**OTHER EXTRA TASK ON SHELL.**

**Write a script that prints the absolute path name of the current working directory**.

current\_directory=$(pwd)

"Current working directory: $current\_directory"

**Write a script that changes the working directory to the user’s home directory.**

#!/bin/bash

# Get the user's home directory

home\_directory=$(eval echo ~)

# Change the working directory to the user's home directory

cd "$home\_directory"

# Print the new working directory

echo "Changed working directory to: $home\_directory"

**Display current directory contents in a long format**

#!/bin/bash

# Get the current directory

current\_directory=$(pwd)

# Display the contents of the current directory in long format

ls -l "$current\_directory"

**Display current directory contents, including hidden files (starting with .). Use the long format.**

#!/bin/bash

# Get the current directory

current\_directory=$(pwd)

# Display the contents of the current directory, including hidden files, in long format

ls -la "$current\_directory

**Display current directory contents.**

* Long format
* with user and group IDs displayed numerically
* And hidden files (starting with .)

**Create a script that creates a directory named my\_first\_directory in the /tmp/ directory.**

#!/bin/bash

# Specify the directory name

directory\_name="my\_first\_directory"

# Specify the parent directory

parent\_directory="/tmp/"

# Create the directory

mkdir "$parent\_directory/$directory\_name"

# Check if the directory was created successfully

if [ $? -eq 0 ]; then

"Directory '$directory\_name' created successfully in '$parent\_directory'"

else

"Failed to create directory '$directory\_name'"

fi

**Move the file betty from /tmp/ to /tmp/my\_first\_directory.**

#!/bin/bash

# Specify the directory name

directory\_name="my\_first\_directory"

# Specify the parent directory

parent\_directory="/tmp/"

# Create the directory

mkdir "$parent\_directory/$directory\_name"

# Check if the directory was created successfully

if [ $? -eq 0 ]; then

"Directory '$directory\_name' created successfully in '$parent\_directory'"

else

"Failed to create directory '$directory\_name'"

fi

Write a script that changes the working directory to the previous one.

#!/bin/bash

# Change the working directory to the previous one

cd -

# Print the new working directory

"Changed working directory to: $(pwd)"

**Write a script that lists all files (even ones with names beginning with a period character, which are normally hidden) in the current directory and the parent of the working directory and the /boot directory (in this order), in long format.**

#!/bin/bash

# Get the current directory

current\_directory=$(pwd)

# Get the parent directory of the working directory

parent\_directory=$(dirname "$current\_directory")

# List files in the current directory, parent directory, and /boot directory in long format

"Listing files in current directory ($current\_directory):"

ls -la "$current\_directory"

"Listing files in parent directory ($parent\_directory):"

ls -la "$parent\_directory"

"Listing files in /boot directory:"

ls -la /boot

**Write a script that prints the type of the file named iamafile. The file iamafile will be in the /tmp directory when we will run your script.**

#!/bin/bash

file\_path="/tmp/iamafile"

if [ -e "$file\_path" ]; then

file\_type=$(file -b --mime-type "$file\_path")

echo "The file type of 'iamafile' is: $file\_type"

else

echo "The file 'iamafile' does not exist in the /tmp directory."

fi

**Create a symbolic link to /bin/ls, named \_\_ls\_\_. The symbolic link should be created in the current working directory.**

ln -s /bin/ls \_\_ls\_\_

**Create a script that copies all the HTML files from the current working directory to the parent of the working directory, but only copy files that did not exist in the parent of the working directory or were newer than the versions in the parent of the working directory.**

#!/bin/bash

current\_dir=$(pwd)

parent\_dir=$(dirname "$current\_dir")

for file in \*.html; do

if [[ ! -e "$parent\_dir/$file" || "$file" -nt "$parent\_dir/$file" ]]; then

cp "$file" "$parent\_dir"

echo "Copied $file to $parent\_dir"

else

echo "$file already exists in $parent\_dir and is newer"

fi

done