Day 7 - Shell Scripting for DevOps

Automation : Automation is a process where youll try to reduce your manual activity. Wherther you are using linux in aws,machine.

to print 1 to 10, we can use echo command. if its 1000 still we can.
but if we increase zeros its difficult to create.

Touch : here we are using Touch command to create files several times.

$ touch sample.sh
$ls -> listing the files / folders
$ls -ltr ->
$man -> Manual (command description)

Touch Command is used in automations, to create several files like 1000's which wont open

Vim Command is used to create and open the files.

Cat Command is used to print data without opening the files

To write a shellscript in file, we can use vim, vi or nano

$ sudo apt install vim -y
$vim --version

#!/bin/<bash-/-dash-/-sh-/-ksh> -> shebang - use shebang command in shell. any of these bash-/-dash-/-sh-/-ksh who takes responsibility for executable.

mostly used are #!/bin/sh -> redirects(softlink & hard link) to bash

- > sh takes request and handover to bash

Ubuntu links sh to dash

#!/bin/dash

The major difference between #!/bin/sh #!/bin/dash

earlier when we use sh its redirected to default bash scripting based on the systems

now systems are modified to redirect to dash, so its advised to share script in dash so that other can run it easily, else they will get errors.

$ vim <file_name>.sh (enter "i" to switch inster mode, to save type "ESC" then "wq!")

to execute the script :

we can use $ sh <file_name>.sh (to print the file - only)

$ ./<file_name>.sh -> prints text

$ cat <file_name>.sh (to print entire shell script)

file permissions in linux :

even though we created the file, we need permission to it.

Granting permission : Chmod (ch stands for change, mod means modification).

chmod is divided into 3 types : 4-2-1 (read-write-execute)

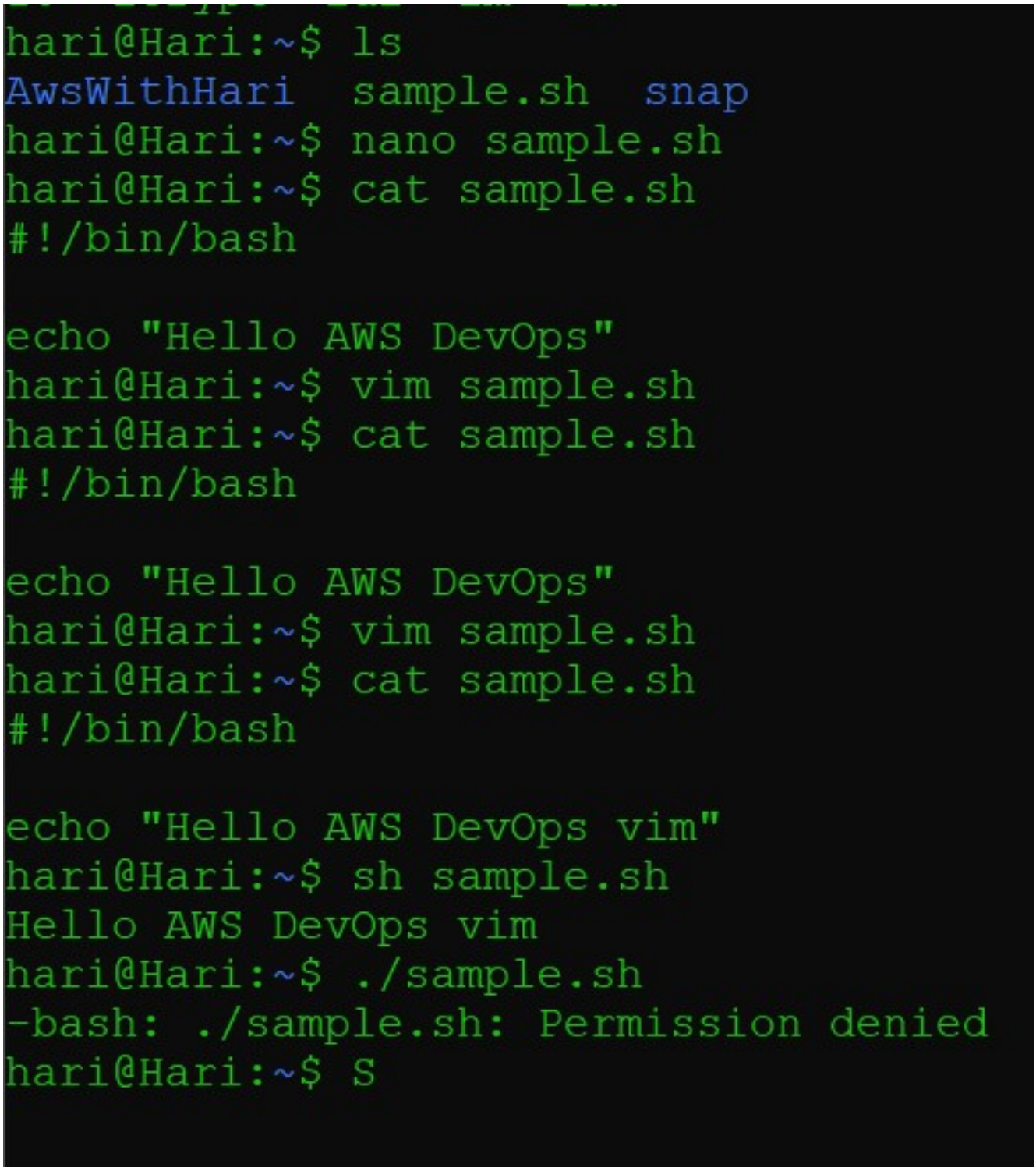
1. what are permissions for admin/user -> 777
2. which group has access -> 77
3. what permissions everyone has -> 7

using number we grand access

$ chmod 444 <file_name>.sh (3 people can read this file)

$ chmod <file_name>.sh (3 people can read this file)

History command : $ history shows the commands you entered so far



Creating a shell script to create file in a folder :

Script file :

#!/bin/bash
#create a folder
mkdir automate_folder

#create two files

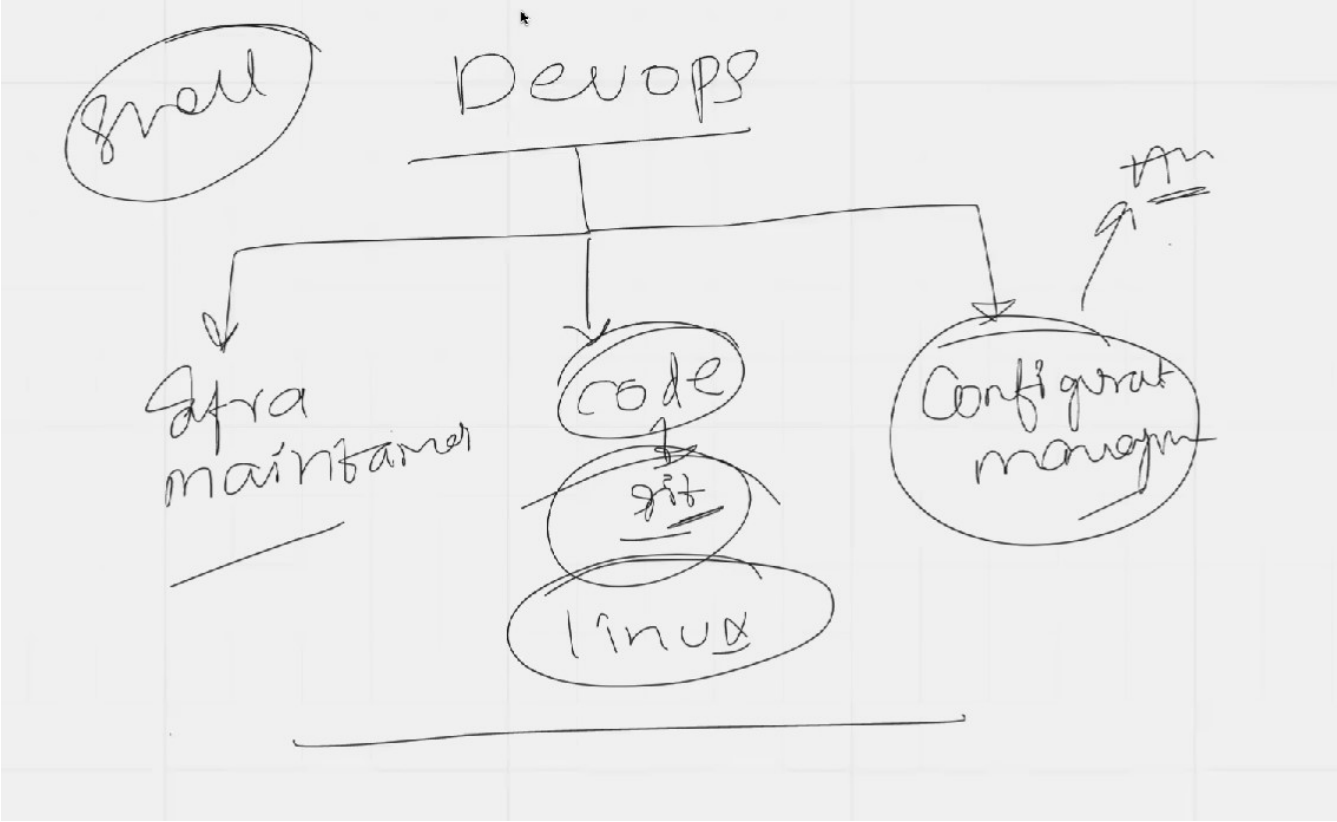
cd automate_folder
touch firstfile secondfile

```
hari@Hari:~$ vim automate_sample.sh
hari@Hari:~$ cat automate_sample.sh
#!/bin/bash
#create a folder
mkdir automate_folder

#create two files
cd automate_folder
touch firstfile secondfile
hari@Hari:~$ ./automate_sample.sh
-bash: ./automate_sample.sh: Permission denied
hari@Hari:~$ chmod 777 automate_sample.sh
hari@Hari:~$ ./automate_sample.sh
hari@Hari:~$ ls
AwsWithHari  automate_folder  automate_sample.sh  sample.sh  snap
hari@Hari:~$ cd automate_folder
hari@Hari:~/automate_folder$ ls
firstfile  secondfile
hari@Hari:~/automate_folder$
```

```
hari@Hari:~/automate_folder$ cd ../..
hari@Hari:/home$ ls
hari
hari@Hari:/home$ cd hari
hari@Hari:~$ ls
AwsWithHari  automate_folder  automate_sample.sh  sample.sh  snap
hari@Hari:~$ chmod 000 automate_sample.sh
hari@Hari:~$ ./automate_sample.sh
-bash: ./automate_sample.sh: Permission denied
hari@Hari:~$
```

```
hari@Hari:~/automate_folder$ cd ../..
hari@Hari:/home$ ls
hari
hari@Hari:/home$ cd hari
hari@Hari:~$ ls
AwsWithHari  automate_folder  automate_sample.sh  sample.sh  snap
hari@Hari:~$ chmod 000 automate_sample.sh
hari@Hari:~$ ./automate_sample.sh
-bash: ./automate_sample.sh: Permission denied
hari@Hari:~$
```


Devops
code
git
linux