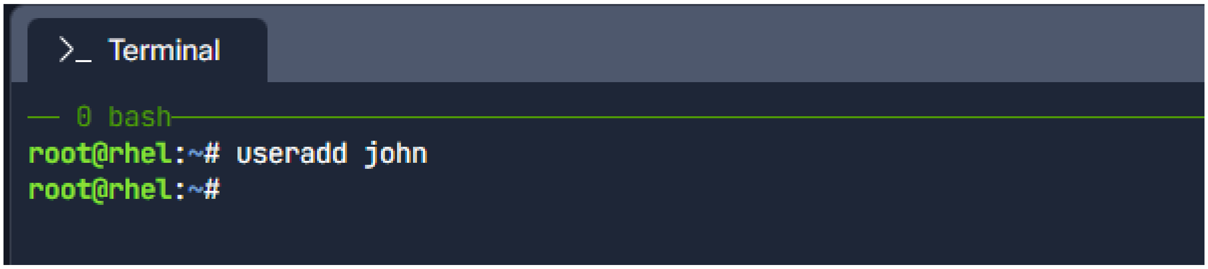
Lab Project - 1

Object: Linux user management lab tasks

1. Create a New User

Q1:-Use the useradd command to create a new user, e.g., john

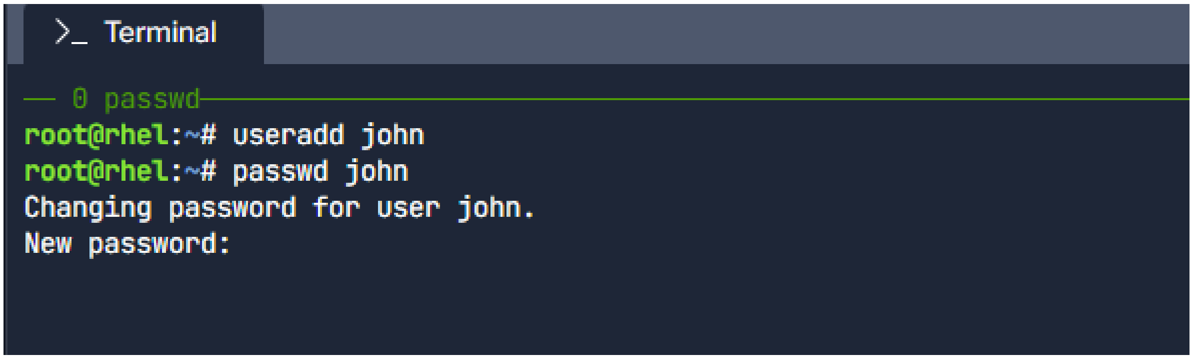
Ans:-



```
root@rhel:~# useradd john
root@rhel:~#
```

Q1:-Set a password for the new user using passwd.

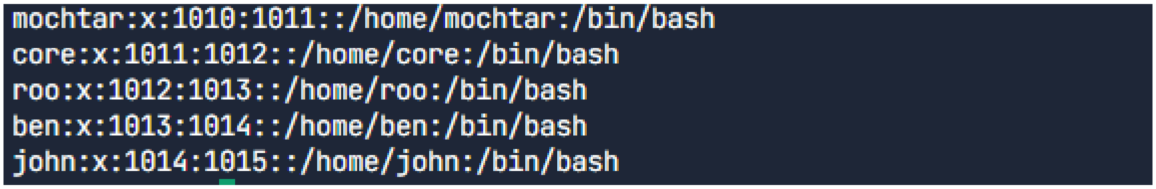
Ans:-



```
root@rhel:~# useradd john
root@rhel:~# passwd john
Changing password for user john.
New password:
```

Verify the new user by checking the /etc/passwd file.

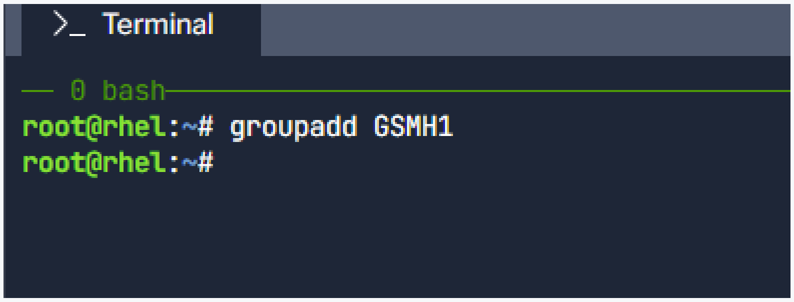
Ans:- cat /etc/passwd



```
mochtar:x:1010:1011::/home/mochtar:/bin/bash
core:x:1011:1012::/home/core:/bin/bash
roo:x:1012:1013::/home/roo:/bin/bash
ben:x:1013:1014::/home/ben:/bin/bash
john:x:1014:1015::/home/john:/bin/bash
```

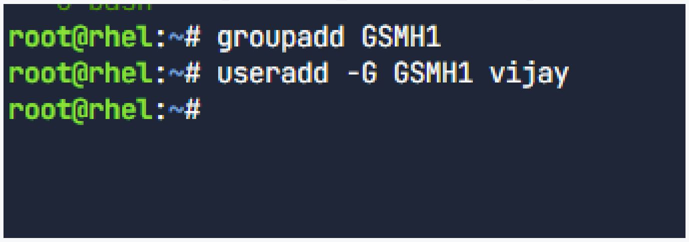
2. Add a User to a Group

1.Create a new group (e.g., developers) using groupadd



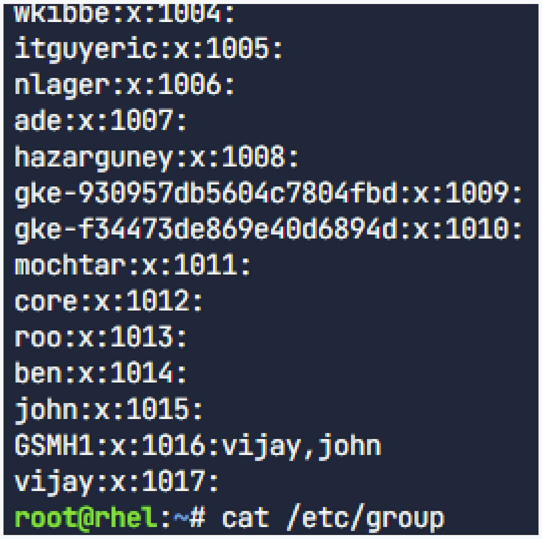
```
root@rhel:~# groupadd GSMH1
root@rhel:~#
```

2 Ans:- Add an existing user (e.g., john) to the group using usermod



```
root@rhel:~# groupadd GSMH1
root@rhel:~# useradd -G GSMH1 vijay
root@rhel:~#
```

3 Verify that the user is added to the group by using the groups command.

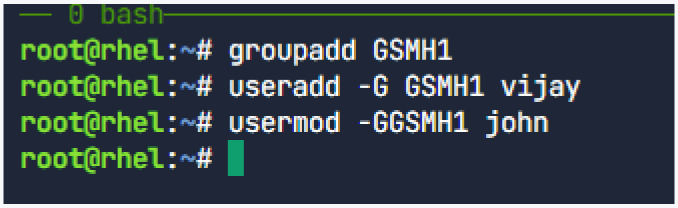


```
wklbbe:x:1004:
itguyeric:x:1005:
nlager:x:1006:
ade:x:1007:
hazarguney:x:1008:
gke-930957db5604c7804fbd:x:1009:
gke-f34473de869e40d6894d:x:1010:
mochtar:x:1011:
core:x:1012:
roo:x:1013:
ben:x:1014:
john:x:1015:
GSMH1:x:1016:vijay,john
vijay:x:1017:
root@rhel:~# cat /etc/group
```

3. Modify User Information

how to modify user attributes.

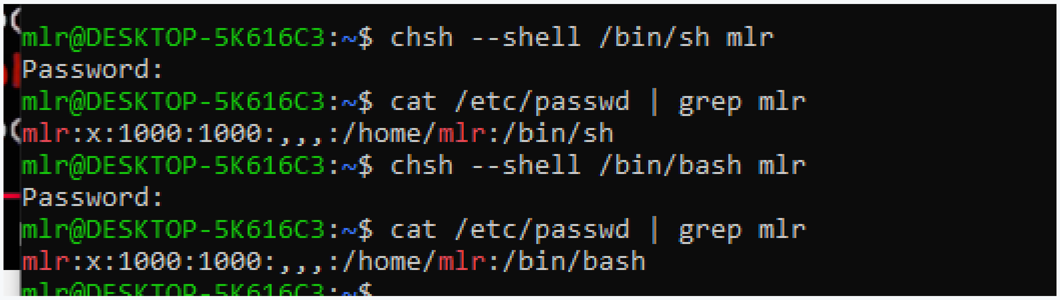
1)Modify the home directory for user john using usermod.



```
── 0 bash──────────────────────────
root@rhel:~# groupadd GSMH1
root@rhel:~# useradd -G GSMH1 vijay
root@rhel:~# usermod -GGSMH1 john
root@rhel:~# 
```

2) Change the default shell for john to /bin/bash.

Man chsh

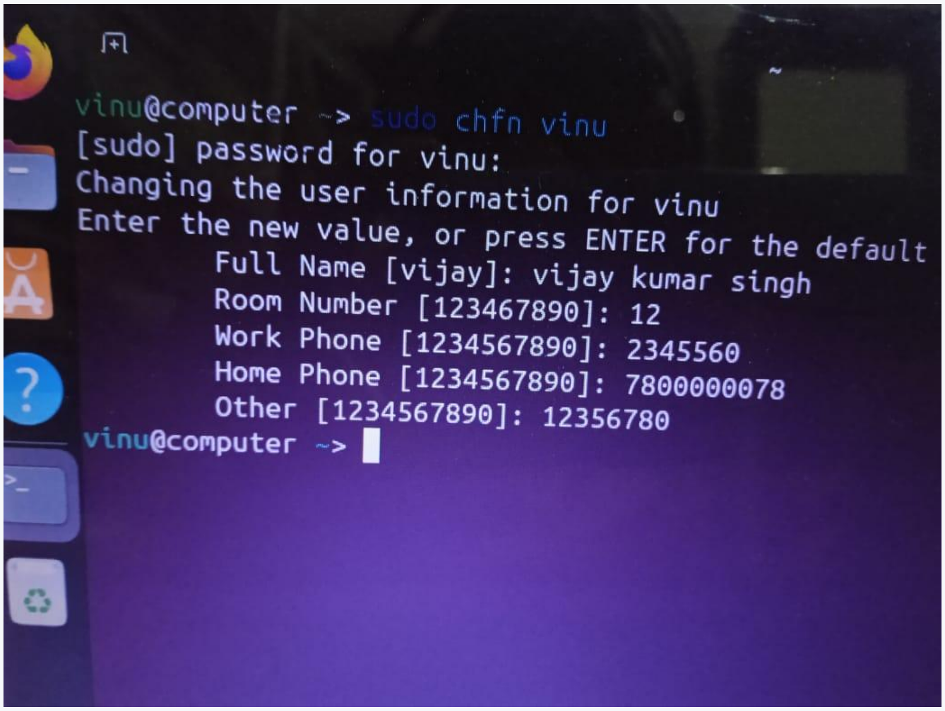


```
mlr@DESKTOP-5K616C3:~$ chsh --shell /bin/sh mlr
Password:
mlr@DESKTOP-5K616C3:~$ cat /etc/passwd | grep mlr
mlr:x:1000:1000:,,,:/home/mlr:/bin/sh
mlr@DESKTOP-5K616C3:~$ chsh --shell /bin/bash mlr
Password:
mlr@DESKTOP-5K616C3:~$ cat /etc/passwd | grep mlr
mlr:x:1000:1000:,,,:/home/mlr:/bin/bash
mlr@DESKTOP-5K616C3:~$
```

Q3:- Change the user's full name using the chfn command.

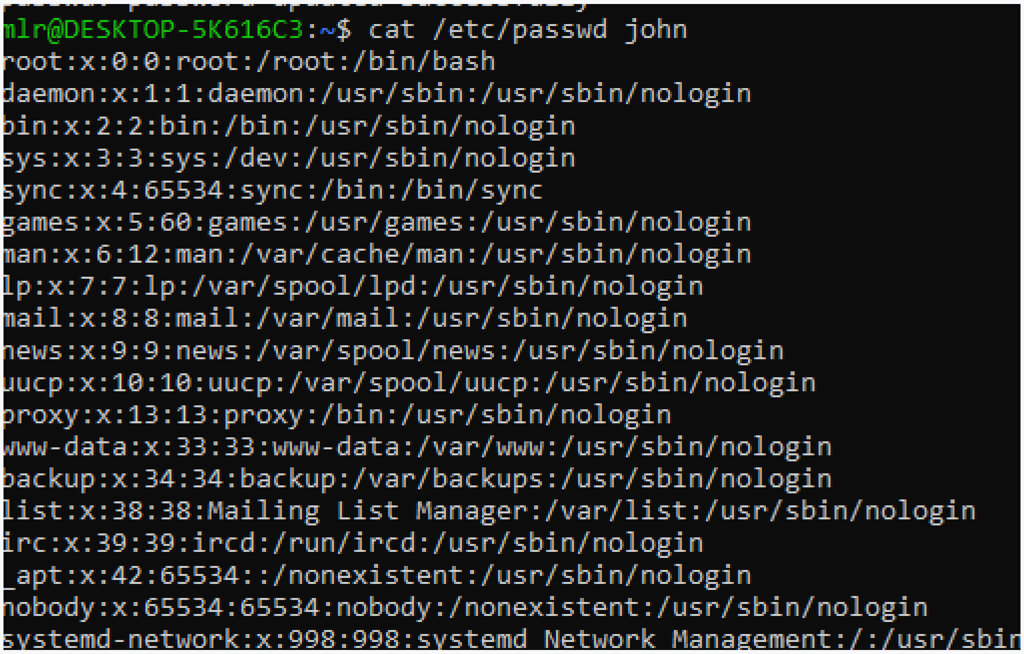
Ans:-

How to Change the Information of a Linux User using CHFN command



```
vinu@computer ~> sudo chfn vinu
[sudo] password for vinu:
Changing the user information for vinu
Enter the new value, or press ENTER for the default
        Full Name [vijay]: vijay kumar singh
        Room Number [123467890]: 12
        Work Phone [1234567890]: 2345560
        Home Phone [1234567890]: 7800000078
        Other [1234567890]: 12356780
vinu@computer ~>
```

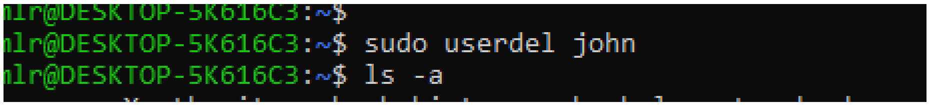
Q:- Verify the changes using grep john /etc/passwd.



```
mlr@DESKTOP-5K616C3:~$ cat /etc/passwd john
root:x:0:0:root:/root:/bin/bash
daemon:x:1:1:daemon:/usr/sbin:/usr/sbin/nologin
bin:x:2:2:bin:/bin:/usr/sbin/nologin
sys:x:3:3:sys:/dev:/usr/sbin/nologin
sync:x:4:65534:sync:/bin:/bin/sync
games:x:5:60:games:/usr/games:/usr/sbin/nologin
man:x:6:12:man:/var/cache/man:/usr/sbin/nologin
lp:x:7:7:lp:/var/spool/lpd:/usr/sbin/nologin
mail:x:8:8:mail:/var/mail:/usr/sbin/nologin
news:x:9:9:news:/var/spool/news:/usr/sbin/nologin
uucp:x:10:10:uucp:/var/spool/uucp:/usr/sbin/nologin
proxy:x:13:13:proxy:/bin:/usr/sbin/nologin
www-data:x:33:33:www-data:/var/www:/usr/sbin/nologin
backup:x:34:34:backup:/var/backups:/usr/sbin/nologin
list:x:38:38:Mailing List Manager:/var/list:/usr/sbin/nologin
irc:x:39:39:ircd:/run/ircd:/usr/sbin/nologin
_apt:x:42:65534::/nonexistent:/usr/sbin/nologin
nobody:x:65534:65534:nobody:/nonexistent:/usr/sbin/nologin
systemd-network:x:998:998:systemd Network Management:/:/usr/sbin
```

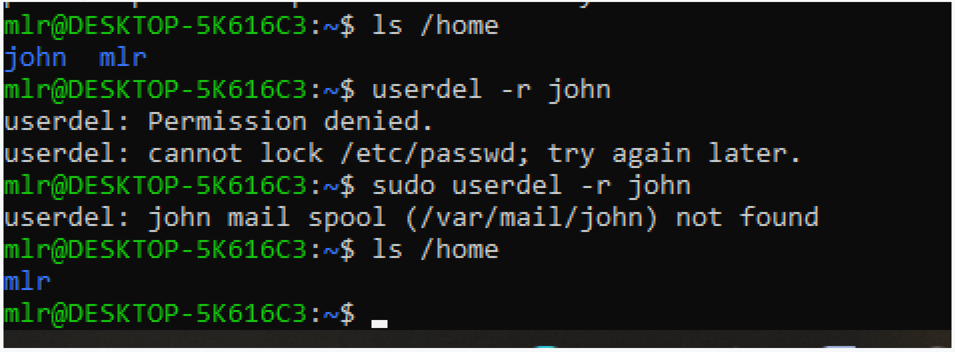
4) Delete a User

Q1:- Delete the user john using the userdel command.



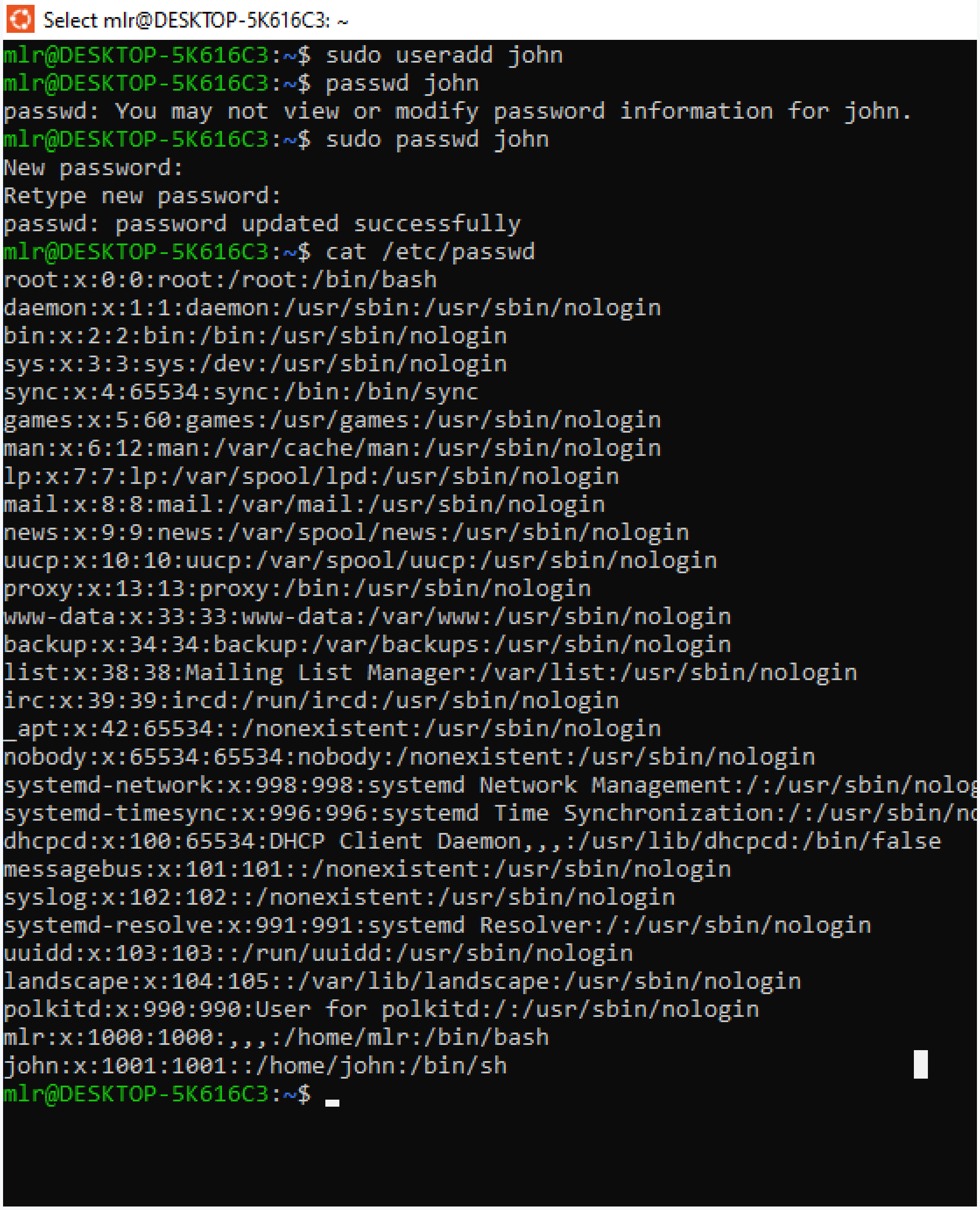
```
mlr@DESKTOP-5K616C3:~$
mlr@DESKTOP-5K616C3:~$ sudo userdel john
mlr@DESKTOP-5K616C3:~$ ls -a
```

Q2:-Ensure the user's home directory and files are removed by using userdel -r



```
mlr@DESKTOP-5K616C3:~$ ls /home
john  mlr
mlr@DESKTOP-5K616C3:~$ userdel -r john
userdel: Permission denied.
userdel: cannot lock /etc/passwd; try again later.
mlr@DESKTOP-5K616C3:~$ sudo userdel -r john
userdel: john mail spool (/var/mail/john) not found
mlr@DESKTOP-5K616C3:~$ ls /home
mlr
mlr@DESKTOP-5K616C3:~$ _
```

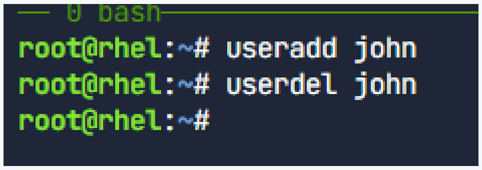
Q3:-Verify the deletion by checking the /etc/passwd file.



```
mlr@DESKTOP-5K616C3:~$ sudo useradd john
mlr@DESKTOP-5K616C3:~$ passwd john
passwd: You may not view or modify password information for john.
mlr@DESKTOP-5K616C3:~$ sudo passwd john
New password:
Retype new password:
passwd: password updated successfully
mlr@DESKTOP-5K616C3:~$ cat /etc/passwd
root:x:0:0:root:/root:/bin/bash
daemon:x:1:1:daemon:/usr/sbin:/usr/sbin/nologin
bin:x:2:2:bin:/bin:/usr/sbin/nologin
sys:x:3:3:sys:/dev:/usr/sbin/nologin
sync:x:4:65534:sync:/bin:/bin/sync
games:x:5:60:games:/usr/games:/usr/sbin/nologin
man:x:6:12:man:/var/cache/man:/usr/sbin/nologin
lp:x:7:7:lp:/var/spool/lpd:/usr/sbin/nologin
mail:x:8:8:mail:/var/mail:/usr/sbin/nologin
news:x:9:9:news:/var/spool/news:/usr/sbin/nologin
uucp:x:10:10:uucp:/var/spool/uucp:/usr/sbin/nologin
proxy:x:13:13:proxy:/bin:/usr/sbin/nologin
www-data:x:33:33:www-data:/var/www:/usr/sbin/nologin
backup:x:34:34:backup:/var/backups:/usr/sbin/nologin
list:x:38:38:Mailing List Manager:/var/list:/usr/sbin/nologin
irc:x:39:39:ircd:/run/ircd:/usr/sbin/nologin
_apt:x:42:65534::/nonexistent:/usr/sbin/nologin
nobody:x:65534:65534:nobody:/nonexistent:/usr/sbin/nologin
systemd-network:x:998:998:systemd Network Management:/:/usr/sbin/nolog
systemd-timesync:x:996:996:systemd Time Synchronization:/:/usr/sbin/no
dhcpcd:x:100:65534:DHCP Client Daemon,,,:/usr/lib/dhcpcd:/bin/false
messagebus:x:101:101::/nonexistent:/usr/sbin/nologin
syslog:x:102:102::/nonexistent:/usr/sbin/nologin
systemd-resolve:x:991:991:systemd Resolver:/:/usr/sbin/nologin
uuidd:x:103:103::/run/uuidd:/usr/sbin/nologin
landscape:x:104:105::/var/lib/landscape:/usr/sbin/nologin
polkitd:x:990:990:User for polkitd:/:/usr/sbin/nologin
mlr:x:1000:1000:,,,:/home/mlr:/bin/bash
john:x:1001:1001::/home/john:/bin/sh
mlr@DESKTOP-5K616C3:~$ 
```

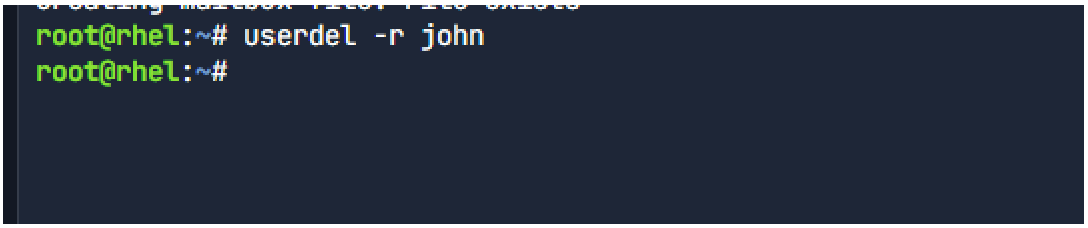
4. Delete a User

Q1. Delete the user john using the userdel command.



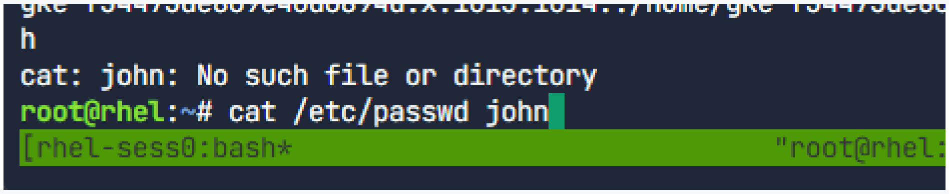
```
── 0 bash──
root@rhel:~# useradd john
root@rhel:~# userdel john
root@rhel:~#
```

Q2:- Ensure the user's home directory and files are removed by using userdel -r.



```
Creating mailbox file: File exists
root@rhel:~# userdel -r john
root@rhel:~#
```

Q3:- Verify the deletion by checking the /etc/passwd file.



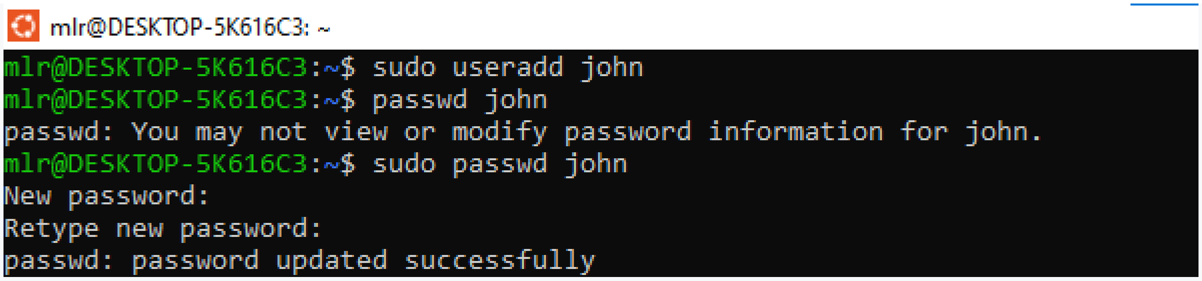
```
h
cat: john: No such file or directory
root@rhel:~# cat /etc/passwd john
[rhel-sess0:bash*                              "root@rhel:
```

5. Create a System User

Task:

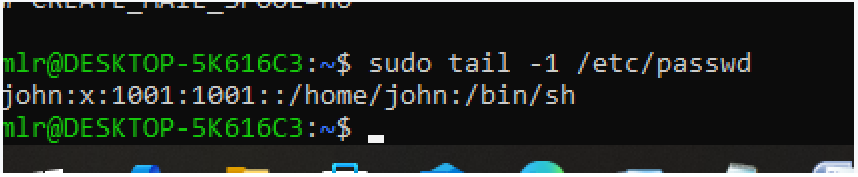
```
1.Create a system user for an application (e.g., www-data for web server
users).
2.Ensure that the system user has no login shell and that no home
directory is created by using useradd -r.
3.Verify the user is created with no login shell by inspecting
/etc/passwd.
```

Q1:- 1.Create a system user for an application (e.g., www-data for web server users).

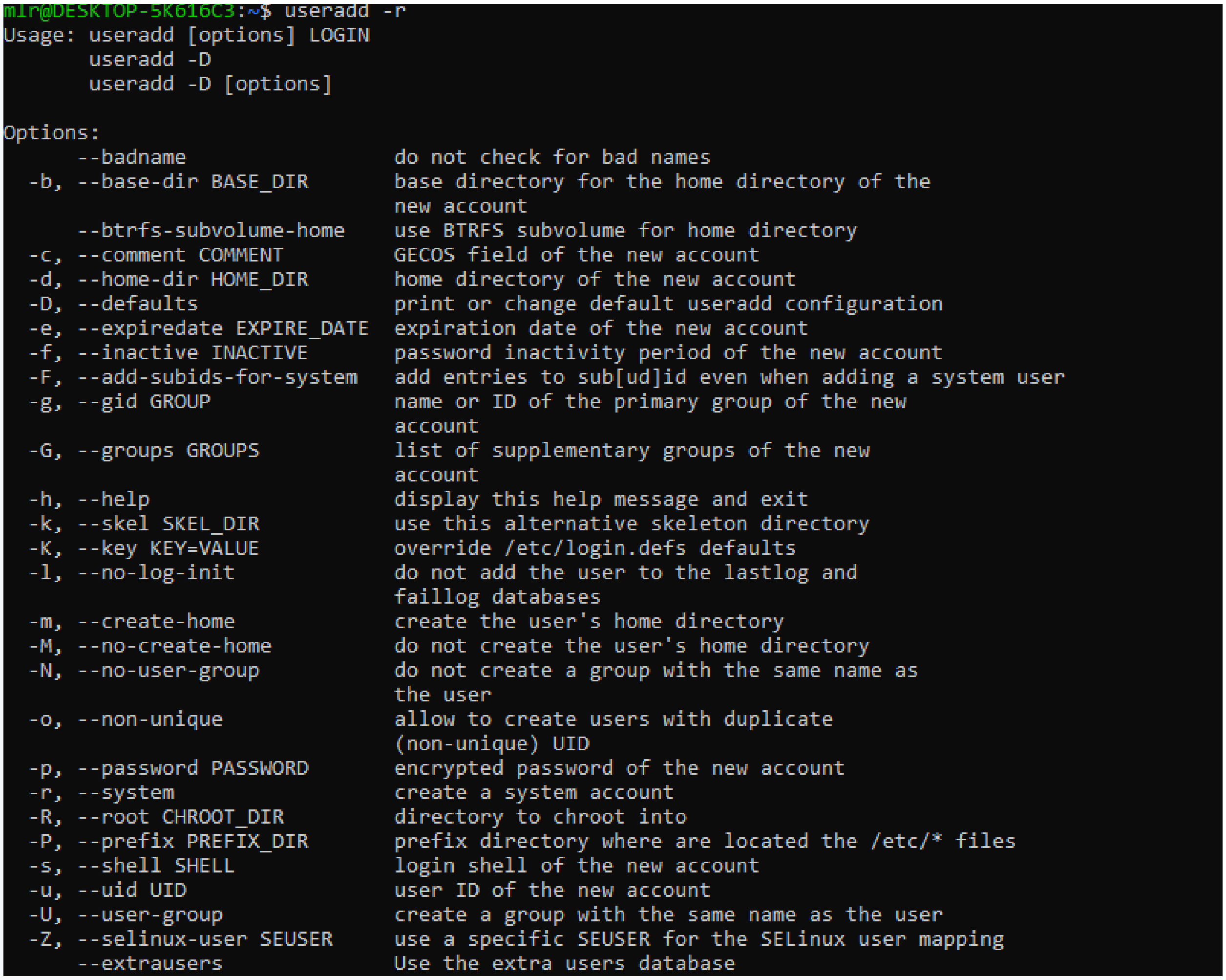


```
mlr@DESKTOP-5K616C3:~$ sudo useradd john
mlr@DESKTOP-5K616C3:~$ passwd john
passwd: You may not view or modify password information for john.
mlr@DESKTOP-5K616C3:~$ sudo passwd john
New password:
Retype new password:
passwd: password updated successfully
```

Q2:- Ensure that the system user has no login shell and that no home directory is created by using useradd -r.

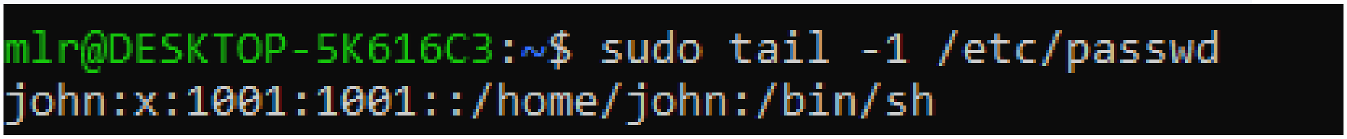


```
# CREATE_MAIL_SPOOL=no

mlr@DESKTOP-5K616C3:~$ sudo tail -1 /etc/passwd
john:x:1001:1001::/home/john:/bin/sh
mlr@DESKTOP-5K616C3:~$ 
```



```
mlr@DESKTOP-5K616C3:~$ useradd -r
Usage: useradd [options] LOGIN
       useradd -D
       useradd -D [options]

Options:
      --badname                 do not check for bad names
  -b, --base-dir BASE_DIR       base directory for the home directory of the
                                new account
      --btrfs-subvolume-home    use BTRFS subvolume for home directory
  -c, --comment COMMENT         GECOS field of the new account
  -d, --home-dir HOME_DIR       home directory of the new account
  -D, --defaults                print or change default useradd configuration
  -e, --expiredate EXPIRE_DATE  expiration date of the new account
  -f, --inactive INACTIVE       password inactivity period of the new account
  -F, --add-subids-for-system   add entries to sub[ud]id even when adding a system user
  -g, --gid GROUP               name or ID of the primary group of the new
                                account
  -G, --groups GROUPS           list of supplementary groups of the new
                                account
  -h, --help                    display this help message and exit
  -k, --skel SKEL_DIR           use this alternative skeleton directory
  -K, --key KEY=VALUE           override /etc/login.defs defaults
  -l, --no-log-init             do not add the user to the lastlog and
                                faillog databases
  -m, --create-home             create the user's home directory
  -M, --no-create-home          do not create the user's home directory
  -N, --no-user-group           do not create a group with the same name as
                                the user
  -o, --non-unique              allow to create users with duplicate
                                (non-unique) UID
  -p, --password PASSWORD       encrypted password of the new account
  -r, --system                  create a system account
  -R, --root CHROOT_DIR         directory to chroot into
  -P, --prefix PREFIX_DIR       prefix directory where are located the /etc/* files
  -s, --shell SHELL             login shell of the new account
  -u, --uid UID                 user ID of the new account
  -U, --user-group              create a group with the same name as the user
  -Z, --selinux-user SEUSER     use a specific SEUSER for the SELinux user mapping
      --extrausers              Use the extra users database
```

Q: -Verify the user is created with no login shell by inspecting /etc/passwd.

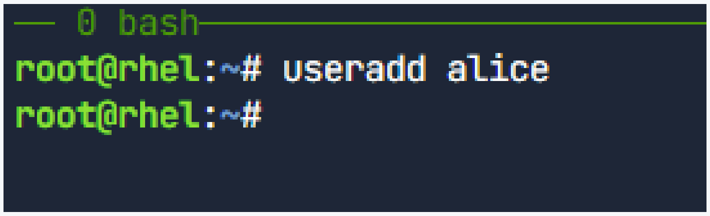


```
mlr@DESKTOP-5K616C3:~$ sudo tail -1 /etc/passwd
john:x:1001:1001::/home/john:/bin/sh
```

6. Managing User Permissions

Objective: Learn how to manage file permissions for users.

Q1:-Create a new user alice.
Ans.

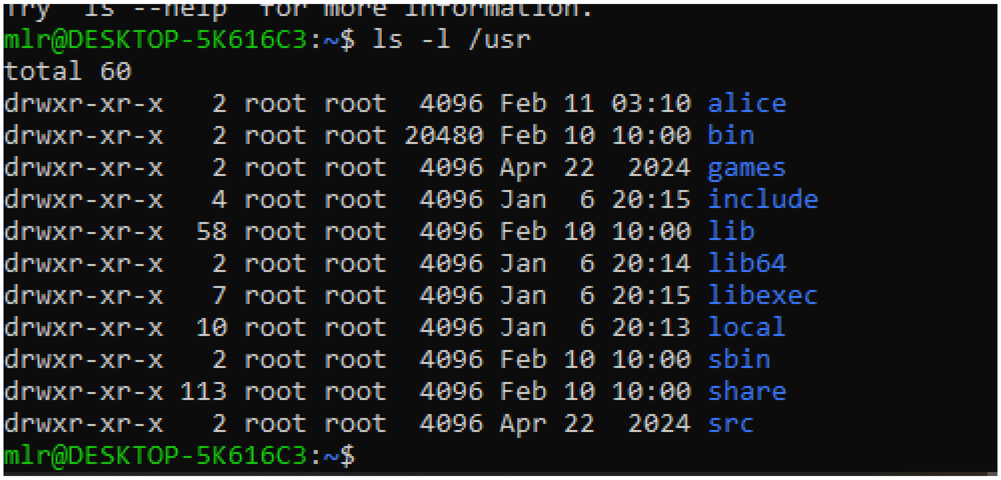


```
── 0 bash────────────────────────────
root@rhel:~# useradd alice
root@rhel:~#
```

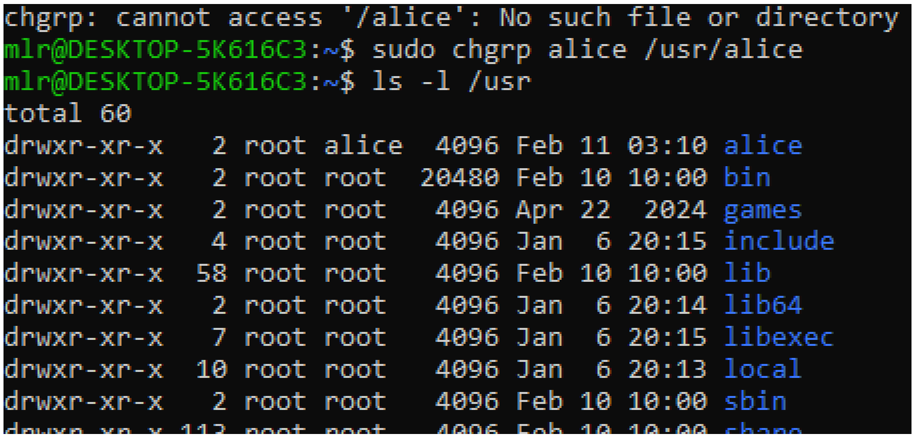
Q2:- Create a directory /home/alice_data and set it as rw for the owner, r for the group, and no permissions for others.

Ans:-

Q3:- Add alice to the group that has access to this directory.
Ans:-

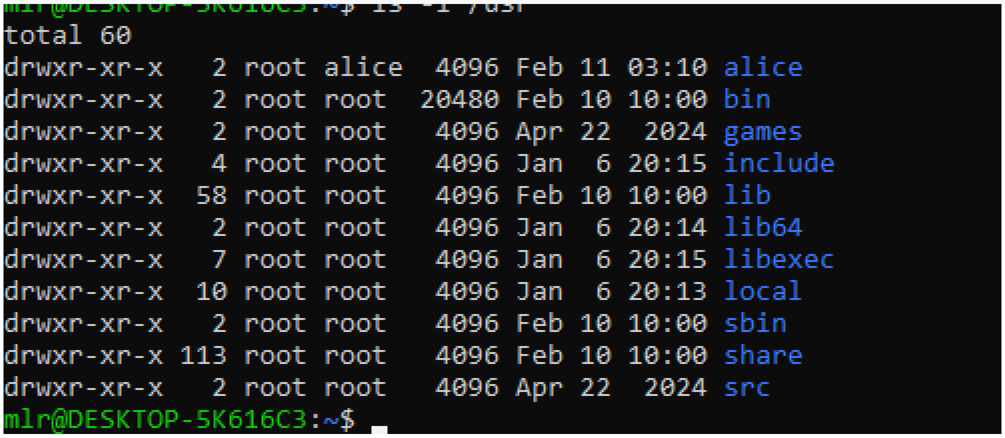


```
Try 'ls --help' for more information.
mlr@DESKTOP-5K616C3:~$ ls -l /usr
total 60
drwxr-xr-x   2 root root  4096 Feb 11 03:10 alice
drwxr-xr-x   2 root root 20480 Feb 10 10:00 bin
drwxr-xr-x   2 root root  4096 Apr 22  2024 games
drwxr-xr-x   4 root root  4096 Jan  6 20:15 include
drwxr-xr-x  58 root root  4096 Feb 10 10:00 lib
drwxr-xr-x   2 root root  4096 Jan  6 20:14 lib64
drwxr-xr-x   7 root root  4096 Jan  6 20:15 libexec
drwxr-xr-x  10 root root  4096 Jan  6 20:13 local
drwxr-xr-x   2 root root  4096 Feb 10 10:00 sbin
drwxr-xr-x 113 root root  4096 Feb 10 10:00 share
drwxr-xr-x   2 root root  4096 Apr 22  2024 src
mlr@DESKTOP-5K616C3:~$
```



```
chgrp: cannot access '/alice': No such file or directory
mlr@DESKTOP-5K616C3:~$ sudo chgrp alice /usr/alice
mlr@DESKTOP-5K616C3:~$ ls -l /usr
total 60
drwxr-xr-x   2 root alice  4096 Feb 11 03:10 alice
drwxr-xr-x   2 root root  20480 Feb 10 10:00 bin
drwxr-xr-x   2 root root   4096 Apr 22  2024 games
drwxr-xr-x   4 root root   4096 Jan  6 20:15 include
drwxr-xr-x  58 root root   4096 Feb 10 10:00 lib
drwxr-xr-x   2 root root   4096 Jan  6 20:14 lib64
drwxr-xr-x   7 root root   4096 Jan  6 20:15 libexec
drwxr-xr-x  10 root root   4096 Jan  6 20:13 local
drwxr-xr-x   2 root root   4096 Feb 10 10:00 sbin
drwxr-xr-x 113 root root   4096 Feb 10 10:00 share
```

Q4 :- Verify the permissions using ls -l.

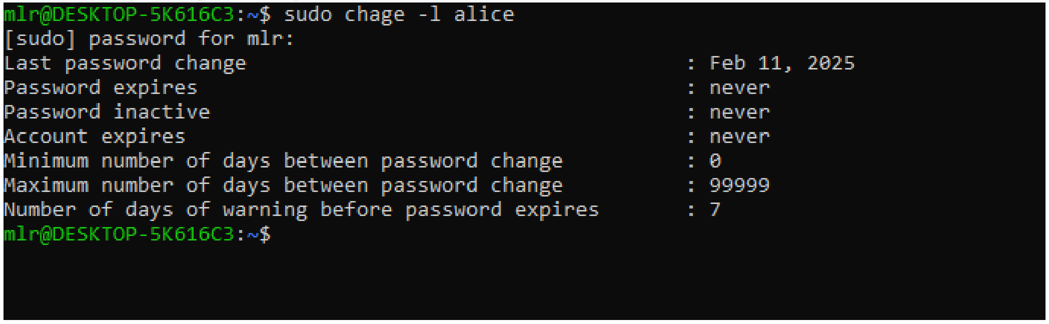


```
mlr@DESKTOP-5K616C3:~$ ls -l /usr
total 60
drwxr-xr-x   2 root alice  4096 Feb 11 03:10 alice
drwxr-xr-x   2 root root  20480 Feb 10 10:00 bin
drwxr-xr-x   2 root root   4096 Apr 22  2024 games
drwxr-xr-x   4 root root   4096 Jan  6 20:15 include
drwxr-xr-x  58 root root   4096 Feb 10 10:00 lib
drwxr-xr-x   2 root root   4096 Jan  6 20:14 lib64
drwxr-xr-x   7 root root   4096 Jan  6 20:15 libexec
drwxr-xr-x  10 root root   4096 Jan  6 20:13 local
drwxr-xr-x   2 root root   4096 Feb 10 10:00 sbin
drwxr-xr-x 113 root root   4096 Feb 10 10:00 share
drwxr-xr-x   2 root root   4096 Apr 22  2024 src
mlr@DESKTOP-5K616C3:~$
```

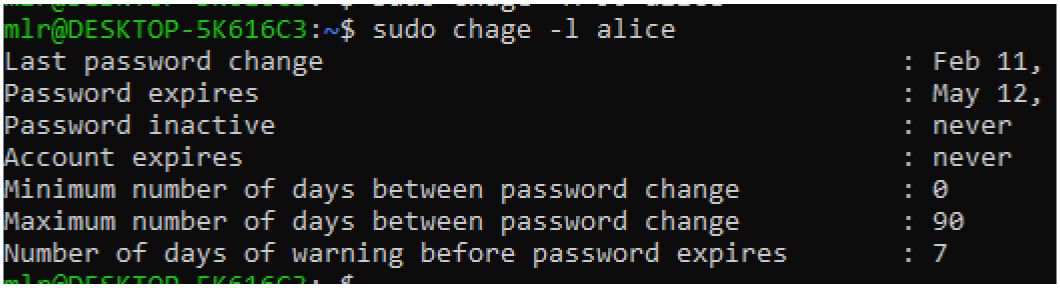
7. Password Aging and Expiry

Objective: Learn how to set password policies for users.

Q1 :- Set a password expiration period of 90 days for user alice using chage.

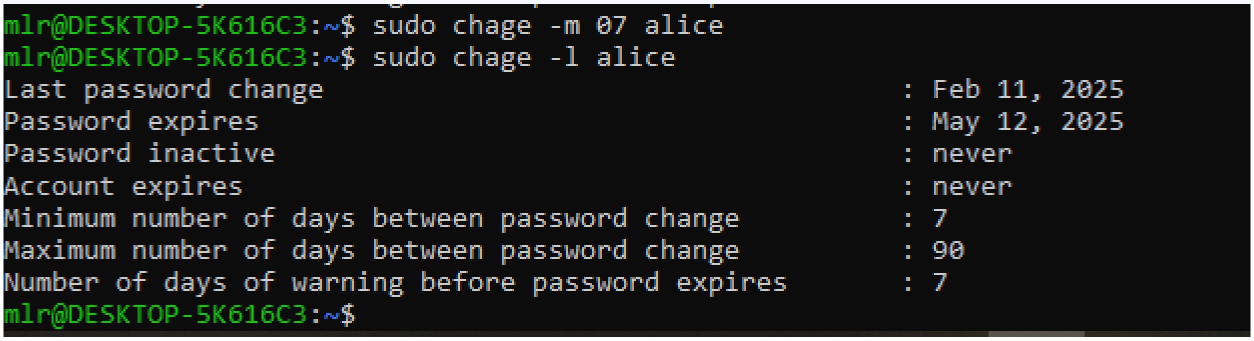


```
mlr@DESKTOP-5K616C3:~$ sudo chage -l alice
[sudo] password for mlr:
Last password change                                    : Feb 11, 2025
Password expires                                        : never
Password inactive                                       : never
Account expires                                         : never
Minimum number of days between password change          : 0
Maximum number of days between password change          : 99999
Number of days of warning before password expires       : 7
mlr@DESKTOP-5K616C3:~$
```



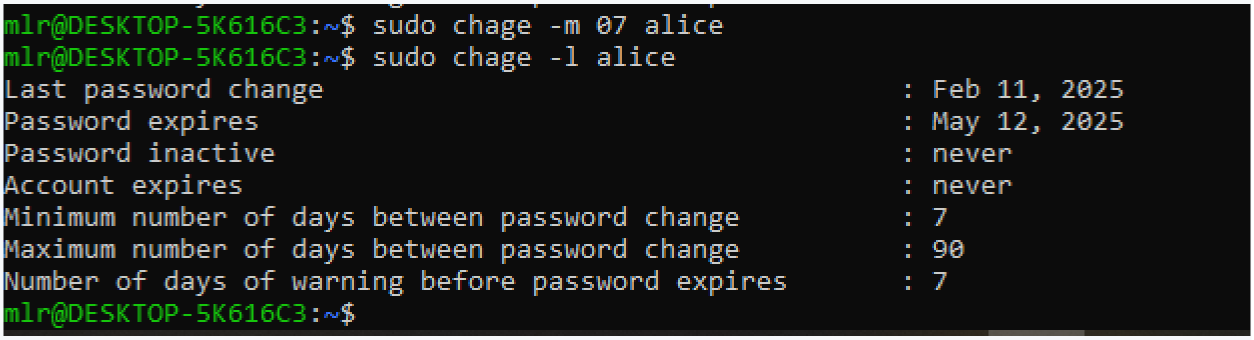
```
mlr@DESKTOP-5K616C3:~$ sudo chage -l alice
Last password change                                    : Feb 11,
Password expires                                        : May 12,
Password inactive                                       : never
Account expires                                         : never
Minimum number of days between password change          : 0
Maximum number of days between password change          : 90
Number of days of warning before password expires       : 7
```

Q2:- Set a warning period to notify the user 7 days before the password expires.



```
mlr@DESKTOP-5K616C3:~$ sudo chage -m 07 alice
mlr@DESKTOP-5K616C3:~$ sudo chage -l alice
Last password change                                    : Feb 11, 2025
Password expires                                        : May 12, 2025
Password inactive                                       : never
Account expires                                         : never
Minimum number of days between password change          : 7
Maximum number of days between password change          : 90
Number of days of warning before password expires       : 7
mlr@DESKTOP-5K616C3:~$
```

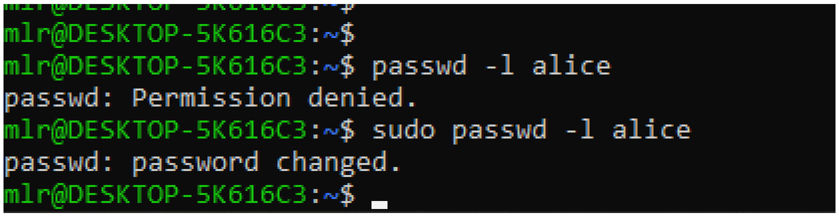
Q3:- Verify the changes using chage -l alice.



```
mlr@DESKTOP-5K616C3:~$ sudo chage -m 07 alice
mlr@DESKTOP-5K616C3:~$ sudo chage -l alice
Last password change                                    : Feb 11, 2025
Password expires                                        : May 12, 2025
Password inactive                                       : never
Account expires                                         : never
Minimum number of days between password change          : 7
Maximum number of days between password change          : 90
Number of days of warning before password expires       : 7
mlr@DESKTOP-5K616C3:~$
```

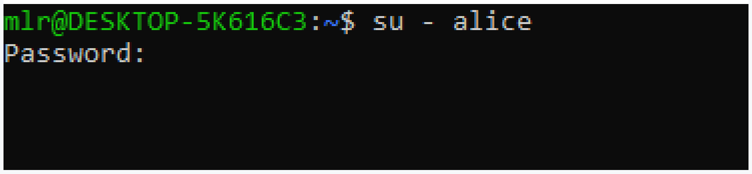
8. Lock and Unlock User Accounts

Q1:- Lock the user account alice by using the passwd -l command.



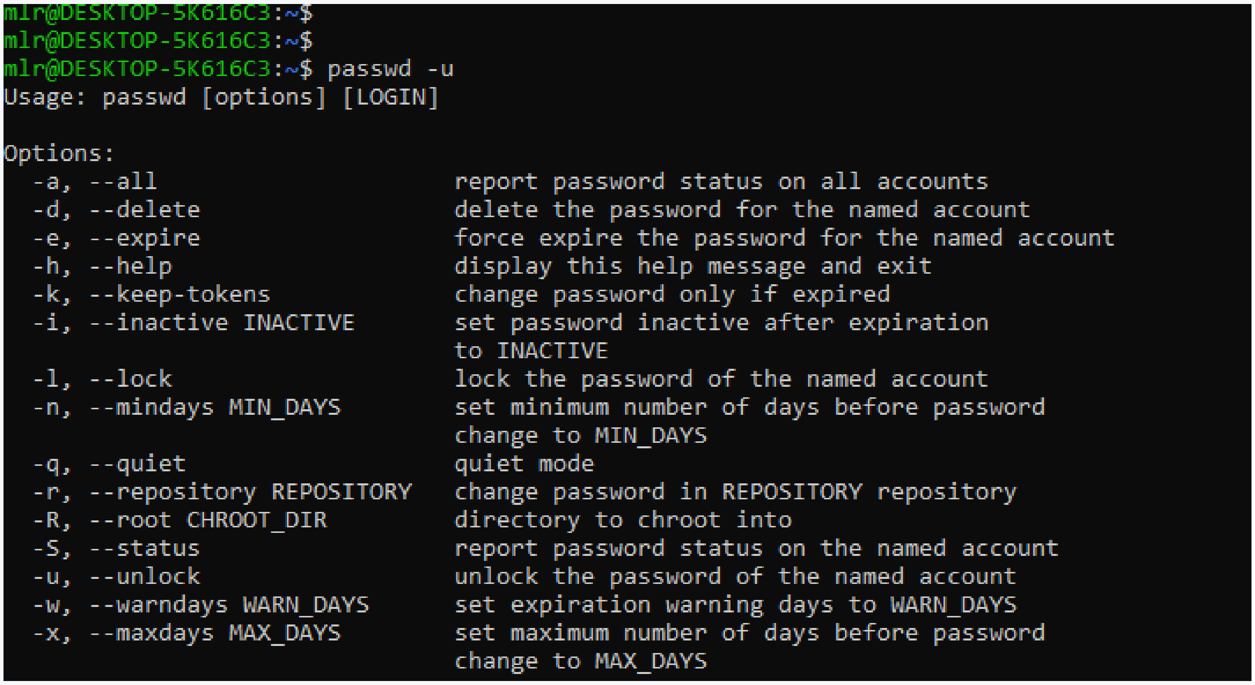
```
mlr@DESKTOP-5K616C3:~$
mlr@DESKTOP-5K616C3:~$ passwd -l alice
passwd: Permission denied.
mlr@DESKTOP-5K616C3:~$ sudo passwd -l alice
passwd: password changed.
mlr@DESKTOP-5K616C3:~$
```

Q2:- Verify that the account is locked by trying to log in as alice.



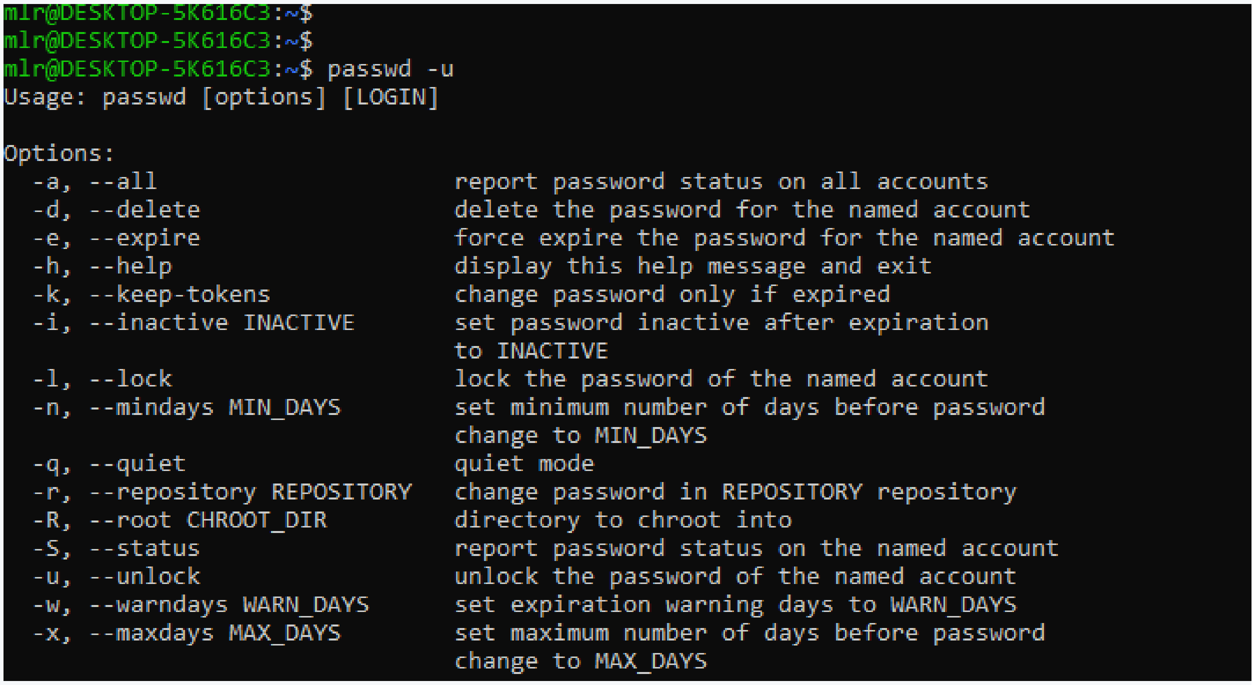
```
mlr@DESKTOP-5K616C3:~$ su - alice
Password:
```

Q3:- Unlock the account using the passwd -u command.



```
mlr@DESKTOP-5K616C3:~$
mlr@DESKTOP-5K616C3:~$
mlr@DESKTOP-5K616C3:~$ passwd -u
Usage: passwd [options] [LOGIN]

Options:
  -a, --all                     report password status on all accounts
  -d, --delete                  delete the password for the named account
  -e, --expire                  force expire the password for the named account
  -h, --help                    display this help message and exit
  -k, --keep-tokens             change password only if expired
  -i, --inactive INACTIVE       set password inactive after expiration
                                to INACTIVE
  -l, --lock                    lock the password of the named account
  -n, --mindays MIN_DAYS        set minimum number of days before password
                                change to MIN_DAYS
  -q, --quiet                   quiet mode
  -r, --repository REPOSITORY   change password in REPOSITORY repository
  -R, --root CHROOT_DIR         directory to chroot into
  -S, --status                  report password status on the named account
  -u, --unlock                  unlock the password of the named account
  -w, --warndays WARN_DAYS      set expiration warning days to WARN_DAYS
  -x, --maxdays MAX_DAYS        set maximum number of days before password
                                change to MAX_DAYS
```

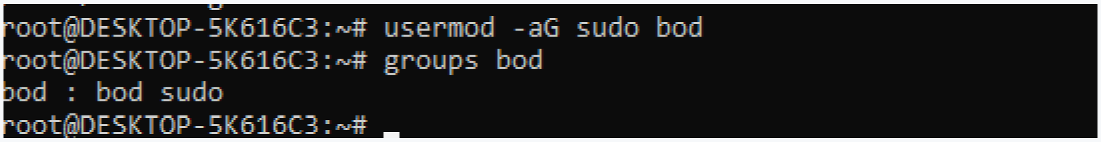
Q4:-Verify the account is unlocked by trying to log in again



```
mlr@DESKTOP-5K616C3:~$
mlr@DESKTOP-5K616C3:~$
mlr@DESKTOP-5K616C3:~$ passwd -u
Usage: passwd [options] [LOGIN]

Options:
  -a, --all                     report password status on all accounts
  -d, --delete                  delete the password for the named account
  -e, --expire                  force expire the password for the named account
  -h, --help                    display this help message and exit
  -k, --keep-tokens             change password only if expired
  -i, --inactive INACTIVE       set password inactive after expiration
                                to INACTIVE
  -l, --lock                    lock the password of the named account
  -n, --mindays MIN_DAYS        set minimum number of days before password
                                change to MIN_DAYS
  -q, --quiet                   quiet mode
  -r, --repository REPOSITORY   change password in REPOSITORY repository
  -R, --root CHROOT_DIR         directory to chroot into
  -S, --status                  report password status on the named account
  -u, --unlock                  unlock the password of the named account
  -w, --warndays WARN_DAYS      set expiration warning days to WARN_DAYS
  -x, --maxdays MAX_DAYS        set maximum number of days before password
                                change to MAX_DAYS
```

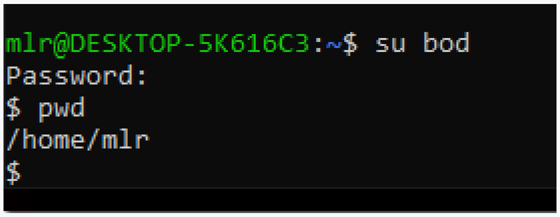
9. Create and Manage Sudo Access

Q1:- Add a user bob to allowing bob to the sudo group, execute commands as root.



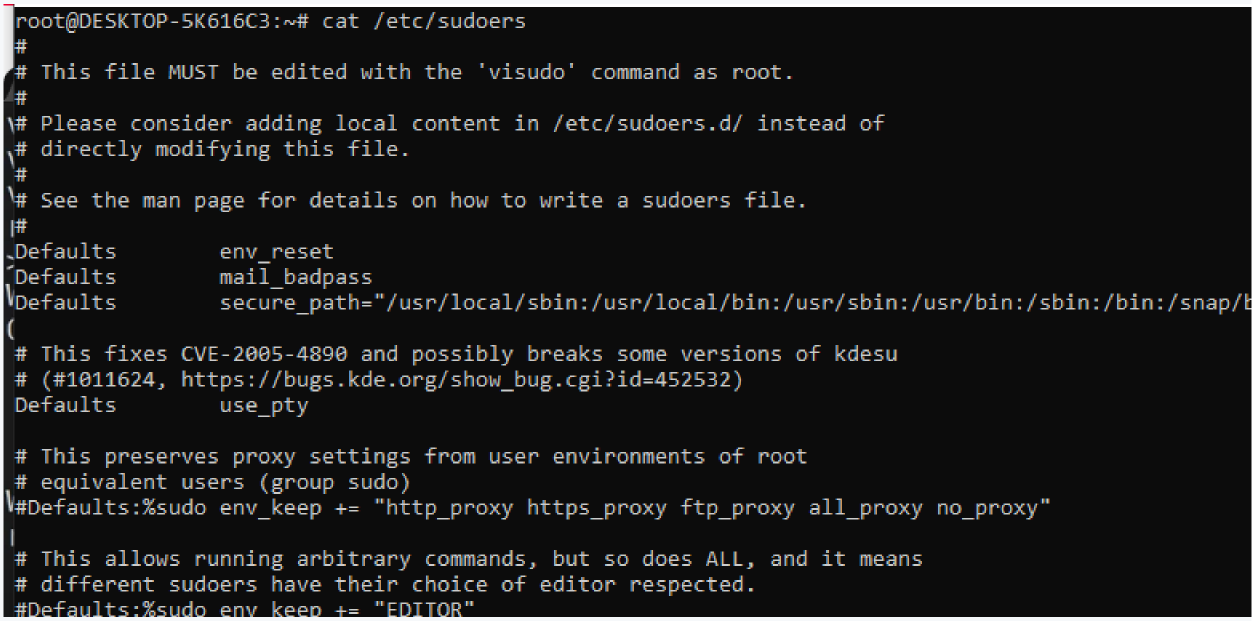
```
root@DESKTOP-5K616C3:~# usermod -aG sudo bod
root@DESKTOP-5K616C3:~# groups bod
bod : bod sudo
root@DESKTOP-5K616C3:~#
```

Q2:- Test by logging in as bob and running a command with sudo.



```
mlr@DESKTOP-5K616C3:~$ su bod
Password:
$ pwd
/home/mlr
$
```

Q3:- Optionally, restrict bob’s sudo access by editing the /etc/sudoers file using visudo (e.g., allow only apt-get commands).

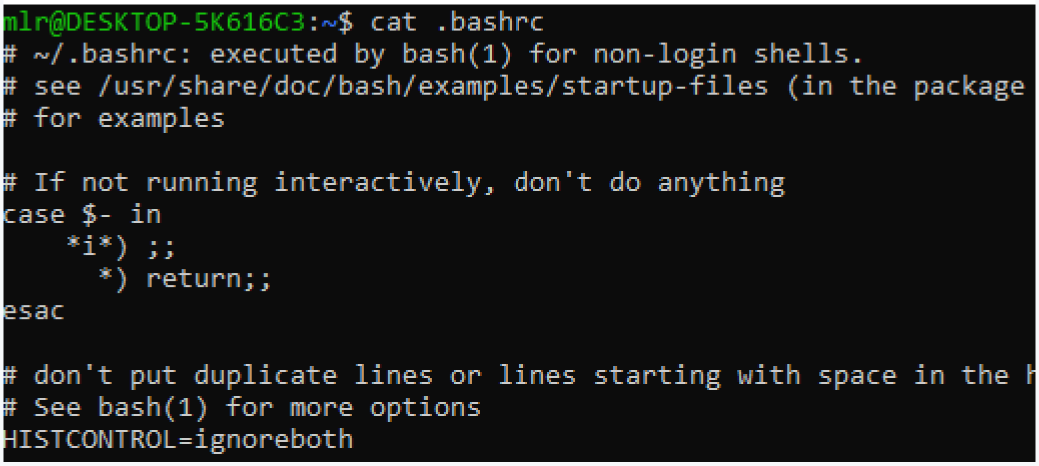


```
root@DESKTOP-5K616C3:~# cat /etc/sudoers
#
# This file MUST be edited with the 'visudo' command as root.
#
# Please consider adding local content in /etc/sudoers.d/ instead of
# directly modifying this file.
#
# See the man page for details on how to write a sudoers file.
#
Defaults        env_reset
Defaults        mail_badpass
Defaults        secure_path="/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin:/snap/b

# This fixes CVE-2005-4890 and possibly breaks some versions of kdesu
# (#1011624, https://bugs.kde.org/show_bug.cgi?id=452532)
Defaults        use_pty

# This preserves proxy settings from user environments of root
# equivalent users (group sudo)
#Defaults:%sudo env_keep += "http_proxy https_proxy ftp_proxy all_proxy no_proxy"

# This allows running arbitrary commands, but so does ALL, and it means
# different sudoers have their choice of editor respected.
#Defaults:%sudo env_keep += "EDITOR"
```

10. Set Up User Environment Variables

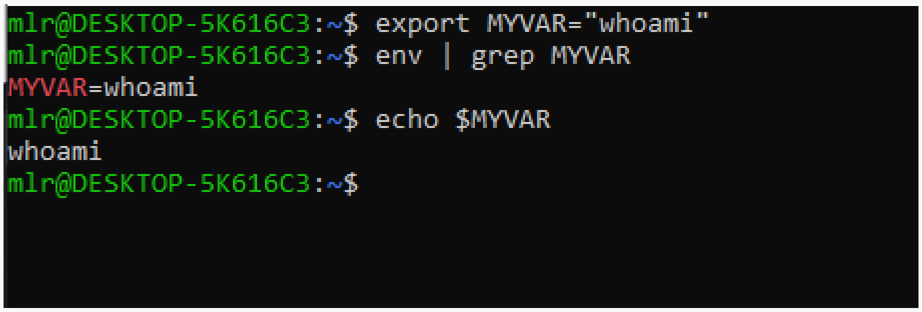
Q1:-Modify the .bashrc file for a user (alice) to set a custom environment variable (e.g., MYVAR=HelloWorld).

Cat .bashrc



```
mlr@DESKTOP-5K616C3:~$ cat .bashrc
# ~/.bashrc: executed by bash(1) for non-login shells.
# see /usr/share/doc/bash/examples/startup-files (in the package
# for examples

# If not running interactively, don't do anything
case $- in
    *i*) ;;
      *) return;;
esac

# don't put duplicate lines or lines starting with space in the h
# See bash(1) for more options
HISTCONTROL=ignoreboth
```

Q2:- Have the user log out and log back in, then check the environment variable using echo $MYVAR.



```
mlr@DESKTOP-5K616C3:~$ export MYVAR="whoami"
mlr@DESKTOP-5K616C3:~$ env | grep MYVAR
MYVAR=whoami
mlr@DESKTOP-5K616C3:~$ echo $MYVAR
whoami
mlr@DESKTOP-5K616C3:~$
```

11. Create and Manage User Quotas

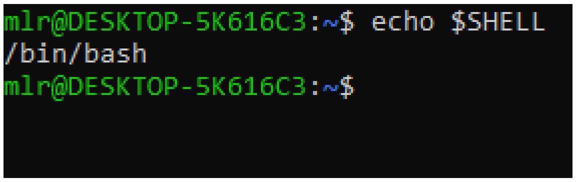
Q1:- Enable disk quotas on a specific file system (/home).

```
Command (m for help): m

Help:

  DOS (MBR)
   a   toggle a bootable flag
   b   edit nested BSD disklabel
   c   toggle the dos compatibility flag

  Generic
   d   delete a partition
   F   list free unpartitioned space
   l   list known partition types
   n   add a new partition
   p   print the partition table
   t   change a partition type
   v   verify the partition table
   i   print information about a partition

  Misc
   m   print this menu
   u   change display/entry units
   x   extra functionality (experts only)

  Script
   I   load disk layout from sfdisk script file
   O   dump disk layout to sfdisk script file

  Save & Exit
   w   write table to disk and exit
   q   quit without saving changes

  Create a new label
   g   create a new empty GPT partition table
   G   create a new empty SGI (IRIX) partition table
   o   create a new empty MBR (DOS) partition table
   s   create a new empty Sun partition table
```

```
Quation Pneding

1.     Enable disk quotas on a specific file system (/home).
2.     Set a soft and hard limit for user alice (e.g., 1 GB for soft, 1.5
GB for hard).
3.     Test the quota by attempting to exceed the disk usage limit.
4.     Verify the user’s quota using the quota command.
```

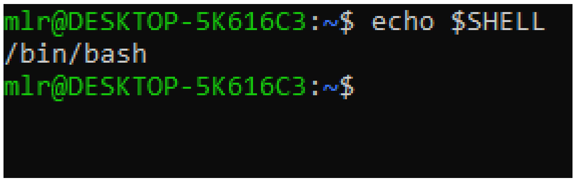
12. Configure User Shells

Q1:- `Create a user eve and set their default shell to /bin/zsh using usermod -s /bin/zsh.`

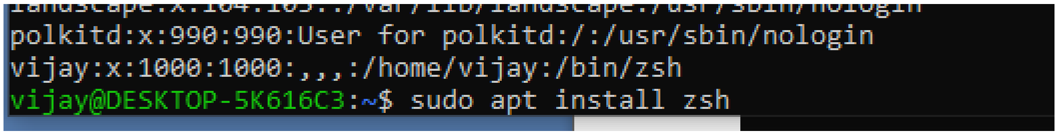


```
mlr@DESKTOP-5K616C3:~$ echo $SHELL
/bin/bash
mlr@DESKTOP-5K616C3:~$
```

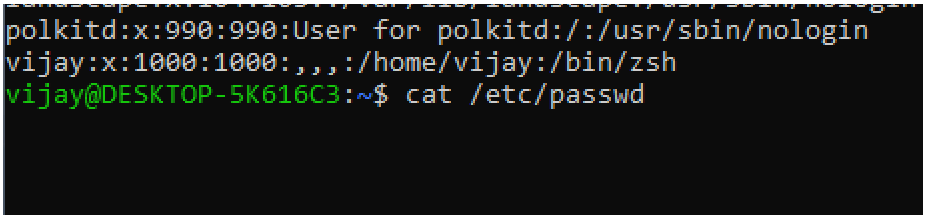
Q2 : - `Verify that eve's default shell is set to Zsh by checking /etc/passwd.`



```
mlr@DESKTOP-5K616C3:~$ echo $SHELL
/bin/bash
mlr@DESKTOP-5K616C3:~$
```



```
landscape:x:104:103::/var/lib/landscape:/usr/sbin/nologin
polkitd:x:990:990:User for polkitd:/:/usr/sbin/nologin
vijay:x:1000:1000:,,,:/home/vijay:/bin/zsh
vijay@DESKTOP-5K616C3:~$ sudo apt install zsh
```

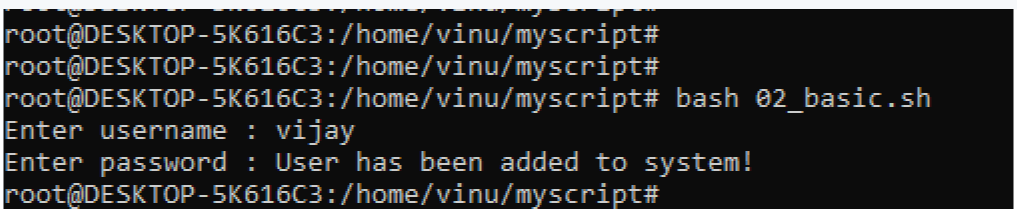


```
landscape:x:104:103::/var/lib/landscape:/usr/sbin/nologin
polkitd:x:990:990:User for polkitd:/:/usr/sbin/nologin
vijay:x:1000:1000:,,,:/home/vijay:/bin/zsh
vijay@DESKTOP-5K616C3:~$ cat /etc/passwd
```

13. Automate User Creation with a Script

Q1 :- Write a Bash script that takes a username and a group as input.
Syntax:-

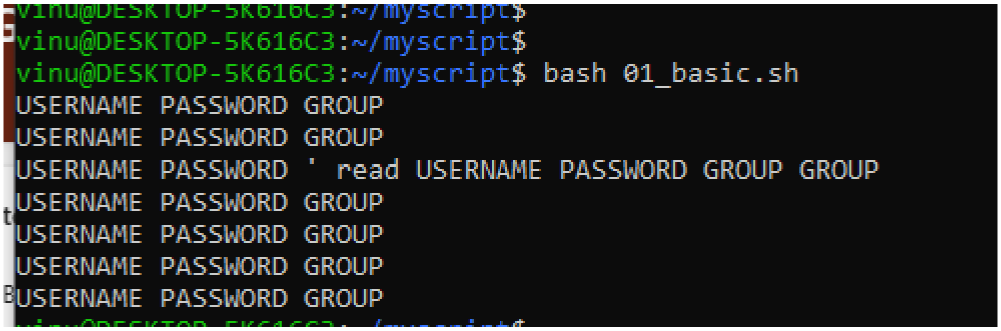
```
if [ $(id -u) -eq 0 ]; then
		read -p "Enter username : " username
		read -s -p "Enter password : " password
		egrep "^$username" /etc/passwd >/dev/null
		if [ $? -eq 0 ]; then
				echo "$username exists!"
				exit 1
		else
				pass=$(perl -e 'print crypt($ARGV[0], "password")' $password)
				useradd -m -p "$pass" "$username"
				[ $? -eq 0 ] && echo "User has been added to system!" || echo "Failed to add a 
user!"
		fi
else
		echo "Only root may add a user to the system."
		exit 2
fi
```



```
root@DESKTOP-5K616C3:/home/vinu/myscript#
root@DESKTOP-5K616C3:/home/vinu/myscript#
root@DESKTOP-5K616C3:/home/vinu/myscript#
root@DESKTOP-5K616C3:/home/vinu/myscript# bash 02_basic.sh
Enter username : vijay
Enter password : User has been added to system!
root@DESKTOP-5K616C3:/home/vinu/myscript#
```

Q2 :- Create the user, create the group if it does not exist, and add the user to the group.

```
#!/bin/bash

FILENAME="01_basic.sh"

while IFS=':' read USERNAME PASSWORD GROUP

do

echo "USERNAME" $USERMANE "PASSWORD" $PASSWORD "GROUP" $GROUP

done<"$FILENAME"
```



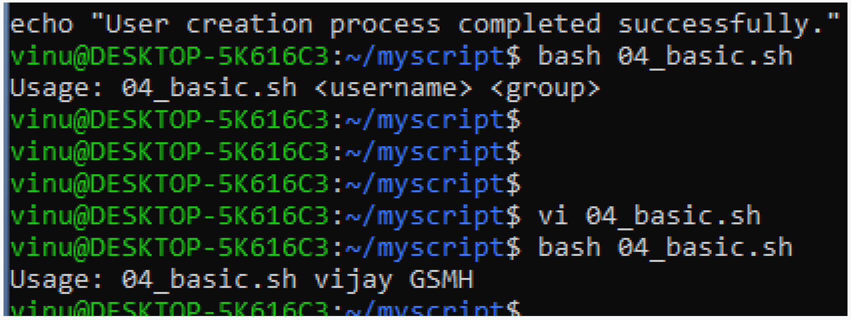
```
vinu@DESKTOP-5K616C3:~/myscript$
vinu@DESKTOP-5K616C3:~/myscript$
vinu@DESKTOP-5K616C3:~/myscript$ bash 01_basic.sh
USERNAME PASSWORD GROUP
USERNAME PASSWORD GROUP
USERNAME PASSWORD ' read USERNAME PASSWORD GROUP GROUP
USERNAME PASSWORD GROUP
USERNAME PASSWORD GROUP
USERNAME PASSWORD GROUP
USERNAME PASSWORD GROUP
vinu@DESKTOP-5K616C3:~/myscript$
```

Q3:- Set a default password for the new user and notify the administrator by email.

ANS:-

```
      #!/bin/bash
          echo "Usage: $0 vijay GSMH"
            exit 1
fi

# Accept username and group as arguments
vijay=$1
GSMH=$2

# Check if group exists, if not create it
if ! grep -q "^$GSMH:" /etc/group; then
          echo "Group '$GROUP' does not exist. Creating the
group..."
            sudo groupadd $GGSMH
    else
              echo "Group '$GSMH' already exists."
fi

# Create user and add them to the group
echo "Creating user '$vijay' and adding to group '$GSMH'..."
sudo useradd -m -g $GROUP -s /bin/bash $vijay

# Set a default password (in this case 'password123')
echo "Setting password for user '$vijay'..."
echo "$vijay:password123" | sudo chpasswd

# Notify administrator by email (using mail command)
ADMIN_EMAIL="vijaykumar5@gmail.com"
```

```
echo "User '$vijay' created and added to group '$GSMH'." | mail
-s "New User Created: $vijay1" $ADMIN_EMAIL

echo "User creation process completed successfully."
```



```
echo "User creation process completed successfully."
vinu@DESKTOP-5K616C3:~/myscript$ bash 04_basic.sh
Usage: 04_basic.sh <username> <group>
vinu@DESKTOP-5K616C3:~/myscript$
vinu@DESKTOP-5K616C3:~/myscript$
vinu@DESKTOP-5K616C3:~/myscript$
vinu@DESKTOP-5K616C3:~/myscript$ vi 04_basic.sh
vinu@DESKTOP-5K616C3:~/myscript$ bash 04_basic.sh
Usage: 04_basic.sh vijay GSMH
vinu@DESKTOP-5K616C3:~/myscript$
```

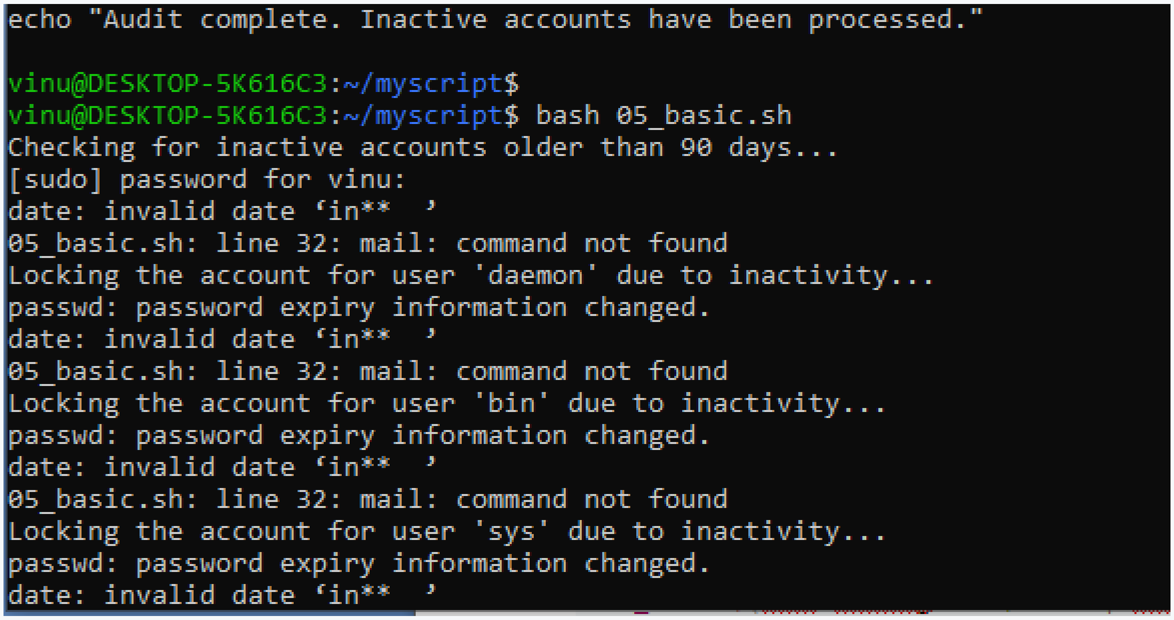
14. User Account Audit

Q1:-Write a script to list all users who have not logged in for the past 90 days.

ANS

```
      #!/bin/bash

# Define the number of days (90 days)
DAYS_INACTIVE=90

# Define the admin email for alert
ADMIN_EMAIL="vinaykumar5367@gmail.com"

# Get today's date in seconds since Unix epoch
CURRENT_DATE=$(date +%s)

# Find users who haven't logged in for more than 90 days
echo "Checking for inactive accounts older than $DAYS_INACTIVE days..."

# Loop through each user in the /etc/passwd file
awk -F: '{ print $1, $3 }' /etc/passwd | while read user uid; do
# Get the last login date of the user (in seconds)
LAST_LOGIN=$(sudo lastlog -u "$user" | awk 'NR==2 {print $4 " " $5 " "
$6}')

# If the user has never logged in, lastlog will return "Never logged in"
if [[ "$LAST_LOGIN" == "Never logged in" ]]; then
LAST_LOGIN_DATE=$CURRENT_DATE
else
LAST_LOGIN_DATE=$(date -d"$LAST_LOGIN" +%s)
fi

# Calculate the difference between the current date and the last login
date
INACTIVE_DAYS=$(( (CURRENT_DATE - LAST_LOGIN_DATE) / 86400 ))

if [ "$INACTIVE_DAYS" -ge "$DAYS_INACTIVE" ]; then
# Optionally send an email alert for the inactive user
echo "User '$user' has been inactive for $INACTIVE_DAYS days." | mail -s
"Inactive Account Alert: $user" $ADMIN_EMAIL

# Lock the user account to disable login
echo "Locking the account for user '$user' due to inactivity..."
sudo passwd -l $user
fi
```

```
done

echo "Audit complete. Inactive accounts have been processed."
```



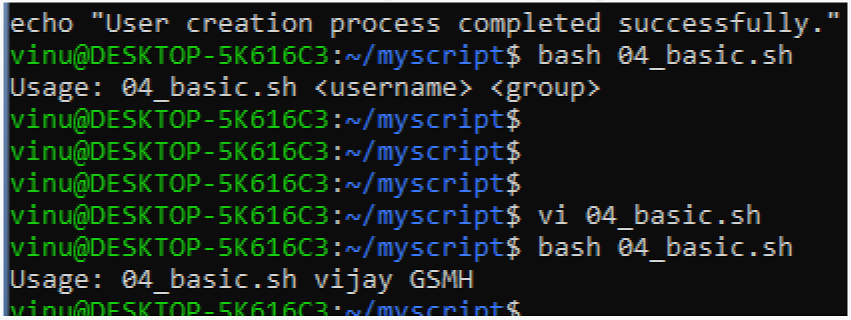
```
echo "Audit complete. Inactive accounts have been processed."

vinu@DESKTOP-5K616C3:~/myscript$
vinu@DESKTOP-5K616C3:~/myscript$ bash 05_basic.sh
Checking for inactive accounts older than 90 days...
[sudo] password for vinu:
date: invalid date ‘in**  ’
05_basic.sh: line 32: mail: command not found
Locking the account for user 'daemon' due to inactivity...
passwd: password expiry information changed.
date: invalid date ‘in**  ’
05_basic.sh: line 32: mail: command not found
Locking the account for user 'bin' due to inactivity...
passwd: password expiry information changed.
date: invalid date ‘in**  ’
05_basic.sh: line 32: mail: command not found
Locking the account for user 'sys' due to inactivity...
passwd: password expiry information changed.
date: invalid date ‘in**  ’
```

Q2:-Optionally, send an email alert for these inactive accounts.

ANS

```
      #!/bin/bash
          echo "Usage: $0 vijay GSMH"
            exit 1
fi

# Accept username and group as arguments
vijay=$1
GSMH=$2

# Check if group exists, if not create it
if ! grep -q "^$GSMH:" /etc/group; then
          echo "Group '$GROUP' does not exist. Creating the
group..."
            sudo groupadd $GGSMH
    else
              echo "Group '$GSMH' already exists."
fi
```

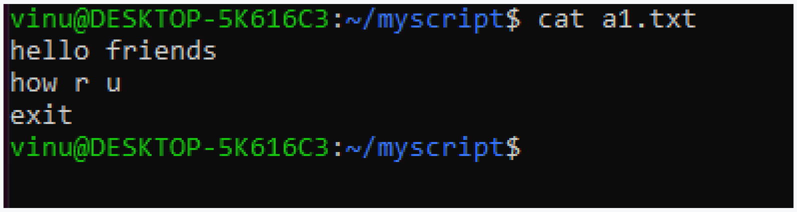
```
# Create user and add them to the group
echo "Creating user '$vijay' and adding to group '$GSMH'..."
sudo useradd -m -g $GROUP -s /bin/bash $vijay

# Set a default password (in this case 'password123')
echo "Setting password for user '$vijay'..."
echo "$vijay:password123" | sudo chpasswd

# Notify administrator by email (using mail command)
ADMIN_EMAIL="vijaykumar5@gmail.com"
echo "User '$vijay' created and added to group '$GSMH'." | mail
-s "New User Created: $vijay1" $ADMIN_EMAIL

echo "User creation process completed successfully."
```



```
echo "User creation process completed successfully."
vinu@DESKTOP-5K616C3:~/myscript$ bash 04_basic.sh
Usage: 04_basic.sh <username> <group>
vinu@DESKTOP-5K616C3:~/myscript$
vinu@DESKTOP-5K616C3:~/myscript$
vinu@DESKTOP-5K616C3:~/myscript$
vinu@DESKTOP-5K616C3:~/myscript$ vi 04_basic.sh
vinu@DESKTOP-5K616C3:~/myscript$ bash 04_basic.sh
Usage: 04_basic.sh vijay GSMH
vinu@DESKTOP-5K616C3:~/myscript$
```

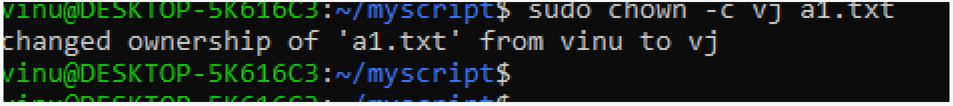
15. Check and Modify User File Permissions

Q1 :- Create a file /home/alice/important_file.txt.



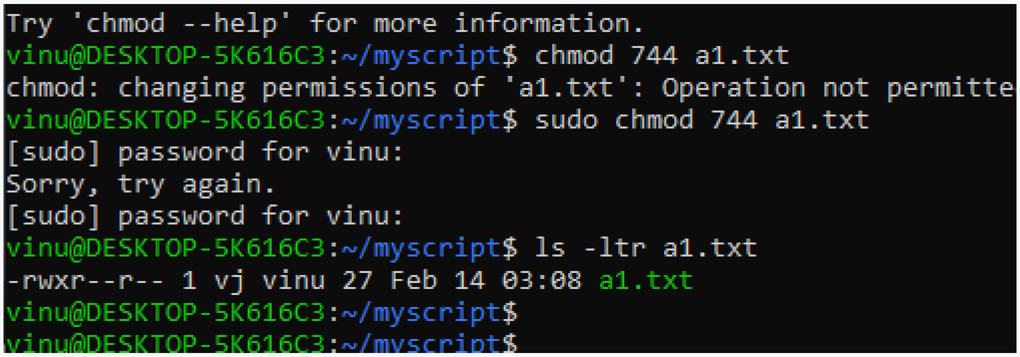
```
vinu@DESKTOP-5K616C3:~/myscript$ cat a1.txt
hello friends
how r u
exit
vinu@DESKTOP-5K616C3:~/myscript$
```

Q2 :- Change the ownership of the file to the user alice using chown.



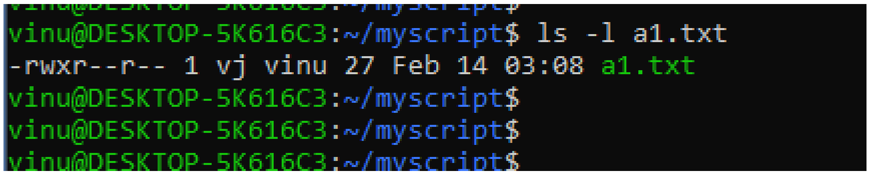
```
vinu@DESKTOP-5K616C3:~/myscript$ sudo chown -c vj a1.txt
hanged ownership of 'a1.txt' from vinu to vj
inu@DESKTOP-5K616C3:~/myscript$
vinu@DESKTOP-5K616C3:~/myscript$
```

Q3:- Set the file permissions so that only alice has read and write access, while others have no access.



```
Try 'chmod --help' for more information.
vinu@DESKTOP-5K616C3:~/myscript$ chmod 744 a1.txt
chmod: changing permissions of 'a1.txt': Operation not permitte
vinu@DESKTOP-5K616C3:~/myscript$ sudo chmod 744 a1.txt
[sudo] password for vinu:
Sorry, try again.
[sudo] password for vinu:
vinu@DESKTOP-5K616C3:~/myscript$ ls -ltr a1.txt
-rwxr--r-- 1 vj vinu 27 Feb 14 03:08 a1.txt
vinu@DESKTOP-5K616C3:~/myscript$
vinu@DESKTOP-5K616C3:~/myscript$
```

Q4:- Verify the permissions using ls -l.



```
vinu@DESKTOP-5K616C3:~/myscript$
vinu@DESKTOP-5K616C3:~/myscript$ ls -l a1.txt
-rwxr--r-- 1 vj vinu 27 Feb 14 03:08 a1.txt
vinu@DESKTOP-5K616C3:~/myscript$
vinu@DESKTOP-5K616C3:~/myscript$
vinu@DESKTOP-5K616C3:~/myscript$
```