1. **Create on Bash script to check if a directory is available or not.**

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

1 additional security update can be applied with ESM Apps.
Learn more about enabling ESM Apps service at https://ubuntu.com/esm


Last login: Sun Apr 27 09:42:03 2025 from 18.206.107.29
ubuntu@ip-172-31-94-134:~$ sudo su -
root@ip-172-31-94-134:~# vi check_directory.sh
root@ip-172-31-94-134:~# chmod +x check_directory.sh
root@ip-172-31-94-134:~# ./check_directory.sh
Usage: ./check_directory.sh <directory_path>
root@ip-172-31-94-134:~# ./check_directory.sh /path/to/directory
Directory '/path/to/directory' does not exist.
root@ip-172-31-94-134:~#
```

root@ip-172-31-94-134: ~

```
#!/bin/bash

# Directory path to check (can be passed as an argument)
DIR="$1"

# Check if directory path was provided
if [ -z "$DIR" ]; then
  echo "Usage: $0 <directory_path>"
  exit 1
fi

# Check if directory exists
if [ -d "$DIR" ]; then
  echo "Directory '$DIR' exists."
else
  echo "Directory '$DIR' does not exist."
fi
~
~
~
~
~
~
~
~
~
~
```

2.Create a bash script for calculator.

```
root@ip-172-31-94-134:~# vi calculator.sh
root@ip-172-31-94-134:~# chmod +x calculator.sh
root@ip-172-31-94-134:~# ./calculator.sh
Welcome to Bash Calculator!
Enter first number:
10
Enter second number:
3
Choose an operation:
1. Addition (+)
2. Subtraction (-)
3. Multiplication (*)
4. Division (/)
1
Result: 13
root@ip-172-31-94-134:~# 2
2: command not found
root@ip-172-31-94-134:~# ./calculator.sh
Welcome to Bash Calculator!
Enter first number:
20
Enter second number:
30
Choose an operation:
1. Addition (+)
2. Subtraction (-)
3. Multiplication (*)
4. Division (/)
2
Result: -10
root@ip-172-31-94-134:~#
```

root@ip-172-31-94-134: ~

```
echo "Enter second number:"
read num2

echo "Choose an operation:"
echo "1. Addition (+)"
echo "2. Subtraction (-)"
echo "3. Multiplication (*)"
echo "4. Division (/)"
read operation

case $operation in
  1)
    result=$(echo "$num1 + $num2" | bc)
    echo "Result: $result"
    ;;
  2)
    result=$(echo "$num1 - $num2" | bc)
    echo "Result: $result"
    ;;
  3)
    result=$(echo "$num1 * $num2" | bc)
    echo "Result: $result"
    ;;
  4)
    if [ "$num2" == "0" ]; then
      echo "Error: Division by zero is not allowed."
    else
      result=$(echo "scale=2; $num1 / $num2" | bc)
      echo "Result: $result"
    fi
    ;;
  *)
    echo "Invalid operation selected."
    ;;
esac

"calculator.sh" 44L, 802B
```

3. **Create bash script to delete last 3 lines for a file.**

```
root@ip-172-31-94-134: ~
root@ip-172-31-94-134:~# vi delete.sh
root@ip-172-31-94-134:~# chmod +x delete.sh
root@ip-172-31-94-134:~# ./delete.sh
Usage: ./delete.sh <file_path>
root@ip-172-31-94-134:~# ./delete.sh yourfile.txt
Error: File 'yourfile.txt' not found!
root@ip-172-31-94-134:~# ./delete_last3.sh /home/username/Documents/myfile.txt
-bash: ./delete_last3.sh: No such file or directory
root@ip-172-31-94-134:~# ls *.txt
print.txt
root@ip-172-31-94-134:~# ./delete.sh print.txt
Successfully deleted the last 3 lines from 'print.txt'.
root@ip-172-31-94-134:~#
```

```
root@ip-172-31-94-134: ~
#!/bin/bash

# Bash script to delete the last 3 lines from a file

FILE="$1"

# Check if file path is provided
if [ -z "$FILE" ]; then
  echo "Usage: $0 <file_path>"
  exit 1
fi

# Check if the file exists
if [ ! -f "$FILE" ]; then
  echo "Error: File '$FILE' not found!"
  exit 1
fi

# Remove the last 3 lines and overwrite the file
head -n -3 "$FILE" > temp_file && mv temp_file "$FILE"

echo "Successfully deleted the last 3 lines from '$FILE'."
~
~
~
~
~
~
~
~
~
~
~
~
"delete.sh" 23L, 450B
```

4. Create a bash script to install nginx in ec2 server.

```
root@ip-172-31-94-134:~#
root@ip-172-31-94-134:~# vi install_nginx.sh
root@ip-172-31-94-134:~# chmod +x install_nginx.sh
root@ip-172-31-94-134:~# ./install_nginx.sh
Updating system packages...
Hit:1 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble InRelease
Get:2 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates InRelease [126 kB]
Get:3 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-backports InRelease [126 kB]
Ign:4 https://pkg.jenkins.io/debian-stable binary/ InRelease
Hit:5 https://pkg.jenkins.io/debian-stable binary/ Release
Get:6 http://security.ubuntu.com/ubuntu noble-security InRelease [126 kB]
Get:7 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates/main amd64 Components [161 kB]
Get:8 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates/universe amd64 Components [368 kB]
Get:9 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates/restricted amd64 Components [212 B]
Get:10 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates/multiverse amd64 Components [940 B]
Get:11 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-backports/main amd64 Components [7092 B]
Get:12 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-backports/universe amd64 Components [16.4 kB]
Get:13 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-backports/restricted amd64 Components [216 B]
Get:14 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-backports/multiverse amd64 Components [212 B]
Get:16 http://security.ubuntu.com/ubuntu noble-security/main amd64 Components [21.5 kB]
Get:17 http://security.ubuntu.com/ubuntu noble-security/universe amd64 Components [52.3 kB]
Get:18 http://security.ubuntu.com/ubuntu noble-security/restricted amd64 Components [212 B]
Get:19 http://security.ubuntu.com/ubuntu noble-security/multiverse amd64 Components [208 B]
Fetched 1007 kB in 1s (959 kB/s)
Reading package lists... Done
Installing Nginx...
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
nginx is already the newest version (1.24.0-2ubuntu7.3).
The following package was automatically installed and is no longer required:
  net-tools
Use 'sudo apt autoremove' to remove it.
0 upgraded, 0 newly installed, 0 to remove and 31 not upgraded.
Starting and enabling Nginx service...
Synchronizing state of nginx.service with SysV service script with /usr/lib/systemd/systemd-sysv-install.
```

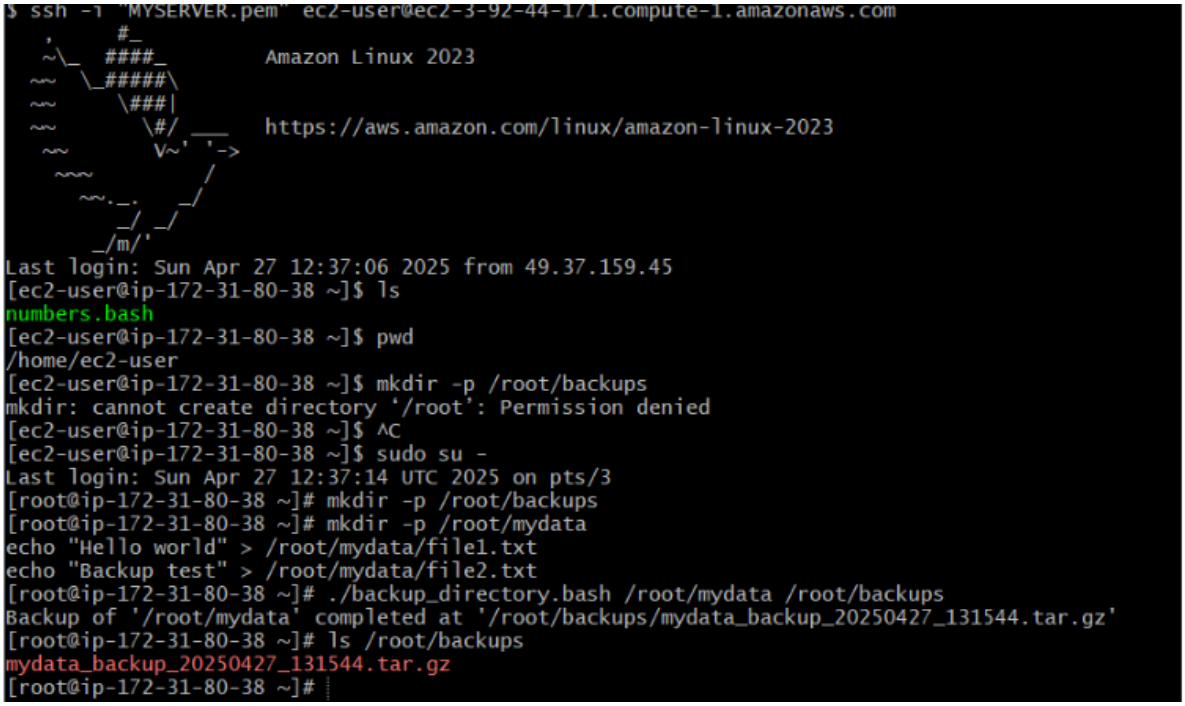
```
Reading package lists... Done
Installing Nginx...
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
nginx is already the newest version (1.24.0-2ubuntu7.3).
The following package was automatically installed and is no longer required:
  net-tools
Use 'sudo apt autoremove' to remove it.
0 upgraded, 0 newly installed, 0 to remove and 31 not upgraded.
Starting and enabling Nginx service...
Synchronizing state of nginx.service with SysV service script with /usr/lib/systemd/systemd-sysv-install.
Executing: /usr/lib/systemd/systemd-sysv-install enable nginx
Checking Nginx status...
● nginx.service - A high performance web server and a reverse proxy server
     Loaded: loaded (/usr/lib/systemd/system/nginx.service; enabled; preset: enabled)
     Active: active (running) since Mon 2025-04-28 09:18:22 UTC; 1h 8min ago
       Docs: man:nginx(8)
   Main PID: 589 (nginx)
      Tasks: 2 (limit: 1129)
     Memory: 3.1M (peak: 3.3M)
        CPU: 21ms
     CGroup: /system.slice/nginx.service
             ├─589 "nginx: master process /usr/sbin/nginx -g daemon on; master_process on;"
             └─593 "nginx: worker process"

Apr 28 09:18:21 ip-172-31-94-134 systemd[1]: Starting nginx.service - A high performance web server and a reverse proxy serve>
Apr 28 09:18:22 ip-172-31-94-134 systemd[1]: Started nginx.service - A high performance web server and a reverse proxy server.
lines 1-14/14 (END)
```

5. Create a bash script to check list if nginx service is running or not,if not running then script should start the service.

```
root@ip-172-31-94-134: ~
root@ip-172-31-94-134:~# vi check_nginx.sh
root@ip-172-31-94-134:~# chmod +x check_nginx.sh
root@ip-172-31-94-134:~# ./check_nginx.sh
Nginx service is already running.
root@ip-172-31-94-134:~#
```

```
root@ip-172-31-94-134: ~
#!/bin/bash

# Bash Script to Check if Nginx is Running and Start it if Needed

# Check if Nginx service exists
if ! systemctl list-units --type=service | grep -q nginx; then
    echo "Nginx service not found. Please install Nginx first."
    exit 1
fi

# Check the status of nginx service
service_status=$(systemctl is-active nginx)

if [ "$service_status" == "active" ]; then
    echo "Nginx service is already running."
else
    echo "Nginx service is not running. Attempting to start..."
    sudo systemctl start nginx

    # Re-check if the service started successfully
    service_status=$(systemctl is-active nginx)
    if [ "$service_status" == "active" ]; then
        echo "Nginx service started successfully!"
    else
        echo "Failed to start Nginx service. Please check manually."
        exit 1
    fi
fi
~
~
~
~
~
~
~
"check_nginx.sh" 29L, 825B
```

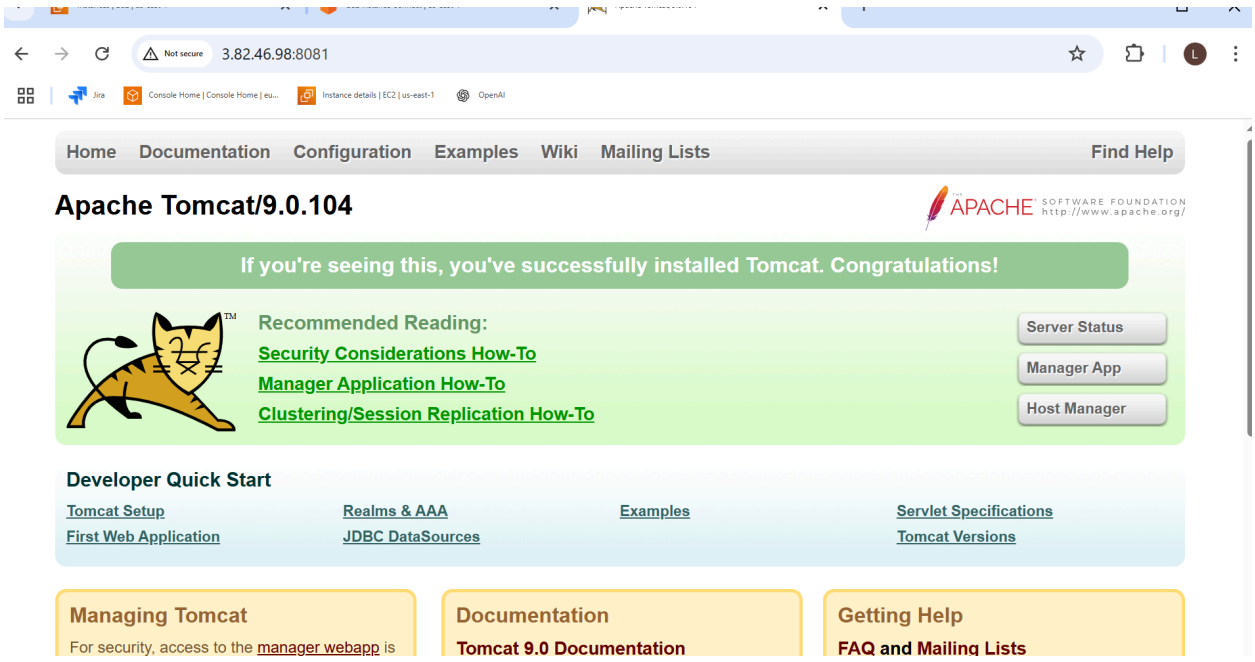
6. Create a bash script to take backup of a directory.



7. Create a bash script to install ApacheTomcat in ec2 server.

root@ip-172-31-90-27:~

```
#!/bin/bash

# Bash Script to Install Apache Tomcat on an EC2 Server

# Set the Tomcat version you want to install
TOMCAT_VERSION="10.1.20"

# Set installation directory
INSTALL_DIR="/opt/tomcat"

# Update the system
echo "Updating system packages..."
if grep -i "amazon linux" /etc/os-release; then
    sudo yum update -y
elif grep -i "ubuntu" /etc/os-release; then
    sudo apt-get update -y
else
    echo "Unsupported OS."
    exit 1
fi

# Install Java (Tomcat requires Java)
echo "Installing Java..."
if grep -i "amazon linux" /etc/os-release; then
    sudo amazon-linux-extras enable corretto11
    sudo yum install java-11-amazon-corretto -y
elif grep -i "ubuntu" /etc/os-release; then
    sudo apt-get install openjdk-11-jdk -y
else
    echo "Unsupported OS for Java installation."
    exit 1
fi

# Verify Java installation
if ! java -version >/dev/null 2>&1; then
    echo "Java installation failed."
    exit 1
fi

-- INSERT --
```

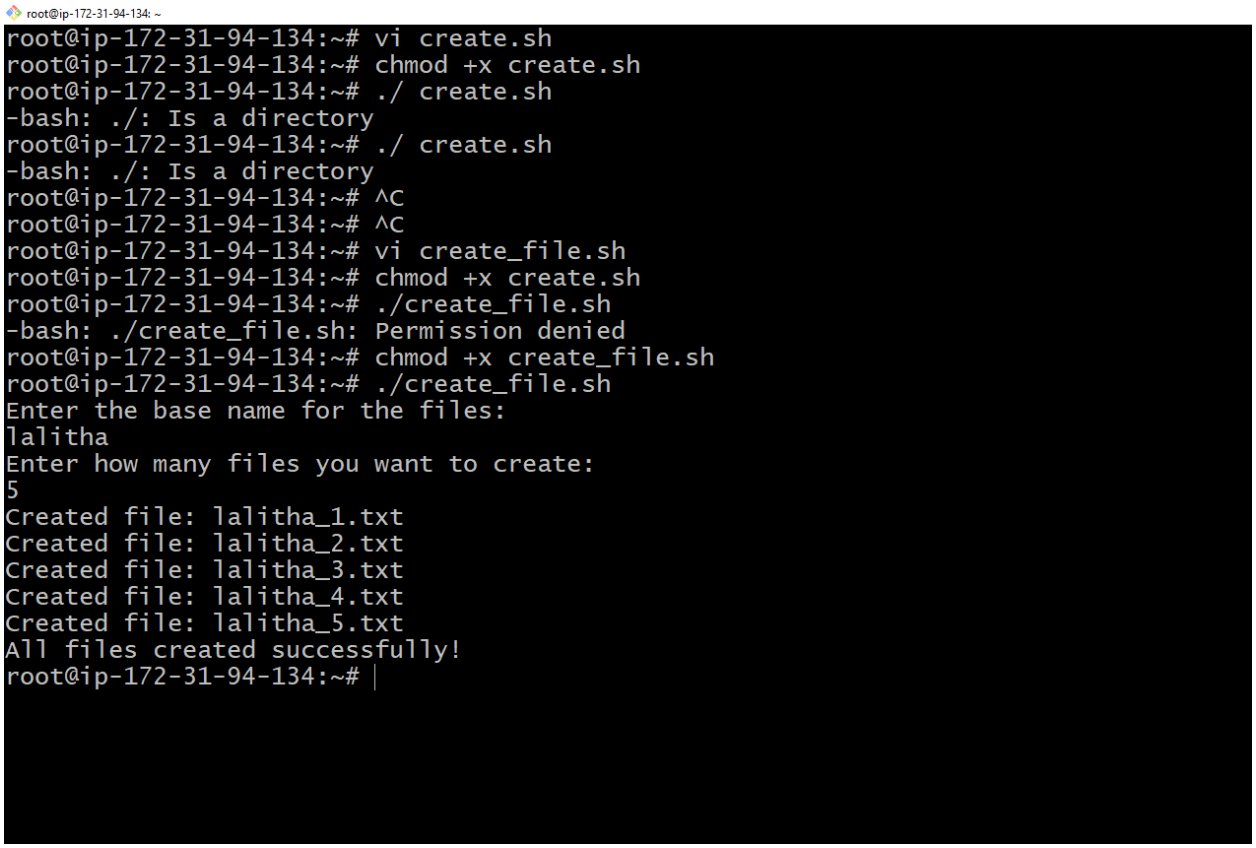


8. Create a bash script to check if directory is avaialble or not,if not then create a directory.

```
#!/bin/bash

# Usage: ./check_create_dir.sh /path/to/directory

DIR="$1"

# Check if directory path was provided
if [[ -z "$DIR" ]]; then
  echo "Usage: $0 /path/to/directory"
  exit 1
fi

# Check if the directory exists
if [[ -d "$DIR" ]]; then
  echo "Directory already exists: $DIR"
else
  echo "Directory not found. Creating: $DIR"
  mkdir -p "$DIR"
  echo "Directory created: $DIR"
fi
```

```
[root@ip-172-31-80-38 ~]# vim check_create_dir.sh
[root@ip-172-31-80-38 ~]# chmod +x check_create_dir.sh
[root@ip-172-31-80-38 ~]# ./check_create_dir.sh /tmp/mynewdir
Directory not found. Creating: /tmp/mynewdir
Directory created: /tmp/mynewdir
[root@ip-172-31-80-38 ~]#
```

9. Create a bash script which will create multiple files.


root@ip-172-31-94-134: ~
root@ip-172-31-94-134:~# vi create.sh
root@ip-172-31-94-134:~# chmod +x create.sh
root@ip-172-31-94-134:~# ./ create.sh
-bash: ./: Is a directory
root@ip-172-31-94-134:~# ./ create.sh
-bash: ./: Is a directory
root@ip-172-31-94-134:~# ^C
root@ip-172-31-94-134:~# ^C
root@ip-172-31-94-134:~# vi create_file.sh
root@ip-172-31-94-134:~# chmod +x create.sh
root@ip-172-31-94-134:~# ./create_file.sh
-bash: ./create_file.sh: Permission denied
root@ip-172-31-94-134:~# chmod +x create_file.sh
root@ip-172-31-94-134:~# ./create_file.sh
Enter the base name for the files:
lalitha
Enter how many files you want to create:
5
Created file: lalitha_1.txt
Created file: lalitha_2.txt
Created file: lalitha_3.txt
Created file: lalitha_4.txt
Created file: lalitha_5.txt
All files created successfully!
root@ip-172-31-94-134:~#

root@ip-172-31-94-134: ~

```
#!/bin/bash

# Bash script to create multiple files

echo "Enter the base name for the files:"
read base_name

echo "Enter how many files you want to create:"
read count

# Check if count is a valid number
if ! [[ "$count" =~ ^[0-9]+$ ]]; then
  echo "Error: Count must be a number."
  exit 1
fi

# Loop to create files
for ((i=1; i<=count; i++))
do
  filename="${base_name}_${i}.txt"
  touch "$filename"
  echo "Created file: $filename"
done

echo "All files created successfully!"
~
~
~
~
~
~
~
~
~
~
"create_file.sh" 26L, 484B
```