**Shell task 08/06/2023**.

1. chmod +x switch\_user.sh
2. chmod +x print\_username.sh
3. #!/bin/bash

# Get the username of the current user

current\_user=$(whoami)

# Get the groups the current user is a part of

groups=$(groups $current\_user)

# Print the group names

echo "Groups for the current user:"

echo $groups

3.#!/bin/bash

# Set the file path

file="hello"

# Set the new owner

new\_owner="betty"

# Change the owner of the file

chown $new\_owner $file

# Print the confirmation message

echo "Owner of the file $file changed to $new\_owner."

4.#!/bin/bash

touch hello

5. #!/bin/bash

chmod u+x hello

6. #!/bin/bash

file="hello"

# Check if the file exists

if [ -e "$file" ]; then

# Add execute permission to the owner and the group owner

chmod u+x,g+x "$file"

# Add read permission to other users

chmod o+r "$file"

echo "Permissions updated for $file"

else

echo "File $file does not exist in the current working directory."

fi

7. #!/bin/bash

file="hello"

# Check if the file exists

if [ -e "$file" ]; then

# Add execute permission to the owner, group owner, and other users

chmod u+x "$file"

chmod g+x "$file"

chmod o+x "$file"

echo "Execution permission added for $file"

else

echo "File $file does not exist in the current working directory."

fi

8. #!/bin/bash

file="hello"

# Check if the file exists

if [ -e "$file" ]; then

# Set no permissions for the owner and the group owner

chmod u-rwx "$file"

chmod g-rwx "$file"

# Set all permissions for other users

chmod o+rwx "$file"

echo "Permissions set for $file"

else

echo "File $file does not exist in the current working directory."

fi

9. chmod 644 hello

10. #!/bin/bash

hello\_path="./hello"

olleh\_path="./olleh"

# Get the mode of the olleh file

olleh\_mode=$(stat -c "%a" "$olleh\_path")

# Set the mode of the hello file to match olleh's mode

chmod "$olleh\_mode" "$hello\_path"

echo "The mode of hello has been set to the same as olleh's mode: $olleh\_mode"

11. #!/bin/bash

# Get the current directory

current\_dir=$(pwd)

# Find all subdirectories recursively and set execute permission

find "$current\_dir" -type d -exec chmod +x {} +

# Print a confirmation message

echo "Execute permission added to all subdirectories in the current directory."

12. #!/bin/bash

# Define the directory name and permissions

dir\_name="my\_dir"

permissions="751"

# Create the directory with the specified permissions

mkdir -m $permissions $dir\_name

echo "Directory '$dir\_name' created with permissions $permissions in the working directory."

13. #!/bin/bash

# Define the file name and group owner

file\_name="hello"

group\_owner="school"

# Change the group owner of the file

chgrp $group\_owner $file\_name

echo "Group owner of '$file\_name' changed to '$group\_owner'."

14. #!/bin/bash

# Define the owner and group owner

owner="vincent"

group\_owner="staff"

# Change the owner and group owner for files and directories in the working directory

chown -R $owner:$group\_owner \*

echo "Owner set to '$owner' and group owner set to '$group\_owner' for all files and directories in the working directory."

15. #!/bin/bash

# Define the file name, owner, and group owner

file\_name="\_hello"

owner="vincent"

group\_owner="staff"

# Change the owner and group owner for the symbolic link

chown -h $owner:$group\_owner $file\_name

echo "Owner set to '$owner' and group owner set to '$group\_owner' for the symbolic link '$file\_name'."

16. #!/bin/bash

# Define the file name, current owner, and desired owner

file\_name="hello"

current\_owner="guillaume"

desired\_owner="betty"

# Check if the file is owned by the current owner

if [[ $(stat -c "%U" $file\_name) == $current\_owner ]]; then

# Change the owner of the file

chown $desired\_owner $file\_name

echo "Owner of '$file\_name' changed to '$desired\_owner'."

else

echo "The file '$file\_name' is not owned by '$current\_owner'. Owner not changed."

fi

17. #!/bin/bash

# Connect to the Star Wars ASCII art server

telnet towel.blinkenlights.nl