

**SVKM'S NMIMS Nilkamal School of Mathematics, Applied Statistics & Analytics**  
**Master of Science (Statistics & Data Science)**  
**Practical-1: Infrastructure as a service using AWS**

Name: Chinmay Shinde Roll No.: A066 SAP ID: 86062300031

- Cloud Computing Architecture
- IAAS
- AWS
- EC2

CHINMAY SHINDE  
MSc. SDS  
86062300031

①

Practical 1: Infrastructure as a service using AWS

1) Cloud Computing Architecture

→ Cloud computing architecture is the blueprint for how cloud services are structured and delivered. It is a multi-layered framework that involves the following:

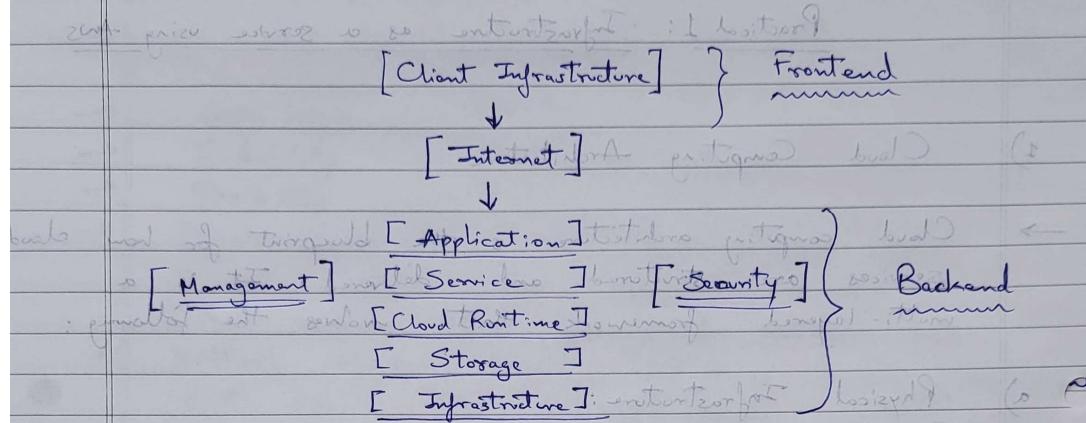
a) Physical Infrastructure: The foundation of the cloud, encompassing physical hardware like servers, storage devices, and networking equipment located in the data centers.

b) Infrastructure as a Service (IaaS): Virtualizes physical resources into on-demand services like virtual machines (VM), storage and networking. The users manage operating systems and the applications.

c) Platform as a Service (PaaS): Provides a cloud-based environment for developing, testing and deploying applications without managing infrastructure. Developers focus on application logic.

d) Software as a Service (SaaS): Delivers fully developed applications over the internet. Users access these applications through a web browser or mobile device without installation.

30 min 2 pages  
2023-24  
12000 85 2023



Cloud Computing Architecture

The key components of this architecture include the following:

- Clients: End users accessing cloud services through devices like computers, smartphones or tablets.
- Applications: Software programs running on the cloud, which is accessible to the clients.
- Cloud Based Delivery: The cloud providers infrastructure, responsible for delivering applications and services.
- Network: The communication bus channel connecting clients to the cloud services.

## 2) IAAS ( Infrastructure as a Service )

→ IAAS provides on demand access to fundamental computing resources, including servers, storage and networking. It's renting to IT infrastructure rather than owning and maintaining it.

(1) The key benefits of IAAS are as follows: (i)

- Cost efficiency
- Scalability
- Flexibility
- Rapid Deployment

The common IAAS offerings are mentioned below.

- Virtual Machines (VM): Virtualized computing environments that behave like physical servers.
- Storage: Various storage options, including block, objects and file storage.
- Networking: Virtual networking, load balancers, firewalls, and other networking components are available at the network (cloud) level.

### (2) AWS (Amazon Web Services) (ii)

→ AWS is a leading cloud platform offering a comprehensive suite of IAAS, PAAS and SAAS services. It provides a scalable and reliable infrastructure for businesses of all sizes.

The core AWS services are as follows:

#### (i) EC2 (Elastic Compute Cloud):

It provides scalable computing capacity in the cloud, allowing users to launch and manage virtual machines.

#### (ii) S3 (Simple Storage Service):

It offers object storage for storing and retrieving any amount of data, with high durability and availability.

(iii) RDS (Relational Database Service):

It manages relational databases, simplifying database administration and scaling.

(iv) Lambda:

This enables serverless computing, allowing you to run code without provisioning or managing the servers.

↑ EC2 (Elastic Compute Cloud):

→ EC2 is a fundamental IaaS service on AWS that provides scalable computing resources. It allows users to launch instances (virtual machines) with the various configurations to meet different workload requirements.

The key EC2 features are mentioned as follows:

- On-Demand Instances: Pay-as-you-go pricing for flexible computing resources.
- Reserved Instances: Discounted pricing for the long-term instance commitments.
- Spot Instances: Bid for idle compute capacity at the potentially lower prices.
- Auto Scaling: Automatically adjusts the number of the instances based on demand.
- Elastic Load Balancing: Distributes incoming traffic across multiple instances for high availability and performance.

It supports wide range of operating systems, frameworks and programming languages, making it suitable for various applications and workloads.

## 1. Implement the windows machine using AWS EC2

The screenshot shows the AWS EC2 dashboard for the Asia Pacific (Mumbai) Region. It displays various EC2 resources: Instances (running) 0, Auto Scaling Groups 0, Dedicated Hosts 0, Elastic IPs 0, Instances 2, Key pairs 2, Load balancers 0, Placement groups 0, Security groups 3, Snapshots 0, and Volumes 0. On the right, it shows the EC2 Free Tier Info, which offers free usage across all AWS Regions. It highlights 1 EC2 free tier offers in use, noting that 0 offers are forecasted to exceed the free tier limit. It also mentions that storage space on EBS is at 1% usage (29.85 GB remaining). A link to View Global EC2 resources is provided.

This screenshot shows the first step of the 'Launch instance' wizard, titled 'Launch an instance'. It asks for the number of instances, which is set to 1. Below this, there's a section for 'Name and tags' where the name 'CS.Prac.1.2' is entered. The next section, 'Application and OS Images (Amazon Machine Image)', contains a search bar and a list of available AMIs. On the right side, the 'Summary' panel shows the configuration: 1 instance, Amazon Linux 2023 AMI 2023.5.2, t2.micro instance type, and a new security group. It also lists storage (1 volume - 8 GiB). At the bottom are 'Cancel', 'Launch instance', and 'Review commands' buttons.

This screenshot shows the second step of the 'Launch instance' wizard, titled 'Quick Start'. It features a grid of quick-launch icons for various operating systems: Amazon Linux, macOS, Ubuntu, Windows, Red Hat, and SUSE Linux. Below this, a specific AMI is selected: 'Microsoft Windows Server 2022 Core Base' (ami-0b58ab51be399f13c). This AMI is marked as 'Free tier eligible'. The 'Summary' panel on the right remains the same as the previous step, showing 1 instance, t2.micro instance type, and a new security group. The 'Launch instance' button is prominently displayed.

**Create key pair**

**Key pair name**  
Key pairs allow you to connect to your instance securely.  
**prac1.4**

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.**

**Create key pair**

**Network settings**

**Summary**

**Number of instances** 1

**Software Image (AMI)**  
Microsoft Windows Server 2022 ...read more  
ami-049f0f6f51145ff40

**Virtual server type (instance type)** t2.micro

**Firewall (security group)**  
New security group

**Storage (volumes)**  
1 volume(s) - 30 GiB

**Launch instance**

**Configure storage**

**Summary**

**Number of instances** 1

**Software Image (AMI)**  
Microsoft Windows Server 2022 ...read more  
ami-0b58ab51be399f13c

**Virtual server type (instance type)** t2.micro

**Firewall (security group)**  
New security group

**Storage (volumes)**  
1 volume(s) - 30 GiB

**Launch instance**

Screenshot of the AWS EC2 Instances launch log showing a successful instance launch:

**Success**  
Successfully initiated launch of instance (i-0d4d109eb82aea787)

**Launch log**

Step	Status
Initializing requests	Succeeded
Creating security groups	Succeeded
Creating security group rules	Succeeded
Launch initiation	Succeeded

Screenshot of the AWS EC2 Instances list page showing two instances:

Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public IPv4 DNS
CS.Prac.1.2	i-00873170c3cb25aa7	Running	t2.micro	Initializing	View alarms +	ap-south-1b	ec2-13-201-166-45.
CS.Prac.1.1	i-0fe0320fde1eea481	Stopped	t2.micro	-	View alarms +	ap-south-1a	-

Screenshot of the AWS EC2 Connect to instance page for instance i-0d4d109eb82aea787 (CS.Prac1.2):

Connect to instance [Info](#)

Connect to your instance i-0d4d109eb82aea787 (CS.Prac1.2) using any of these options

Session Manager | RDP client | EC2 serial console

Instance ID: i-0d4d109eb82aea787 (CS.Prac1.2)

Connection Type:

- Connect using RDP client  
Download a file to use with your RDP client and retrieve your password.
- Connect using Fleet Manager  
To connect to the instance using Fleet Manager Remote Desktop, the SSM Agent must be installed and running on the instance. For more information, see [Working with SSM Agent](#).

You can connect to your Windows instance using a remote desktop client of your choice, and by downloading and running the RDP shortcut file below:

[Download remote desktop file](#)

When prompted, connect to your instance using the following username and password:

CloudShell | Feedback | © 2024, Amazon Web Services, Inc. or its affiliates. | Privacy | Terms | Cookie preferences

Screenshot of the AWS EC2 Connect to instance page for instance i-0d4d109eb82aea787 (CS.Prac1.2), showing a modal for entering credentials:

Windows Security

Enter your credentials

These credentials will be used to connect to ec2-13-201-166-45.ap-south-1.compute.amazonaws.com.

Administrator

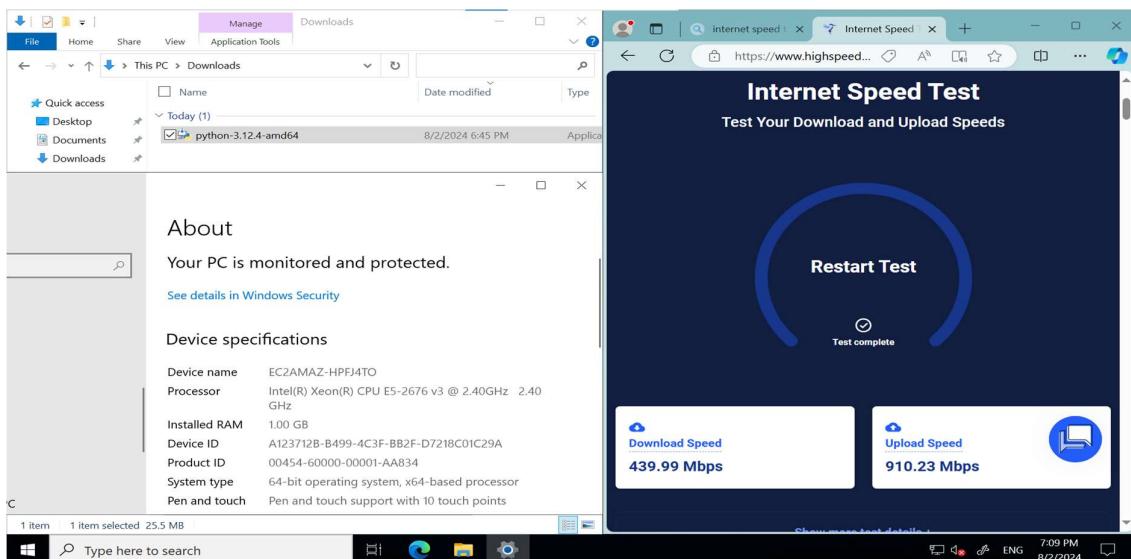
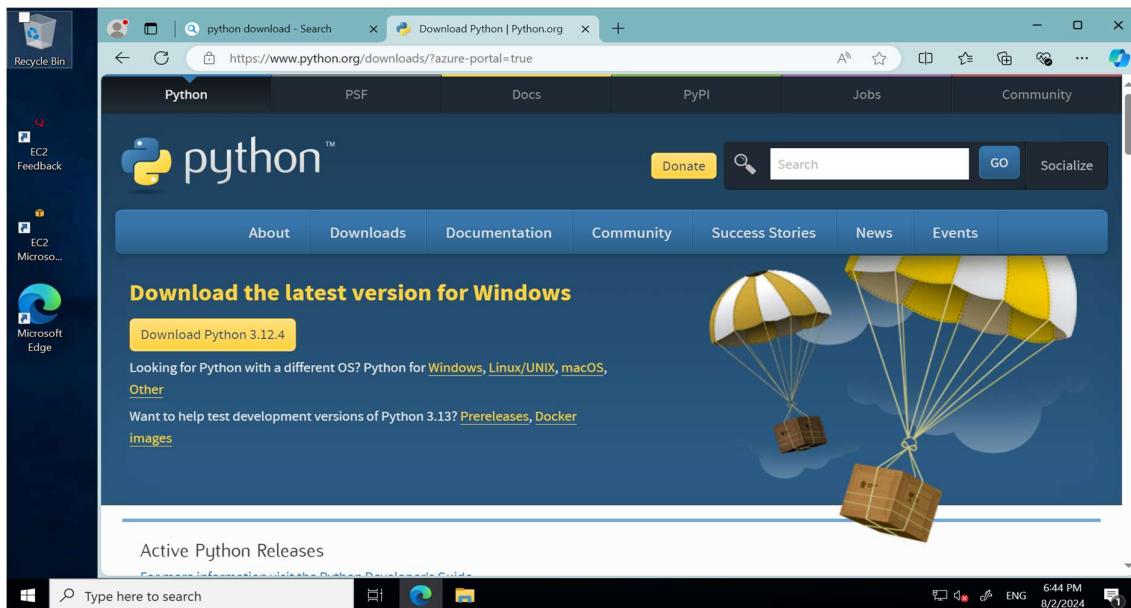
Password: CHINMAY-GALAXYB\Administrator

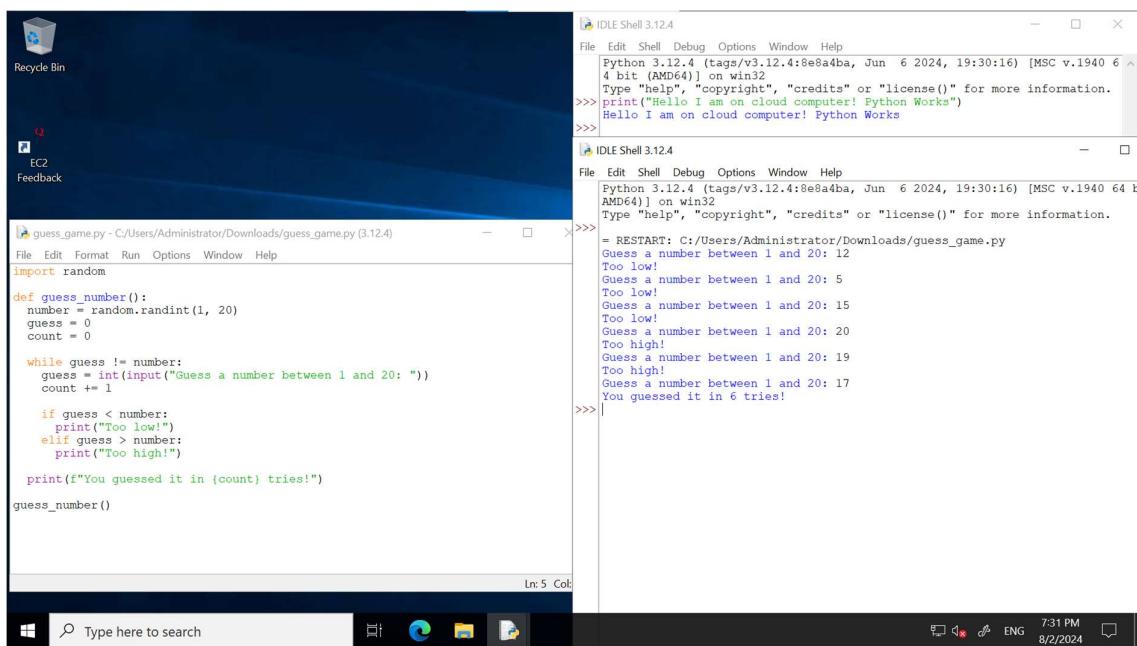
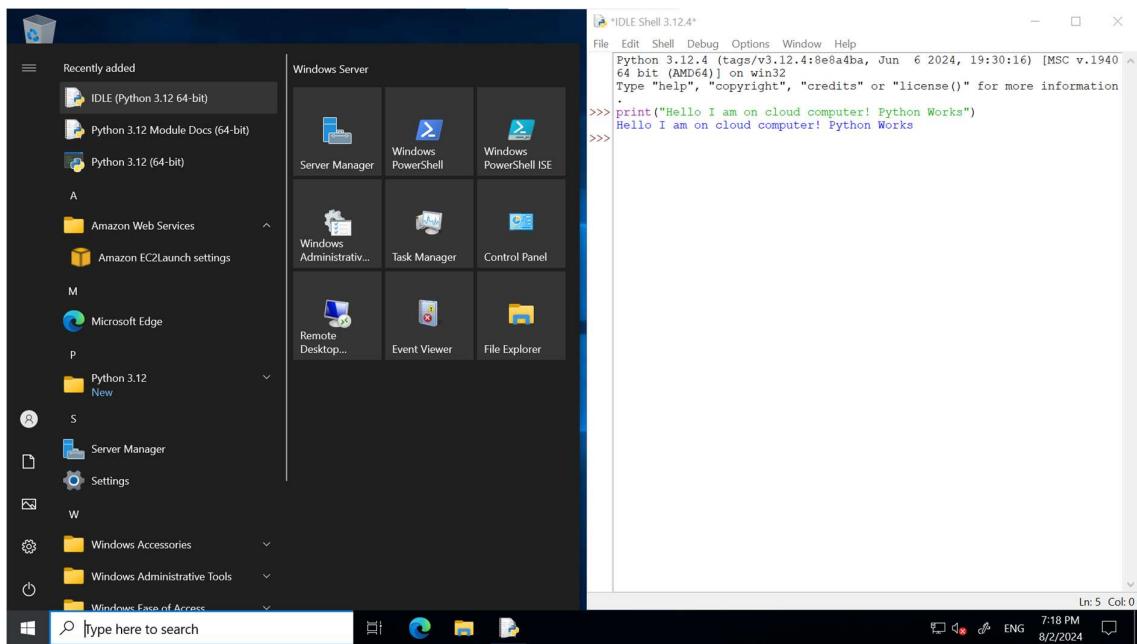
Remember me

More choices

OK | Cancel

CloudShell | Feedback | © 2024, Amazon Web Services, Inc. or its affiliates. | Privacy | Terms | Cookie preferences





## 2. Implement Ubuntu machine using AWS EC2 and execute Linux commands.

The screenshot shows the AWS EC2 Dashboard. On the left sidebar, under the 'Instances' section, there is a 'Launch instance' button. This button is highlighted with a yellow box, indicating it is the next step in the process. Below this button, there is a 'Migrate a server' button. A note below the 'Launch instance' button states: 'Note: Your instances will launch in the Asia Pacific (Mumbai) Region'. To the right of the main content area, there is a 'Service health' section showing 'AWS Health Dashboard' and a status message: 'This service is operating normally.' There is also a 'Storage space on EBS' section showing 1% usage with 29.85 GB remaining. The bottom of the dashboard includes links for 'Explore AWS', 'Privacy', 'Terms', and 'Cookie preferences'.

The screenshot shows the 'Launch an instance' wizard. The first step is 'Name and tags'. It has a 'Name' field containing 'CS.Prac1.1' and a 'Add additional tags' link. Below this, there is a section titled 'Application and OS Images (Amazon Machine Image)' with a search bar. The search bar placeholder is 'Search our full catalog including 1000s of application and OS images'. The bottom of the screen includes 'CloudShell', 'Feedback', 'Privacy', 'Terms', and 'Cookie preferences' links.

The screenshot shows the 'Launch an instance' wizard. The second step is 'Application and OS Images (Amazon Machine Image)'. It displays a grid of AMI icons: Amazon Linux, macOS, Ubuntu, Windows, Red Hat, and SUSE Linux. An 'ubuntu' icon is highlighted with a yellow box. To the right of the grid, there is a search icon and a link to 'Browse more AMIs'. Below the grid, there is a section for 'Ubuntu Server 24.04 LTS (HVM), SSD Volume Type' with details: 'Free tier eligible', 'ami-0ad21ae1d0696ad58 (64-bit (x86)) / ami-01f6c796d6dbc1e36 (64-bit (Arm))', 'Virtualization: hvm', 'ENA enabled: true', and 'Root device type: ebs'. At the bottom, there are sections for 'Architecture' (64-bit (x86)), 'AMI ID' (ami-0ad21ae1d0696ad58), and 'Verified provider'. The bottom of the screen includes 'CloudShell', 'Feedback', 'Privacy', 'Terms', and 'Cookie preferences' links.

**Create key pair**

**Key pair name**  
Key pairs allow you to connect to your instance securely.

**Key pair type**  
 RSA RSA encrypted private and public key pair  
 ED25519 ED25519 encrypted private and public key pair

**Private key file format**  
 .pem For use with OpenSSH  
 .ppk For use with PuTTY

**Create key pair**

**Network**

vpc-02215a291e8b77ba7

**Subnet**  
No preference (Default subnet in any availability zone)

Auto-assign public IP  Enable

Additional charges apply when outside of free tier allowance

**Firewall (security groups)**  Create security group  Select existing security group

We'll create a new security group called 'launch-wizard-3' with the following rules:

Allow SSH traffic from Anywhere 0.0.0.0/0  
 Allow HTTPS traffic from the internet To set up an endpoint, for example when creating a web server  
 Allow HTTP traffic from the internet To set up an endpoint, for example when creating a web server

**Summary**

Number of instances

Software Image (AMI)  
Canonical, Ubuntu, 24.04 LTS, ...[read more](#)  
ami-0ad21ae1d0696ad58

Virtual server type (instance type)  
t2.micro

Firewall (security group)  
New security group

Storage (volumes)  
1 volume(s) - 8 GiB

**Launch instance**

Screenshot of the AWS EC2 Instances launch confirmation page. It shows a green success message: "Successfully initiated launch of instance (i-0fe0320fde1eea481)".

Screenshot of the AWS EC2 Instances list page. It shows one instance named "CS.Prac1.1" with the ID "i-0fe0320fde1eea481". The instance is running and assigned to the t2.micro type.

Screenshot of the AWS EC2 Instance summary page for instance "i-0fe0320fde1eea481". It provides detailed information such as Public IPv4 address (13.201.124.116), Private IPv4 address (172.31.45.149), Public IPv4 DNS (ec2-13-201-124-116.ap-south-1.compute.amazonaws.com), and VPC ID (vpc-02215a291e8b77ba7).

Screenshot of the AWS EC2 Connect to instance page. It shows the instance ID "i-0fe0320fde1eea481" and two connection options: "Connect using EC2 Instance Connect" (selected) and "Connect using EC2 Instance Connect Endpoint". A warning message states: "Port 22 (SSH) is open to all IPv4 addresses. Port 22 (SSH) is currently open to all IPv4 addresses, indicated by 0.0.0.0/0 in the inbound rule in your security group. For increased security, consider restricting access to only the EC2 Instance Connect service. IP addresses for your Region: 13.233.177.0/29.".

```
Welcome to Ubuntu 24.04 LTS (GNU/Linux 6.8.0-1009-aws x86_64)

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

System information as of Fri Aug 2 11:38:31 UTC 2024

System load: 0.03 Processes: 107
Usage of /: 22.7% of 6.71GB Users logged in: 0
Memory usage: 20% IPv4 address for enx0: 172.31.45.149
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
```

i-0fe0320fde1eea481 (CS.Prac1.1)  
Public IPs: 13.201.124.116 Private IPs: 172.31.45.149

a) List directory contents and disk usage

```
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.

ubuntu@ip-172-31-45-149:~$ ls -la
total 28
drwxr-x--- 4 ubuntu ubuntu 4096 Aug 2 11:38 .
drwxr-xr-x 3 root root 4096 Aug 2 11:34 ..
-rw-r--r-- 1 ubuntu ubuntu 220 Mar 31 08:41 .bash_logout
-rw-r--r-- 1 ubuntu ubuntu 3771 Mar 31 08:41 .bashrc
drwx----- 2 ubuntu ubuntu 4096 Aug 2 11:38 .cache
-rw-r--r-- 1 ubuntu ubuntu 807 Mar 31 08:41 .profile
drwx----- 2 ubuntu ubuntu 4096 Aug 2 11:34 .ssh
ubuntu@ip-172-31-45-149:~$ df -h
Filesystem      Size  Used Avail Use% Mounted on
/dev/root       6.8G  5.2G  23% /
tmpfs          479M    0  479M  0% /dev/shm
tmpfs          192M  868K  191M  1% /run
tmpfs          5.0M    0  5.0M  0% /run/lock
/dev/xvda16     881M   76M  744M  10% /boot
/dev/xvda15    105M   6.1M  99M   6% /boot/efi
tmpfs          96M   12K   96M   1% /run/user/1000
ubuntu@ip-172-31-45-149:~$
```

b) Create a Python file using a text editor (nano factorial\_cs.py )

```
GNU nano 7.2          factorial_cs.py
```

[ New File ]

^G Help ^C Write Out ^W Where Is ^K Cut ^E Execute ^C Location M-U Undo M-A Set Mark
^X Exit ^R Read File ^A Replace ^P Paste ^J Justify ^Y Go To Line M-E Redo M-C Copy

c) Write the Python code for factorial of a given number and save the file

```
GNU nano 7.2                                     factorial.cs.py *
def factorial(n):
    if n == 0:
        return 1
    else:
        return n * factorial(n - 1)

number = int(input("Enter a number: "))
print(f"The factorial of {number} is {factorial(number)}")
```

The screenshot shows a terminal window titled "factorial.cs.py \*". It contains a Python script that defines a factorial function using recursion. It then prompts the user for a number, calculates its factorial, and prints the result. The terminal interface includes standard keyboard shortcuts like Help, Exit, Write Out, Read File, Cut, Paste, Execute, Justify, Location, Go To Line, Undo, Redo, Set Mark, and Copy.

d) Run the Python code

```
drwxr-xr-x 3 root root 4096 Aug 2 11:34 ..
-rw-r--r-- 1 ubuntu ubuntu 220 Mar 31 08:41 .bash_logout
-rw-r--r-- 1 ubuntu ubuntu 3771 Mar 31 08:41 .bashrc
drwx----- 2 ubuntu ubuntu 4096 Aug 2 11:38 .cache
-rw-r--r-- 1 ubuntu ubuntu 807 Mar 31 08:41 .profile
drwx----- 2 ubuntu ubuntu 4096 Aug 2 11:34 .ssh
ubuntu@ip-172-31-45-149:~$ df -h
Filesystem      Size   Used Avail Use% Mounted on
/dev/root       6.8G  1.6G  5.2G  23% /
tmpfs          479M     0  479M  0% /dev/shm
tmpfs          192M  868K  191M  1% /run
tmpfs          5.0M     0  5.0M  0% /run/lock
/dev/xvda16     881M  76M  744M  10% /boot
/dev/xvda15     105M  6.1M  99M  6% /boot/efi
tmpfs          96M   12K  96M  1% /run/user/1000
ubuntu@ip-172-31-45-149:~$ nano --version
GNU nano, version 7.2
(C) 2023 the Free Software Foundation and various contributors
Compiled options: --disable-libmagic --enable-utf8
ubuntu@ip-172-31-45-149:~$ nano factorial.cs.py
ubuntu@ip-172-31-45-149:~$ python factorial.cs.py
Command 'python' not found, did you mean:
  command 'python3' from deb python3
  command 'python' from deb python-is-python3
ubuntu@ip-172-31-45-149:~$ python3 factorial.cs.py
Enter a number: 10
The factorial of 10 is 3628800
ubuntu@ip-172-31-45-149:~$
```

The screenshot shows the terminal output of running the Python script. It first lists system files and memory usage. Then it shows the nano editor being used to edit the script. Finally, it runs the script and prints the factorial of 10, which is 3628800.

e) Check system process and resource usage then power off

```
Tasks: 104 total, 1 running, 103 sleeping, 0 stopped, 0 zombie
%Cpu(s): 0.0 us, 0.0 sy, 0.0 ni,100.0 id, 0.0 wa, 0.0 hi, 0.0 si, 0.0 st
MiB Mem : 957.4 total, 573.7 free, 295.2 used, 239.0 buff/cache
MiB Swap: 0.0 total, 0.0 free, 0.0 used. 662.2 avail Mem

 PID USER      PR  NI    VIRT    RES    SHR S %CPU %MEM TIME+ COMMAND
 1 root      20  0 22052 13240 9528 S  0.0  1.4 0:03.93 systemd
 2 root      20  0     0     0     0 S  0.0  0.0 0:00.00 kthreadd
 3 root      20  0     0     0     0 S  0.0  0.0 0:00.00 pool_workqueue_release
 4 root     -20  0     0     0     0 I  0.0  0.0 0:00.00 kworker/R-rcu_g
 5 root     -20  0     0     0     0 I  0.0  0.0 0:00.00 kworker/R-rcu_p
 6 root     -20  0     0     0     0 I  0.0  0.0 0:00.00 kworker/R-slub
 7 root     -20  0     0     0     0 I  0.0  0.0 0:00.00 kworker/ksoftirqd
 8 root     -20  0     0     0     0 I  0.0  0.0 0:00.00 kworker/0:0-events
 9 root     -20  0     0     0     0 I  0.0  0.0 0:00.04 kworker/0:1-events
10 root     -20  0     0     0     0 I  0.0  0.0 0:00.00 kworker/0:0h-events_highpri
11 root     -20  0     0     0     0 I  0.0  0.0 0:00.00 kworker/u30:0-flush-202:0
12 root     -20  0     0     0     0 I  0.0  0.0 0:00.00 kworker/R-mm_p
13 root     -20  0     0     0     0 I  0.0  0.0 0:00.00 rctasks rude_kthread
14 root     -20  0     0     0     0 I  0.0  0.0 0:00.00 rctasks_trace_kthread
15 root     -20  0     0     0     0 S  0.0  0.0 0:00.00 ksoftirqd/0
16 root     -20  0     0     0     0 I  0.0  0.0 0:00.00 rctasks_sched
17 root     -51  0     0     0     0 S  0.0  0.0 0:00.00 migration/0
18 root     -51  0     0     0     0 S  0.0  0.0 0:00.00 idle Select/0
19 root     -20  0     0     0     0 S  0.0  0.0 0:00.00 cpuhp/0
20 root     -20  0     0     0     0 S  0.0  0.0 0:00.00 kdevtmpfs
21 root     -20  0     0     0     0 I  0.0  0.0 0:00.00 kworker/R-inet_
```

```
ubuntu@ip-172-31-45-149:~$ sudo shutdown now
Broadcast message from root@ip-172-31-45-149 on pts/1 (Fri 2024-08-02 12:20:48 UTC):
the system will power off now!
```

The screenshot shows the terminal output of checking system processes and resource usage using the top command. It then shows the sudo shutdown command being run, which triggers a broadcast message indicating the system will power off.