Ahmed Hossam

Part A :

1. Create three new users using `adduser` command:

```
################################
# Actions here done by root user
################################

# alice (interactive creation with full prompts)
adduser alice

#bob (with custom home directory `/opt/users/bob`)
useradd bob --home /opt/users/bob

# charlie (system user for services)
useradd --system charlie
```

2. Set passwords for alice and bob:

```
################################
# Actions here done by root user
################################

# Set passwords for alice
passwd alice

# Set passwords for bob
passwd bob

# Force alice to change password at next login
passwd -e alice
```

3. View user information:

- Display alice's entry from `/etc/passwd`

```
[root@localhost /]# cat etc/passwd | grep "alice"
alice:x:1000:1000::/home/alice:/bin/bash
[root@localhost /]#
```

- Show bob's password aging information using `chage -l bob`

```
root@localhost:/
[root@localhost /]# chage -l bob
Last password change                                    : Oct 14, 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
[root@localhost /]#
[root@localhost /]#
[root@localhost /]#
[root@localhost /]# chage -l alice
Last password change                                    : password must be changed
Password expires                                        : password must be changed
Password inactive                                       : password must be changed
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
[root@localhost /]#
[root@localhost /]#
```

Part B: Group Management

4. Create two groups:
   - `developers` (regular group)
   - `project-team` (using addgroup command)

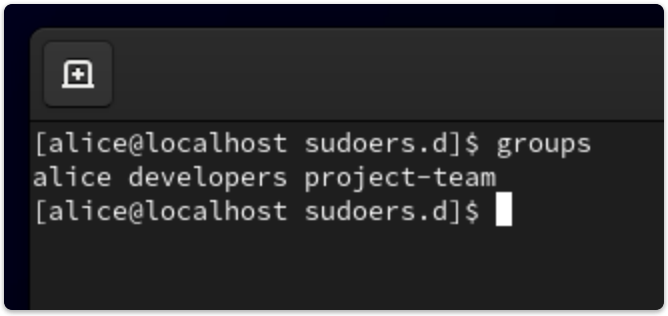
```
########################################################################
# Actions done by alice after adding his account to sudo group
########################################################################

# developers (regular group)
# didn't get it, anyway created with add group :D
# used groupadd as i'm using RedHat 9 ^^
sudo groupadd developers

# project-team (using addgroup command)
```

```
sudo groupadd project-team
```

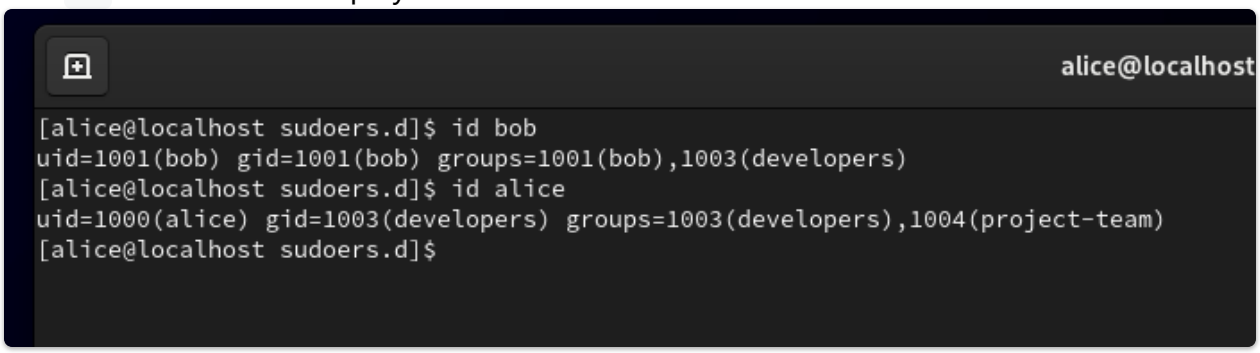
5. Add users to groups:
   - Add alice to both `developers` and `project-team`
   - Add bob to `developers` only
   - Make `developers` alice's primary group

```
# Add alice to both developers and project-team
usermod -aG developers alice
usermod -aG project-team alice

# Add bob to developers only
usermod -aG developers bob

# Make developers alice's primary group
sudo usermod -g developers alice
```

6. Verify group memberships:
   - Use `groups` command to show alice's groups



   - Use `id` command to display bob's UID and GID information



Part C: File Permissions and Ownership (25 points)

7. Create test files and directories:

```
# Create directory /tmp/lab1_test
mkdir lab1_test

# Create file /tmp/lab1_test/data.txt with some content
cat > data.txt
 Hello from data file
 ^C

 # Create file /tmp/lab1_test/script.sh
 touch script.sh
```

8. Set permissions using symbolic notation:

```
# Set data.txt permissions to rw-r--r-- (owner: read/write, group: read, others: read)
sudo chmod 644 data.txt

# Set script.sh permissions to rwxr-xr-x (owner: full access, group/others: read/execute)
sudo chmod 755 script.sh
```

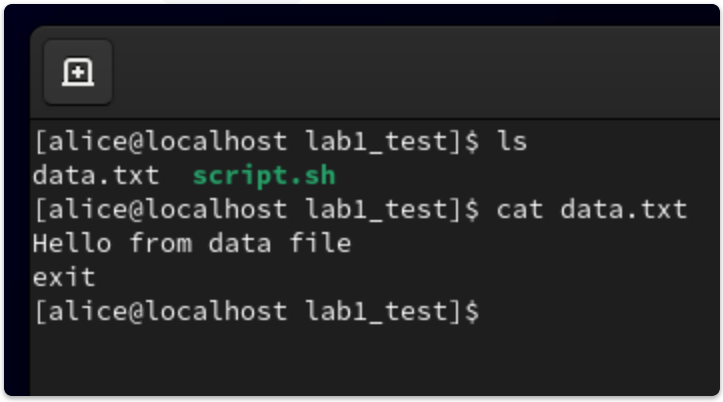
9. Change ownership:

```
# Change owner of data.txt to alice and group to developer
sudo chown alice:developers data.txt

# Change owner of entire /tmp/lab1_test directory to bob and group to project-team (recursive)
sudo chown -R bob:project-team  lab1_test
```

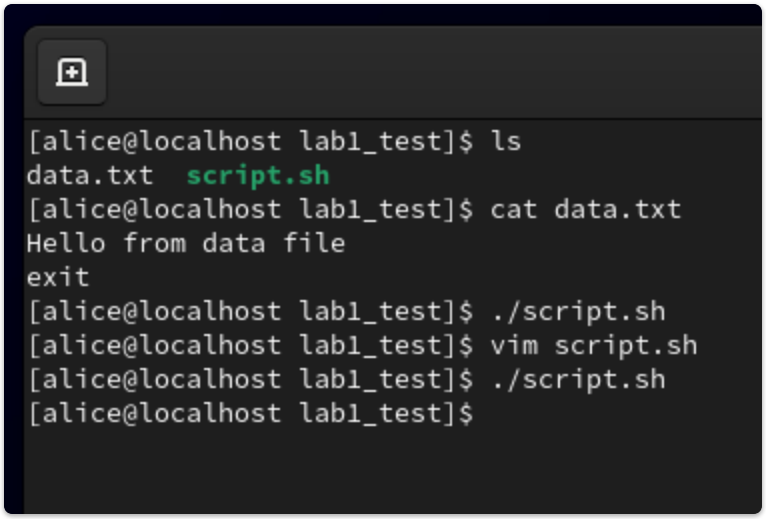
Part D: Verification and Testing

10. Test permissions:

- Try to read `data.txt` (data us readable with no issues)

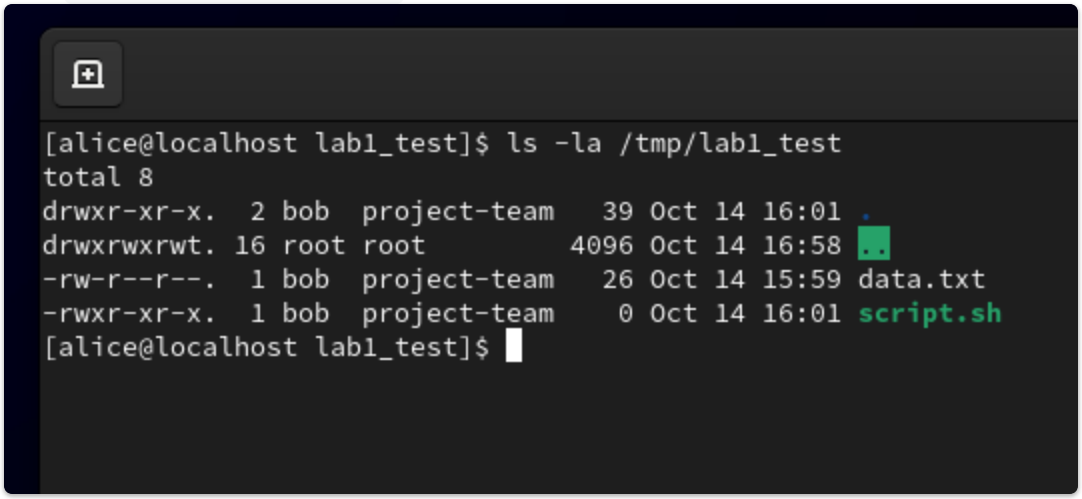


- Try to execute `script.sh` (fie is excitable but we can't add content inside because no write permission given to alice)



11. Display final state:

- Use `ls -la /tmp/lab1_test` to show all permissions and ownership



- Show group memberships for all created users