

Assignment 2

Name – Sati, Ankit

Date – 2/20/2022

Section – 001

SID – as14128

Total in points (Maximum 100 points)–

Professors Comments –

Affirmation of Independent Effort – Ankit Sati

Step 1 - Making an Azure Student account.

- ➔ Post setting up the Aws account there are only a small key difference in the Microsoft Azure account.
- ➔ We have to setup the storage and server instances as done in the amazon accounts.
 - a. Before that we need to take care of a few dependencies.
 - Python 3.7 (As this will be used in the future for the ArchNav)

```
ankit@LAPTOP-S2U1QMGB:/mnt/c/Users/ankit$ sudo apt install python3-pip
[sudo] password for ankit:
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following packages were automatically installed and are no longer required:
  bridge-utils dns-root-data dnsmasq-base libidn11 ubuntu-fan
Use 'sudo apt autoremove' to remove them.
The following additional packages will be installed:
  libexpat1-dev libpython3-dev libpython3.8 libpython3.8-dev libpython3.8-minimal libpython3.8-stdlib python-pip-whl
  python3-dev python3-wheel python3.8 python3.8-dev python3.8-minimal zlib1g-dev
Suggested packages:
  Unpacking python3.8-dev (3.8.10-0ubuntu1~20.04.2) ...
  Selecting previously unselected package python3-dev.
  Preparing to unpack .../11-python3-dev_3.8.2-0ubuntu2_amd64.deb ...
  Unpacking python3-dev (3.8.2-0ubuntu2) ...
  Selecting previously unselected package python3-wheel.
  Preparing to unpack .../12-python3-wheel_0.34.2-1_all.deb ...
  Unpacking python3-wheel (0.34.2-1) ...
  Selecting previously unselected package python3-pip.
  Preparing to unpack .../13-python3-pip_20.0.2-5ubuntu1.6_all.deb ...
  Unpacking python3-pip (20.0.2-5ubuntu1.6) ...
  Setting up libpython3.8-minimal:amd64 (3.8.10-0ubuntu1~20.04.2) ...
  Setting up python3-wheel (0.34.2-1) ...
  Setting up libexpat1-dev:amd64 (2.2.9-1build1) ...
  Setting up zlib1g-dev:amd64 (1:1.2.11.dfsg-2ubuntu1.2) ...
  Setting up python3.8-minimal (3.8.10-0ubuntu1~20.04.2) ...
  Setting up python-pip-whl (20.0.2-5ubuntu1.6) ...
  Setting up libpython3.8-stdlib:amd64 (3.8.10-0ubuntu1~20.04.2) ...
  Setting up python3.8 (3.8.10-0ubuntu1~20.04.2) ...
```

- a.
 - b. Post this we need to install the JupyterLabs and set the appropriate path.

```
ankit@LAPTOP-S2U1QMGB:/mnt/c/Users/ankit$ pip install jupyterlab
Collecting jupyterlab
  Downloading jupyterlab-3.2.9-py3-none-any.whl (8.5 MB)
    |#####| 8.5 MB 6.6 MB/s
Collecting jupyterlab-server~=2.3
  Downloading jupyterlab_server-2.10.3-py3-none-any.whl (61 kB)
    |#####| 61 kB 7.3 MB/s
Requirement already satisfied: Jinja2>=2.1 in /usr/lib/python3/dist-packages (from jupyterlab) (2.10.1)
Collecting nbclassic~=0.2
  Downloading nbclassic-0.3.5-py3-none-any.whl (25 kB)
Collecting tornado>=6.1.0
  Downloading tornado-6.1-cp38-cp38-manylinux2010_x86_64.whl (427 kB)
    |#####| 427 kB 10.4 MB/s
Collecting jupyter-server~=1.4
  Downloading jupyter_server-1.13.5-py3-none-any.whl (397 kB)
    |#####| 397 kB 11.7 MB/s
Collecting packaging
  Downloading packaging-21.3-py3-none-any.whl (40 kB)
    |#####| 40 kB 7.1 MB/s
Collecting jupyter-core
```

- c. We should have the Docker on our machine and it should be up and running.

```
To get more help with docker, check out our guides at https://docs.docker.com/go/guides/

ankit@LAPTOP-S2U1QMGB:/mnt/c/Users/ankit$ docker --version
Docker version 20.10.12, build e91ed57
ankit@LAPTOP-S2U1QMGB:/mnt/c/Users/ankit$ service docker status
* Docker is running
ankit@LAPTOP-S2U1QMGB:/mnt/c/Users/ankit$ |
```

- d. Post this, as mentioned in the assignment, we need to run the command (docker run -i -t -p 8888:8888 dbgannon/tutorial).

```
ankit@LAPTOP-S2U1QMGB:/mnt/c/Users/ankit$ service docker status
* Docker is running
ankit@LAPTOP-S2U1QMGB:/mnt/c/Users/ankit$ docker run -i -t -p 8888:8888 dbgannon/tutorial
cp: omitting directory '/tutorial_notebooks/graph'
/home/jovyan/work
total 16
drwxr-xr-x 1 jovyan users 4096 Feb 18 06:27 .
drwxr-xr-x 1 jovyan users 4096 Jul 8 2017 ..
drwxr-xr-x 2 jovyan users 4096 Feb 18 06:27 notebooks
[I 06:27:29.522 NotebookApp] Writing notebook server cookie secret to /home/jovyan/.local/share/jupyter/runtime/notebook_cookie_secret
[I 06:27:30.200 NotebookApp] JupyterLab alpha preview extension loaded from /opt/conda/lib/python3.5/site-packages/jupyterlab
[I 06:27:30.209 NotebookApp] Serving notebooks from local directory: /home/jovyan/work
[I 06:27:30.209 NotebookApp] 0 active kernels
[I 06:27:30.209 NotebookApp] The Jupyter Notebook is running at: https://[all ip addresses on your system]:8888/
[I 06:27:30.209 NotebookApp] Use Control-C to stop this server and shut down all kernels (twice to skip confirmation).
```

- e. Finally, just for the deployment purposes we need to get the kubectl instance up and running with the docker daemons already started.

➔ Now lets get started with setting up the Azure account.

Subscription type – Azure account

Payment – Pay as you go

Account information.

Subscription ID

5c3b1518-756b-449a-a9cb-424eb50a0f71

Directory

nyu.edu (nyu.edu)

Status

Disabled

Parent management group

7b331012-87a1-4a16-8b0f-a4605b1f3d7f

Step 2 and 3 – SSH

SSH pair key
resource group as14128

Connection Key - HostName=as141286.azure-devices.net;DeviceId=myEdgeDevice;SharedAccessKey=DUQ43decm2Rqt8D5u01ZEVDZ52gq7elK8qdiUKkwBs=

```
connectionState : "Disconnected",
"connectionStateUpdatedTime": "0001-01-01T00:00:00",
"deviceId": "myEdgeDevice",
"deviceScope": "ms-azure-iot-edge://myEdgeDevice-63772533601409849",
"etag": "MjIzOTY3MzAy",
"generationId": "63772533601409849",
"lastActivityTime": "0001-01-01T00:00:00",
"parentScopes": [],
"status": "enabled",
"statusReason": null,
"statusUpdatedTime": "0001-01-01T00:00:00"
}
ankit@Azure:~$ az iot hub device-identity connection-string show --device-id myEdgeDevice --hub-name as141286
{
  "connectionString": "HostName=as141286.azure-devices.net;DeviceId=myEdgeDevice;SharedAccessKey=DUQ43decm2Rqt8D5u01ZEVDZ52gq7elK8qdiUKkwBs="
}
ankit@Azure:~$ ^C
```

The CSV file- <https://github.com/domoritz/random-csv>

Step 4 – Creating a storage account with BLOB and tables.

Creating the BLOB containers to import the CSV file.

- First step is to create the file in the blob containers.
- In my case I used the same containers that I had on docker and imported the CSV file onto them.
- Post this data was ready to be exported to Azure.

Creating the BLOB containers to import the CSV file.

- Setting up the BLOB in a new storage account.
- In my case I had made the cheapest account that was possible and hence uploaded a very small csv file to capture the data in the BLOB containers.

Microsoft Azure Search resources, services, and docs (3 v7)

Home > Create a resource > Marketplace > Storage account

Create a storage account

Basics Advanced Networking Data protection Encryption Tags Review + create

manage your storage account together with other resources.

Subscription * Azure subscription 1

Resource group * (New) BLOBContainers [Create new](#)

Instance details

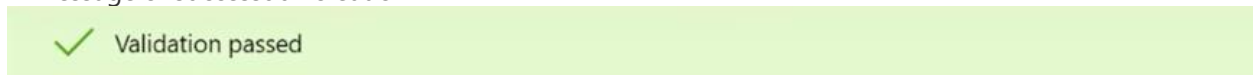
If you need to create a legacy storage account type, please click [here](#).

Storage account name *

Region * (US) East US

Performance * ☒ Standard: Recommended for most scenarios (general-purpose v2 account) ☐ Premium: Recommended for scenarios that require low latency.

- Message of successful creation



Basics Networking Advanced Tags **Review + create**

Basics

Subscription Visual Studio Enterprise

Resource group (New) tablestorage-intro

BLOB Container properties.

Data storage

- Containers
- File shares
- Queues
- Tables

Security + networking

- Networking
- Azure CDN
- Access keys

Static website

Lifecycle management

Azure search

Settings

- Configuration
- Resource sharing (CORS)
- Advisor recommendations
- Endpoints
- Locks

Monitoring

- Insights
- Alerts
- Metrics

Properties Monitoring Capabilities (7) Recommendations Tutorials Developer Tools

Blob service

Property	Value
Hierarchical namespace	Disabled
Default access tier	Hot
Blob public access	Enabled
Blob soft delete	Disabled
Container soft delete	Disabled
Versioning	Disabled
Change feed	Disabled

Security

Property	Value
Secure transfer required	Enabled
Storage account key access	Enabled
Minimum TLS version	Version 1.2
Infrastructure encryption	Disabled

Networking

Property	Value
Allow access from	All networks

Blob service

Resource ID: /subscriptions/7ad168af-d6a9-4286-a218-afc724130a43/resourceGroups/azsjdstorage/providers/Microsoft...

Blob service: <https://azsjdstoragenew.blob.core.windows.net/>

File service

Resource ID: /subscriptions/7ad168af-d6a9-4286-a218-afc724130a43/resourceGroups/azsjdstorage/providers/Microsoft...

File service: <https://azsjdstoragenew.file.core.windows.net/>

Queue service

Resource ID: /subscriptions/7ad168af-d6a9-4286-a218-afc724130a43/resourceGroups/azsjdstorage/providers/Microsoft...

Queue service: <https://azsjdstoragenew.queue.core.windows.net/>

Table service

Resource ID: /subscriptions/7ad168af-d6a9-4286-a218-afc724130a43/resourceGroups/azsjdstorage/providers/Microsoft...

Table service: <https://azsjdstoragenew.table.core.windows.net/>

Step 5 and 6– Creating a table with the URL links to each of the BLOB and visualizing the data to azure.

<https://docs.microsoft.com/en-us/azure/cosmos-db/graph/graph-visualization-partners>

Post this we need to push the same amount of files as well as the structure of the BLOB containers should be similar to those in which the CSV files were stored locally.

Note – I have left the file sharer and queues unperformed as they are not required for this assignment.

Adding the URL to the CSV

The image shows a Jupyter Notebook interface with a code cell and a form below it.

Jupyter Notebook Code:

```
ures-3.3.0 idna-2.10 python-dateutil-2.8.2 requests-2.27.1 six-1.16.0 urllib3-1.26.8
You are using pip version 9.0.1, however version 22.0.3 is available.
You should consider upgrading via the 'pip install --upgrade pip' command.

In [2]: import csv
import sys
import azure.storage
from azure.storage.table import TableService, Entity
from azure.storage.blob import BlockBlobService
from azure.storage.blob import PublicAccess
import time
print('hello at' + time.asctime(time.localtime(time.time())))

hello atThu Feb 24 03:36:11 2022

follow the instructions in the book to create a storage account. the one used here is "tutorial", but you will want to pick something else and replace that name
everywhere below
```

Form Fields:

BlobType	BlockBlob
LeaseState	available
LeaseStatus	unlocked
ServerEncrypted	true
Uri	https://a4edemostor.blob.core.windows.net/public/queue.png
AccessTier	Hot
AccessTierInferred	true
AccessTierChangeTime	

Now, Once the URL to each of the csv file is added, we need to create a new container for a new CSV file.

Once that is done we need to repeat the same steps as above to create and add new URL so that we can reach the same BLOB Containers.

Tables

```
In [4]: tice = TableService(account_name=account,
                           account_key=account_key)

In [5]: if table_service.create_table('DataTable'):
        print("table created")
        else:
        print("table already there")

table already there

Now you need a directory with a csv file. For example

'experiment1', '1', '3/15/2002', 'exp1', 'this is the comment'
'experiment1', '2', '3/15/2002', 'exp2', 'this is the comment2'
'experiment2', '3', '3/16/2002', 'exp3', 'this is the comment3'
'experiment3', '4', '3/16/2002', 'exp4', 'this is the comment233'

now you need a directory datafiles that has four blobs of any type. call them exp1, exp2, exp3, exp4.
we have one ready for you if you are using the tutorial container. It is stored in /datadir. If you are running this somewhere else you will need to make your own.
let's see what is there.

In [6]: with open('/datadir/experiments.csv', 'rb') as csvfile:
        csvf = csv.reader(csvfile, delimiter=',', quotechar='"')
        for item in csvf:
            print(item)
```

Uploading data

The screenshot shows a Jupyter Notebook with the following code and output:

```
[ 'experiment1', ' id1', ' 12/1/2016', 'a.jpg', ' "here is a view of what is to be made"' ]
[ 'experiment1', ' id2', ' 12/3/2016', 'b.jpg', ' "map reduce picture"' ]
[ 'experiment2', ' id3', ' 12/4/2016', 'c.jpg', ' "sample notebook"' ]
[ 'experiment3', ' id4', ' 12/5/2016', 'd.jpg', ' "workers"' ]
[ 'experiment1', ' id5', ' 12/6/2016', 'e.jpg', ' "bio samples"' ]

Next let's do a query for experiment1 and project onto the urls. it should print the urls created above

In [8]: tasks = table_service.query_entities('DataTable', filter="PartitionKey eq 'experiment1'", select="url")
        for task in tasks:
            print(task.url)
```

Below the code, a table view displays the data:

PARTITIONKEY	ROWKEY	TIMESTAMP	DATE	URL	DESCRIPTION
experiment1	id1	2021-02-17T03:06:10.845647Z	12/1/2016	https://oclab2.blob.core.windows.net/datacont/a.jpg	"here is a view of what is to be made"
experiment1	id2	2021-02-17T03:06:10.9867438Z	12/3/2016	https://oclab2.blob.core.windows.net/datacont/b.jpg	"map reduce picture"
experiment1	id5	2021-02-17T03:06:11.2919557Z	12/6/2016	https://oclab2.blob.core.windows.net/datacont/e.jpg	"bio samples"
experiment2	id3	2021-02-17T03:06:11.0968201Z	12/4/2016	https://oclab2.blob.core.windows.net/datacont/c.jpg	"sample notebook"
experiment3	id4	2021-02-17T03:06:11.1728713Z	12/5/2016	https://oclab2.blob.core.windows.net/datacont/d.jpg	"workers"

FINAL PART _ CLEANING UP THE RESOURCES

Final screenshot before deletion

Microsoft Azure

Search resources, services, and docs (G+)

Azure services

- Create a resource
- Resource groups
- All resources
- Quickstart Center
- Virtual machines
- App Services
- Storage accounts
- SQL databases
- Azure Cosmos DB
- More services

Recent resources

Name	Type	Last Viewed
as141286	IoT Hub	11 minutes ago
as14128	Resource group	26 minutes ago

Navigate

- Subscriptions
- Resource groups
- All resources
- Dashboard

Tools

- Microsoft Learn: Learn Azure with free online training from Microsoft
- Azure Monitor: Monitor your apps and infrastructure
- Security Center: Secure your apps and infrastructure
- Cost Management: Analyze and optimize your cloud spend for free

Post Deletion

Microsoft Azure

Search resources, services, and docs (G+)

Home >

Resource not found

Resource not found

Get support Perform self-diagnostics

ds72b6da062e466ab15d65d358039f13 /subscriptions/63c50fad-40c7-4df4-9f7d-39c17a16b5a...

Extension
HubsExtension

Content
ResourceMenuBlade

Error code
404

Details
The resource was not found, it may have been deleted. If this was launched from a pinned tile on the dashboard, it should be removed.

Resource ID: /subscriptions/63c50fad-40c7-4df4-9f7d-39c17a16b5a0/resourcegroups/as14128/providers/Microsoft.Devices/IotHubs/as141286

Status Code: 404

Status Message: Resource group 'as14128' could not be found.

Activity log snippet

Notifications



[More events in the activity log →](#)

[Dismiss all](#)

✓ Deleted resource group as14128

Deleted resource group as14128

2 minutes ago

✓ Set Modules

Successfully updated IoT Edge settings for device myEdgeDevice.

13 minutes ago

✓ Deployment succeeded

Deployment 'as141286-1114192541' to resource group 'as14128' was successful.

[Go to resource](#)

[Pin to dashboard](#)

53 minutes ago

Again repeating the same steps on AWS.

1. Setting up the AWS account with the required instances.

Required instances.

- ➔ **EC2(Compute server)** – This is basically a **regular server** instance that is used to deploy and the required resources over the VM as per the choices made by the users.
 - This is used to deploy the VM.
 - Manage resources over those VM's.
 - Finally to migrate services and monitor volumes.
- ➔ **S3(Storage utility)** – This is a basic protocol that acts like a storage bucket.
 - The prime feature of this protocol is to deal with the data as per service request.
 - We need this to store the data in the **data buckets** which are later used to store and move the data across volumes created.

Initial account setup.

aws

Services

Search for services, features, blogs, docs, and more

[Alt+S]

📄

🔔

🔍

N. Virginia

Ankit1996

AWS Management Console

AWS services

▶ Recently visited services

▶ All services

Build a solution

Get started with simple wizards and automated workflows.

Launch a virtual machine

With EC2

2-3 minutes

Build a web app

With Elastic Beanstalk

6 minutes

Build using virtual servers

With Lightsail

1-2 minutes

Register a domain

With Route 53

Connect an IoT device

With AWS IoT

Start migrating to AWS

With AWS MGN

New AWS Console Home

See valuable insights for your account and services with the new customizable Console Home experience. [Learn more](#)

Switch now

Stay connected to your AWS resources on-the-go

AWS Console Mobile App now supports four additional regions. Download the AWS Console Mobile App to your iOS or Android mobile device. [Learn more](#)

Explore AWS

Machine Learning University

Find self-service machine learning training from

Feedback

English (US)

© 2022, Amazon Web Services, Inc. or its affiliates.

Privacy

Terms

Cookie preferences

Launch Status

✓

Your instances are now launching
The following instance launches have been initiated: i-045e181e24b79803b [View launch log](#)

ⓘ

Get notified of estimated charges
Create billing alerts to get an email notification when estimated charges on your AWS bill exceed an amount you define (for example, if you exceed the free usage tier).

How to connect to your instances

Your instances are launching, and it may take a few minutes until they are in the **running** state, when they will be ready for you to use. Usage hours on your new Instances will start immediately and continue to accrue until you stop or terminate your instances.

Click **View Instances** to monitor your instances' status. Once your instances are in the **running** state, you can **connect** to them from the Instances screen. [Find out](#) how to connect to your instances.

▼ Here are some helpful resources to get you started

- How to connect to your Linux instance
- Learn about AWS Free Usage Tier
- Amazon EC2: User Guide
- Amazon EC2: Discussion Forum

While your instances are launching you can also

- Create status check alarms to be notified when these instances fail status checks. (Additional charges may apply)
- Create and attach additional EBS volumes (Additional charges may apply)
- Manage security groups

aws

Services

Search for services, features, blogs, docs, and more

[Alt+S]

📄

🔔

🔍

N. Virginia

Ankit199

New EC2 Experience

Tell us what you think

EC2 Dashboard

EC2 Global View

Events

Tags

Limits

▼ Instances

Instances **New**

Instance Types

Launch Templates

Spot Requests

Savings Plans

Reserved Instances **New**

Dedicated Hosts

Scheduled Instances

Capacity Reservations

▼ Images

AMIs **New**

AMI Catalog

▼ Elastic Block Store

EC2 > Instances > i-045e181e24b79803b

Instance summary for i-045e181e24b79803b

Info

Updated less than a minute ago

🔄

Connect

Instance state ▼

Actions ▼

Instance ID

i-045e181e24b79803b

IPv6 address

-

Hostname type

IP name: ip-172-31-86-38.ec2.internal

Instance type

t2.micro

AWS Compute Optimizer finding

Opt-in to AWS Compute Optimizer for recommendations. [Learn more](#)

Public IPv4 address

3.95.225.191 | [open address](#)

Instance state

Running

Private IP DNS name (IPv4 only)

ip-172-31-86-38.ec2.internal

Elastic IP addresses

-

IAM Role

-

Private IPv4 addresses

172.31.86.38

Public IPv4 DNS

ec2-3-95-225-191.compute-1.amazonaws.com | [open address](#)

Answer private resource DNS name

-

VPC ID

vpc-0f3baf8a7df1d49c1 [🔗](#)

Subnet ID

subnet-0963b1e174ce8e45f [🔗](#)

Details

Security

Networking

Storage

Status checks

Monitoring

Tags

▼ Instance details

Info

Platform

SUSE Linux (Inferred)

AMI ID

ami-08895422b5f3aa64a

Monitoring

disabled

2. Step 2 and 3 – SSH into the instance that we have created.

```
ankit@LAPTOP-S2U1QMGB:/mnt/c/Users/ankit$ sudo apt-get install -y kubectl
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following NEW packages will be installed:
  kubectl
0 upgraded, 1 newly installed, 0 to remove and 112 not upgraded.
Need to get 8929 kB of archives.
After this operation, 46.6 MB of additional disk space will be used.
Get:1 https://packages.cloud.google.com/apt/kubernetes-xenial/main amd64 kubectl amd64 1.23.3-00 [8929 kB]
Fetched 8929 kB in 1s (10.5 MB/s)
Selecting previously unselected package kubectl.
(Reading database ... 32226 files and directories currently installed.)
Preparing to unpack .../kubectl_1.23.3-00_amd64.deb ...
Unpacking kubectl (1.23.3-00) ...
Setting up kubectl (1.23.3-00) ...
ankit@LAPTOP-S2U1QMGB:/mnt/c/Users/ankit$ |
```

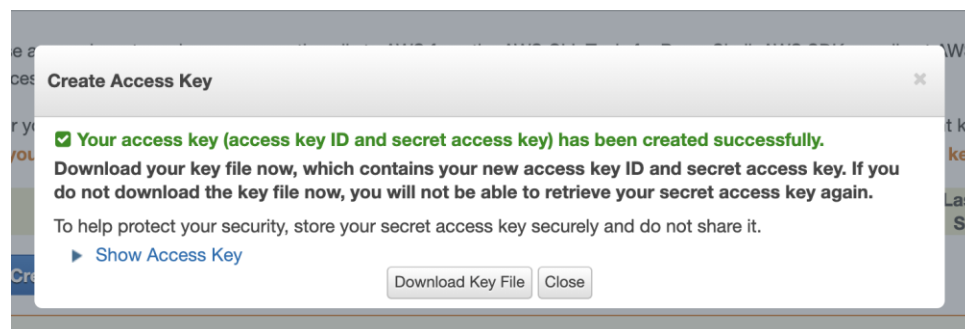
➔ Saving the required keys as mentioned in the assignment.

Key.pub identification

Type – RSA 4096

Key – Screenshot attached (Blurred the actual key for privacy)

```
ankit@LAPTOP-S2U1QMGB:~/tce-linux-amd64-v0.9.1$ ssh-keygen -t rsa -b 4096 -C "jct@archemyl.com"
Generating public/private rsa key pair.
Enter file in which to save the key (/home/ankit/.ssh/id_rsa): Key
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in Key
Your public key has been saved in Key.pub
The key fingerprint is:
SHA256:JhV8DJE4Jv00XxwKwNcVU...
The key's randomart image is:
+---[RSA 4096]-----+
|  . o*=XBB00=|
|  . =o+.o*Bo|
|  o +. =.o.*|
|  .. . . 000|
|  . S      ..|
|  o       E   |
|              |
+---[SHA256]-----+
ankit@LAPTOP-S2U1QMGB:~/tce-linux-amd64-v0.9.1$ |
```



➔ The final build needs to be created on the GUI first.

➔ Post that we need to setup the EC2 instance and the S3 storage bucket

➔ Finally, we can **ssh** into the created instance.

- **Screenshot of the final build**

```

ankit@LAPTOP-S2U1QMGB:/mnt/c/Users/ankit$ sudo apt-get install -y kubectl
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following NEW packages will be installed:
  kubectl
0 upgraded, 1 newly installed, 0 to remove and 112 not upgraded.
Need to get 8929 kB of archives.
After this operation, 46.6 MB of additional disk space will be used.
Get:1 https://packages.cloud.google.com/apt/kubernetes-xenial/main amd64 kubectl amd64 1.23.3-00 [8929 kB]
Fetched 8929 kB in 1s (10.5 MB/s)
Selecting previously unselected package kubectl.
(Reading database ... 32226 files and directories currently installed.)
Preparing to unpack .../kubectl_1.23.3-00_amd64.deb ...
Unpacking kubectl (1.23.3-00) ...
Setting up kubectl (1.23.3-00) ...
ankit@LAPTOP-S2U1QMGB:/mnt/c/Users/ankit$

```

The CSV file- <https://github.com/domoritz/random-csv>

Step 4 – NoSQL DB in AWS

```

Provisioned read capacity units    1
Provisioned write capacity units   1
Last decrease time                 -
Last increase time                 -
Storage size (in bytes)            0 bytes
Item count                        0
Region                             US East (N. Virginia)
Amazon Resource Name (ARN)         arn:aws:dynamodb:us-east-1:367622474624:table/simplilearn_users

```

Storage size and item count are not updated in real-time. They are updated periodically, roughly every six hours.

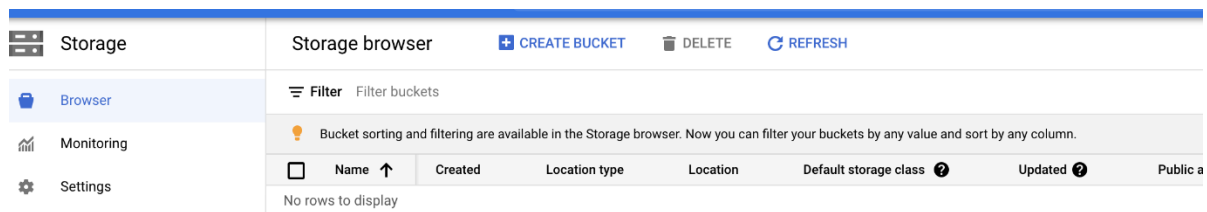
After running the notebook, go to aws console and go to dynamodb from there. Click on tables and under that click on DataTable and click on items tab to view the contents of the table.

The screenshot shows the AWS Management Console interface for a DynamoDB table named 'DataTable'. The 'Items' tab is selected, displaying a list of 5 items. The table has two primary keys: 'PartitionKey' and 'RowKey'. The 'Items' tab shows a list of items with columns for PartitionKey, RowKey, date, description, and uri.

	PartitionKey	RowKey	date	description	uri
<input type="checkbox"/>	experiment1	id1	12/1/2016	"here is a view of what is to be made"	https://cc-lab2.s3.
<input type="checkbox"/>	experiment1	id2	12/3/2016	"map reduce picture"	https://cc-lab2.s3.
<input type="checkbox"/>	experiment1	id5	12/6/2016	"bio samples"	https://cc-lab2.s3.
<input type="checkbox"/>	experiment2	id3	12/4/2016	"sample notebook"	https://cc-lab2.s3.
<input type="checkbox"/>	experiment3	id4	12/5/2016	"workers"	https://cc-lab2.s3.

GCP

Login to GCP account and go to Storage from the navigation panel.



Click on create bucket and give a valid name.

Once the bucket is created, it should be visible

Once done, go to terminal and run the following command:

```
docker run -i -t -p 8888:8888 dbgannon/tutorial
```

After this go to local host link and run the python notebook called gcloud.ipynb after making necessary edits related to bucket name and file paths.

```

In [22]: client = storage.Client()

In [23]: from gcloud import datastore
         clientds = datastore.Client()

In [29]: import csv

In [53]: bucket = client.bucket('cc-lab2')
         key = clientds.key('book-table')

In [55]: with open('/datadir/experiments.csv', 'rt') as csvfile:
         csvf = csv.reader(csvfile, delimiter=',', quotechar='"')
         for item in csvf:
             print(item)
             blob = bucket.blob(item[3])
             data = open("/datadir/"+item[3], 'rb')
             blob.upload_from_file(data)
             blob.make_public()
             url = "https://console.cloud.google.com/storage/browser/cc-lab2/"+item[3]
             entity = datastore.Entity(key=key)
             entity['experiment-name'] = item[0]
             entity['experiment-id'] = item[1]
             entity['date'] = item[2]
             entity['description'] = item[4]
             entity['url'] = url

```

Once the above steps are executed, go to GCP again and go to storage to see the uploaded objects.

Buckets > cc-lab2

[UPLOAD FILES](#)
[UPLOAD FOLDER](#)
[CREATE FOLDER](#)
[MANAGE HOLDS](#)
[DOWNLOAD](#)
[DELETE](#)

Filter by name prefix only Filter objects and folders

<input type="checkbox"/>	Name	Size	Type	Created time	Storage class	Last modified	Public access	Encryption	Retention
<input type="checkbox"/>	a	54.5 KB	application/octet-stream	18 Feb 2021, ...	Standard	18 Feb 202...	Public to Internet	Copy URL	Google-managed key
<input type="checkbox"/>	b	68 KB	application/octet-stream	18 Feb 2021, ...	Standard	18 Feb 202...	Public to Internet	Copy URL	Google-managed key
<input type="checkbox"/>	c	86.2 KB	application/octet-stream	18 Feb 2021, ...	Standard	18 Feb 202...	Public to Internet	Copy URL	Google-managed key
<input type="checkbox"/>	d	41.8 KB	application/octet-stream	18 Feb 2021, ...	Standard	18 Feb 202...	Public to Internet	Copy URL	Google-managed key
<input type="checkbox"/>	e	114.9 KB	application/octet-stream	18 Feb 2021, ...	Standard	18 Feb 202...	Public to Internet	Copy URL	Google-managed key

Check the datastore to have a look at created table

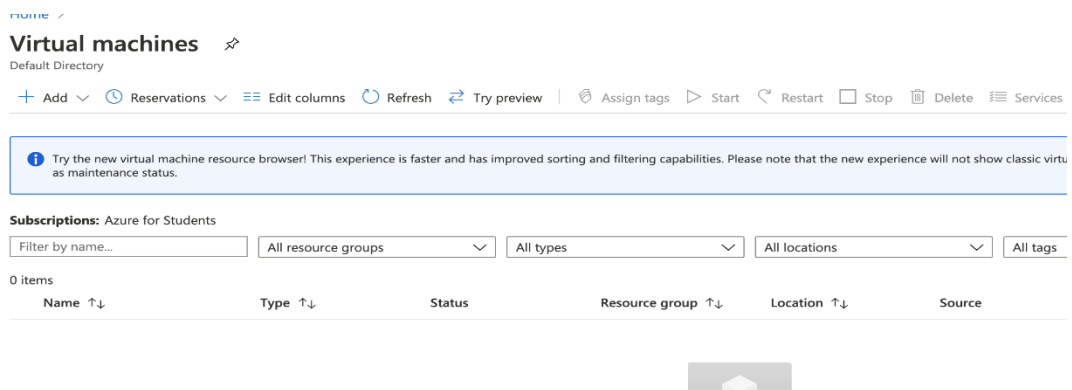
<div> <div>Search</div> <div>Data</div> <div>Indexes</div> <div>Import/Export</div> <div>Usage</div> <div>Security rules</div> </div>	<div> <div>/ > book-table > __id5644004762845184__</div> <div>Root</div> <div> <div>+ START COLLECTION</div> <div>book-table</div> </div> </div>	<div> <div>book-table</div> <div> <div>+ ADD DOCUMENT</div> <div> <div>__id5632499082330112__</div> <div>__id5634161670881280__</div> <div>__id5642368648740864__</div> <div>__id5644004762845184__</div> <div>__id5704568633556992__</div> </div> </div> </div>	<div> <div>__id5644004762845184__</div> <div> <div>+ START COLLECTION</div> <div>+ ADD FIELD</div> <div> <div>date: '12/3/2016'</div> <div>description: "map reduce picture"</div> <div>experiment-id: 'id2'</div> <div>experiment-name: 'experiment1'</div> <div>url: 'https://console.cloud.google.com/storage/browser/cc...</div> </div> </div> </div>
---	--	--	---

PART 2

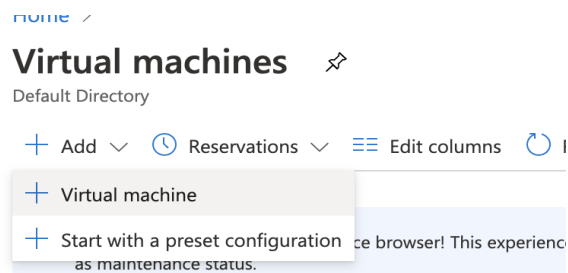
Generate ssh key using ssh-key gen and following interactive instructions as follows:

```
ankit@LAPTOP-S2U1QMG8:~/tce-linux-amd64-v0.9.1$ ssh-keygen -t rsa -b 4096 -C "jet@archemyl.com"
Generating public/private rsa key pair.
Enter file in which to save the key (/home/ankit/.ssh/id_rsa): Key
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in Key
Your public key has been saved in Key.pub
The key fingerprint is:
SHA256:JhV8DJE4Jv00XXwkWNCVU [REDACTED]
The key's randomart image is:
+---[RSA 4096]-----+
|  . o*+=XBB00=|
|  . =o+.=o*Bo|
|  o +.= . =o.*|
|  .. . . 000|
|  . S      ..|
|  o      E   |
|  |          |
+---[SHA256]-----+
ankit@LAPTOP-S2U1QMG8:~/tce-linux-amd64-v0.9.1$
```

Go to azure portal and go to virtual machines from the navigation menu



Click on Add button and after that click on Virtual machine,

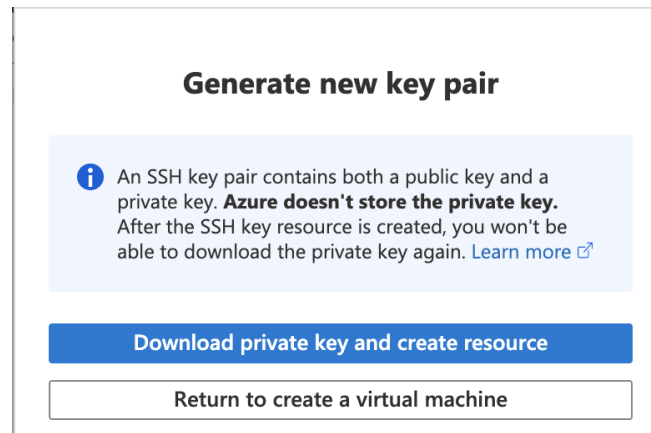


Enter necessary details on the creation page and then click on Review + create

After validation is passed, click on create

It will prompt to download the generated key pairs, download it.

Once the download is completed, click on "Go to resource"



Notice the public IP address of the VM. In order to ssh to this vm, open the terminal and type the command as shown in figure.

```
Welcome to Ubuntu 20.04.3 LTS (GNU/Linux 5.11.0-1028-azure x86_64)

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

System information as of Fri Feb 25 00:56:23 UTC 2022

System load:  0.06          Processes:           130
Usage of /:   4.8% of 28.9GB Users logged in:      0
Memory usage: 3%           IPv4 address for eth0: 10.0.0.4
Swap usage:   0%

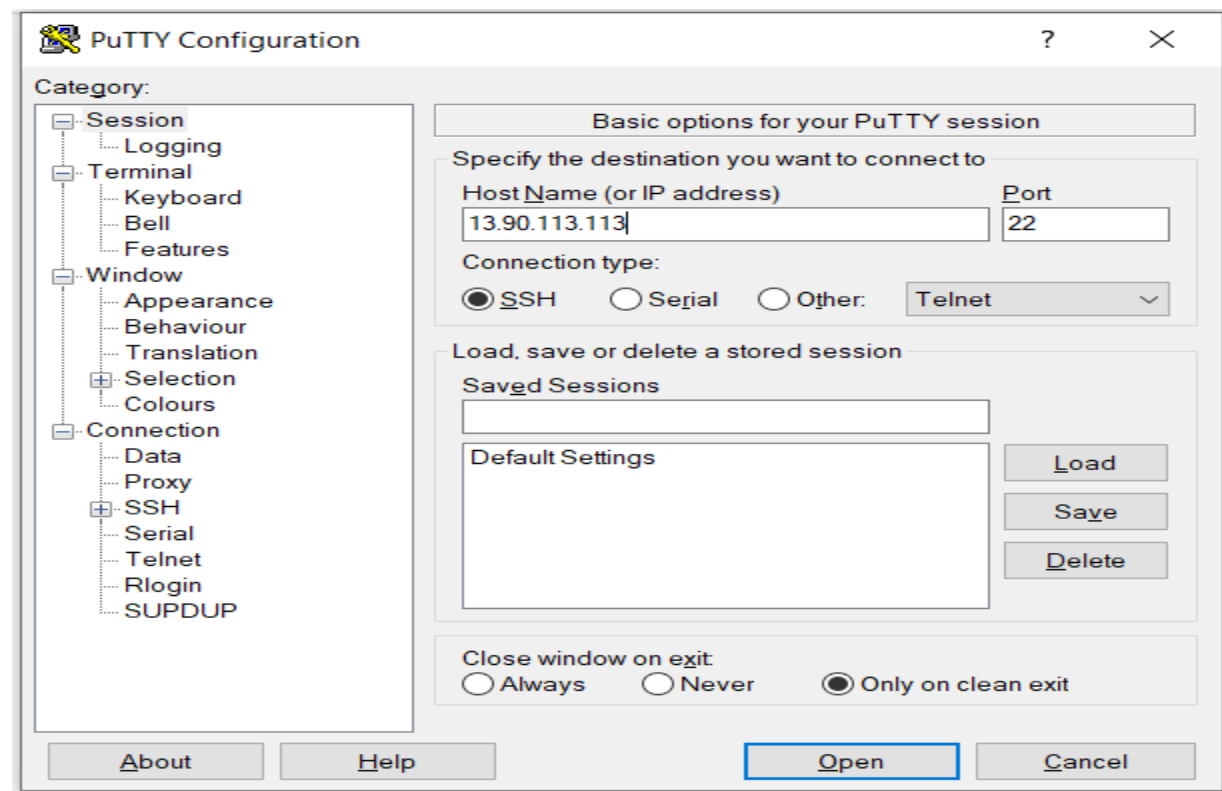
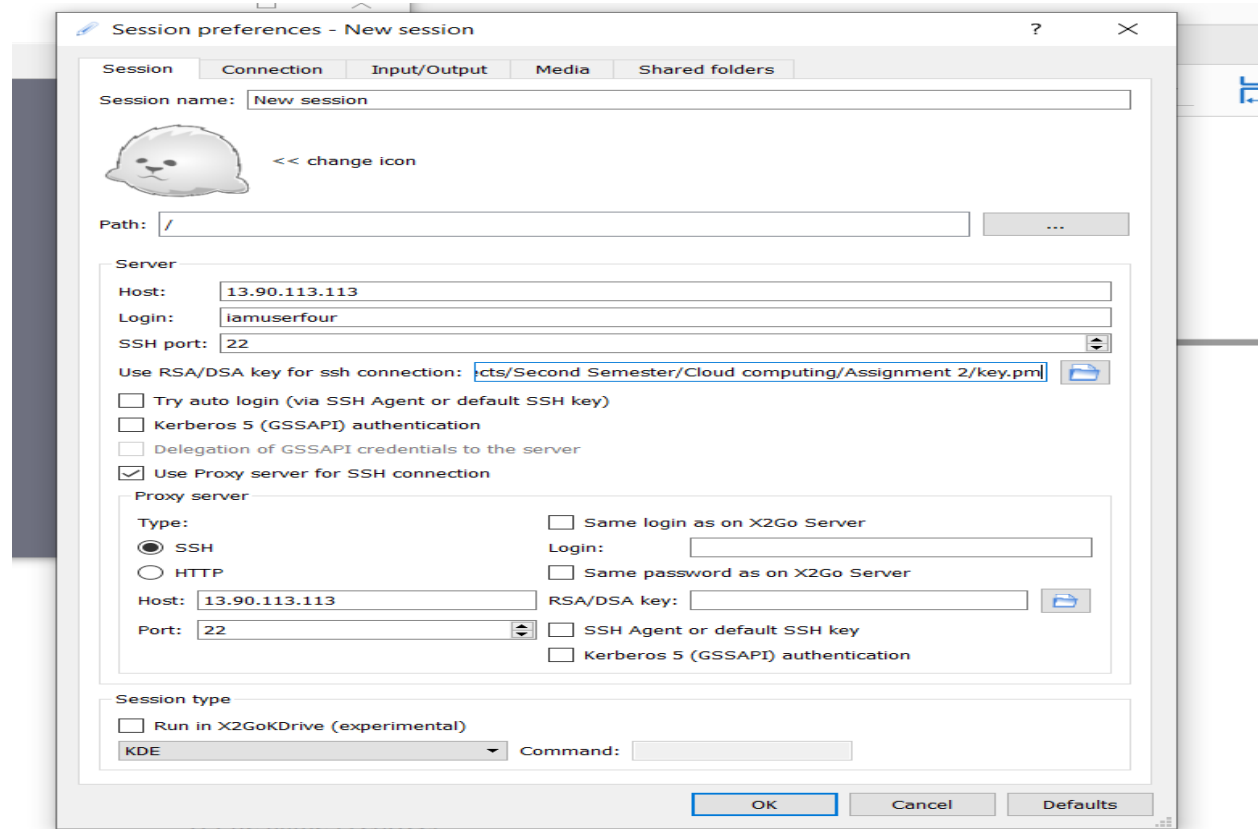
1 update can be applied immediately.
To see these additional updates run: apt list --upgradable

The list of available updates is more than a week old.
To check for new updates run: sudo apt update

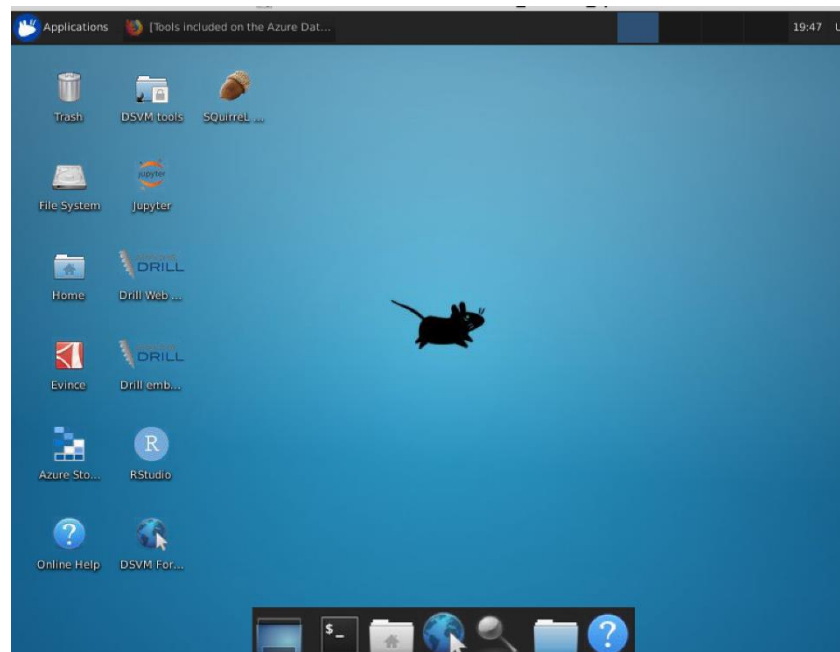
The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

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


Download the **x2go** client software as well the **PuTTY** client for ssh.



Start the session, a window will appear as below:



Enter the username and password of the virtual machine in order to login to jupyter hub

Once done, jupyter notebooks should be visible

