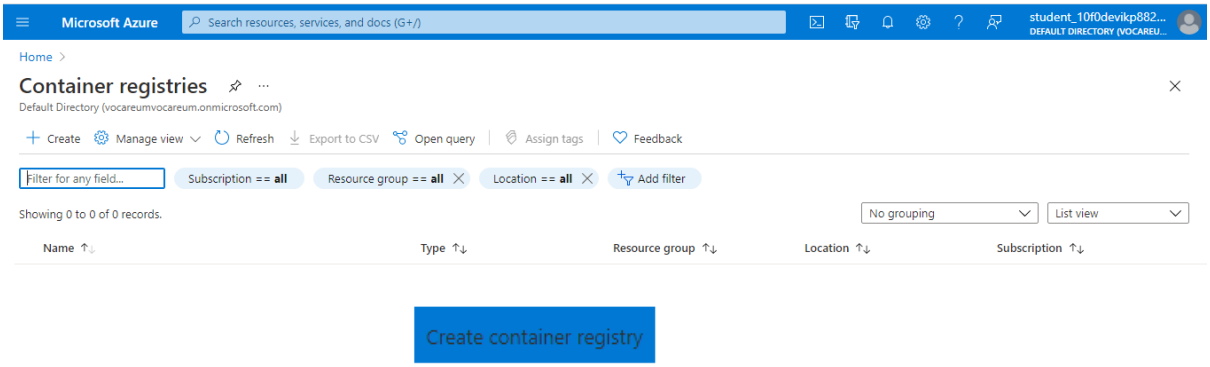
Contents

1 Azure container Registry and creation

Azure Container Registry allows you to build, store, and manage container images and artifacts in a private registry for all types of container deployments. Use Azure container registries with your existing container development and deployment pipelines. Use Azure Container Registry Tasks to build container images in Azure on-demand, or automate builds triggered by source code updates, updates to a container's base image, or timers. *Documentation* : https://docs.microsoft.com/en-us/azure/container-registry/

Create container Register



Home > Container registries >

Create container registry

Subscription * Production 1

Resource group * Regroup_3mku

Create new

Instance details

Registry name * SathiContRegDemo .azurecr.io

Location * West US

Availability zones Enabled

Availability zones are enabled on premium registries and in regions that support availability zones. Learn more

SKU * Basic

Review + create | < Previous | Next: Networking >

Validation passed

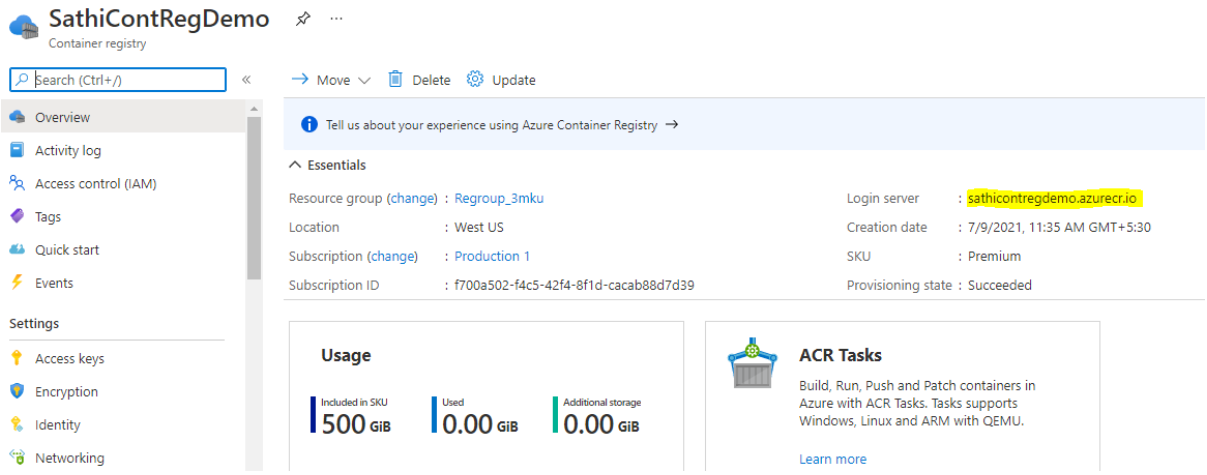
Basics | Networking | Encryption | Tags | Review + create

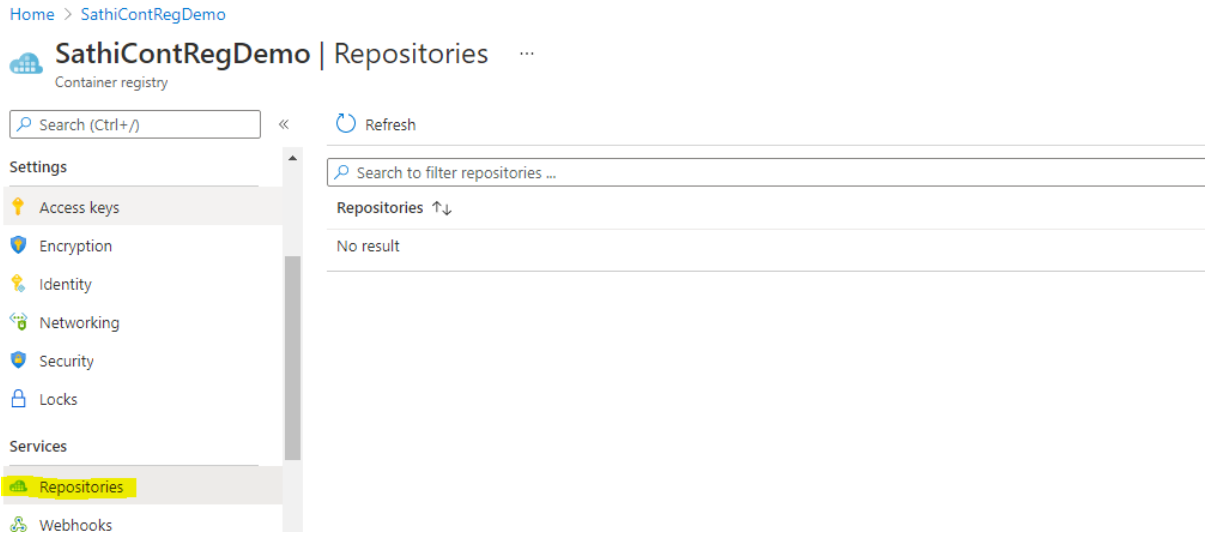
Registry details

Basics

Registry name	SathiContRegDemo
Subscription	Production 1
Resource Group	Regroup_3mku
Location	West US
Availability zones	Disabled
SKU	Premium

Networking

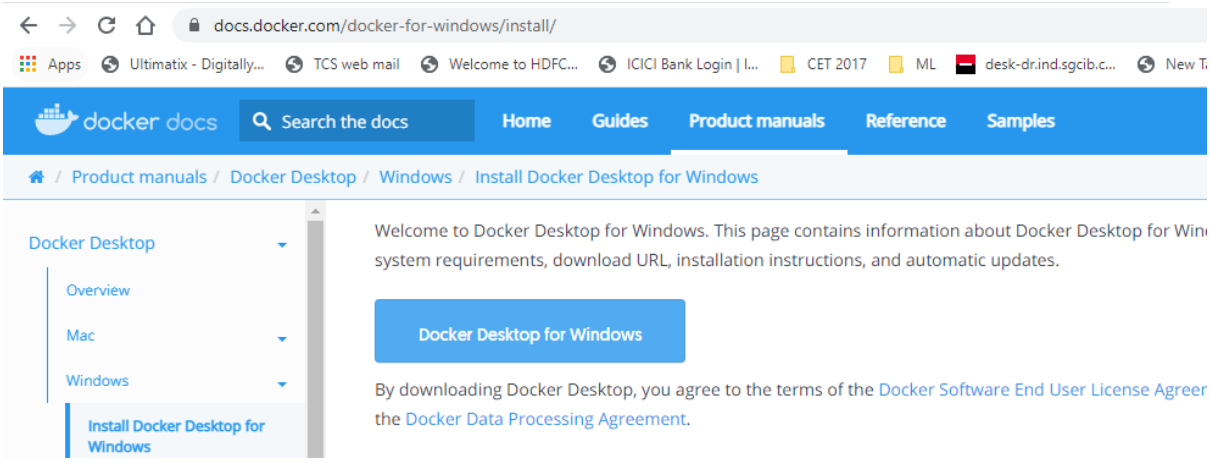




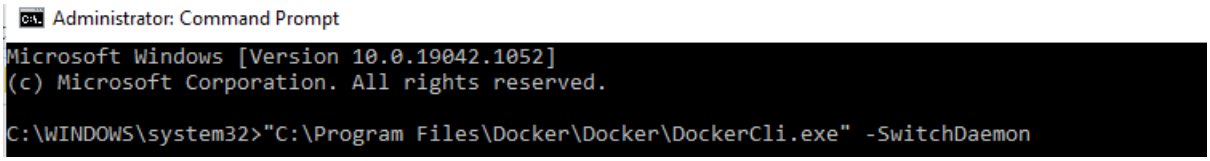
2 Docker setup on local machine

https://docs.microsoft.com/en-us/windows/wsl/install-win10
https://docs.docker.com/docker-for-windows/install/
https://docs.docker.com/get-started/

Download and Install wsl and docker

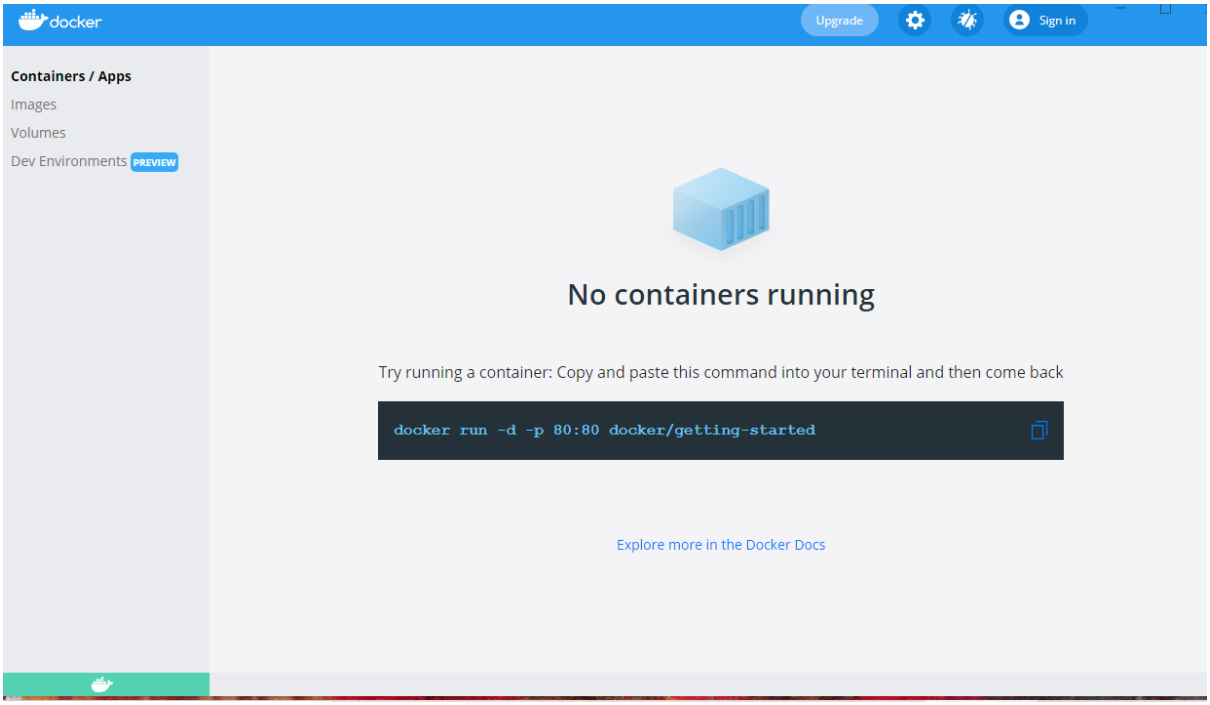


Run Daemon

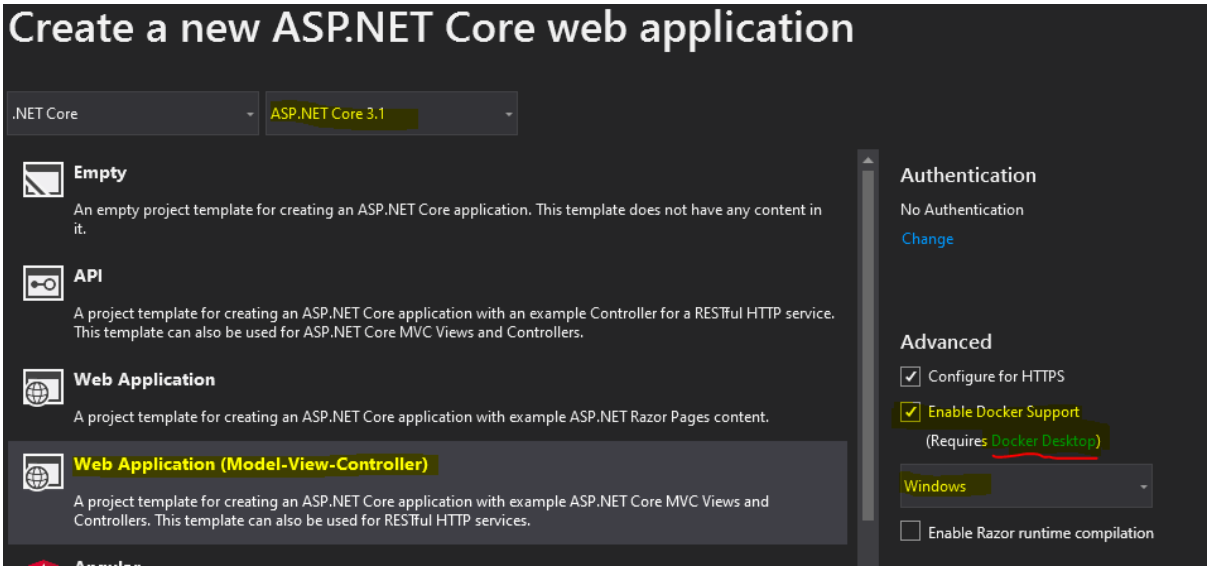


Check Version

Run –Green indicator shows its successsfully runing



3 Create sample .net core MVC app with docker file



```
Command Prompt

C:\Users\Smt.Narayanamma\source\repos\WebContainerDemo\WebContainerDemo>dotnet build
Microsoft (R) Build Engine version 16.8.0+126527ff1 for .NET
Copyright (C) Microsoft Corporation. All rights reserved.

  Determining projects to restore...
  All projects are up-to-date for restore.
  WebContainerDemo -> C:\Users\Smt.Narayanamma\source\repos\WebContainerDemo\WebContainerDemo\bin\Debug\netcoreapp3.1\WebContainerDemo.dll
  WebContainerDemo -> C:\Users\Smt.Narayanamma\source\repos\WebContainerDemo\WebContainerDemo\bin\Debug\netcoreapp3.1\WebContainerDemo.Views.dll

Build succeeded.
    0 Warning(s)
    0 Error(s)

Time Elapsed 00:00:03.38
```

DOCKER FILE

```
Dockerfile

 5
 6  FROM mcr.microsoft.com/dotnet/core/aspnet:3.1 AS base
 7  WORKDIR /app
 8  EXPOSE 80
 9  EXPOSE 443
10
11  FROM mcr.microsoft.com/dotnet/core/sdk:3.1 AS build
12  WORKDIR /src
13  COPY ["WebContainerDemo/WebContainerDemo.csproj", "WebContainerDemo/"]
14  RUN dotnet restore "WebContainerDemo/WebContainerDemo.csproj"
15  COPY . .
16  WORKDIR "/src/WebContainerDemo"
17  RUN dotnet build "WebContainerDemo.csproj" -c Release -o /app/build
18
19  FROM build AS publish
20  RUN dotnet publish "WebContainerDemo.csproj" -c Release -o /app/publish
21
22  FROM base AS final
23  WORKDIR /app
24  COPY --from=publish /app/publish .
25  ENTRYPOINT ["dotnet", "WebContainerDemo.dll"]
```

Solution Explorer

Search Solution Explorer (Ctrl+;)

Solution 'WebContainerDem
WebContainerDemo
Connected Services
Dependencies
Properties
wwwroot
Controllers
Models
Views
appsettings.json
Dockerfile
Program.cs
Startup.cs

Solution Explorer | Git Changes

Properties

3 Create and Push image to azure container registry

```
C:\Users\Smt.Narayanamma\source\repos\WebContainerDemo\WebContainerDemo>docker images
REPOSITORY   TAG       IMAGE ID   CREATED   SIZE
```

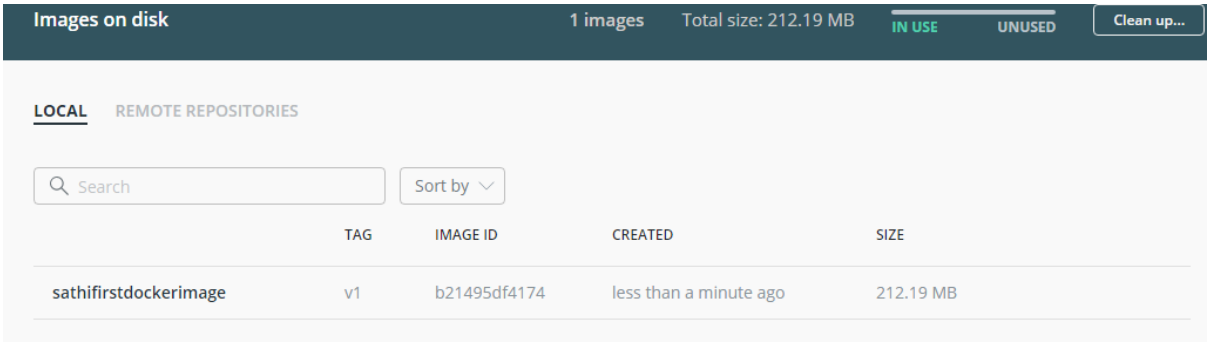
```
e\repos\WebContainerDemo>docker build . -t sathifirstdockerimage:v1 -f WebContainerDemo/DockerFile
```

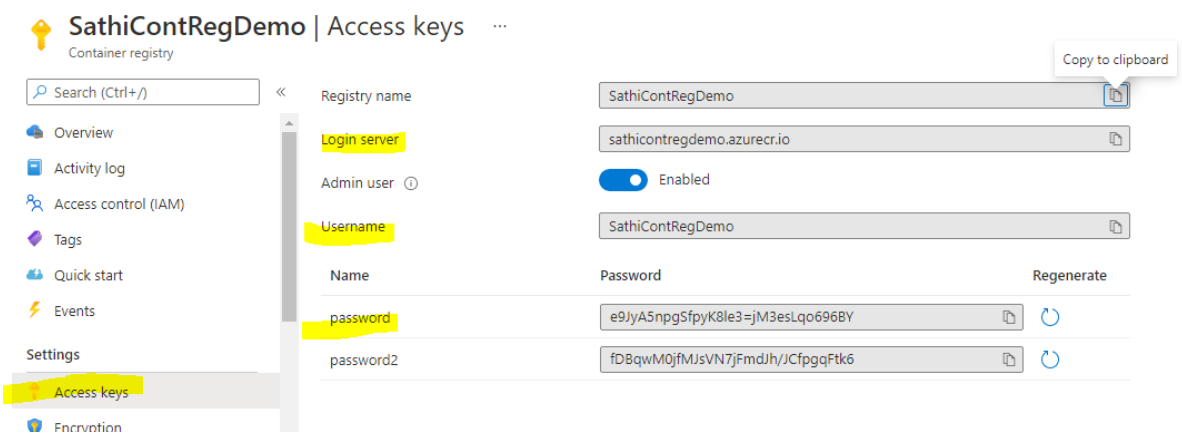
When docker file and solution are not in same folder , naviagte to solution level folder and specfiy docker file path eg: *-f WebContainerDemo/Dockerfile* , if not you wiill get cache key error . Also make sure that your docker is running and internet is up to download metadata from microsoft . *build.* says that source is from same folder *–t sathisfirstdockerimage* will be image name , v1 will be version tag. First time build should take good time as it will download .net run time(212MB) along with our project which is prerequest /dependency

```
=> [build 1/7] FROM mcr.microsoft.com/dotnet/core/sdk:3.1@sha256:93683294b914c3370bf7871d9397812452f4f05af7bfec854af149b36366e31c
```

```
=> exporting to image
=> => exporting layers
=> => writing image sha256:b21495df4174758b52ebfd8c9a17331c1f3fdd48dd036370ba704bde13829d09
=> => naming to docker.io/library/sathifirstdockerimage:v1
```

```
C:\Users\Smt.Narayanamma\source\repos\WebContainerDemo>docker images
REPOSITORY              TAG       IMAGE ID       CREATED         SIZE
sathifirstdockerimage   v1        b21495df4174   2 minutes ago   212MB
```





docker login –-username SathiContRegDemo sathicontregdemo.azurecr.io

```
C:\Users\Smt.Narayanamma\source\repos\WebContainerDemo>docker login --username SathiContRegDemo sathicontregdemo.azurecr.io
Password:
Error response from daemon: Get https://sathicontregdemo.azurecr.io/v2/: unauthorized: authentication required, visit https://aka.ms/acr/authorization for more informa
tion.

C:\Users\Smt.Narayanamma\source\repos\WebContainerDemo>docker login --username SathiContRegDemo sathicontregdemo.azurecr.io --password e9JyA5npgSfpyK8le3=jM3esLqo696BY

WARNING! Using --password via the CLI is insecure. Use --password-stdin.
Login Succeeded

C:\Users\Smt.Narayanamma\source\repos\WebContainerDemo>
```

Tag the image b/n local and remote location

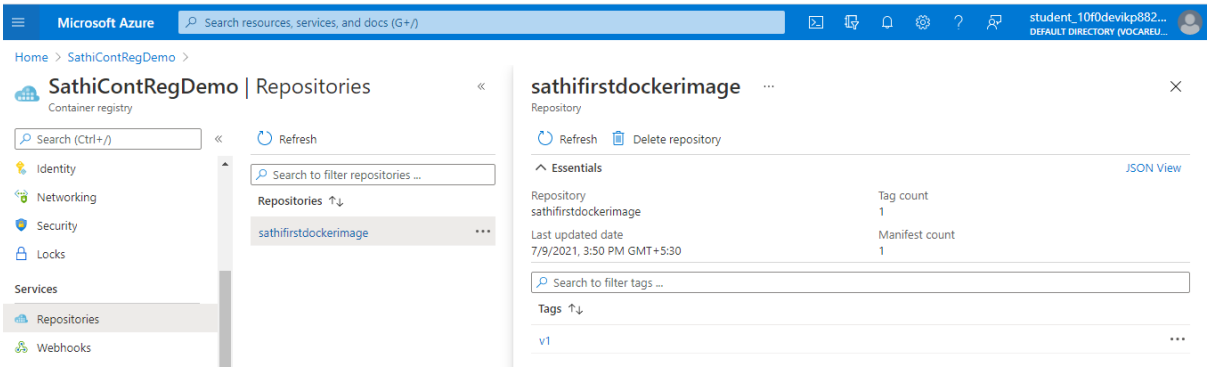
```
C:\Users\Smt.Narayanamma\source\repos\WebContainerDemo>docker images
REPOSITORY              TAG       IMAGE ID       CREATED          SIZE
sathifirstdockerimage   v1        b21495df4174   27 minutes ago   212MB
```

docker tag sathifirstdockerimage:v1 sathicontregdemo.azurecr.io/sathifirstdockerimage:v1

```
C:\Users\Smt.Narayanamma\source\repos\WebContainerDemo>docker images
REPOSITORY              TAG       IMAGE ID       CREATED         SIZE
sathifirstdockerimage   v1        b21495df4174   30 minutes ago   212MB

C:\Users\Smt.Narayanamma\source\repos\WebContainerDemo>docker tag sathifirstdockerimage:v1 sathicontregdemo.azurecr.io/sathifirstdockerimage:v1

C:\Users\Smt.Narayanamma\source\repos\WebContainerDemo>docker images
REPOSITORY                                      TAG       IMAGE ID       CREATED         SIZE
sathifirstdockerimage                           v1        b21495df4174   30 minutes ago   212MB
sathicontregdemo.azurecr.io/sathifirstdockerimage   v1        b21495df4174   30 minutes ago   212MB
```

docker push sathicontregdemo.azurecr.io/sathifirstdockerimage:v1

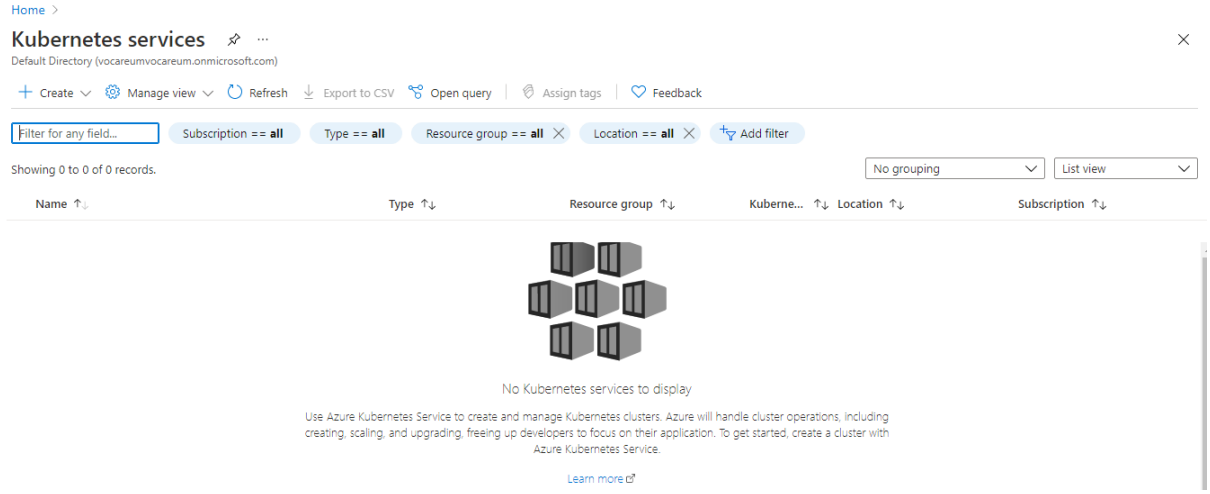
```
C:\Users\Smt.Narayanamma\source\repos\WebContainerDemo>docker push sathicontregdemo.azurecr.io/sathifirstdockerimage:v1
The push refers to repository [sathicontregdemo.azurecr.io/sathifirstdockerimage]
53da8099f52b: Pushed
5f70bf18a086: Pushed
d91ccedca273: Pushed
540b35ee11b3: Pushed
b2c669501e5b: Pushed
11d081e2b7c0: Pushed
4987333e5cc2: Pushed
764055ebc9a7: Pushed
v1: digest: sha256:95f2ecc543ce4652a993ba0febb4986fca949c701c2c42b4a03697085182c71b size: 1999

C:\Users\Smt.Narayanamma\source\repos\WebContainerDemo>
```

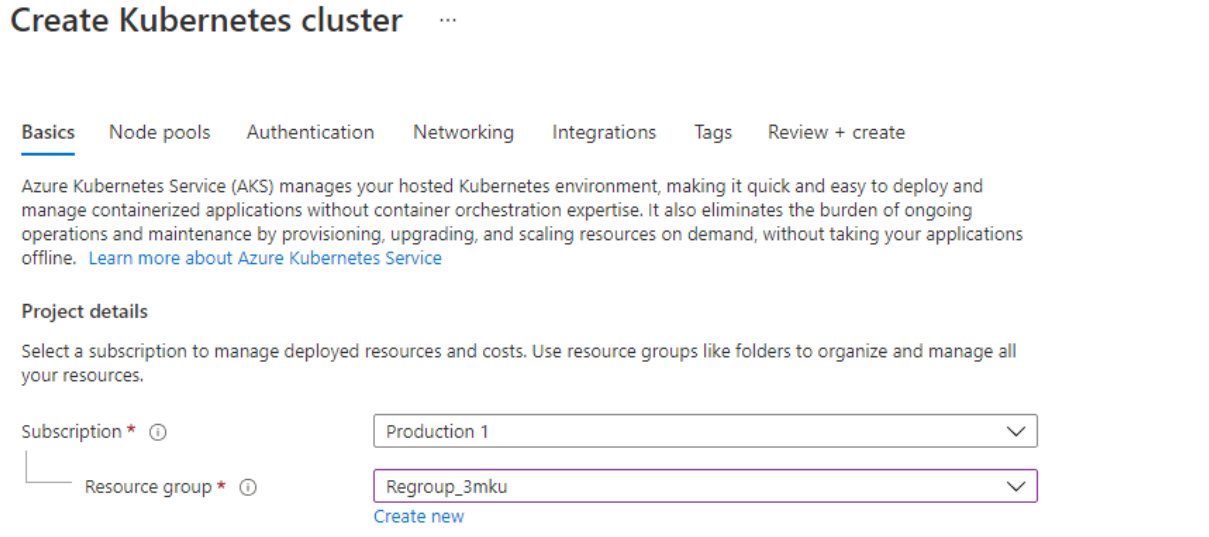


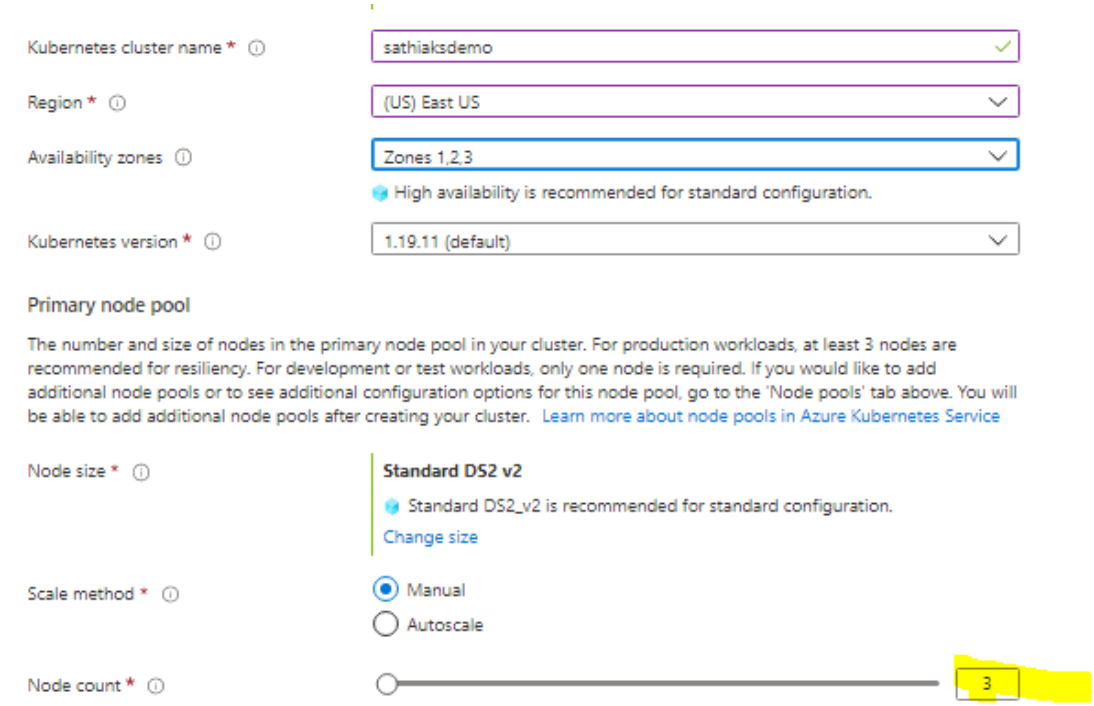
4 AKS – create and deploy ACR image to cluster/pod

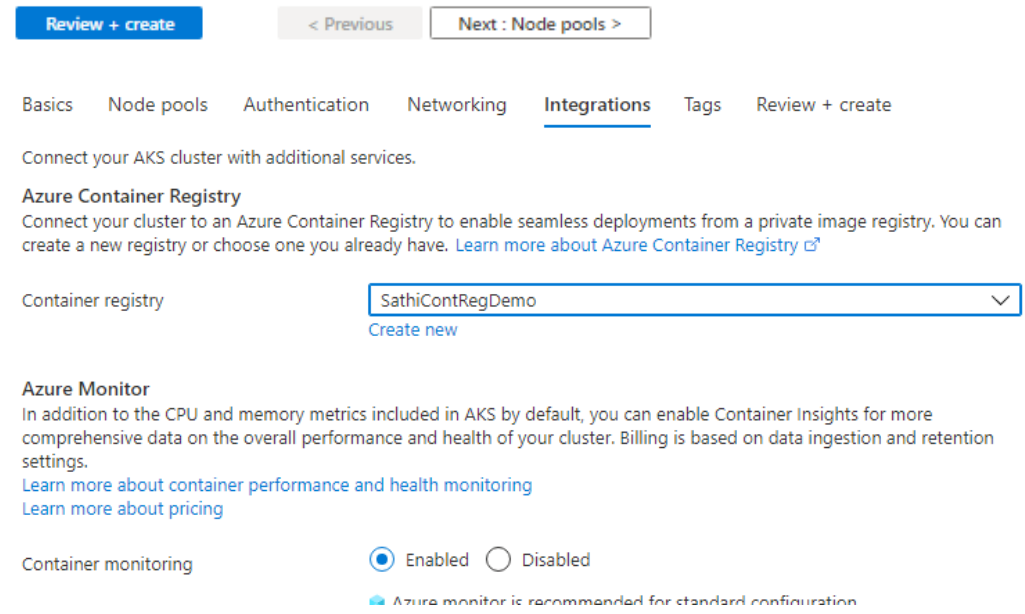
Azure Kubernetes Service (AKS) is a managed Kubernetes service that lets you quickly deploy and manage clusters:

Kubernetes Service.

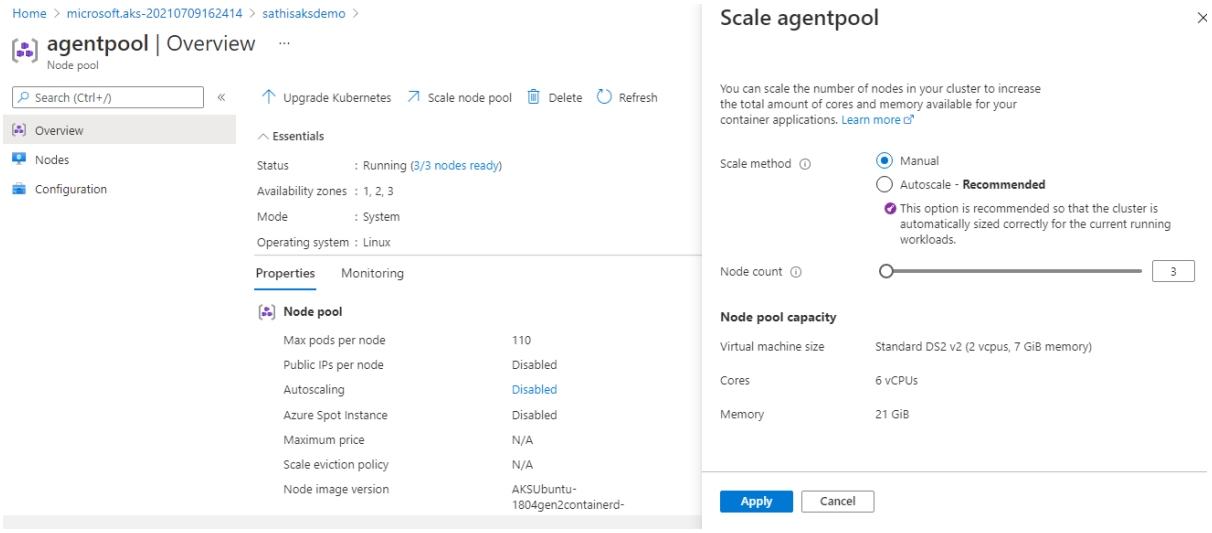








Disable above container moitoring as it not working

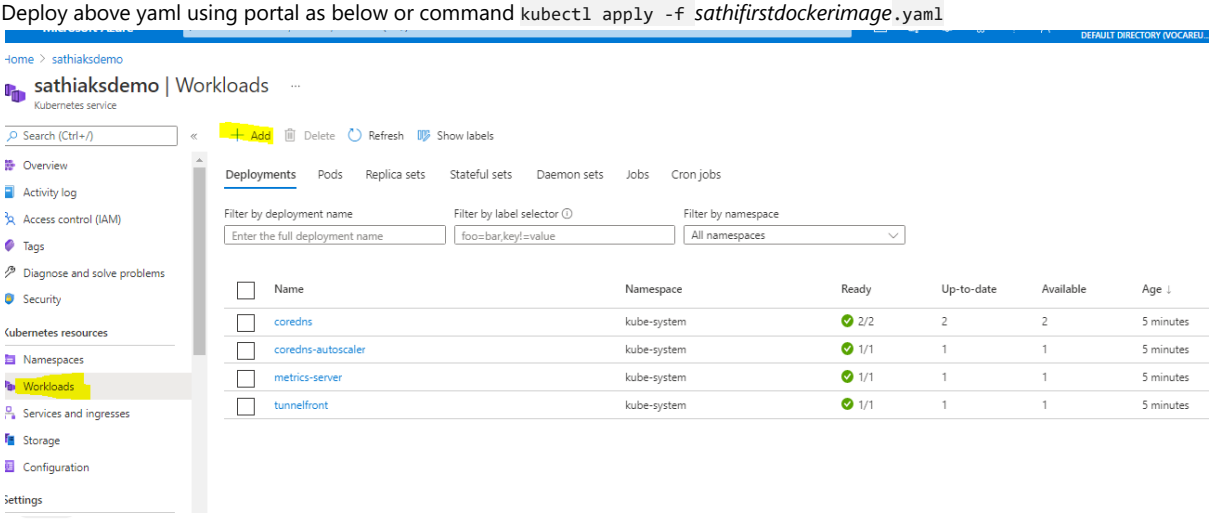


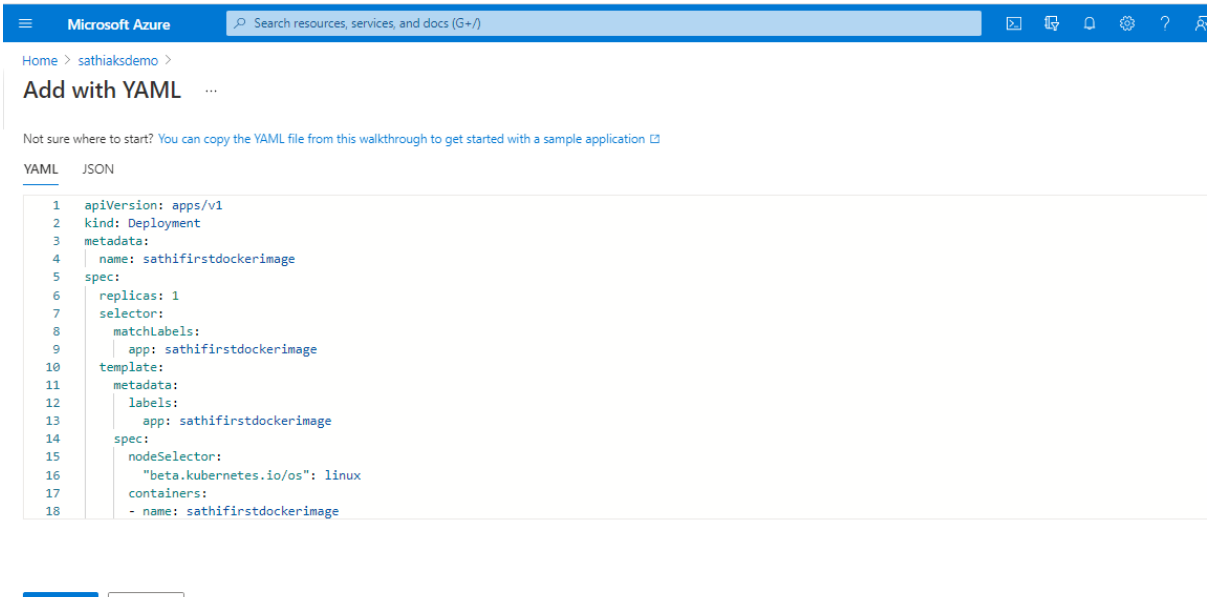
Create Yaml File : https://docs.microsoft.com/en-us/azure/aks/kubernetes-walkthrough-portal#run-the-application

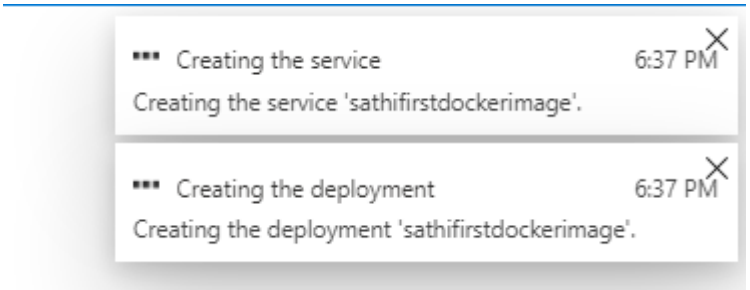
```
apiVersion: apps/v1
kind: Deployment
metadata:
  name: sathifirstdockerimage
spec:
  replicas: 1
  selector:
    matchLabels:
      app: sathifirstdockerimage
  template:
    metadata:
      labels:
        app: sathifirstdockerimage
    spec:
      nodeSelector:
        "beta.kubernetes.io/os": linux
      containers:
      - name: sathifirstdockerimage
        image: sathicontregdemo.azurecr.io/sathifirstdockerimage:v1
        resources:
          requests:
            cpu: 100m
            memory: 128Mi
          limits:
            cpu: 250m
            memory: 256Mi
        ports:
        - containerPort: 80
        env:
        - name: REDIS
          value: "sathifirstdockerimage"
---
apiVersion: v1
kind: Service
metadata:
  name: sathifirstdockerimage
spec:
  type: LoadBalancer
  ports:
  - port: 80
  selector:
    app: sathifirstdockerimage
```

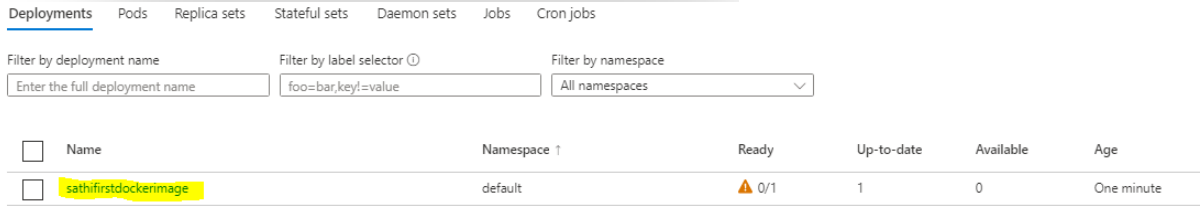
https://docs.microsoft.com/en-us/azure/aks/tutorial-kubernetes-prepare-app

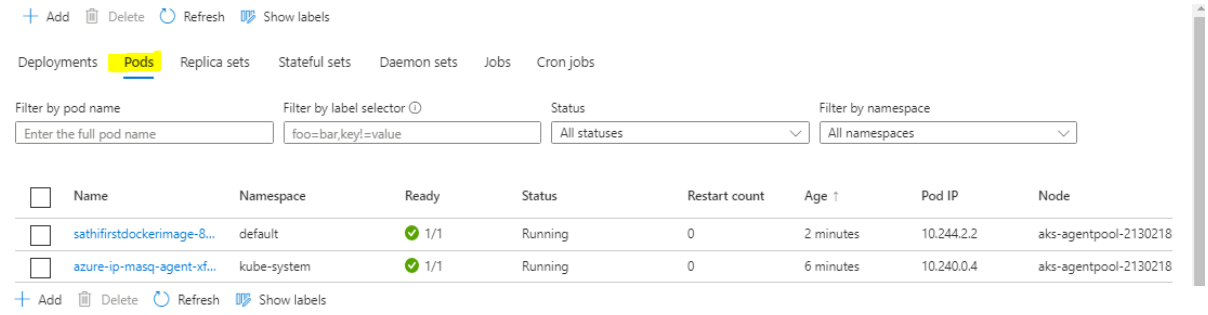
Deploy above yaml using portal as below or command `kubectl apply -f sathifirstdockerimage.yaml`

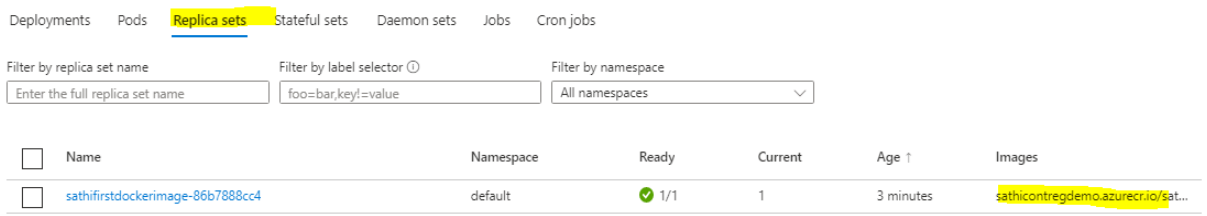


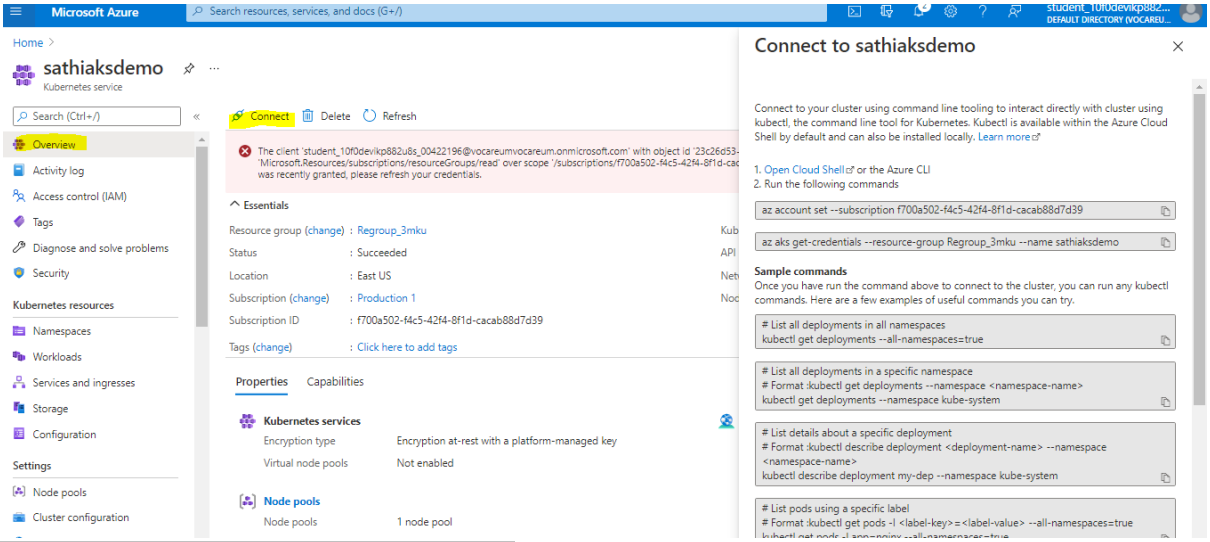










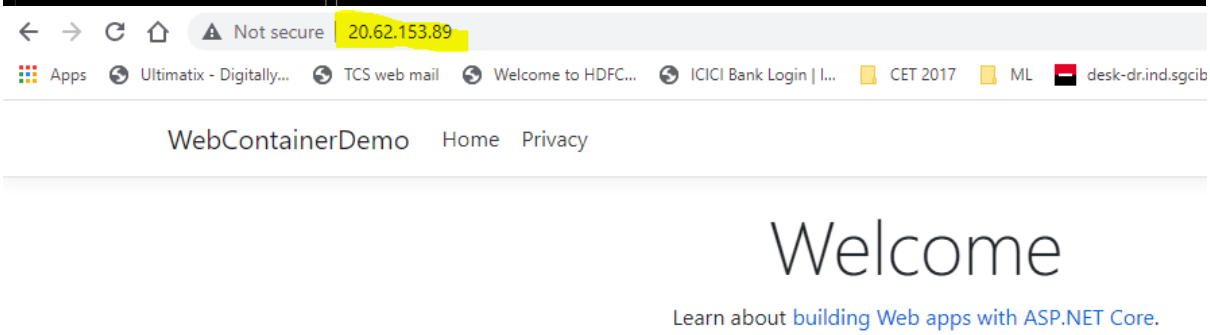


Connect To Azure CLI – (Enter required storage details)

```
student_10f0devikp882u8s@Azure:~$ az account set --subscription f700a502-f4c5-42f4-8f1d-cacab88d7d39
student_10f0devikp882u8s@Azure:~$ az aks get-credentials --resource-group Regroup_3mku --name sathiaksdemo
Merged "sathiaksdemo" as current context in /home/student_10f0devikp882u8s/.kube/config
student_10f0devikp882u8s@Azure:~$ # List all deployments in all namespaces
student_10f0devikp882u8s@Azure:~$ kubectl get deployments --all-namespaces=true
NAMESPACE     NAME                   READY   UP-TO-DATE   AVAILABLE   AGE
default       sathifirstdockerimage  1/1     1            1           13m
kube-system   coredns                2/2     2            2           19m
kube-system   coredns-autoscaler     1/1     1            1           19m
kube-system   metrics-server         1/1     1            1           19m
kube-system   tunnelfront            1/1     1            1           19m
student_10f0devikp882u8s@Azure:~$
```

`kubectl get service` *sathifirstdockerimage* `--watch`

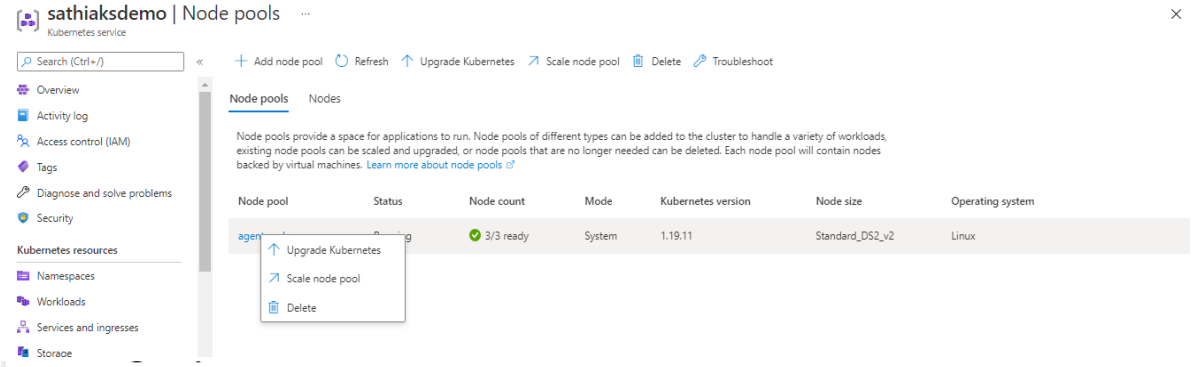
```
student_10f0devikp882u8s@Azure:~$ kubectl get service sathifirstdockerimage --watch
NAME                   TYPE           CLUSTER-IP     EXTERNAL-IP     PORT(S)          AGE
sathifirstdockerimage  LoadBalancer   10.0.93.211    20.62.153.89    80:31339/TCP     15m
```

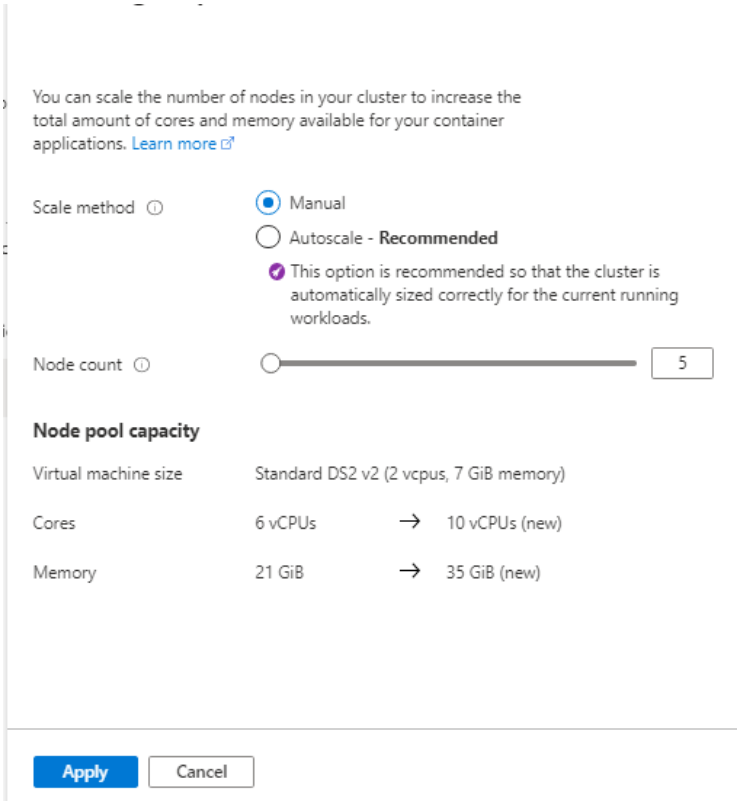


5 POD : SCALING

A pod is the smallest unit in Kubernetes that you create or deploy. A pod represents the application as a running process on your cluster.

Manually scale the number of Kubernetes pods from 3 to 5

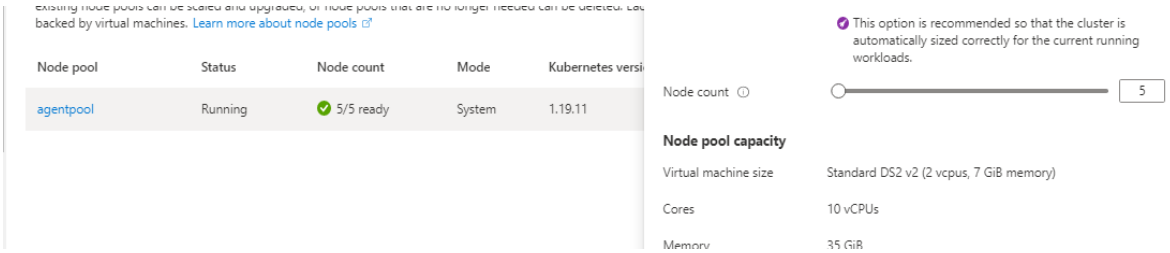




Do not apply above changes, we will apply via command

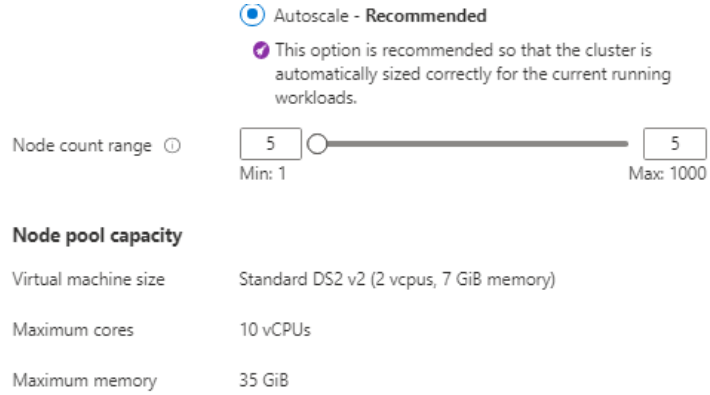
az aks scale --resource-group Regroup_3mku --name sathiaksdemo --node-count 5

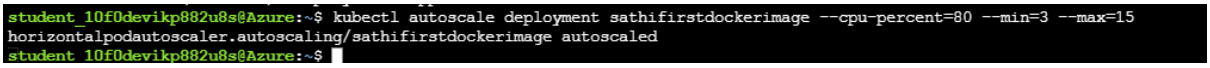
```
student_10f0devikp882u8s@Azure:~$ az aks scale --resource-group Regroup_3mku --name sathiaksdemo --node-count 5
{
  "aadProfile": null,
  "addonProfiles": {
    "azurepolicy": {
```



Create an autoscaling policy where the number of nodes is increased when the CPU utilization of the cluster exceeds 80% of their capacity up to a maximum of 15 pods
Console command :

```
kubectl autoscale deployment sathifirstdockerimage --cpu-percent=80 --min=3 --max=15
```

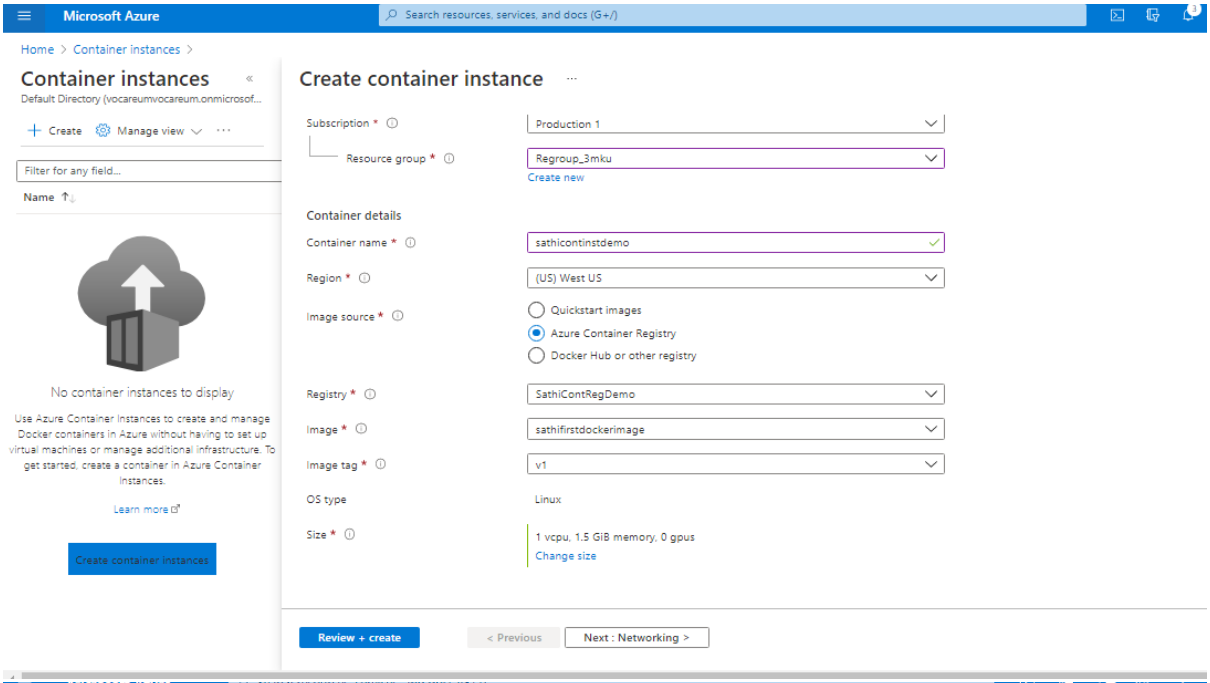


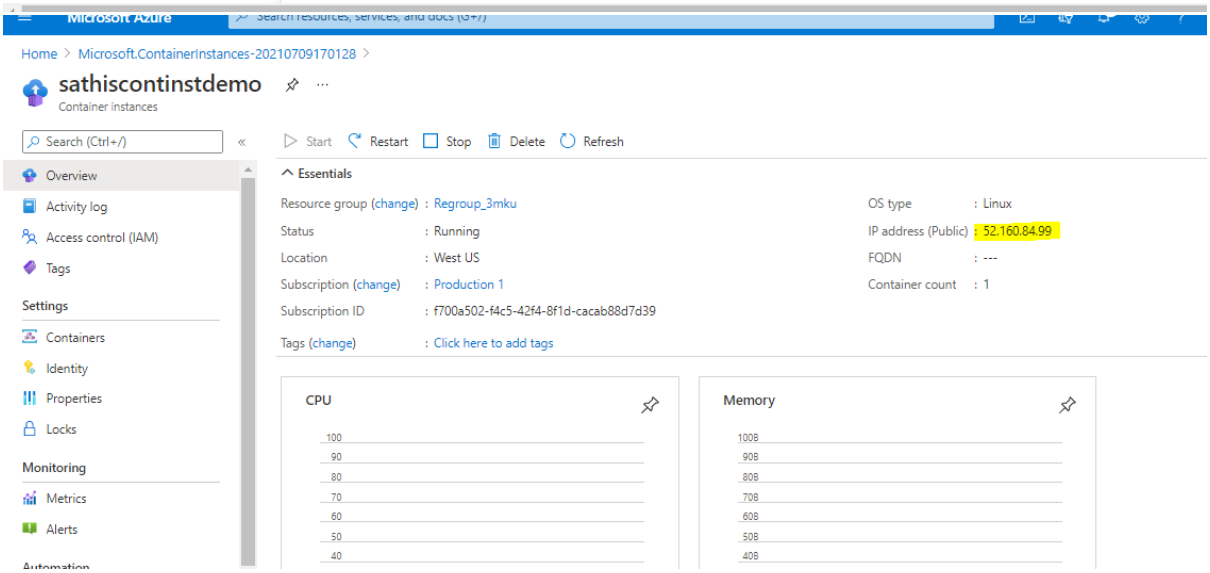


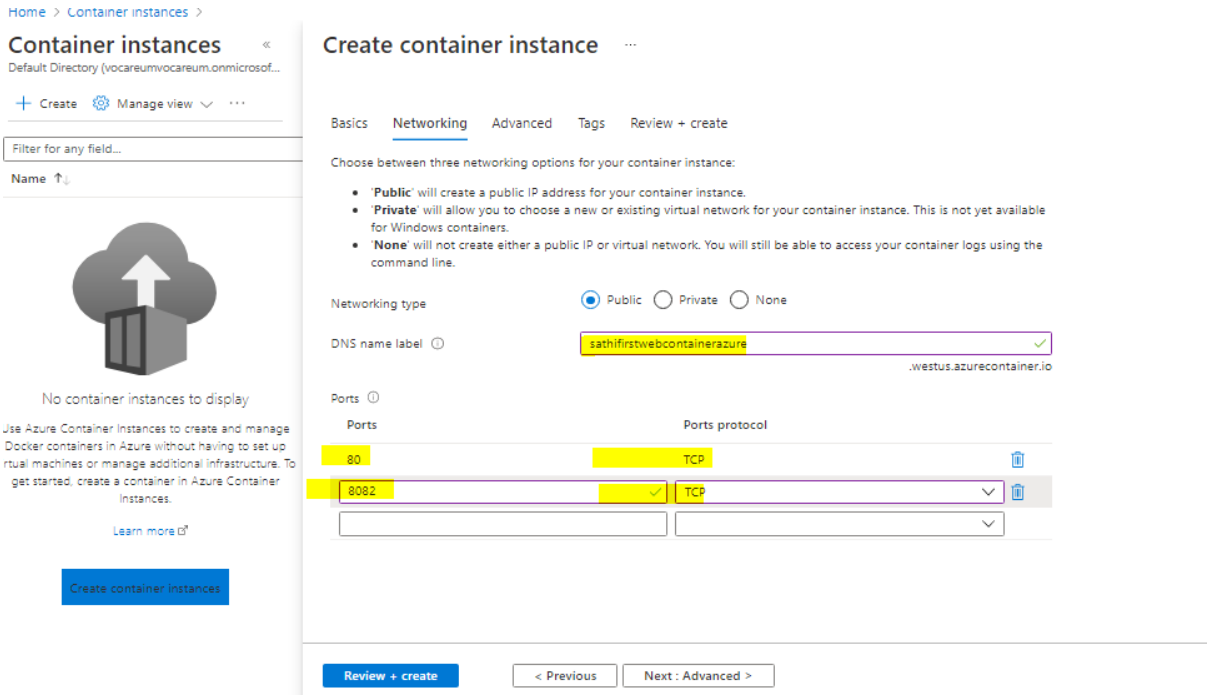
`Opening the files, editing and re deploying`
https://docs.microsoft.com/en-us/azure/aks/tutorial-kubernetes-app-update?tabs=azure-cli

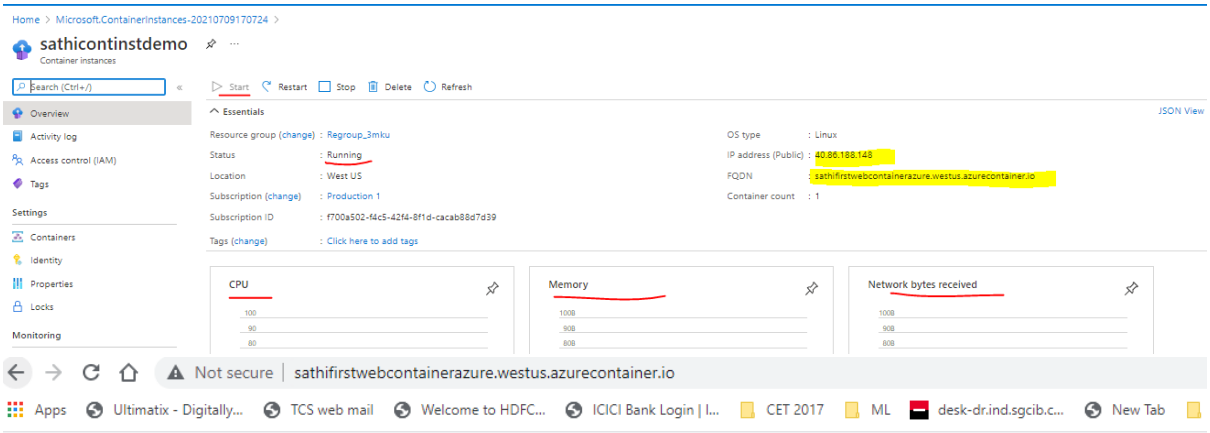
Additional Information

Hosting image in container instance

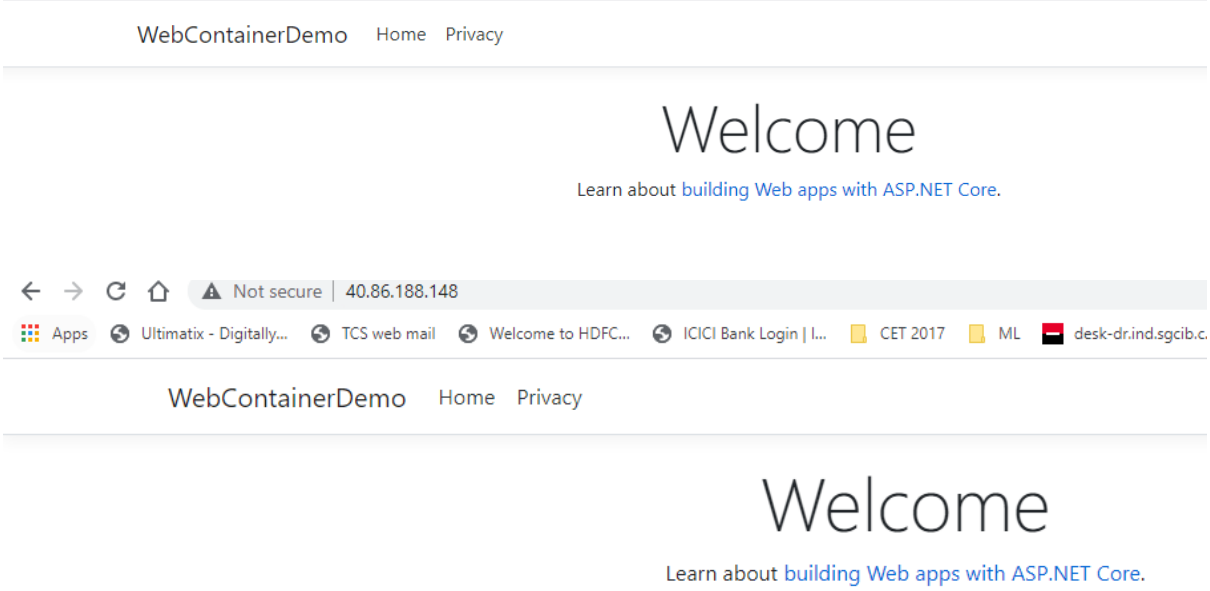

Microsoft Azure
Search resources, services, and docs (G+/)
Home > Container instances >
Container instances
Default Directory (vocareumvocareum.onmicrosof...
Create
Manage view
Filter for any field...
Name
No container instances to display
Use Azure Container Instances to create and manage Docker containers in Azure without having to set up virtual machines or manage additional infrastructure. To get started, create a container in Azure Container Instances.
Learn more
Create container instances
Create container instance
Subscription * Production 1
Resource group * Regroup_3mku
Create new
Container details
Container name * sathicontinstdemo
Region * (US) West US
Image source * Quickstart images
Azure Container Registry
Docker Hub or other registry
Registry * SathiContRegDemo
Image * sathifirstdockerimage
Image tag * v1
OS type Linux
Size * 1 vcpu, 1.5 GiB memory, 0 gpus
Change size
Review + create
< Previous
Next : Networking >


Home > Microsoft.ContainerInstances-20210709170128 >
sathiscontinstdemo
Container instances
Search (Ctrl+/)
Overview
Activity log
Access control (IAM)
Tags
Settings
Containers
Identity
Properties
Locks
Monitoring
Metrics
Alerts
Start
Restart
Stop
Delete
Refresh
Essentials
Resource group (change) : Regroup_3mku
Status : Running
Location : West US
Subscription (change) : Production 1
Subscription ID : f700a502-f4c5-42f4-8f1d-cacab88d7d39
Tags (change) : Click here to add tags
OS type : Linux
IP address (Public) : 52.160.84.99
FQDN : ---
Container count : 1
CPU
Memory


Home > Container instances >
Container instances
Default Directory (vocareumvocareum.onmicrosof...
Create
Manage view
Filter for any field...
Name
No container instances to display
Use Azure Container Instances to create and manage Docker containers in Azure without having to set up virtual machines or manage additional infrastructure. To get started, create a container in Azure Container Instances.
Learn more
Create container instances
Create container instance
Basics
Networking
Advanced
Tags
Review + create
Choose between three networking options for your container instance:
'Public' will create a public IP address for your container instance.
'Private' will allow you to choose a new or existing virtual network for your container instance. This is not yet available for Windows containers.
'None' will not create either a public IP or virtual network. You will still be able to access your container logs using the command line.
Networking type
Public
Private
None
DNS name label
sathifirstwebcontainerazure
.westus.azurecontainer.io
Ports
Ports
Ports protocol
80
TCP
8082
TCP
Review + create
< Previous
Next : Advanced >


Home > Microsoft.ContainerInstances-20210709170724 >
sathicontinstdemo
Container instances
Search (Ctrl+/)
Overview
Activity log
Access control (IAM)
Tags
Settings
Containers
Identity
Properties
Locks
Monitoring
Start
Restart
Stop
Delete
Refresh
Essentials
JSON View
Resource group (change) : Regroup_3mku
Status : Running
Location : West US
Subscription (change) : Production 1
Subscription ID : f700a502-f4c5-42f4-8f1d-cacab88d7d39
Tags (change) : Click here to add tags
OS type : Linux
IP address (Public) : 40.86.188.148
FQDN : sathifirstwebcontainerazure.westus.azurecontainer.io
Container count : 1
CPU
Memory
Network bytes received

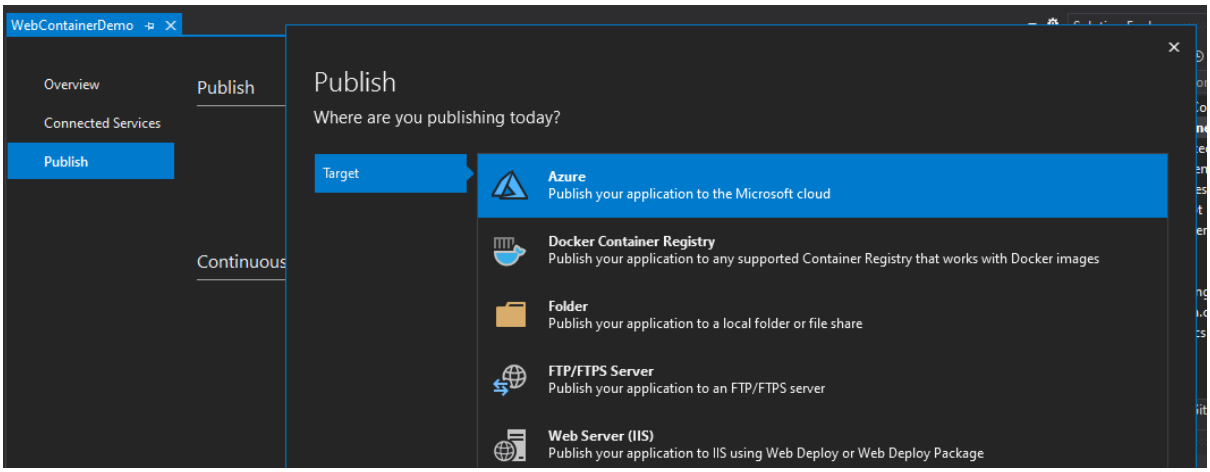
Not secure
sathifirstwebcontainerazure.westus.azurecontainer.io
Apps
Ultimatix - Digitally...
TCS web mail
Welcome to HDFC...
ICICI Bank Login | I...
CET 2017
ML
desk-dr.ind.sgcib.c...
New Tab

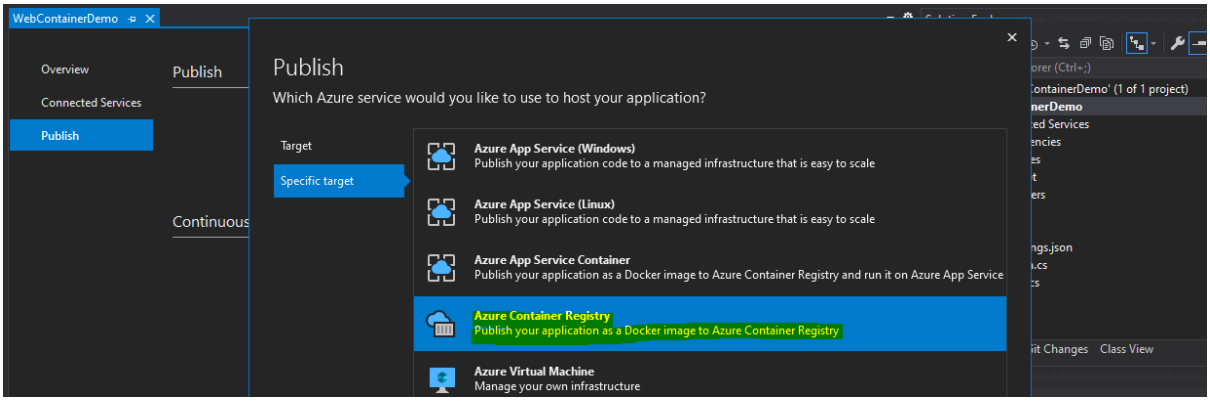

WebContainerDemo
Home
Privacy
Welcome
Learn about building Web apps with ASP.NET Core.

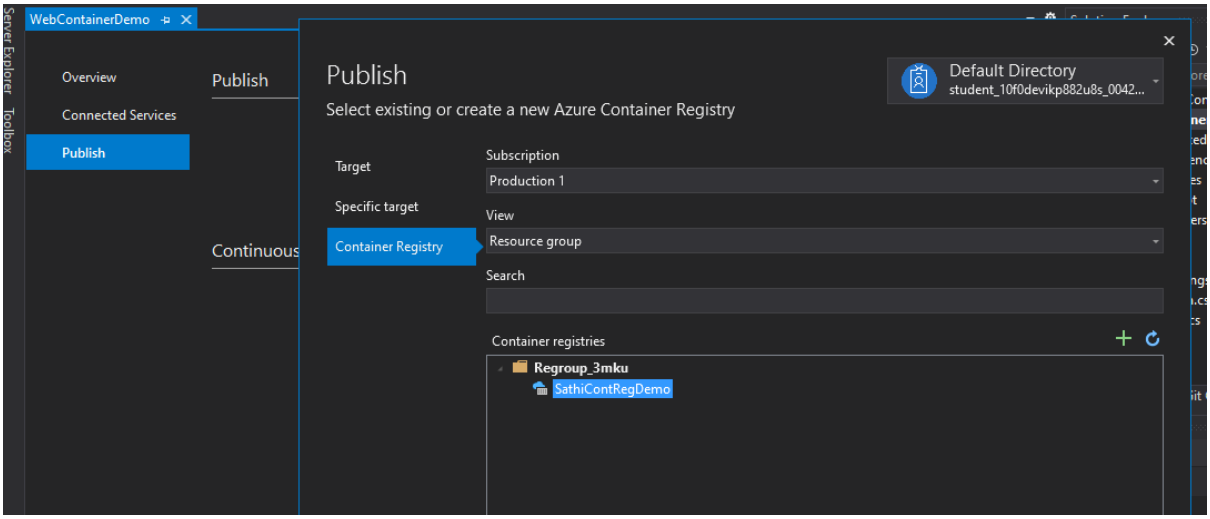
Not secure
40.86.188.148
Apps
Ultimatix - Digitally...
TCS web mail
Welcome to HDFC...
ICICI Bank Login | I...
CET 2017
ML
desk-dr.ind.sgcib.c...

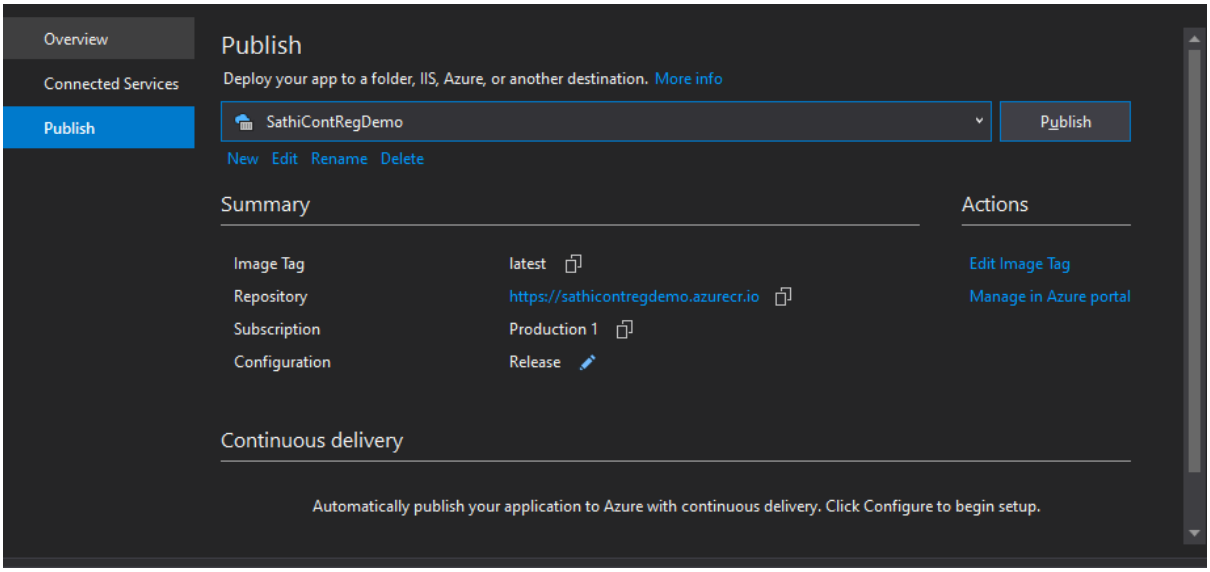
WebContainerDemo
Home
Privacy
Welcome
Learn about building Web apps with ASP.NET Core.

Using Visual Studio:








Overview
Connected Services
Publish
Publish
Deploy your app to a folder, IIS, Azure, or another destination. More info
SathiContRegDemo
Publish
New Edit Rename Delete
Summary
Actions
Image Tag
latest
Repository
https://sathicontregdemo.azurecr.io
Subscription
Production 1
Configuration
Release
Edit Image Tag
Manage in Azure portal
Continuous delivery
Automatically publish your application to Azure with continuous delivery. Click Configure to begin setup.