

1) Host a simple nodejs application using custom data

> create a virtual machine (VM) by adding nodejsfile in custom data (advanced section)

The screenshot shows the Microsoft Azure portal interface. The top navigation bar includes the Microsoft Azure logo and a search bar. The main content area displays the details of a virtual machine named 'training'. The left sidebar contains a navigation menu with options like Overview, Activity log, Access control (IAM), Tags, Diagnose and solve problems, Favorites, Connect, Networking, Settings, Disks, Extensions + applications, Operating system, Configuration, Advisor recommendations, Properties, Locks, Availability + scale, and Availability + scaling. The main content area is divided into two sections: 'Essentials' and 'Properties'. The 'Essentials' section provides a summary of the VM's configuration, including its resource group, status, location, subscription, and public IP address. The 'Properties' section provides a detailed view of the VM's configuration, including its computer name, operating system, VM generation, and VM architecture. The 'Networking' section shows the public IP address and the network interface.

Essentials	
Resource group	training
Status	Running
Location	Central India (Zone 1)
Subscription	87d67179-ba15-4d91-81d4-f06832ca7714
Subscription ID	87d67179-ba15-4d91-81d4-f06832ca7714
Availability zone	1
Operating system	Linux (ubuntu 24.04)
Size	Standard B1s (1 vcpu, 1 GiB memory)
Public IP address	20.197.45.170
Virtual network/subnet	training-vnet/default
DNS name	Not configured
Health state	-
Time created	8/12/2024, 11:20 AM UTC

Properties	
Computer name	training
Operating system	Linux (ubuntu 24.04)
VM generation	V2
VM architecture	x64

Networking	
Public IP address	20.197.45.170 (Network interface training511_z1)
Public IP address (IPv6)	-
Private IP address	10.0.0.4
Private IP address	-

> connect the VM to putty

> type vi /home/azureuser/myapp/index.js

> type the following inside index.js

> var express = require('express')

var app = express()

var os = require('os');

app.get('/', function (req, res) {

res.send('Hello World from host ' + os.hostname() + '!')

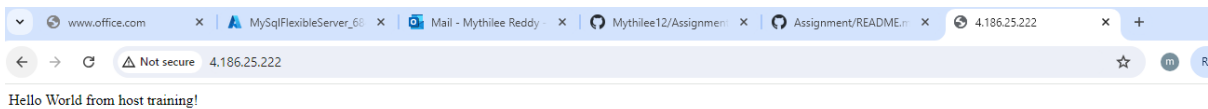
})

app.listen(3000, function () {

console.log('Hello world app listening on port 3000!')

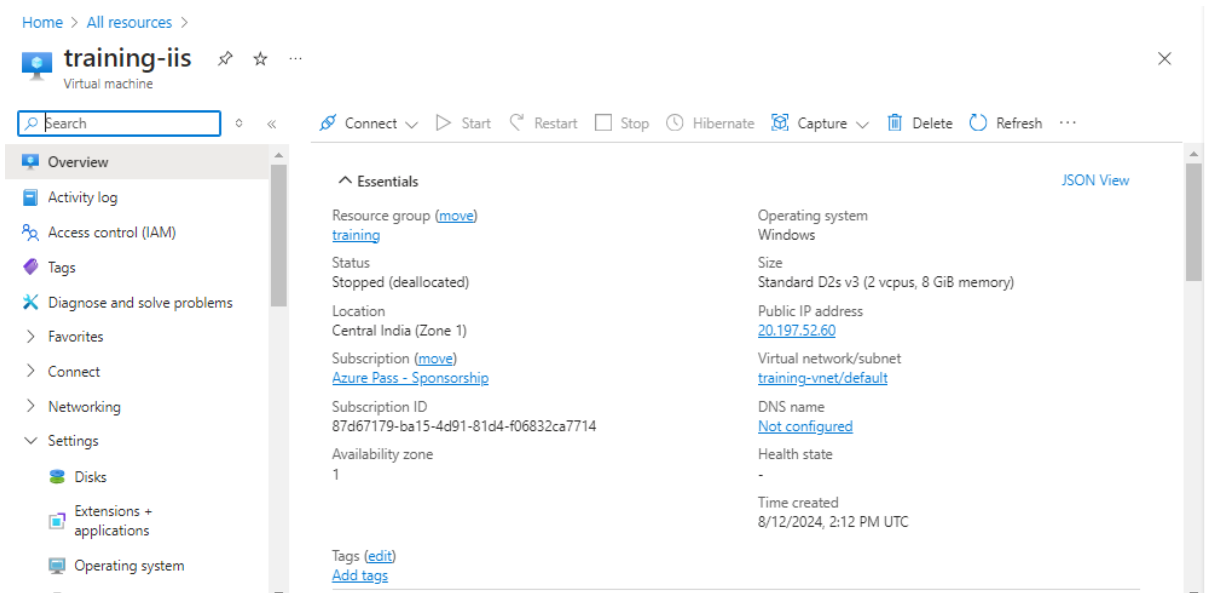
})

> nohup nodejs index.js &

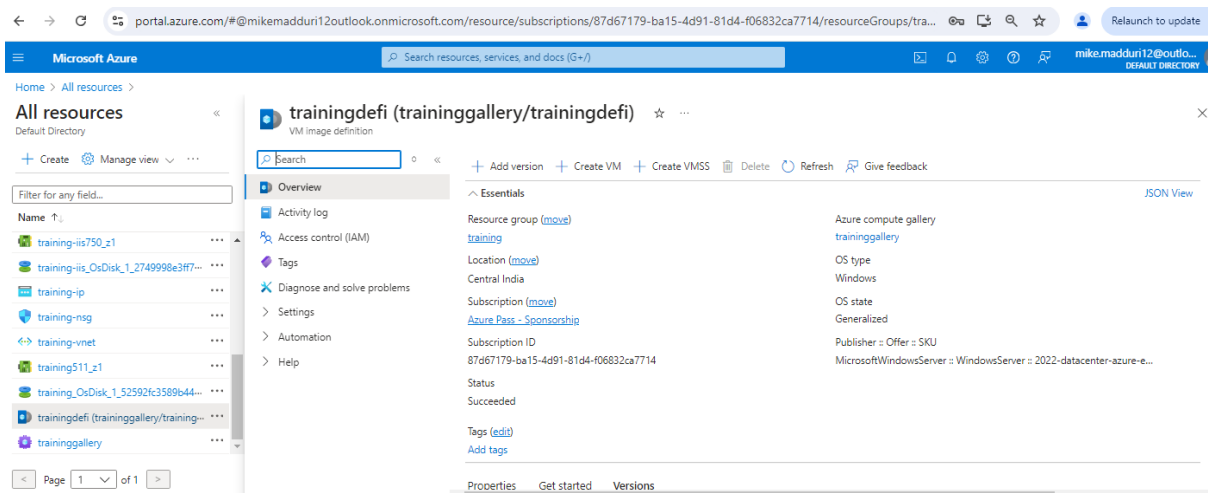
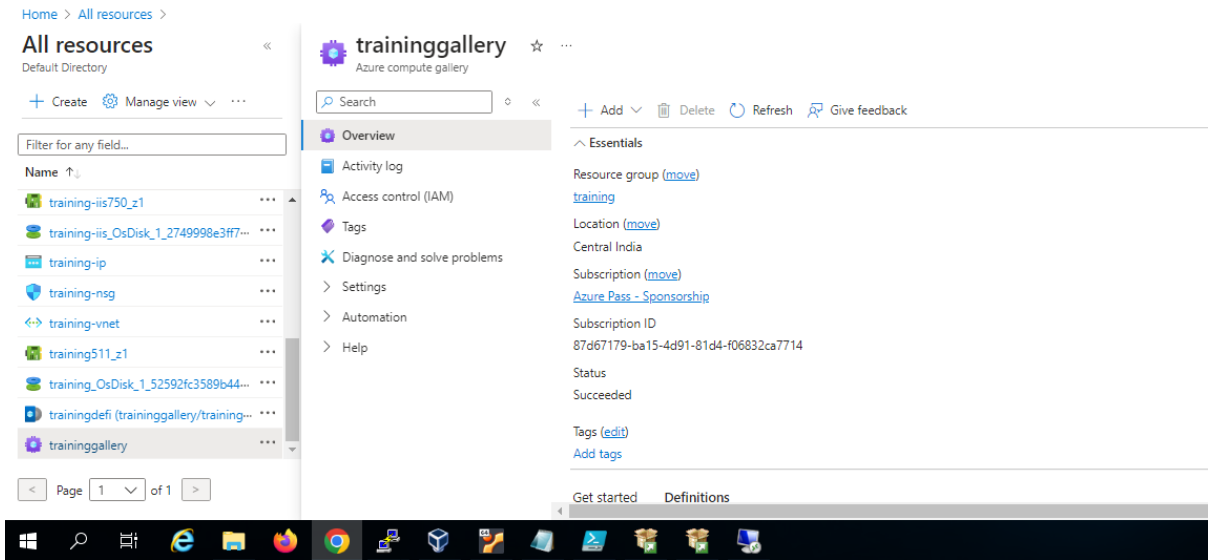


2)Create a window image with IIS pre-installed

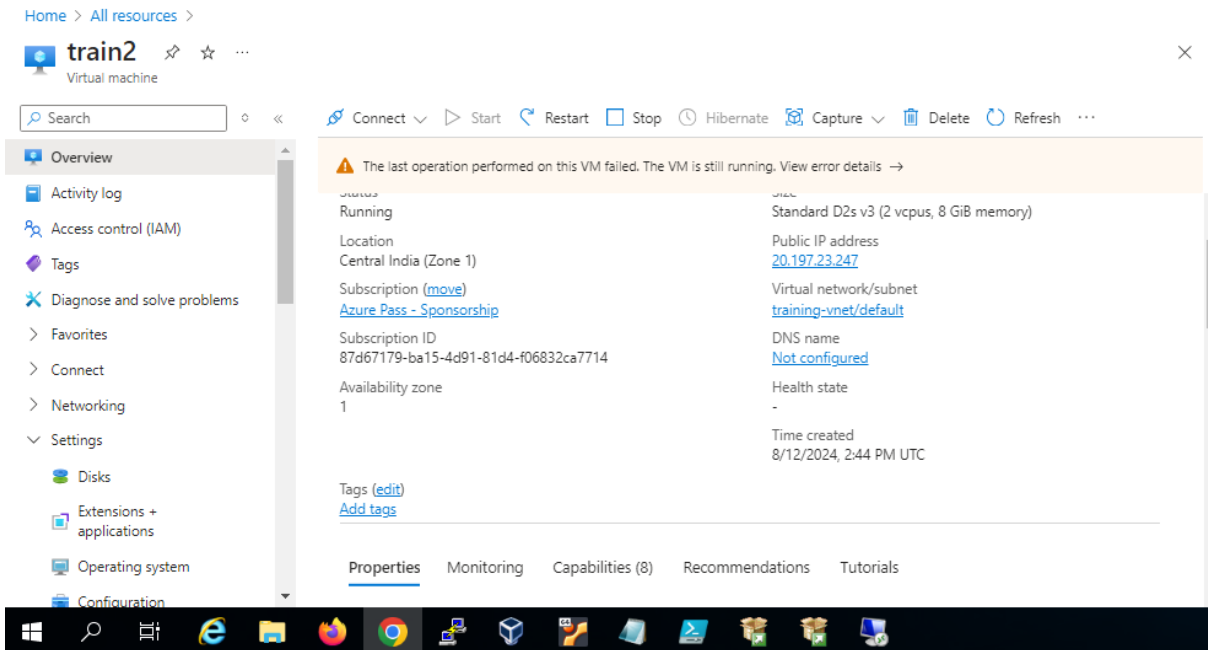
- >create resource group and a VM with RDP port
- >copy the IP address of VM in RDP
- >select add roles and features in search bar
- >select webserver IIS and install



- >click on capture on created VM and create an image



>click on create VM



>copy the IP of created VM and connect to RDP

