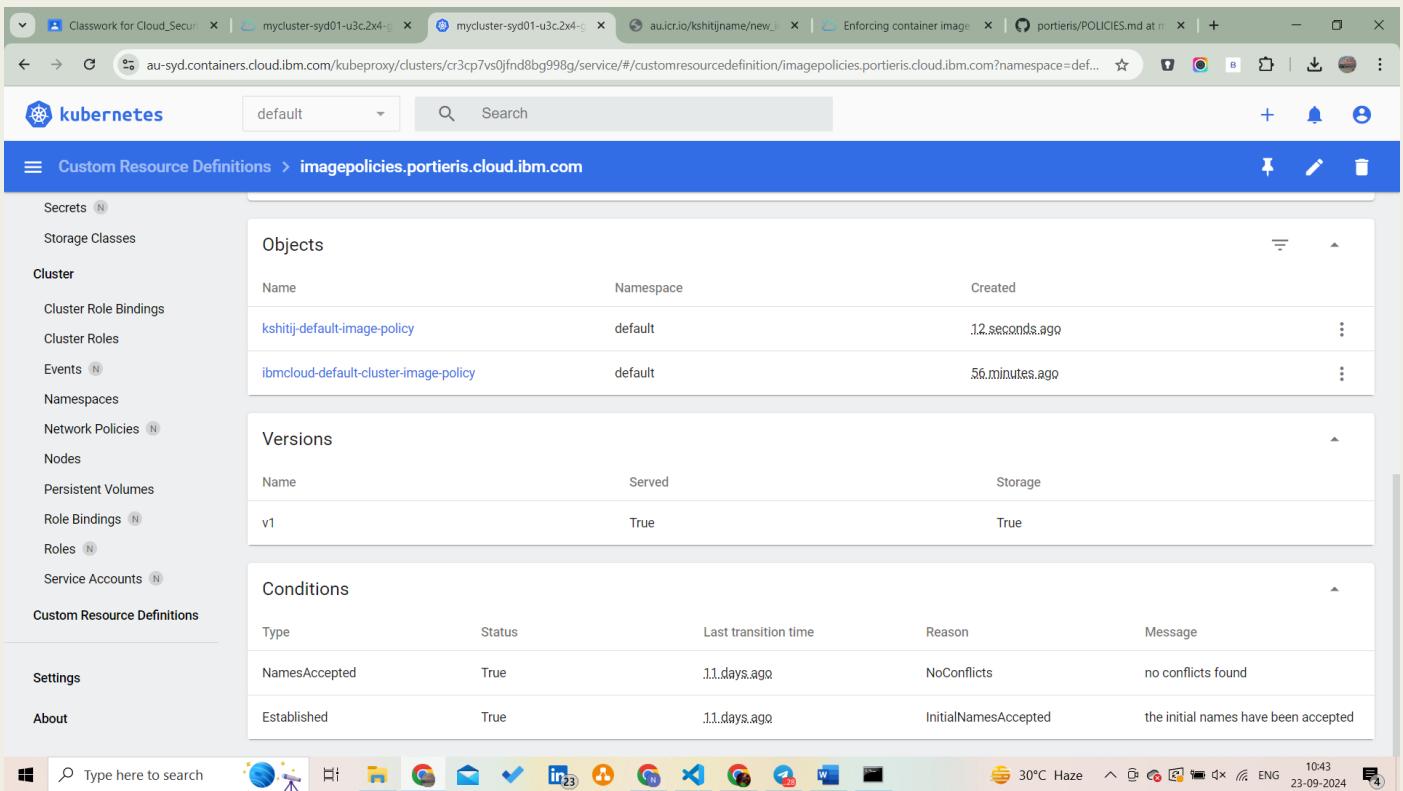
In this, we set vulnerability adviser to check vulnerability.



The screenshot shows the VS Code interface with the following details:

- File Explorer:** Shows a folder named "PR6" containing a file named "docker.yaml".
- Code Editor:** Displays the "docker.yaml" file content, which includes a policy section for vulnerability checking across various registries.
- Status Bar:** Shows "Ln 2, Col 7" and "YAML".
- Taskbar:** Shows the Windows taskbar with various pinned icons like File Explorer, Edge, and File Explorer.

```
apiVersion: portieris.cloud.ibm.com/v1
kind: ImagePolicy
metadata:
  name: ibmcloud-default-cluster-image-policy
spec:
  repositories:
    # Docker hub Container Registry
    - name: "docker.io/*"
    | policy:
      # CoreOS Container Registry
      - name: "quay.io/*"
      | policy:
        # Google Container Registry
        - name: "gcr.io/*"
        | policy:
          # Azure Container Registry
          - name: "*azurecr.io/*"
          | policy:
            # IBM Container Registry
            - name: "au.icr.io/*"
            | policy:
              vulnerability:
                ICCRVA:
                  enabled: true
```



The screenshot shows the Kubernetes UI (Dashboard) with the following details:

- Sidebar:** Shows navigation options like Secrets, Storage Classes, Cluster, Cluster Role Bindings, Cluster Roles, Events, Namespaces, Network Policies, Nodes, Persistent Volumes, Role Bindings, Roles, Service Accounts, and Custom Resource Definitions.
- Custom Resource Definitions:** Under "imagepolicies.portieris.cloud.ibm.com", it shows two objects:
 - kshitij-default-image-policy**: Namespace: default, Created: 12.seconds.ago
 - ibmcloud-default-cluster-image-policy**: Namespace: default, Created: 56.minutes.ago
- Versions:** Shows a single version v1.
- Conditions:** Shows two conditions:
 - NamesAccepted**: Status: True, Last transition time: 11.days.ago, Reason: NoConflicts, Message: no conflicts found
 - Established**: Status: True, Last transition time: 11.days.ago, Reason: InitialNamesAccepted, Message: the initial names have been accepted
- Status Bar:** Shows the Windows taskbar with various pinned icons.

The screenshot shows the Visual Studio Code interface with the following details:

- Explorer View:** Shows a project named "PR4" containing files like "app.js", "Dockerfile", "package-lock.json", "package.json", "service.YAML", and "Practical_4.rar".
- Code Editor:** An open file is titled "deployment.YAML". The code is as follows:

```
apiVersion: apps/v1
kind: Deployment
metadata:
  name: kshitijpr6-deployment
spec:
  selector:
    matchLabels:
      app: node1-007
  replicas: 1
  template:
    metadata:
      labels:
        app: node1-007
    spec:
      containers:
        - name: nodecontainer
```

- Terminal:** Displays the command "kubectl apply -f deployment.YAML" being run in a PowerShell window. The output shows:

```
The configuration for cr3cp7vs0jfn8bg998 was downloaded successfully.
Added context for cr3cp7vs0jfn8bg998 to the current kubeconfig file.
You can now execute 'kubectl' commands against your cluster. For example, run 'kubectl get nodes'.
PS D:\SEM-7\CS\PR4> kubectl apply -f deployment.YAML
error: the path "deployment.YAML" does not exist
PS D:\SEM-7\CS\PR4> kubectl apply -f deployment.YAML
* History restored
```

- Bottom Status Bar:** Shows the file path "PS D:\SEM-7\CS\PR4", line count "Ln 10, Col 12", spaces "Spaces: 2", encoding "UTF-8", and file type "CRLF YAML".

There is no vulnerability in over file so its giving no error.

