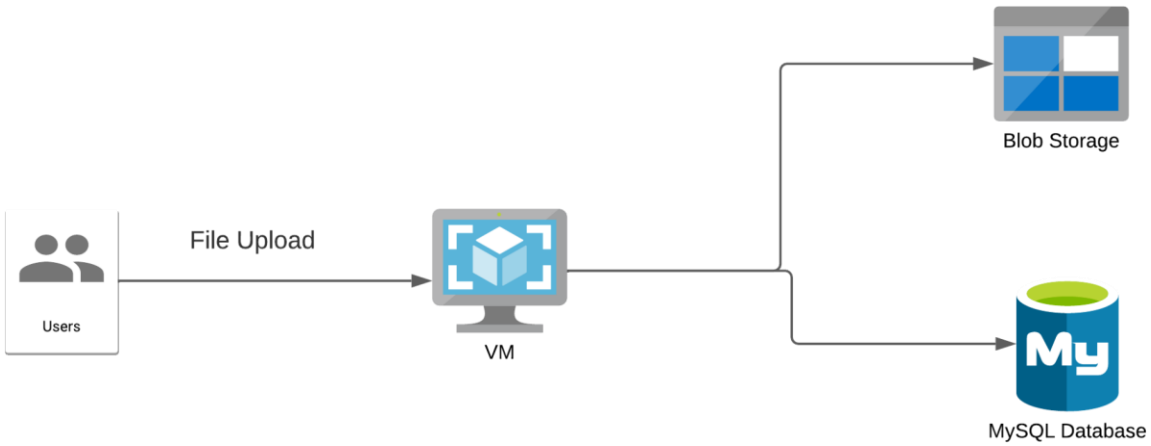
Architecture diagram:

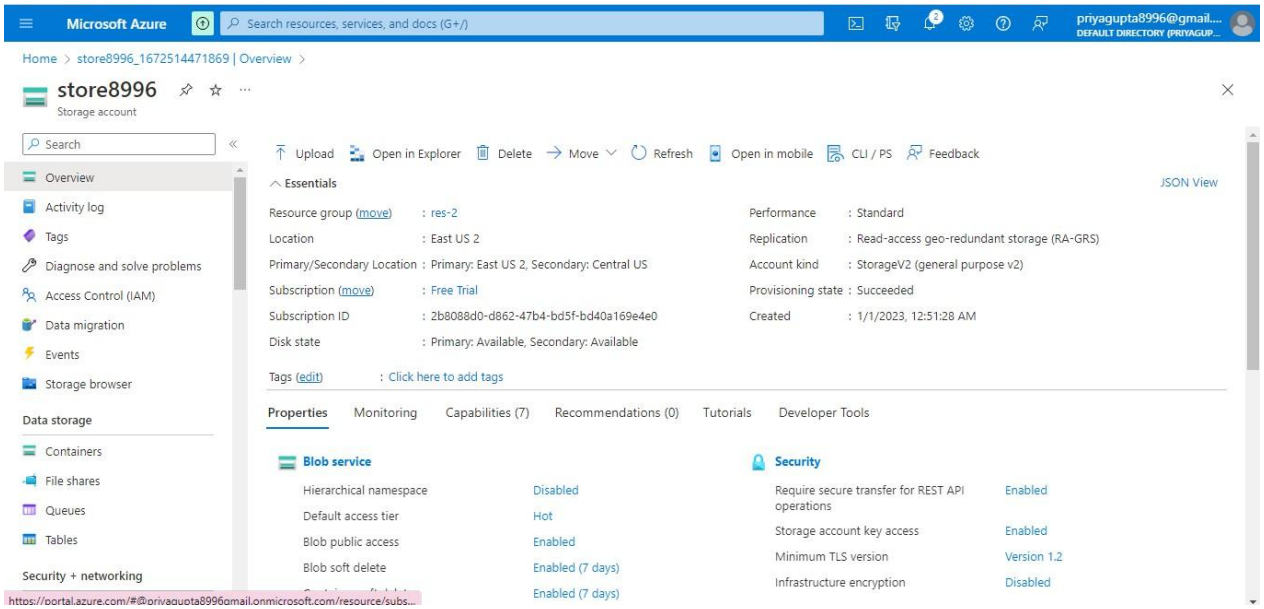


Architecture Implementation	
1	Upload the custom program and provided text file to a VM created using Ubuntu
2	Createa MySQL server using Azure Database service
3	Createa database inside the MySQL server created above
4	Running the custom program will convert the text file into a CSV file,upload it to blob storage and send he data to the MySQL server.

Step1:Creationof resources

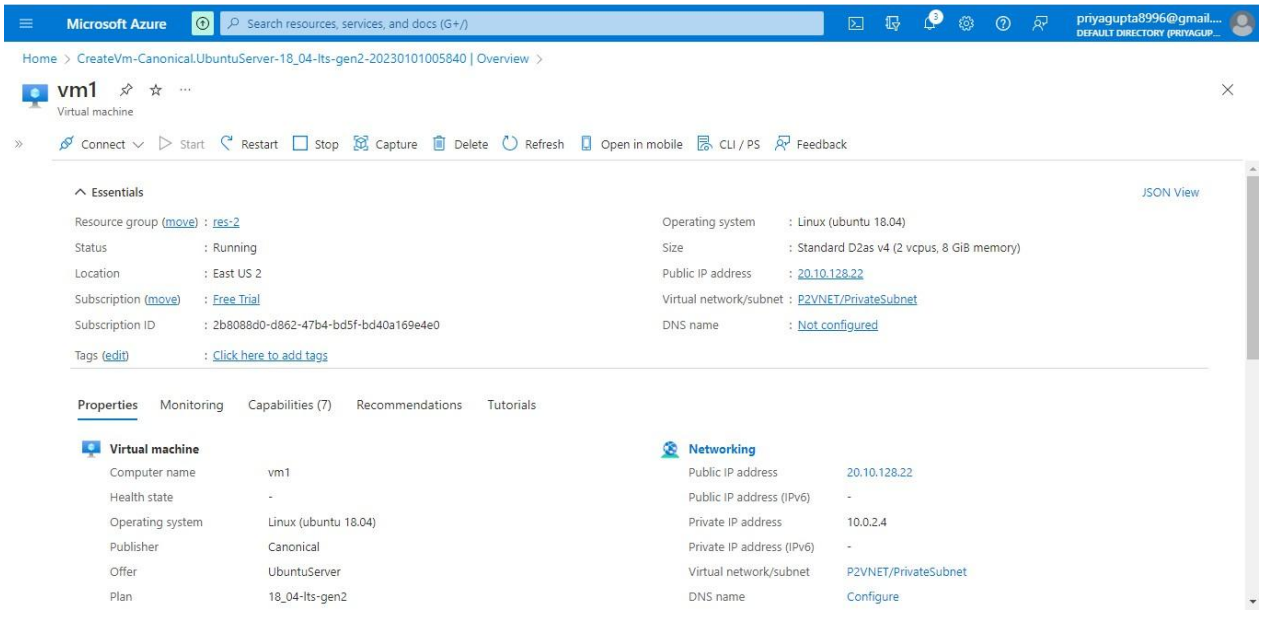
Stepnumber	a
Stepname	CreationofResourcegroupandblobstorage
Expectedscreenshots	Screen showing created storage account

Storage Account Created:



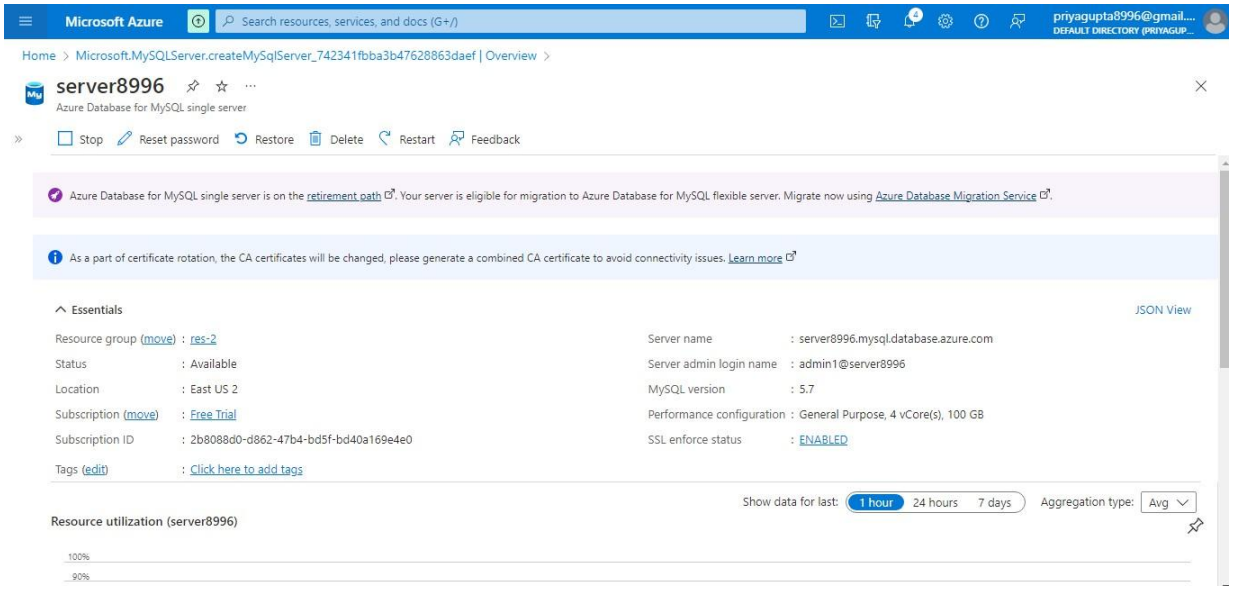
Stepnumber	b
Stepname	Creation of VM
Expected screenshots	CreatedVM

Created VM



Stepnumber Stepname	C Creation of MySQL server
Expectedscreenshots	1)Overview screen of the created database server.

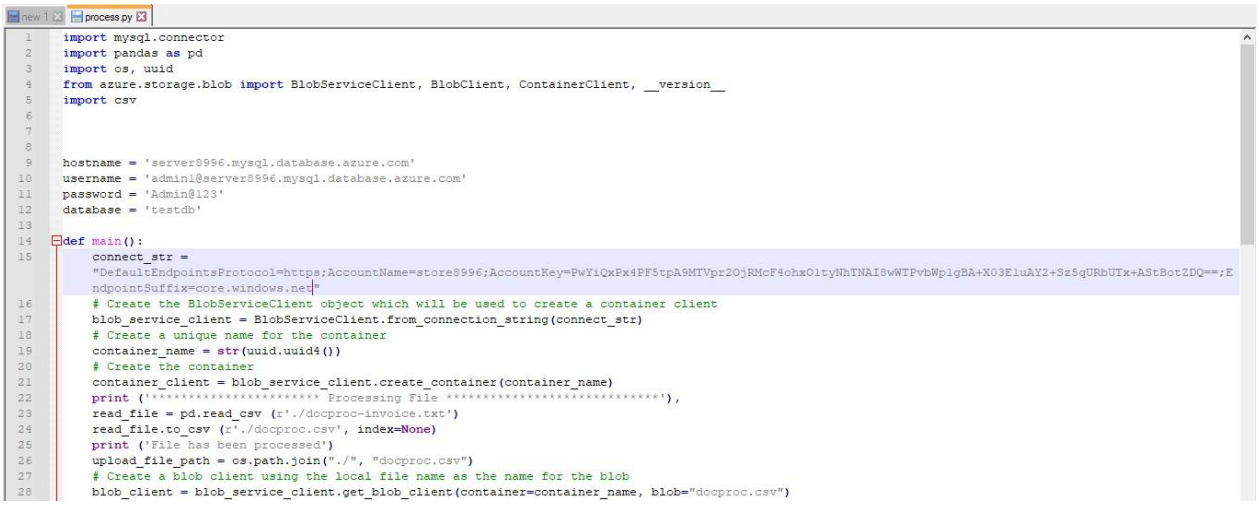
Database Server:



Step2:RunthecustomprogramintheVM

Stepnumber	a
Stepname	Environmentsetup
Expected screenshots	Screenshot of the process.py file after completing Step3 above Copying the files using scp Screenshot after completing Step above.

process.py file



```
import mysql.connector
import pandas as pd
import os, uuid
from azure.storage.blob import BlobServiceClient, BlobClient, ContainerClient, __version__
import csv



hostname = 'server8996.mysql.database.azure.com'
username = 'admin1@server8996.mysql.database.azure.com'
password = 'Admin@123'
database = 'testdb'

def main():
    connect_str =
    "DefaultEndpointsProtocol=https;AccountName=store8996;AccountKey=PwYiQxPx4PF5tpA9MTVpr2OjRMcF4ohxOltyNhTNAI8wWTPvbWplgBA+X03EluAY2+Sz5qURbUTx+AStBotZDQ==;E
    ndpointSuffix=core.windows.net"
    # Create the BlobServiceClient object which will be used to create a container client
    blob_service_client = BlobServiceClient.from_connection_string(connect_str)
    # Create a unique name for the container
    container_name = str(uuid.uuid4())
    # Create the container
    container_client = blob_service_client.create_container(container_name)
    print ('********************** Processing File ******************************'),
    read_file = pd.read_csv (r'./docproc-invoice.txt')
    read_file.to_csv (r'./docproc.csv', index=None)
    print ('File has been processed')
    upload_file_path = os.path.join("./", "docproc.csv")
    # Create a blob client using the local file name as the name for the blob
    blob_client = blob_service_client.get_blob_client(container=container_name, blob="docproc.csv")
```

Copying the files using scp

```
lenovo@DESKTOP-1332UR7 MINGW64 ~/Downloads
$ scp -i vm1_key.pem docproc-invoice.txt azureuser@20.10.128.22:/home/azureuser
docproc-invoice.txt

lenovo@DESKTOP-1332UR7 MINGW64 ~/Downloads
$ scp -i vm1_key.pem process.py azureuser@20.10.128.22:/home/azureuser
process.py

lenovo@DESKTOP-1332UR7 MINGW64 ~/Downloads
$
```

```
-rw-r--r-- 1 azureuser azureuser  251 Dec 31 20:18 docproc-invoice.txt
-rw-r--r-- 1 azureuser azureuser 2302 Dec 31 20:19 process.py
azureuser@vm1:~$
```

Post completion step:

```
azureuser@vm1: ~
Building dependency tree
Reading state information... Done
python3 is already the newest version (3.6.7-1~18.04).
0 upgraded, 0 newly installed, 0 to remove and 0 not upgraded.
azureuser@vm1:~$ sudo apt install python3-pip
Reading package lists... Done
Building dependency tree
Reading state information... Done
E: Unable to locate package python3-pip
azureuser@vm1:~$ sudo pip3 install pandas
sudo: pip3: command not found
azureuser@vm1:~$ sudo pip3 install azure-storage-blob
sudo: pip3: command not found
azureuser@vm1:~$ sudo pip3 install mysql-connector-python
sudo: pip3: command not found
azureuser@vm1:~$ sudo apt install mysql-client-core-5.7
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following additional packages will be installed:
  libaio1
The following NEW packages will be installed:
  libaio1 mysql-client-core-5.7
0 upgraded, 2 newly installed, 0 to remove and 0 not upgraded.
Need to get 6761 kB of archives.
After this operation, 30.3 MB of additional disk space will be used.
Do you want to continue? [Y/n] y
Get:1 http://azure.archive.ubuntu.com/ubuntu bionic-updates/main amd64 libaio1 amd64 0.3.110-5ubuntu0.1 [6476 B]
Get:2 http://azure.archive.ubuntu.com/ubuntu bionic-updates/main amd64 mysql-client-core-5.7 amd64 5.7.40-0ubuntu0.18.04.1 [6755 kB]
Fetched 6761 kB in 0s (25.1 MB/s)
Selecting previously unselected package libaio1:amd64.
(Reading database ... 59059 files and directories currently installed.)
Preparing to unpack .../libaio1_0.3.110-5ubuntu0.1_amd64.deb ...
Unpacking libaio1:amd64 (0.3.110-5ubuntu0.1) ...
Selecting previously unselected package mysql-client-core-5.7.
Preparing to unpack .../mysql-client-core-5.7_5.7.40-0ubuntu0.18.04.1_amd64.deb ...
Unpacking mysql-client-core-5.7 (5.7.40-0ubuntu0.18.04.1) ...
Setting up libaio1:amd64 (0.3.110-5ubuntu0.1) ...
Setting up mysql-client-core-5.7 (5.7.40-0ubuntu0.18.04.1) ...
Processing triggers for man-db (2.8.3-2ubuntu0.1) ...
Processing triggers for libc-bin (2.27-3ubuntu1.6) ...
azureuser@vm1:~$
```

Stepnumber	b
Stepname	Configure the database
Expected screenshots	1)Screenshot after completing Step above

Database Configuration: