

Name: Purva Burundkar  
Sap Id: 86092300018  
Roll no: A018

## Practical 1: Infrastructure as a Service using AWS

**SVKM'S NMIM'S Nilkamal School of Mathematics,  
Applied Statistics & Analytics  
Master of Science (Data Science)**

Practical-1 Infrastructure as a service using AWS.

Writeup:-

- Cloud Computing architecture
- IAAS
- AWS
- EC2

1. Implement the windows machine using AWS ec2.
2. Implement Ubuntu machine using AWS ec2 and execute the Linux commands.

### Cloud computing - IAAS(Infrastructure as storage)

- STORAGE → COMPUTING → NETWORKING → VIRTUALIZATION → O.S → RUNTIME → APPLICATION → **DATA** (layers of CC)
- FROM VIRTUALIZATION TO DATA IT WILL BE **IAAS**
- DATA + APPLICATION WILL BE **PAAS**
- DATA CAN BE ACCESSED BY **SAAS**

### WRITE UP:

#### **Cloud Computing Architecture:**

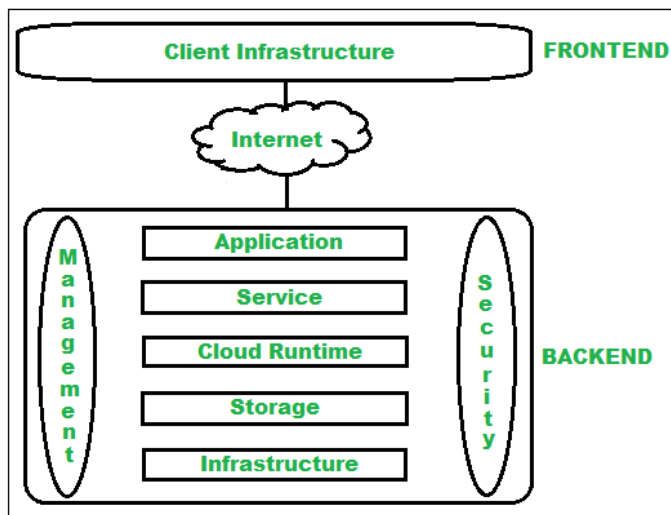
As far as we are aware, cloud computing technology allows both big and small businesses to store data on the cloud and retrieve it whenever and from wherever they have an internet connection.

Event-driven and service-oriented architectures are combined in cloud computing architecture.

The architecture of cloud computing is separated into the following two sections:

Front End  
Back End

The below diagram shows the architecture of cloud computing -



### Front End

The client uses the front end. It has the applications and client-side interfaces needed to access cloud computing platforms. Web servers (such as Internet Explorer, Firefox, and Chrome), thin and fat clients, tablets, and mobile devices make up the front end.

### Back End

The service provider uses the back end. It oversees every resource needed to deliver cloud computing services. A vast amount of data storage, servers, virtual machines, traffic control mechanisms, deployment models, and security measures are all included.

Through a network, usually the internet connection, the front end and back end are connected to each other.

### Elements of the Architecture of Cloud Computing

The architecture of cloud computing consists of the following elements:

#### 1. Client Infrastructure

One component of the front end is client infrastructure. It offers a graphical user interface (GUI) for cloud interaction.

#### 2. Application

Any program or platform that a client wishes to use can be the application.

#### 3. Service

Depending on the needs of the client, a cloud service controls the kind of service you can access.

Cloud computing offers the following three type of services:

**i. Software as a Service (SaaS)** – It is also known as cloud application services. Mostly, SaaS applications run directly through the web browser means we do not require to download and install these applications. Some important example of SaaS is given below – Example: Google Apps, Salesforce Dropbox, Slack, Hubspot, Cisco WebEx.

**ii. Platform as a Service (PaaS)** – It is also known as cloud platform services. It is quite similar to SaaS, but the difference is that PaaS provides a platform for software creation, but using SaaS, we can access software over the internet without the need of any platform. Example: Windows Azure, Force.com, Magento Commerce Cloud, OpenShift.

**iii. Infrastructure as a Service (IaaS)** – It is also known as cloud infrastructure services. It is responsible for managing applications data, middleware, and runtime environments.

Example: Amazon Web Services (AWS) EC2, Google Compute Engine (GCE), Cisco Metapod.

#### **4. Cloud Runtime**

The virtual machines' execution and runtime environment is supplied by Runtime Cloud.

#### **5. Storage**

One of the key elements of cloud computing is storage. It offers enormous cloud storage capacity for managing and storing data.

#### **6. Establishment of Infrastructure**

It offers services at the network, application, and host levels. In order to support the cloud computing model, cloud infrastructure consists of hardware and software elements like servers, storage, network devices, virtualization software, and additional storage resources.

#### **7. Management**

Management is the process of overseeing and coordinating various backend components, including applications, services, runtime clouds, infrastructure, storage, and other security concerns.

#### **8. Security**

The cloud computing system comes with built-in security. It puts in place a security measure on the back end.

#### **9. Internet**

Front end and back end can communicate and interact with one another via the Internet.

### **IAAS:**

Infrastructure as a service (IaaS) is the on-demand availability of highly scalable computing resources as services over the internet. It eliminates the need for enterprises to procure, configure, or manage infrastructure themselves, and they only pay for what they use.

IaaS in cloud computing is when you rent access to cloud infrastructure resources as individual services from a cloud service provider (CSP), including servers, virtual machines,

networking resources, and storage. IaaS helps eliminate much of the complexity and costs associated with building and maintaining physical infrastructure in an on-premises data centre.

The CSP is responsible for managing and maintaining the infrastructure, so you can concentrate on installing, configuring, and managing software and keeping your data secure. IaaS providers also offer additional services, such as detailed billing management, logging, monitoring, storage resiliency, and security.

You can access IaaS resources using a pay-as-you-go basis, allowing you to only pay to consume the resources that you need. In other words, you can easily increase or decrease resources, allowing you to pay less when needed or instantly provision and scale out resources to meet new demand.

## **AWS:**

AWS stands for Amazon Web Services. It is an expanded cloud computing platform provided by Amazon Company. AWS provides a wide range of services with a pay-as-per-use pricing model over the Internet such as Storage, Computing power, Databases, Machine Learning services, and much more. AWS facilitates for both businesses and individual users with effectively hosting the applications, storing the data securely, and making use of a wide variety of tools and services improving management flexibility for IT resources.

AWS comes up with its own network infrastructure on establishing the datacentres in different regions mostly all over the world. Its global Infrastructure acts as a backbone for operations and services provided by AWS. It facilitates the users on creating secure environments using Amazon VPCs (Virtual Private Clouds). Essential services like Amazon EC2 and Amazon S3 for utilizing the compute and storage service with elastic scaling. It supports the dynamic scaling of the applications with the services such as Auto Scaling and Elastic Load Balancing (AWS ELB). It provides a good user-friendly AWS Management Console facilitating seamless configuration and management of AWS services to the Users. Its Architecture ensures high availability, fault tolerance making AWS as a versatile powerful Cloud Computing Platform.

The following are some of the main fundamentals of AWS:

**Regions:** AWS provides the services with respective division of regions. The regions are divided based on geographical areas/locations and will establish data centres. Based on need and traffic of users, the scale of data centres is depended to facilitate users with low-latencies of services.

**Availability Zones (AZ):** To prevent the Data centres for the Natural Calamities or any other disasters. The Datacentres are established as sub sections with isolated locations to enhance fault tolerance and disaster recovery management.

**Global Network Infrastructure:** AWS ensures the reliability and scalability of services through setting up its own AWS Network Infrastructure globally. It helps in better management of data transmissions for optimized performance and security reliance.

## Top AWS Services

In the rapid revolution of Cloud Computing, AWS facilitates with wide variety of services respect to the fields and needs. The following are the top AWS services that are in wide usage:

**Amazon EC2(Elastic Compute Cloud):** Amazon EC2, part of Amazon Web Services, offers resizable compute capacity in the cloud through virtual servers called instances. Users can choose from various instance types, pre-configured machine images (AMIs), and pricing models. EC2 provides features like security groups, Virtual Private Cloud (VPC), elastic load balancing, auto scaling, and Elastic Block Store (EBS). Widely used for diverse applications, EC2 enables on-demand scaling, flexibility, and cost-effectiveness in managing compute resources.

**Amazon S3 (Simple Storage Service):** It offers scalable object Storage as a Service with high durability for storing and retrieving any amount of data.

**AWS Lambda:** It is a service in Serverless Architecture with Function as a Service facilitating serverless computing i.e., running the code on response to the events, the background environment management of servers is handled by aws automatically. It helps the developers to completely focus on the logic of code build.

**Amazon RDS (Relational Database Service):** This is an aws service that simplifies the management of database providing high available relational databases in the cloud.

**Amazon VPC (Virtual Private Cloud):** It enables the users to create isolated networks with option of public and private expose within the AWS cloud, providing safe and adaptable configurations of their resources.

## IAAS USING EC2 (Elastic Compute Cloud)

### 1. Implement Windows Machine using AWS EC2.

Go to [aws.amazon.com](https://aws.amazon.com) and Sign Up by providing all the information.



## Congratulations

Thank you for signing up for AWS.

We are activating your account, which should only take a few minutes. You will receive an email when this is complete.

[Go to the AWS Management Console](#)

[Sign up for another account](#) or [contact sales](#).



## After signing in, Login and go to EC2 Service

### All services

#### Services by category

##### Compute

EC2  
Lightsail  
Lambda  
Batch  
Elastic Beanstalk  
Serverless Application  
Repository  
AWS Outposts  
EC2 Image Builder  
AWS App Runner  
AWS SimSpace Weaver

##### Containers

Elastic Container Registry  
Elastic Container Service  
Elastic Kubernetes Service  
Red Hat OpenShift Service on AWS

##### Storage

S3  
EFS  
FSx  
S3 Glacier  
Storage Gateway  
AWS Backup  
AWS Elastic Disaster

##### Developer Tools

CodeStar  
CodeCommit  
CodeBuild  
CodeDeploy  
CodePipeline  
Cloud9  
CloudShell  
X-Ray  
AWS FIS  
CodeArtifact  
Amazon CodeCatalyst  
AWS AppConfig  
Amazon CodeWhisperer  
Application Composer

##### Customer Enablement

AWS IQ  
Managed Services  
Activate for Startups  
Support  
AWS re:Post Private

##### Robotics

AWS RoboMaker

##### Blockchain

Amazon Managed

##### Machine Learning

Amazon SageMaker  
Amazon Augmented AI  
Amazon CodeGuru  
Amazon DevOps Guru  
Amazon Comprehend  
Amazon Forecast  
Amazon Fraud Detector  
Amazon Kendra  
Amazon Personalize  
Amazon Polly  
Amazon Rekognition  
Amazon Textract  
Amazon Transcribe  
Amazon Translate  
AWS DeepComposer  
AWS DeepLens  
AWS DeepRacer  
AWS Panorama  
Amazon Monitron  
AWS HealthLake  
Amazon Lookout for Vision  
Amazon Lookout for Equipment  
Amazon Lookout for Metrics  
Amazon Lex  
Amazon Comprehend Medical  
Amazon Rekognition

##### Cloud Financial Management

AWS Marketplace  
Subscriptions  
AWS Billing Conductor  
Billing and Cost  
Management

##### Front-end Web & Mobile

AWS Amplify  
AWS AppSync  
Device Farm  
Amazon Location Service

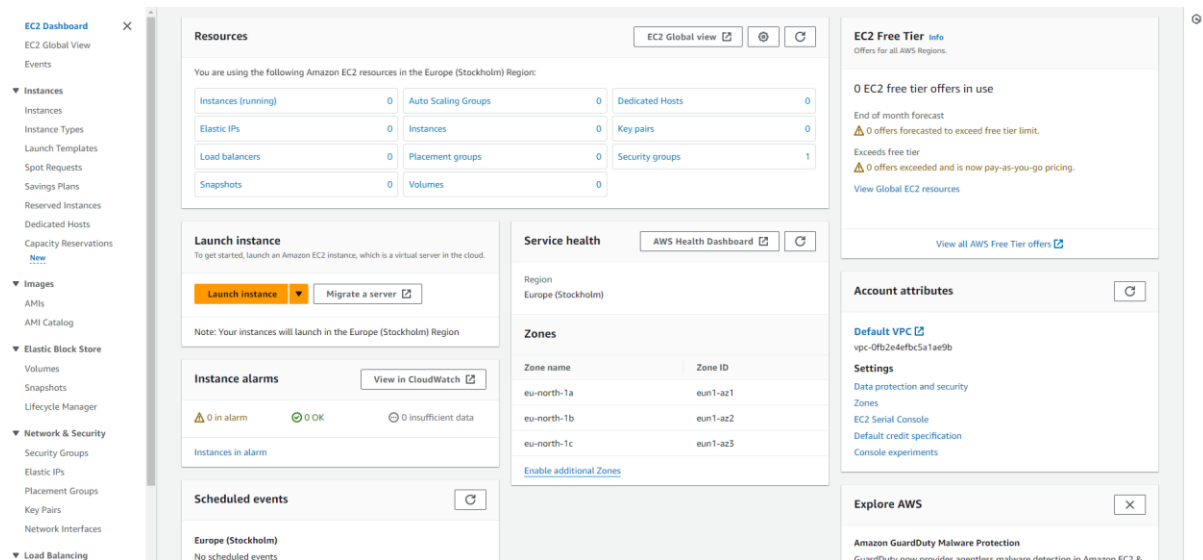
##### Application Integration

Step Functions  
Amazon AppFlow  
Amazon EventBridge  
Amazon MQ  
Simple Notification Service  
Simple Queue Service  
SWF  
Managed Apache Airflow  
AWS B2B Data Interchange

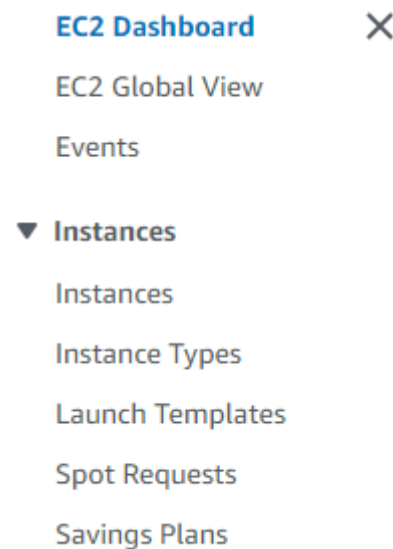
##### Business Applications

Amazon Connect

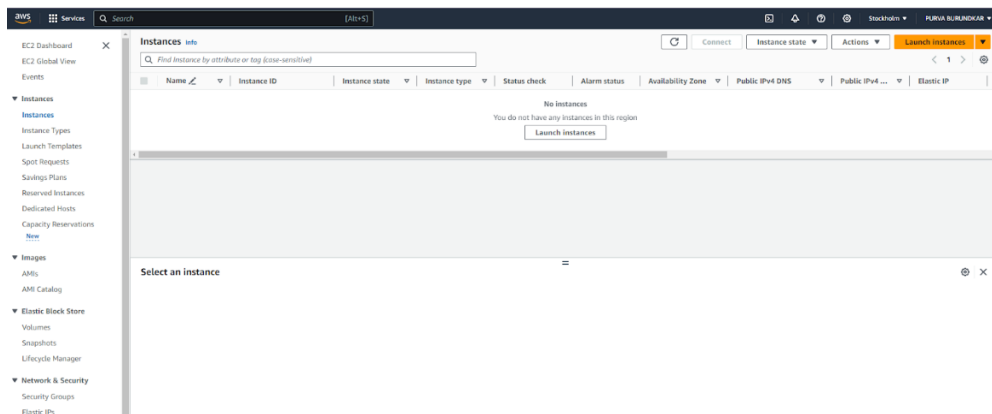
On the left side bar, click on Instances



## CHOOSE INSTANCES



Now click on Launch Instance:



## LAUNCH INSTANCE

### Launch instance

To get started, launch an Amazon EC2 instance, which is a virtual server in the cloud.

Launch instance

▼

Migrate a server

**Note:** Your instances will launch in the Europe (Stockholm) Region

EC2 > Instances > Launch an Instance

#### Launch an instance [Info](#)

Amazon EC2 allows you to create virtual machines, or instances, that run on the AWS Cloud. Quickly get started by following the simple steps below.

**Name and tags** [Info](#)

Name

MyWindows123

Add additional tags

**▼ Application and OS Images (Amazon Machine Image)** [Info](#)

An AMI is a template that contains the software configuration (operating system, application server, and applications) required to launch your instance. Search or Browse for AMIs if you don't see what you are looking for below

Q Search our full catalog including 1000s of application and OS images

Quick Start

Amazon Linux

macOS

Ubuntu

Windows

Red Hat

SUSE Linux

Browse more AMIs

Including AMIs from AWS, Marketplace and the Community

**▼ Summary**

Number of instances [Info](#)

1

Software Image (AMI)

Microsoft Windows Server 2022 ...read more  
ami-099a7b25010174ee4

Virtual server type (instance type)

t3.micro

Firewall (security group)

New security group

Storage (volumes)

1 volume(s) - 30 GiB

**Free tier:** In your first year includes 750 hours of t2.micro (or t3.micro in the Regions in which t2.micro is unavailable) instance usage on free tier AMIs per month, 30 GiB of EBS storage, 2 million I/Os, 1 GB of snapshots, and 100 GB of bandwidth to the internet.

Cancel

Launch instance

[Review commands](#)

## CREATE KEY PAIR

**▼ Key pair (login)** [Info](#)

You can use a key pair to securely connect to your instance. Ensure that you have access to the selected key pair before you launch the instance.

Key pair name - *required*

Select

▼

[Create new key pair](#)

For Windows instances, you use a key pair to decrypt the administrator password. You then use the decrypted password to connect to your instance.



Create key pair

Key pair name

Key pairs allow you to connect to your instance securely.

WindowsKey

The name can include up to 255 ASCII characters. It can't include leading or trailing spaces.

Key pair type

☒ RSA  
RSA encrypted private and public key pair

☐ ED25519  
ED25519 encrypted private and public key pair (Not supported for Windows instances)

Private key file format

☒ .pem  
For use with OpenSSH

☐ .ppk  
For use with PuTTY

⚠ When prompted, store the private key in a secure and accessible location on your computer. You will need it later to connect to your instance. [Learn more](#)

Cancel

Create key pair

Number of instances [Info](#)

1

#### Software Image (AMI)

Microsoft Windows Server 2022 ...[read more](#)  
ami-099a7b25010174ee4

#### Virtual server type (instance type)

t3.micro

#### Firewall (security group)

New security group

#### Storage (volumes)

1 volume(s) - 30 GiB

i

Free tier: In your first year includes 750 hours of t2.micro (or t3.micro in the Regions in which t2.micro is unavailable) instance usage on free tier AMIs per month, 30 GiB of EBS storage, 2 million IOs, 1 GB of snapshots, and 100 GB of bandwidth to the internet.

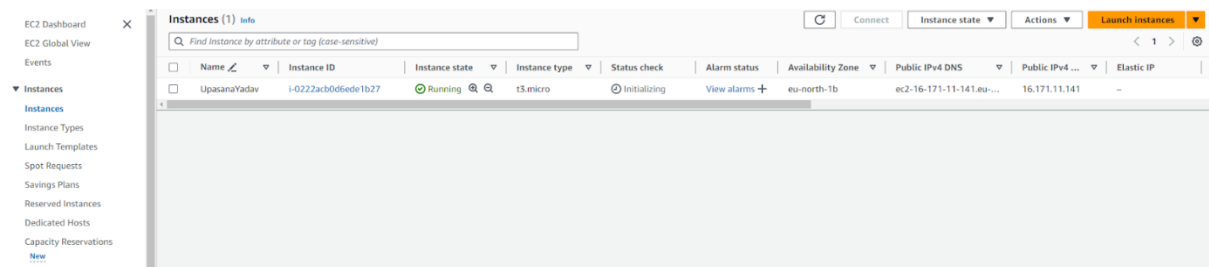
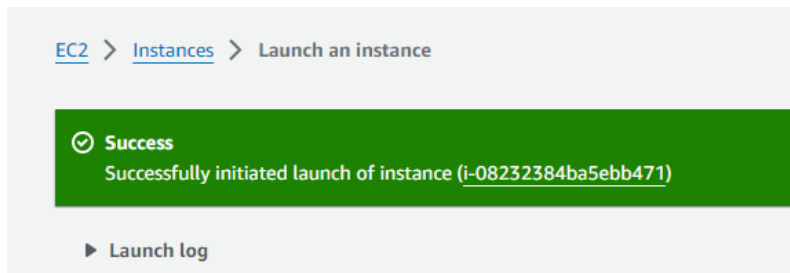
×

Cancel

Launch instance

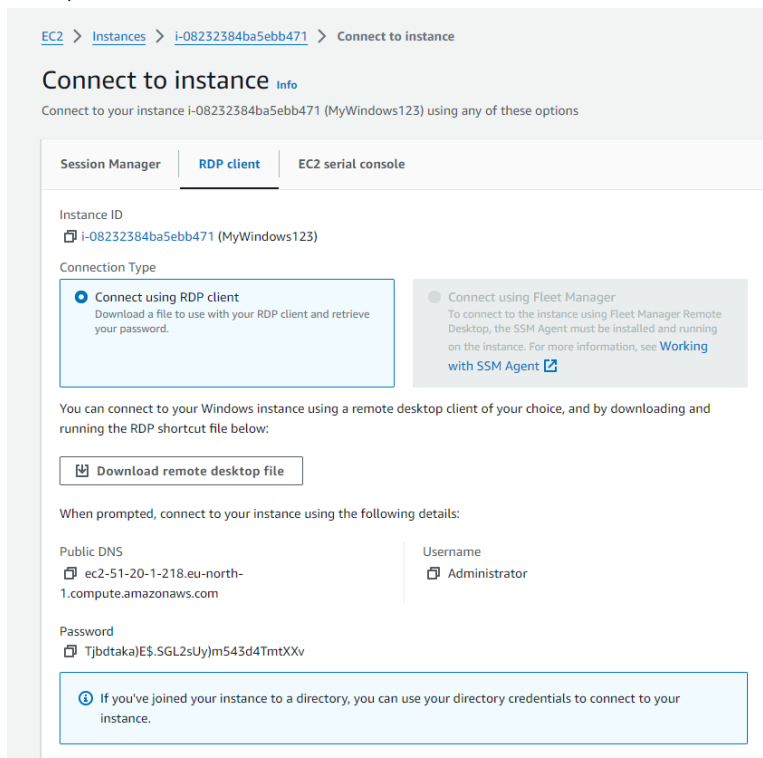
[Review commands](#)

After successfully launching instance, you shall see following message.



Connect :

Now, Click on RDP client and click on Get Password also download remote desktop file.



After clicking on get password, upload your private key file that you created and click on decrypt password.

EC2 > Instances > i-08232384ba5ebb471 > Get Windows password

## Get Windows password [Info](#)

Use your private key to retrieve and decrypt the initial Windows administrator password for this instance.

Instance ID  
i-08232384ba5ebb471 (MyWindows123)

Key pair associated with this instance  
WindowsKey

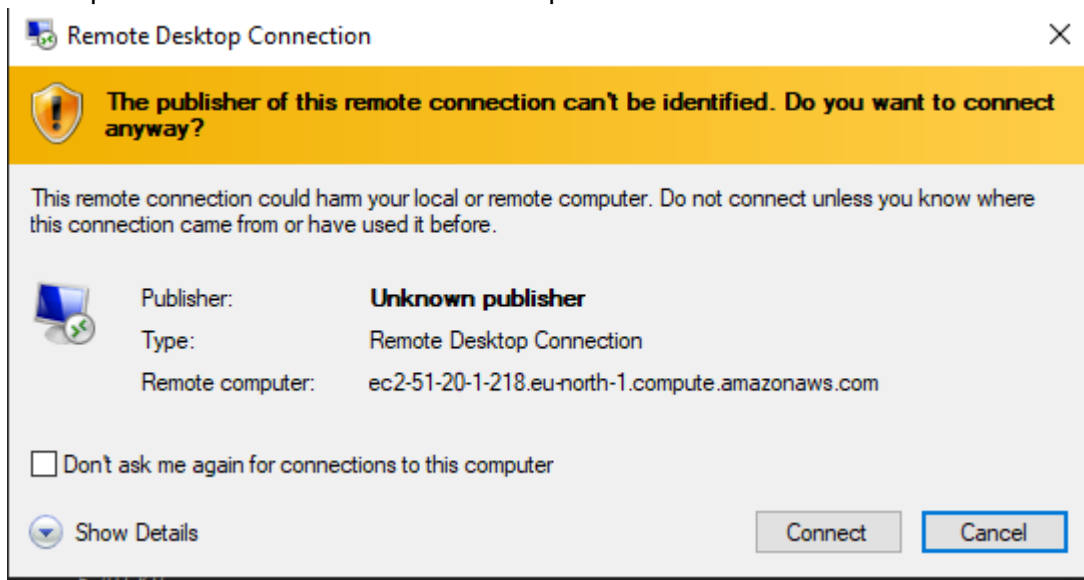
Private key  
Either upload your private key file or copy and paste its contents into the field below.

WindowsKey.pem  
1.678KB

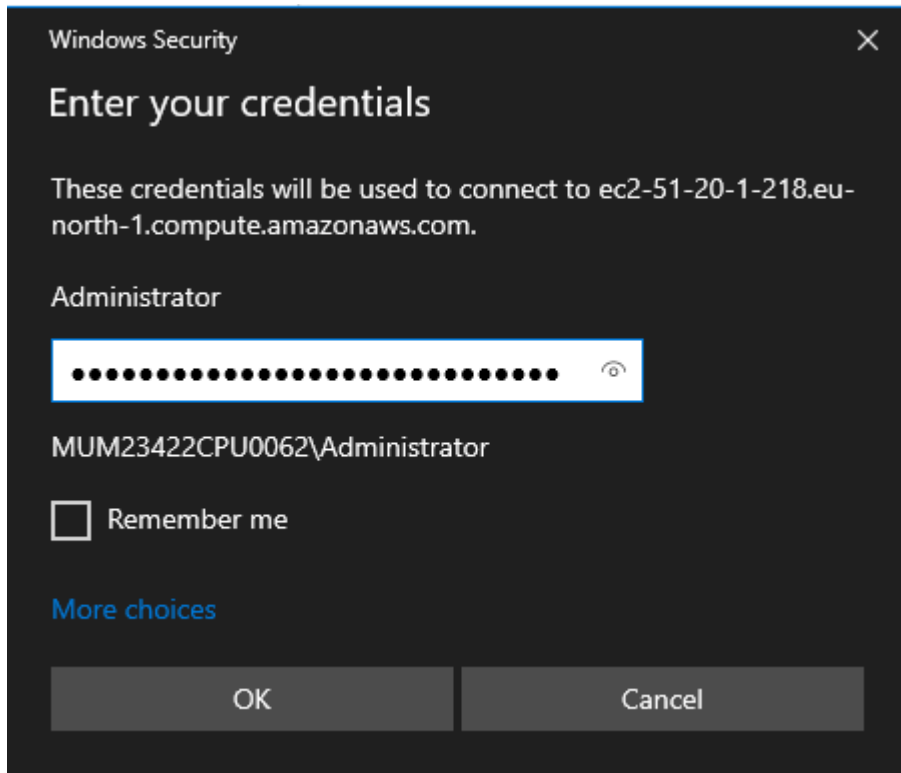
Private key contents - optional

```
-----BEGIN RSA PRIVATE KEY-----
MIIEpQIBAAKCAQEAsoDLP+ni1RtLsypkhab2Ciatr37JICRMdDmveJTSqRy2Dg
5N4Fr6/diBXmCjia3KoOlouQSh5Wo4dyatKYpnhErGudA0gI1rNVc87ovDLk3dCv
4KNLlibLMVFaoQwjeGLUPCiasdhu289aCzWX5nB67gsB7FFXAl0cxRY1RJ0wvhwB
dtlbjStUgrQ+IUkgGb49DYByTjJmBQSn5ndieSx8Xa/OzKqdVtwORYB5oU/VLqNM
KwJEmcdJ4ol0/rSFPYv7Rqd921fXnj6Lw7kSFzDhUUmJq1KLv/LCFXJV1js+Kon
rNggFFK6jHpyhA4PVLdYhNeAvgtln3aNIIMcZKQIDAQABAQCDXg8PPcWK1w/n
GpXXILftswLdM3gY5TBQFGAckZhBKINqCImaS8Y1kdRfQg8Q+cBFQ8K8X9B5n0uA
-----
```

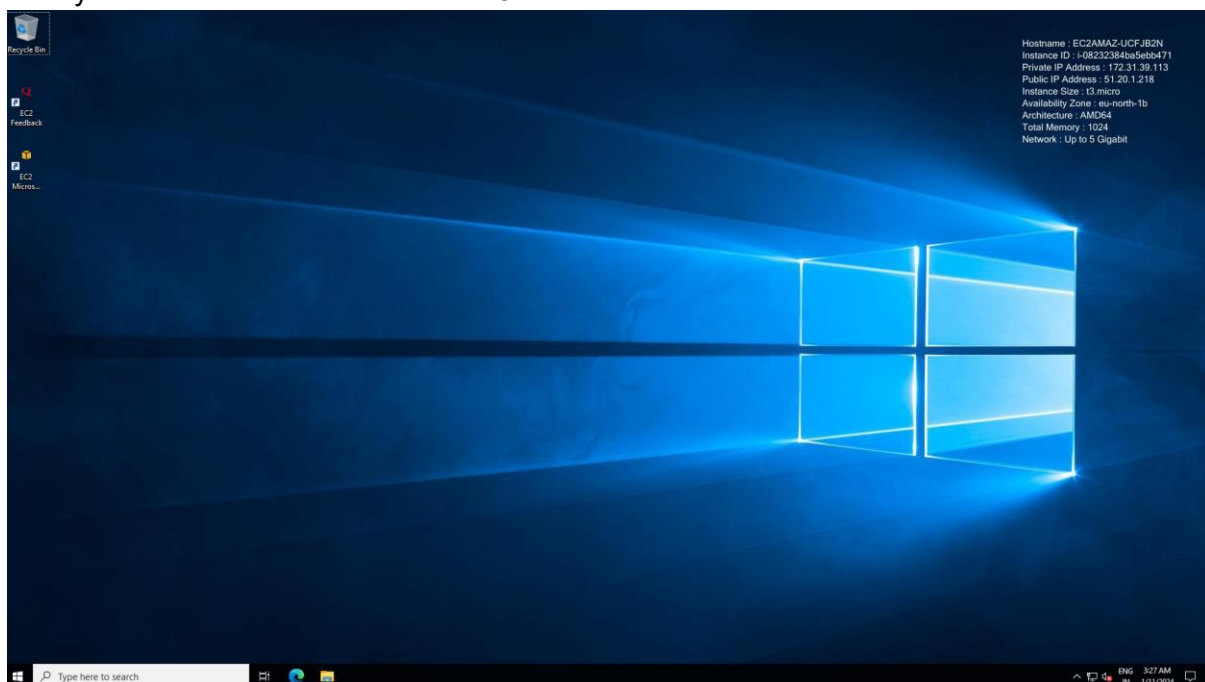
Now open the downloaded remote desktop file.



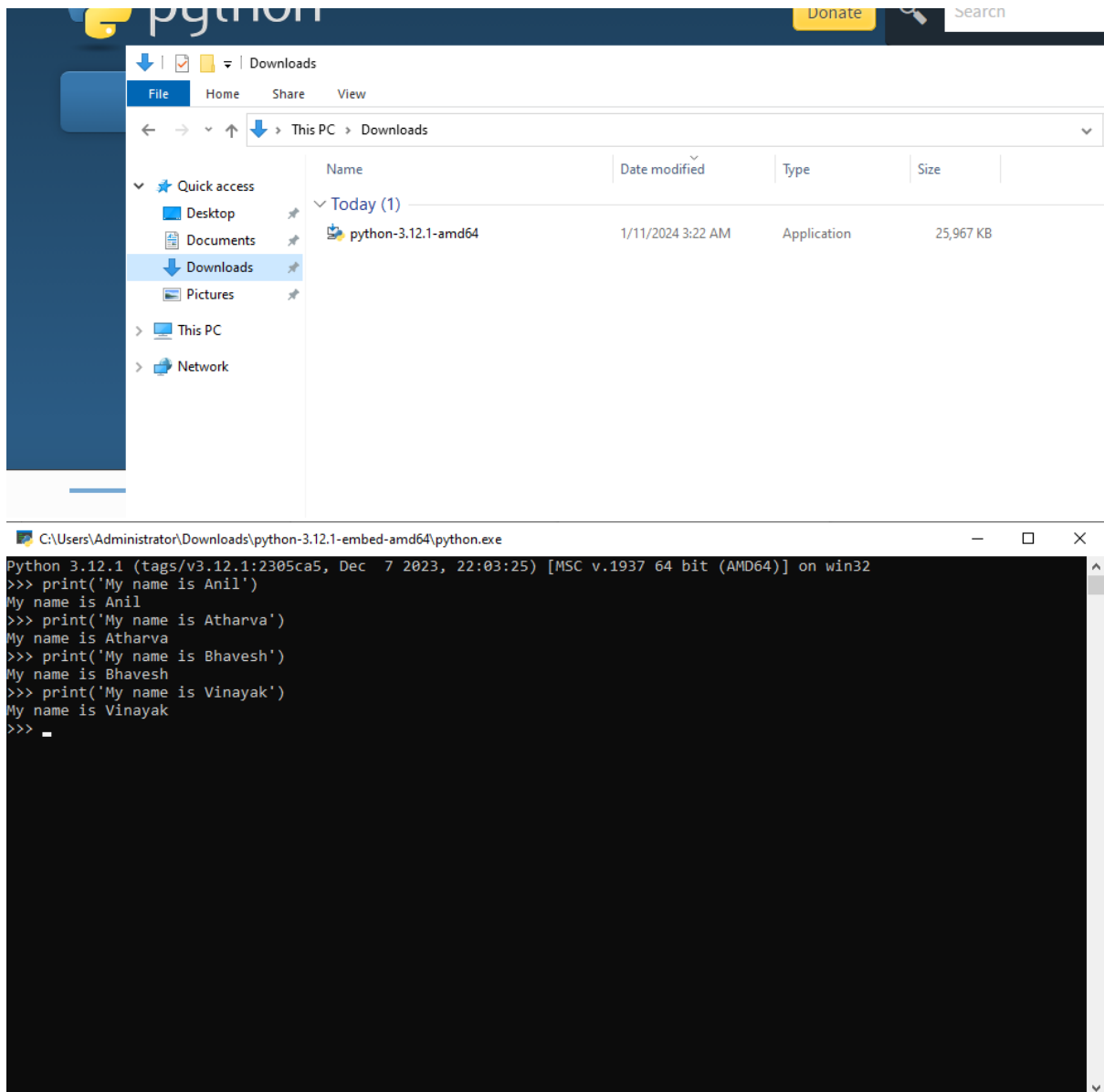
Enter the Password that you decrypted.



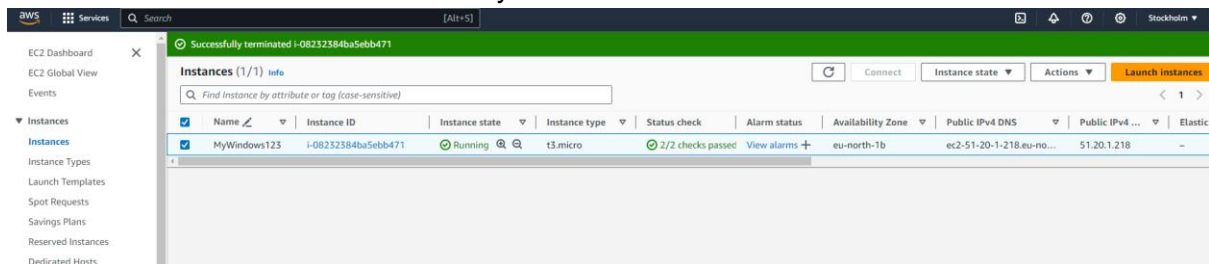
Now you'll be able to see a Windows OS as follows which is a Virtual Machine



Now install python in that machine and run it in the Command Prompt



After this Terminate the Instance that you created.



## 2)Implement the Ubuntu Machine using AWS EC2 and Execute the Linux Commands

- Disk Information in Human Readable form
- Create a file with your name
- Create a file with you CourseName and add a text file in it
- Display the created file
- Copy the contents of the created file in another file and print it.
- Install Chrome Browser/ Python3

**Step 1:** Create an EC2 instance

Launch an instance [Info](#)

Amazon EC2 allows you to create virtual machines, or instances, that run on the AWS Cloud. Quickly get started by following the simple steps below.

Name and tags [Info](#)


Name


MyUbuntuServer


[Add additional tags](#)▼ Application and OS Images (Amazon Machine Image) [Info](#)


An AMI is a template that contains the software configuration (operating system, application server, and applications) required to launch your instance. Search or Browse for AMIs if you don't see what you are looking for below


## Quick Start


 Amazon Linux


 macOS

 Ubuntu

 Windows

 Red Hat

 SUSE Linux

 Browse more AMIs  
Including AMIs from AWS, Marketplace and the Community

## ▼ Summary

Number of instances [Info](#)

1

Software Image (AMI)

Canonical, Ubuntu, 22.04 LTS, ...[read more](#)  
ami-0014ce3e52359afbd

Virtual server type (instance type)

t3.micro

Firewall (security group)

New security group

Storage (volumes)

1 volume(s) - 8 GiB

**Free tier:** In your first year includes 750 hours of t2.micro (or t3.micro in the Regions in which t2.micro is unavailable) instance usage on free tier AMIs per month, 30 GiB of EBS storage, 2 million I/Os, 1 GB of snapshots, and 100 GB of bandwidth to the internet.

Cancel

Launch instance

[Review commands](#)

## Key pair name

Key pairs allow you to connect to y

purva linux

The name can include up to 255 A

## Key pair type



RSA

RSA encrypted private and pair

## Private key file format



.pem

For use with OpenSSH



.ppk

For use with PuTTY



When prompted, store your computer. You wi

We'll create a new security group called 'launch



Allow SSH traffic from

Helps you connect to your instance

Anyw

0.0.0.0



Allow HTTPS traffic from the internet

To set up an endpoint, for example when creating ;



Allow HTTP traffic from the internet

To set up an endpoint, for example when creating ;



Rules with source of 0.0.0.0/0 allow all I

## Connect to instance [Info](#)

Connect to your instance i-010877623d8df0676 (Atharva3) using any of these options

**EC2 Instance Connect**

Session Manager

SSH client

EC2 serial console

Instance ID

i-010877623d8df0676 (Atharva3)

Connection Type

☒ Connect using EC2 Instance Connect

Connect using the EC2 Instance Connect browser-based client, with a public IPv4 address.

☐ Connect using EC2 Instance Connect Endpoint

Connect using the EC2 Instance Connect browser-based client, with a private IPv4 address and a VPC endpoint.

Public IP address

13.51.168.196

Username

Enter the username defined in the AMI used to launch the instance. If you didn't define a custom username, use the default username, ubuntu.

Q ubuntu

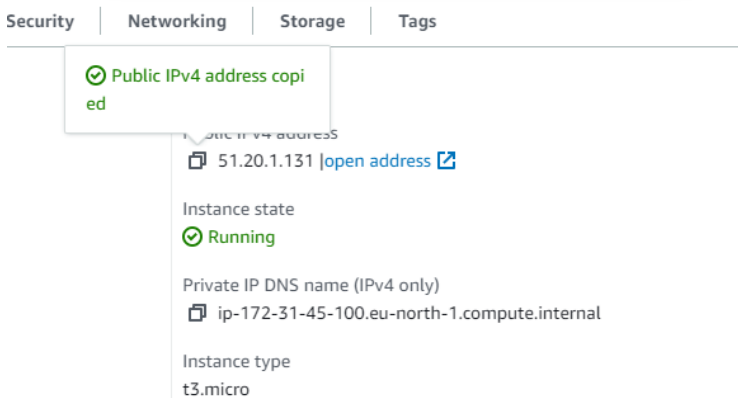
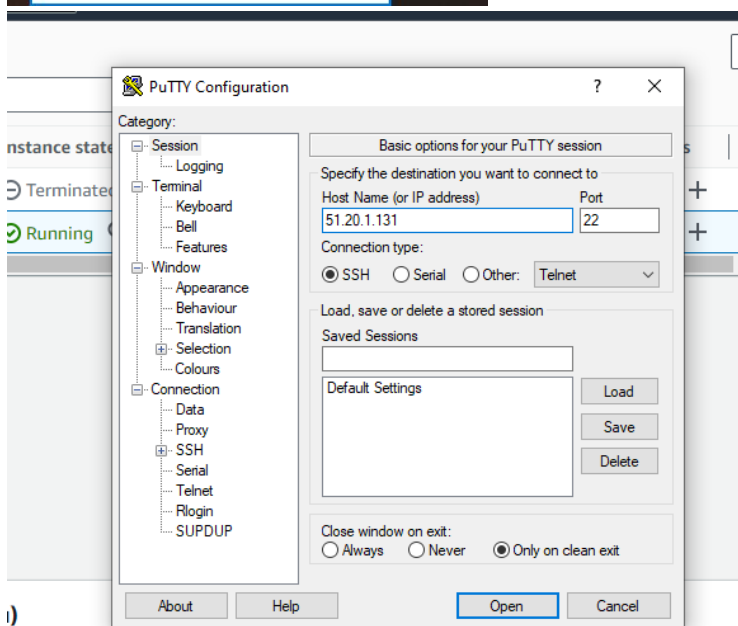
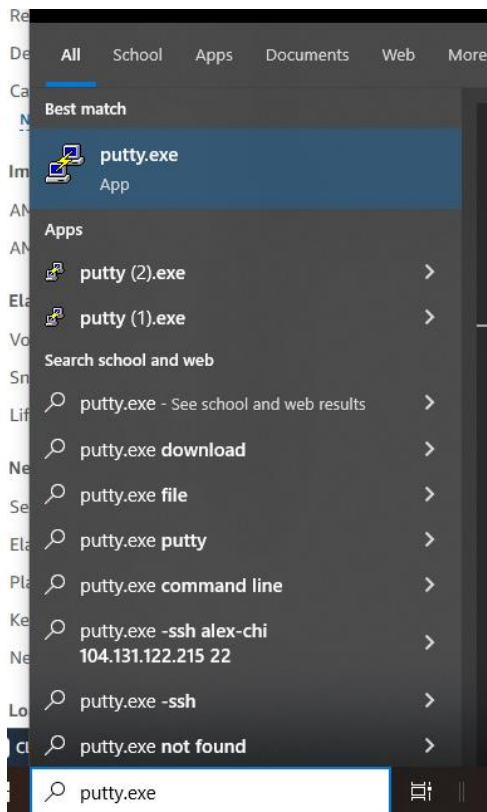
**Note:** In most cases, the default username, ubuntu, is correct. However, read your AMI usage instructions to check if the AMI owner has changed the default AMI username.

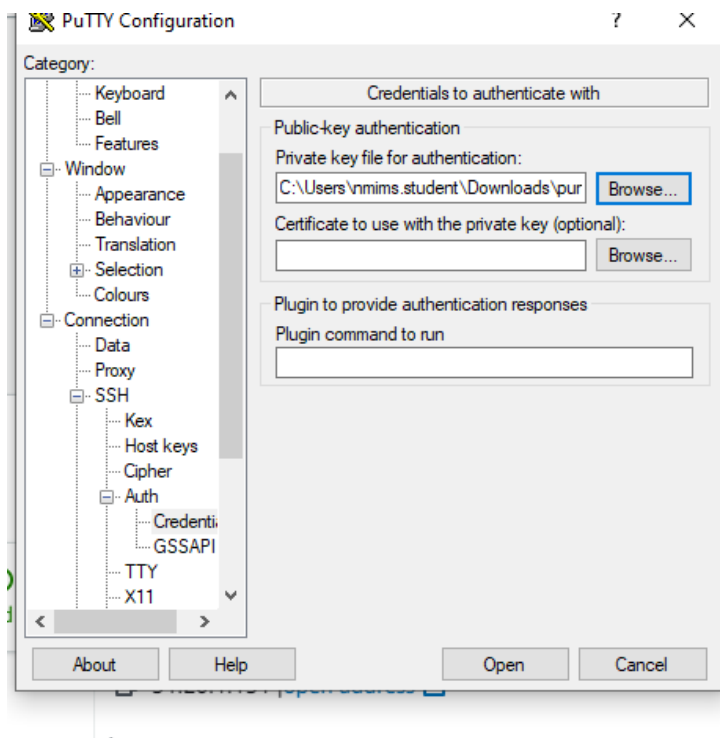
Cancel

Connect

**Step 2:** After the instance is created open putty.exe file that we have downloaded







**Step 4:** Now a command prompt will be opened type your user name here

```
ubuntu@ip-172-31-41-166: ~  
login as: ubuntu  
Authenticating with public key "Atharva28"  
Welcome to Ubuntu 22.04.3 LTS (GNU/Linux 6.2.0-1017-aws x86_64)  
  
* Documentation:  https://help.ubuntu.com  
* Management:    https://landscape.canonical.com  
* Support:        https://ubuntu.com/advantage  
  
System information as of Sat Jan 20 02:45:28 UTC 2024  
  
System load:  0.0               Processes:            99  
Usage of /:   20.6% of 7.57GB   Users logged in:     0  
Memory usage: 21%              IPv4 address for ens5: 172.31.41.166  
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 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.
```

**Step 5:** Now enter the commands here

```
ubuntu@ip-172-31-41-166:~$ ls
ubuntu@ip-172-31-41-166:~$ mkdir msc
ubuntu@ip-172-31-41-166:~$ ls
msc
ubuntu@ip-172-31-41-166:~$ cd msc
ubuntu@ip-172-31-41-166:~/msc$ touch cloud.txt
ubuntu@ip-172-31-41-166:~/msc$ ls
cloud.txt
ubuntu@ip-172-31-41-166:~/msc$
```

```
ubuntu@ip-172-31-41-166:~/msc$ cat>cloud.txt
Bhaveh pashte only sonapapdi loverubuntu@ip-172-31-41-166:~/msc$
ubuntu@ip-172-31-41-166:~/msc$ cat cloud.txt
Bhaveh pashte only sonapapdi loverubuntu@ip-172-31-41-166:~/msc$
ubuntu@ip-172-31-41-166:~/msc$
```

**Step 6:** Now install python in cmd

```
ubuntu@ip-172-31-41-166:~/msc$ sudo apt install python3
ubuntu@ip-172-31-41-166:~/msc$ sudo apt install python3
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
python3 is already the newest version (3.10.6-1~22.04).
python3 set to manually installed.
0 upgraded, 0 newly installed, 0 to remove and 0 not upgraded.
ubuntu@ip-172-31-41-166:~/msc$
```

Because the python was already installed

**Step 7:** Now type python3 and then you can run python code on it

```
ubuntu@ip-172-31-41-166:~$ python3
Python 3.10.12 (main, Nov 20 2023, 15:14:05) [GCC 11.4.0] on linux
Type "help", "copyright", "credits" or "license" for more information.
>>> print('Hello World')
Hello World
>>>
```

**Step 8** Create a file write your address and read the file

```
ubuntu@ip-172-31-41-166:~$ nano city
ubuntu@ip-172-31-41-166:~$ cat city
Atharva kulkarni kattar puneekar 411033
ubuntu@ip-172-31-41-166:~$
```

### Step 9: Copy content from 1 file to another

```
ubuntu@ip-172-31-41-166:~$ cp city city1
ubuntu@ip-172-31-41-166:~$ cat city1
Atharva kulkarni kattar punekar 411033
```

### Step 10: install chrome browser in ubuntu

```
ubuntu@ip-172-31-41-166:~$ wget https://dl.google.com/linux/direct/google-chrome-stable_current_amd64.deb
--2024-01-20 03:06:02-- https://dl.google.com/linux/direct/google-chrome-stable_current_amd64.deb
Resolving dl.google.com (dl.google.com)... 142.250.74.142, 2a00:1450:400f:802::200e
Connecting to dl.google.com (dl.google.com)|142.250.74.142|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 105000828 (100M) [application/x-debian-package]
Saving to: 'google-chrome-stable_current_amd64.deb'

google-chrome-stable_current_amd 100%[=====>] 100.14M  182MB/s  in 0.6s

2024-01-20 03:06:02 (182 MB/s) - 'google-chrome-stable_current_amd64.deb' saved [105000828/105000828]

ubuntu@ip-172-31-41-166:~$
```

### Step 11: install Firefox in ubuntu

Using code: `sudo apt install firefox`.