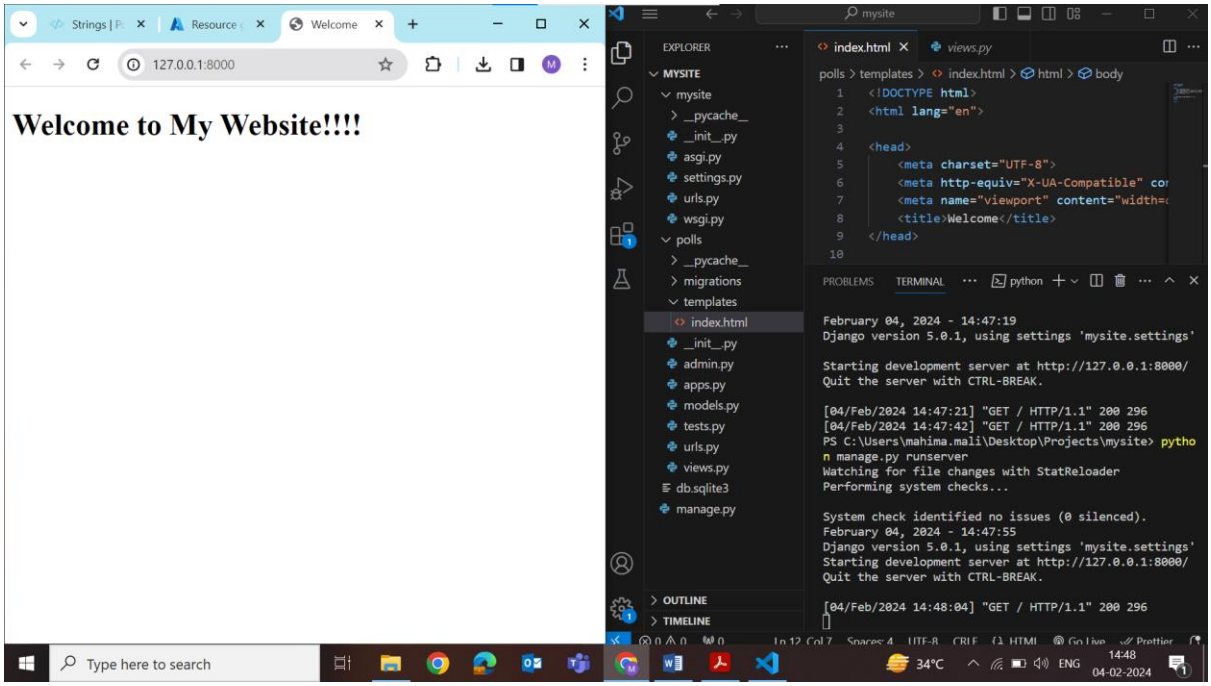
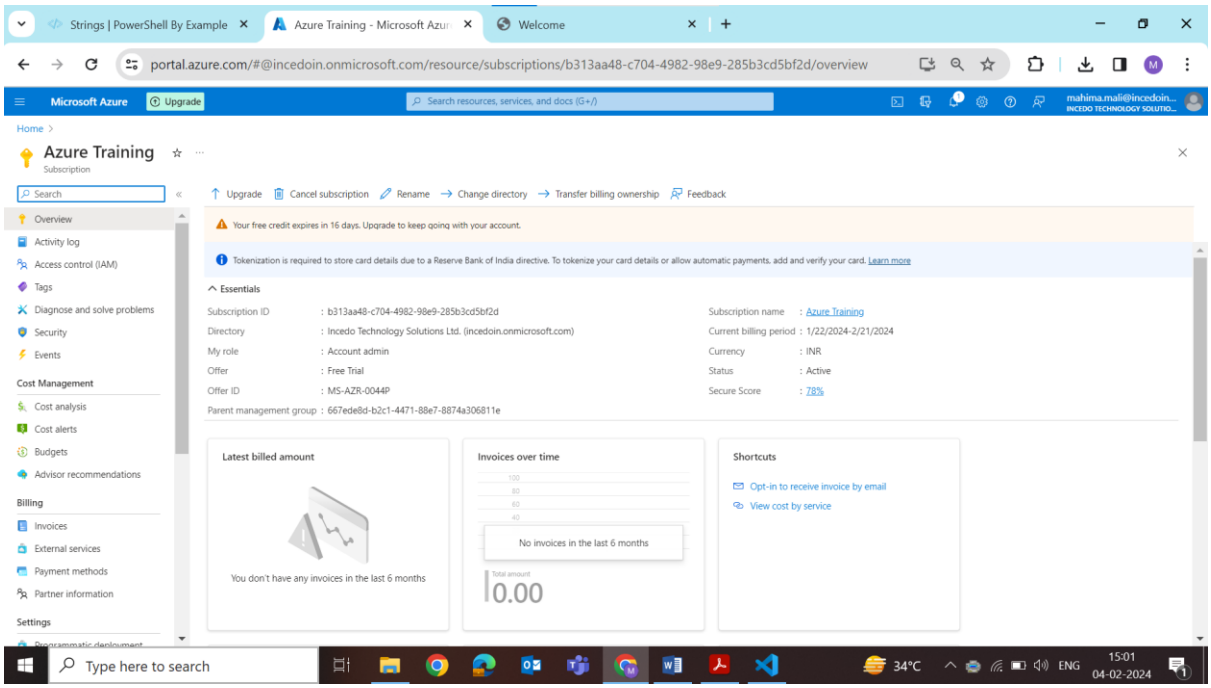
Project 1: Django Application Dokerization

1. Prerequisites

a. Django Application run in local environment



b. Azure Portal



2. Order and Connect Azure VM

a. Configuration of VM Specifications

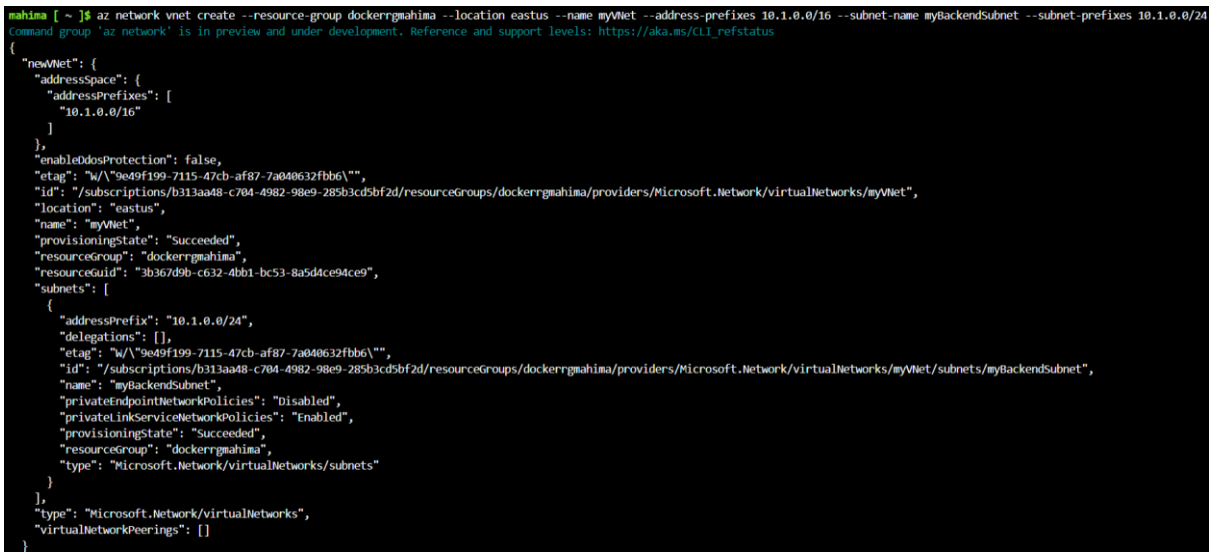
RAM: 16GB

Vnet: 10.0.0.0/16
Subnets: 10.0.0.0/24
Authentication: SSH with Existing Key

Create RG:
az group create --name dockerrgmahima --location eastus

```
mahima [ ~ ]$ az group create --name dockerrgmahima --location eastus
{
  "id": "/subscriptions/b313aa48-c704-4982-98e9-285b3cd5bf2d/resourceGroups/dockerrgmahima",
  "location": "eastus",
  "managedBy": null,
  "name": "dockerrgmahima",
  "properties": {
    "provisioningState": "Succeeded"
  },
  "tags": null,
  "type": "Microsoft.Resources/resourceGroups"
}
```

Create Virtual Network:
az network vnet create --resource-group dockerrgmahima --location eastus --name myVNet --address-prefixes 10.1.0.0/16 --subnet-name myBackend Subnet --subnet-prefixes 10.1.0.0/24



```
mahima [ ~ ]$ ssh-keygen -t rsa -b 2048 -f ~/.ssh/mahima_lock
Generating public/private rsa key pair.
/home/mahima/.ssh/mahima_lock already exists.
Overwrite (y/n)? y
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /home/mahima/.ssh/mahima_lock
Your public key has been saved in /home/mahima/.ssh/mahima_lock.pub
The key fingerprint is:
SHA256:0XfxbdgJfAcp7cqppKsiRAlMBy8fPqI+XiT5GhltqDY mahima@SandboxHost-638426141942405382
The key's randomart image is:
+---[RSA 2048]----+
|oo..        .ooo |
|..o      .  ..+*+|
| o +    . . .o+.*|
|  X .    . . ... |
| B B    S  . o   |
|o X .     . +    |
|oE o     o .     |
|o.* .   . .      |
|.+.. .....       |
+----[SHA256]-----+
```

b. Connect Through SSH key

az vm create --resource-group dockerrgmahima --name myVNet --image Ubuntu2204 --size Standard_DS3_v2 --admin-username azureuser --authentication-type ssh --ssh-key-value ~/.ssh/mahima_lock.pub

```
mahima [ ~ ]$ az vm create --resource-group dockerrgmahima --name myVNet --image Ubuntu2204 --size
Standard_DS3_v2 --admin-username azureuser --authentication-type ssh --ssh-key-value ~/.ssh/mahima_
lock.pub

{
  "fqdns": "",
  "id": "/subscriptions/b313aa48-c704-4982-98e9-285b3cd5bf2d/resourceGroups/dockerrgmahima/provider
s/Microsoft.Compute/virtualMachines/myVNet",
  "location": "eastus",
  "macAddress": "00-0D-3A-8C-75-19",
  "powerState": "VM running",
  "privateIpAddress": "10.1.0.4",
  "publicIpAddress": "20.124.208.236",
  "resourceGroup": "dockerrgmahima",
  "zones": ""
}
```

ssh -i ~/.ssh/mahima_lock azureuser@20.124.208.236

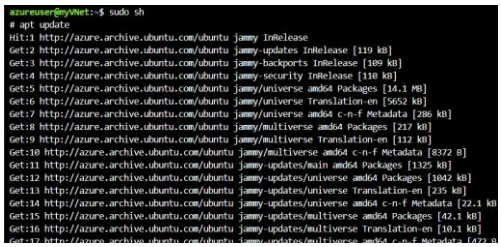
```
mahima [ ~ ]$ ssh -i ~/.ssh/mahima_lock azureuser@20.124.208.236
Welcome to Ubuntu 22.04.3 LTS (GNU/Linux 6.2.0-1019-azure x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/pro

  System information as of Sun Feb  4 10:53:46 UTC 2024

  System load:  0.080078125       Processes:             122
  Usage of /:   5.1% of 28.89GB   Users logged in:       0
  Memory usage: 2%                IPv4 address for eth0: 10.1.0.4
  Swap usage:   0%

Expanded Security Maintenance for Applications is not enabled.

0 updates can be applied immediately.

Enable ESM Apps to receive additional future security updates.
See https://ubuntu.com/esm or run: sudo pro status



The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

To run a command as administrator (user "root"), use "sudo <command>".
See "man sudo_root" for details.
```

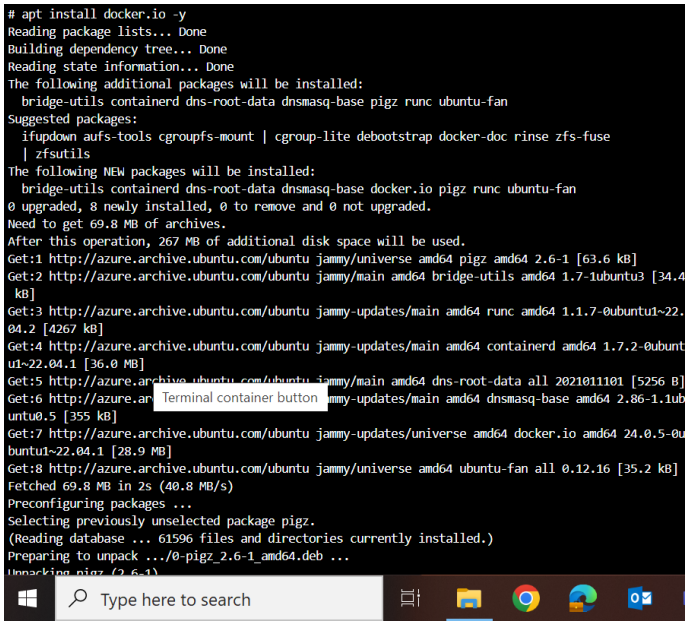
```
azureuser@myVNet:~$ exit
logout
Connection to 20.124.208.236 closed.
```

3. Installing Docker on Azure VM

az ssh vm --resource-group dockerrgmahima --vm-name myVNet --subscription b313aa48-c704-4982-98e9-285b3cd5bf2d

sudo sh
apt update
apt install docker,io -y





docker --version
docker --info

```
# docker --version
Docker version 24.0.5, build 24.0.5-0ubuntu1~22.04.1
```

docker network create mahima-project

```
# docker network create mahima-project
9a593ebb5ec191ab09c8b8a34760b80775eb7552fae007ab016bbb27961a045e
# docker network ls
NETWORK ID     NAME              DRIVER    SCOPE
37dd1b90b625   bridge            bridge    local
b32e1e52e4c2   host              host      local
9a593ebb5ec1   mahima-project    bridge    local
be7df3c41827   none              null      local
```

4. Django Application Dockerization

Pull Ubuntu LTS Image

sudo docker pull ubuntu:latest

```
azureuser@myVNet:~$ sudo docker pull ubuntu:latest
latest: Pulling from library/ubuntu
57c139bbda7e: Pull complete
Digest: sha256:e9569c25505f33ff72e88b2990887c9dcf230f23259da296eb814fc2b41af999
Status: Downloaded newer image for ubuntu:latest
docker.io/library/ubuntu:latest
```

Copy Django Project in Docker Container

docker run --rm --net mahima-project --name dock -v C:\Users\mahima.mali\Desktop\Projects\mysite –it ubuntu:latest

```
docker.io/library/ubuntu:latest
# docker images
REPOSITORY   TAG       IMAGE ID       CREATED      SIZE
ubuntu       latest    fd1d8f58e8ae   9 days ago   77.9MB
# docker run --rm --net mahima-project --name dock -v C:\Users\mahima.mali\Desktop\Projects\mysite -
docker: invalid reference format.
See 'docker run --help'.
# docker run --rm --net mahima-project --name dock -v C:\Users\mahima.mali\Desktop\Projects\mysite -

"docker run" requires at least 1 argument.
See 'docker run --help'.

Usage:  docker run [OPTIONS] IMAGE [COMMAND] [ARG...]

Create and run a new container from an image
# docker run --rm --net mahima-project --name dockers -it ubuntu:latest
root@fc3a91476eb4:/#  -v C:\Users\mahima.mali\Desktop\Projects\mysite
bash: -v: command not found
root@fc3a91476eb4:/# exit
exit
```

docker ps

```
# docker --version
Docker version 24.0.5, build 24.0.5-0ubuntu1~22.04.1
# docker run --rm --net mahima-project --name dockers -d ubuntu:latest
d4ba7346a2f05740e53aafb94399c814ccda37f5655ce021f7bfc44bc661996d
# docker ps
CONTAINER ID   IMAGE     COMMAND   CREATED   STATUS    PORTS     NAMES
# docker images
REPOSITORY   TAG       IMAGE ID       CREATED       SIZE
ubuntu       latest    fd1d8f58e8ae   10 days ago   77.9MB
# docker pull nginx:latest
latest: Pulling from library/nginx
c57ee5000d61: Pull complete
9b0163235c08: Pull complete
f24a6f652778: Pull complete
9f3589a5fc50: Pull complete
f0bd99a47d4a: Pull complete
398157bc5c51: Pull complete
1ef1c1a36ec2: Pull complete
Digest: sha256:5f44022eab9198d75939d9eaa5341bc077eca16fa51d4ef32d33f1bd4c8cbe7d
Status: Downloaded newer image for nginx:latest
docker.io/library/nginx:latest
# docker ps
CONTAINER ID   IMAGE     COMMAND   CREATED   STATUS    PORTS     NAMES
# docker run --rm --net mahima-project --name dockers -d nginx:latest
8b4751f0fc57530cf4cb96a2961cffca11dc1d41e4a45c78525d1e15e2c0b6c6
# docker ps
CONTAINER ID   IMAGE          COMMAND                  CREATED         STATUS         PORTS     NAMES
8b4751f0fc57   nginx:latest   "/docker-entrypoint.…"   4 seconds ago   Up 3 seconds   80/tcp    dockers
```

export cid=

docker inspect $cid | grep IPAddress

docker run --net mahima-project –it busybox sh

docker images

Access in another Container

5. Networking and Security Considerations

6. Scaling and Load Balancing

Create LB → Health Probe (HP) → LB Route

Create LB:
az network lb create --resource-group dockerrgmahima --name mahimaLoadBalancer --sku Standard --public-ip-address myPublicIP --frontend-ip-name myFrontEnd --backend-pool-name myBackEndPool

Create HP:
az network lb probe create --resource-group dockerrgmahima --lb-name mahimaLoadBalancer --name myHealthProbe --protocol tcp --port 80

```
az network lb rule create \
        --resource-group dockerrgmahima \
        --lb-name mahimaLoadBalancer \
        --name myHTTPRule \
        --protocol tcp --frontend-port 80 \
        --backend-port 80 \
        --frontend-ip-name myFrontEnd \
        --backend-pool-name myBackEndPool \
        --probe-name myHealthProbe \
        --disable-outbound-snat true \
        --idle-timeout 15 --enable-tcp-reset true
```

Network Security Group (NSG) → NSG Rules

```
az network nsg create \
        --resource-group dockerrgmahima \
        --name myNSG

az network nsg rule create \
        --resource-group dockerrgmahima \
```

```
    --nsg-name myNSG \
    --name myNSGRuleHTTP \
    --protocol '*' --direction inbound \
    --source-address-prefix '*' \
    --source-port-range '*' \
    --destination-address-prefix '*' \
    --destination-port-range 80 \
    --access allow --priority 200
```

Bastion Host

Create Public IP address – myBastionIP:

```
az network public-ip create --resource-group dockerrgmahima  --name myBastionIP --sku
Standard --zone 1 2 3
```

Create Bastion Subnet:

```
az network vnet subnet create \
    --resource-group dockerrgmahima \
    --name AzureBastionSubnet \
    --vnet-name myVNet \
    --address-prefixes 10.1.1.0/27
```

Create bastion host

```
az network bastion create \
    --resource-group dockerrgmahima \
    --name myBastionHost \
    --public-ip-address myBastionIP \
    --vnet-name myVNet \
    --location eastus
```

Backend Subnet

Create NIC with name as MyNicVM1; MyNicVM2

```
array=(myNicVM1 myNicVM2)
  for vmnic in "${array[@]}"
  do
    az network nic create \
      --resource-group Dockerrgmahima\
      --name $vmnic \
      --vnet-name myVNet \
      --subnet myBackEndSubnet \
      --network-security-group myNSG
  done
```

Create VM1 and VM2

```
az vm create \
    --resource-group Dockerrgmahima\
    --name myVM1 \
    --nics myNicVM1 \
    --image win2019datacenter \
```

```
        --admin-username azureuser \
        --zone 1 --no-wait

az vm create \
        --resource-group Dockerrgmahima\
        --name myVM2 \
        --nics myNicVM2 \
        --image win2019datacenter \
        --admin-username azureuser \
        --zone 2 --no-wait
```

Attach LB ← VM1, VM2 ← MyNicVM1, MyNicVM2

```
array=(myNicVM1 myNicVM2)
  for vmnic in "${array[@]}"
  do
    az network nic ip-config address-pool add \
      --address-pool myBackendPool \
      --ip-config-name ipconfig1 \
      --nic-name $vmnic \
      --resource-group Dockerrgmahima\
      --lb-name myLoadBalancer1
  Done
```

NAT Gateway

Create MyNATGWIp

```
az network public-ip create \
        --resource-group Dockerrgmahima\
        --name myNATgatewayIP \
        --sku Standard \
        --zone 1 2 3
```

Create MyNATGW

```
az network nat gateway create \
        --resource-group Dockerrgmahima\
        --name myNATgateway \
        --public-ip-addresses myNATgatewayIP \
        --idle-timeout 10
```

Update MyNATGW → BackendSubnet

```
az network vnet subnet update \
    --resource-group Dockerrgmahima\
    --vnet-name myVNet \
    --name myBackendSubnet \
    --nat-gateway myNATgateway
```

WebServer (IIS) ← HelloWorld ← VM1, VM2

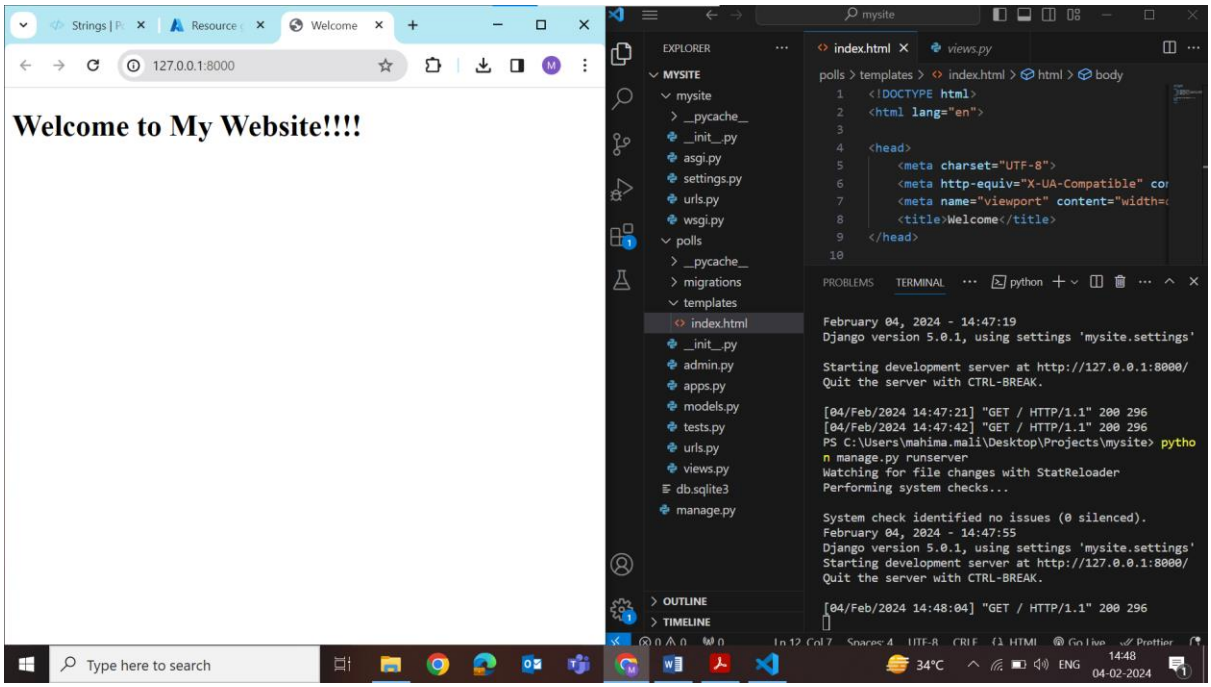
```
array=(myVM1 myVM2)
  for vm in "${array[@]}"
  do
    az vm extension set \
      --publisher Microsoft.Compute \
      --version 1.8 --name CustomScriptExtension \
      --vm-name $vm --resource-group Dockerrgmahima\
      --settings '{"commandToExecute":"powershell Add-WindowsFeature Web-Server;
powershell Add-Content -Path \"C:\\inetpub\\wwwroot\\Default.htm\" -Value
$($env:computername)"}'
  done
```

Test

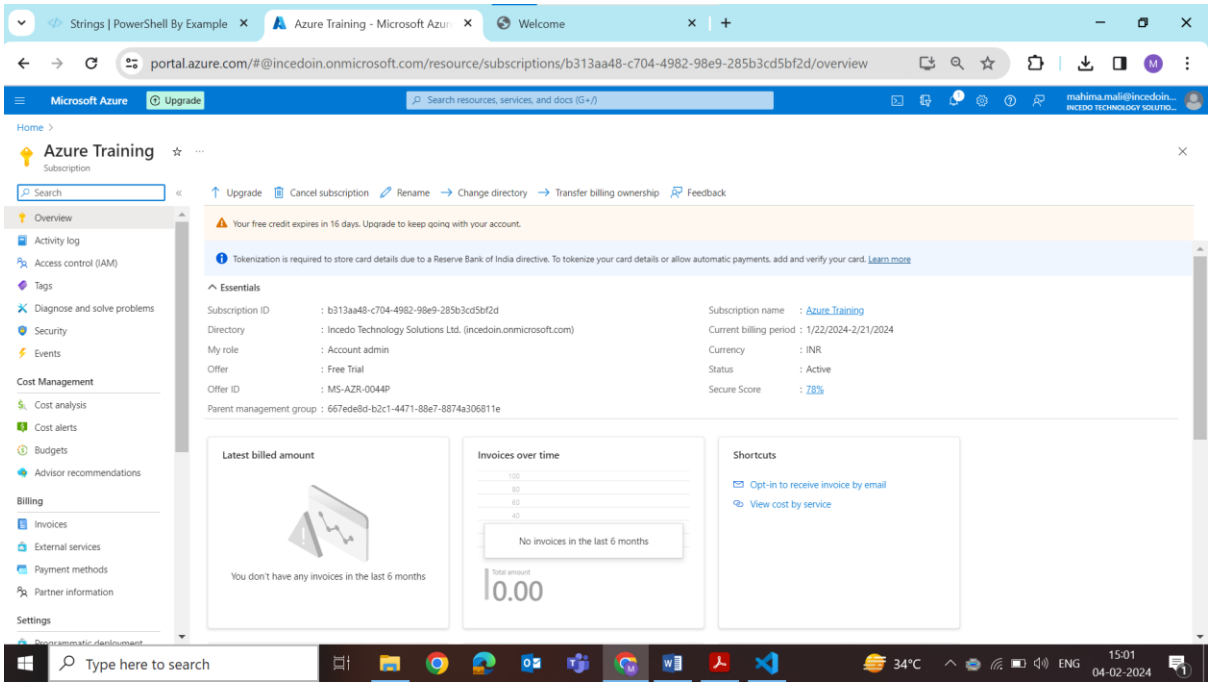
```
az network public-ip show \
      --resource-group Dockerrgmahima\
      --name myPublicIP \
      --query ipAddress \
      --output tsv
```

7. Deployment and Documentation

a. Django Application run in local environment



b. Azure Portal


Strings | PowerShell By Example
Azure Training - Microsoft Azur
Welcome
portal.azure.com/#@incedoin.onmicrosoft.com/resource/subscriptions/b313aa48-c704-4982-98e9-285b3cd5bf2d/overview
Microsoft Azure
Upgrade
Search resources, services, and docs (G+/)
mahima.mali@incedoin...
INCEDO TECHNOLOGY SOLUTIO...
Home
Azure Training
Subscription
Search
Overview
Activity log
Access control (IAM)
Tags
Diagnose and solve problems
Security
Events
Cost Management
Cost analysis
Cost alerts
Budgets
Advisor recommendations
Billing
Invoices
External services
Payment methods
Partner information
Settings
Upgrade
Cancel subscription
Rename
Change directory
Transfer billing ownership
Feedback
Your free credit expires in 16 days. Upgrade to keep going with your account.
Tokenization is required to store card details due to a Reserve Bank of India directive. To tokenize your card details or allow automatic payments, add and verify your card. Learn more
Essentials
Subscription ID : b313aa48-c704-4982-98e9-285b3cd5bf2d
Directory : Incedo Technology Solutions Ltd. (incedoin.onmicrosoft.com)
My role : Account admin
Offer : Free Trial
Offer ID : MS-AZR-0044P
Parent management group : 667ede8d-b2c1-4471-88e7-8874a306811e
Subscription name : Azure Training
Current billing period : 1/22/2024-2/21/2024
Currency : INR
Status : Active
Secure Score : 78%
Latest billed amount
You don't have any invoices in the last 6 months
Invoices over time
No invoices in the last 6 months
Total amount
0.00
Shortcuts
Opt-in to receive invoice by email
View cost by service
Type here to search
34°C
ENG
15:01
04-02-2024