Exercise: Shell scripts

1. Write a shell script to get the current date, time, username and current working directory.

nano myinfo.sh

#!/bin/bash

echo "Hello, $LOGNAME"

echo "Current date is `date`"

echo "User is `whoami`"

echo "Current directory `pwd`"

Then we press CTRL + S to save and CTRL + X to exit the nano editor.

Then we write a command to make the script executable and then run it.

Chmod +x myinfo.sh

./myinfo.sh

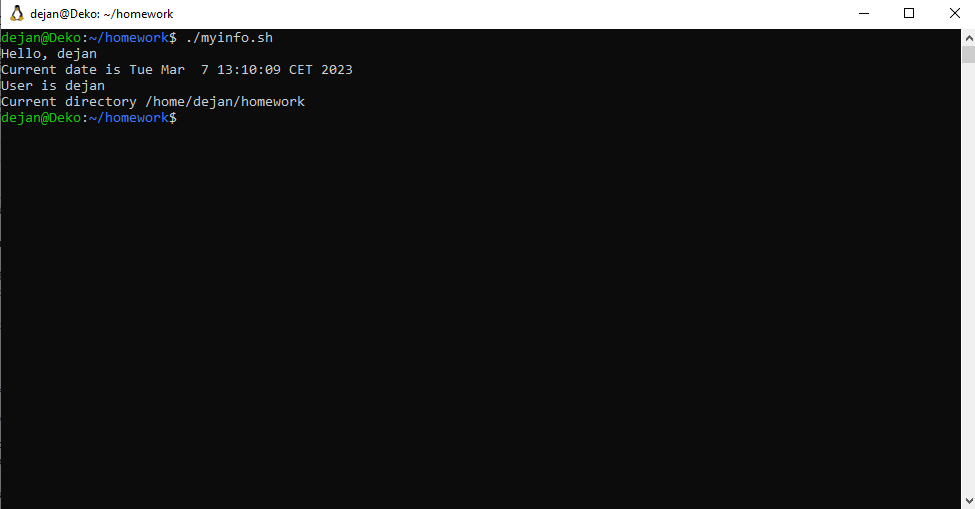
And here is the output:

Hello, dejan

Current date is Tue Mar 7 13:00:56 CET 2023

User is dejan

Current directory /home/dejan/homework



========================================================

1. Write a shell script that prints “I love learning about DevOps” on the screen. Message should be a variable.

nano IloveDevOps.sh

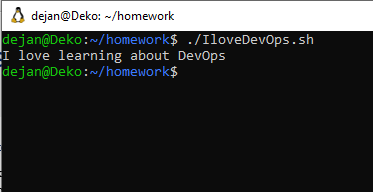
#!/bin/bash

love='I love learning about DevOps'

echo $love

chmod +x IloveDevOps.sh

./IloveDevOps.sh



========================================================

1. Write a shell script that displays “plan code build test release deploy” on the screen with each appearing on a separate line.

nano plan.sh

echo "plan"

echo "code"

echo "build"

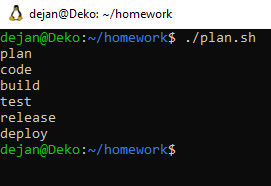
echo "test"

echo "release"

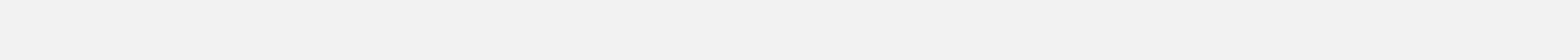
echo "deploy"

chmod +x plan.sh

./plan.sh



========================================================



69

Bulgaria Blvd., Infinity Tower B, fl. 8, Triaditza district, 1404 Sofia, Bulgaria | +359 2424 6484 |

sales@scalefocus.com

| www.scalefocus.com



1. Write a shell script that prompts the user for a name of a file or directory and reports if it is a

regular file, a directory, or another type of file. Also perform a ls command against the file or directory with the long listing option.

nano isfile.sh

#!/bin/bash

read -p "Enter a file or directory name: " file

if [ -f "$file" ]; then

echo "$file is a regular file"

elif [ -d "$file" ]; then

echo "$file is a directory"

else

echo "$file is another type of file"

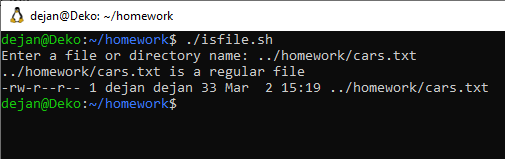
fi

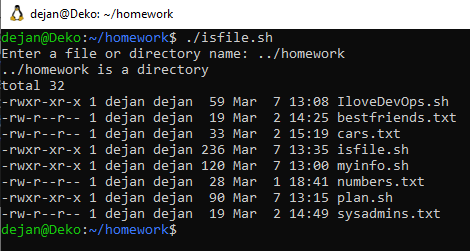
ls -l "$file"

chmod +x isfile.sh

./file.sh

And here are some outputs:





========================================================

1. Use arguments in a script. Total number of arguments should be three.

nano arg.sh

#!/bin/bash

if [ "$#" -ne 3 ]; then

echo "Error: Invalid number of arguments. Expected 3, got $#"

exit 1

fi

echo "The first argument is: $1"

echo "The second argument is: $2"

echo "The third argument is: $3"

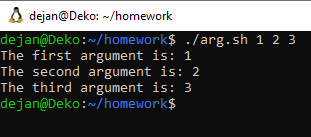
chmod +x arguments.sh

./arguments.sh

The first argument is: 1

The second argument is: 2

The third argument is: 3



========================================================

1. Write a script that till output your name out of a variable and will display the server uptime

nano servuptime.sh

name=”Dejan”

echo “Hello, $name!”

chmod +x servuptime.sh

./servuptime.sh

Hello, Dejan!

13:55:22 up 1:10, 0 users, load average: 0.00, 0.00, 0.00

