

# CompTIA Linux+

## Work with the Flow Control Constructs

- Introduction
- Lab Topology
- Exercise 1 - Work with the Flow Control Constructs
- Review

## Introduction

Welcome to the Work with the Flow Control Constructs Practice Lab. In this module you will be provided with the instructions and devices needed to develop your hands-on skills.

Flow Control Constructs

Loop

Linux System

## Learning Outcomes

In this module, you will complete the following exercise:

- Exercise 1 - Work with the Flow Control Constructs

After completing this lab, you will be able to:

- Use the if...else statement
- Use the while loop
- Use the until loop
- Use the for loop

## Exam Objectives

The following exam objectives are covered in this lab:

- LPI: 105.2 Customize or write simple scripts
- CompTIA: 5.1 Given a scenario, deploy and execute basic BASH scripts.

## **Exercise 1 - Work with the Flow Control Constructs**

A server can hold more than one hard drive. Due to an unforeseen error, there are changes that the server may crash, which can also impact the hard drive by making it unusable. The server administrator must regularly check the filesystem health, which is installed on the hard drive. CentOS provides several commands for this purpose.

In this exercise, you will learn to view the hard drive details.

### **Learning Outcomes**

After completing this exercise, you will be able to:

- Log into a Linux system
- Use if...else statements

### **Your Devices**

You will be using the following device in this lab. Please power this on now.

- PLABLINUX01 (CentOS Server)



### **Task 1 - Use if...else statements**

The if...else statements are mainly used for making a decision and therefore, are known as decision-making statements. When you have to make a choice from two given choices, the if...else statements work well. However, there may be a scenario in which you will have to work with multiple conditions to choose from, then how to use the case...esac statements.

In this task, you will learn to use if...else statements. To use the if...else statements, perform the following steps:

### ***Step 1***

On the desktop, right-click and select Open Terminal.

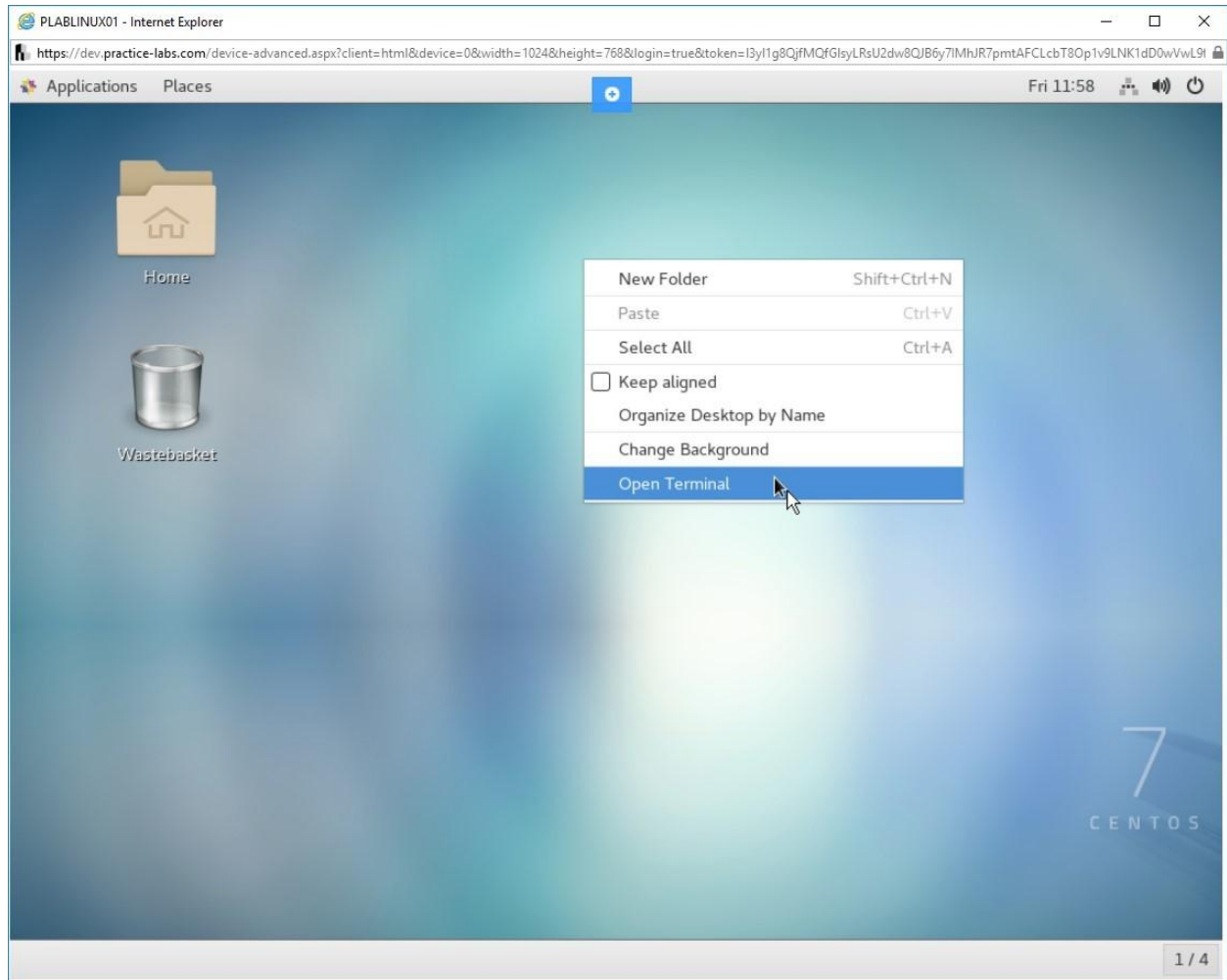


Figure 1.1 Screenshot of PLABLINUX01: Selecting the Open Terminal option from the context menu.

## ***Step 2***

In shell scripting, you can use three different types of if...else statements. They are the following:

- if...fi statement
- if...else...fi statement
- if...elif...else...fi statement

Let's first create a new shell script with the following command:

```
gedit if-fi.sh
```

Press Enter.

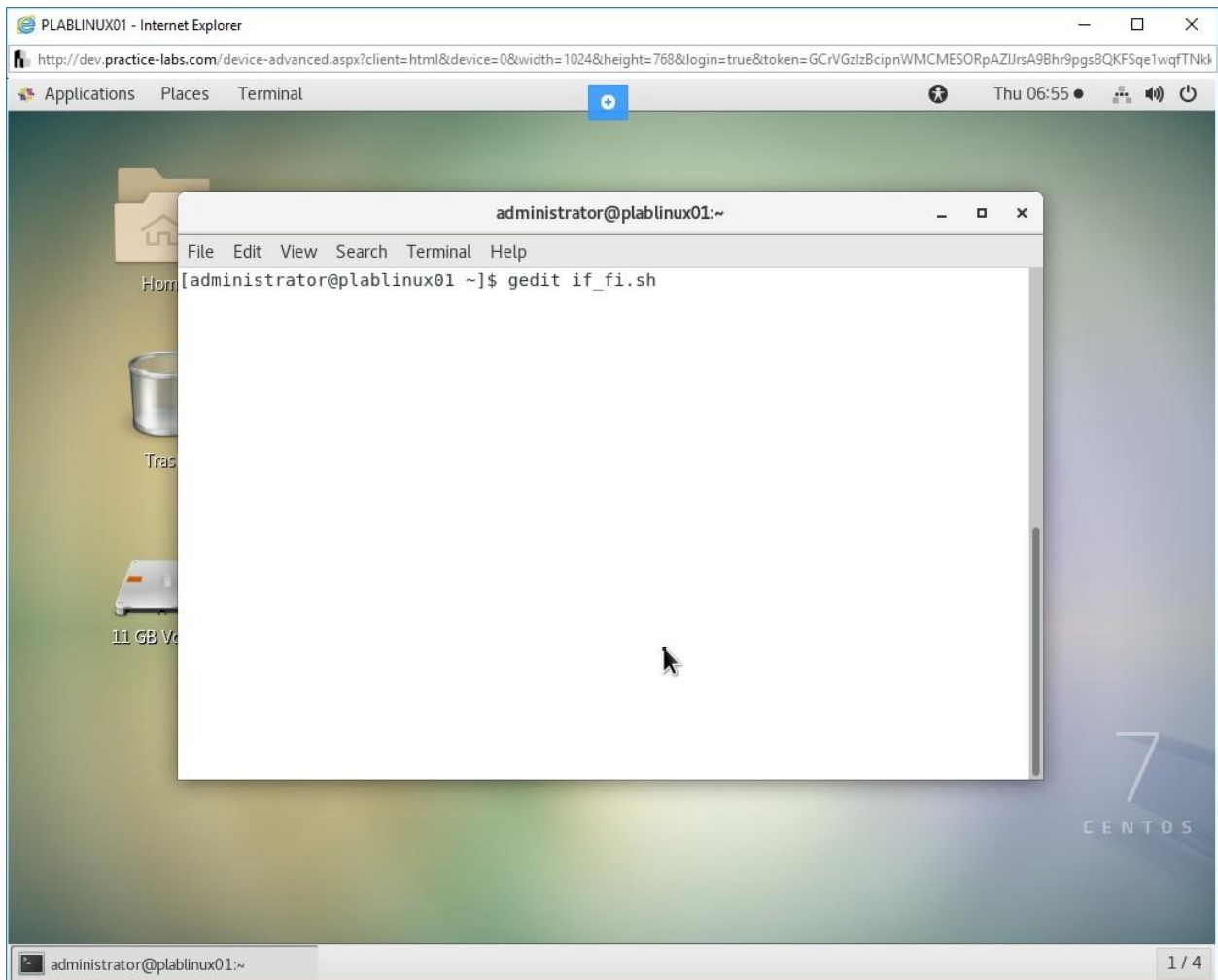


Figure 1.2 Screenshot of PLABLINUX01: Opening a new file in gedit.

### ***Step 3***

Notice a new file is created with the name `if_fi.sh`. Type the following:

```
#!/bin/sh
x=10
z=20
if [ $x == $z ]
then
    echo "x is same as z"
```

```
elif [ $x -gt $z ]
then
    echo "x is more than z"
elif [ $x -lt $z ]
then
    echo "x is less than z"
else
    echo "None of them match"
fi
```

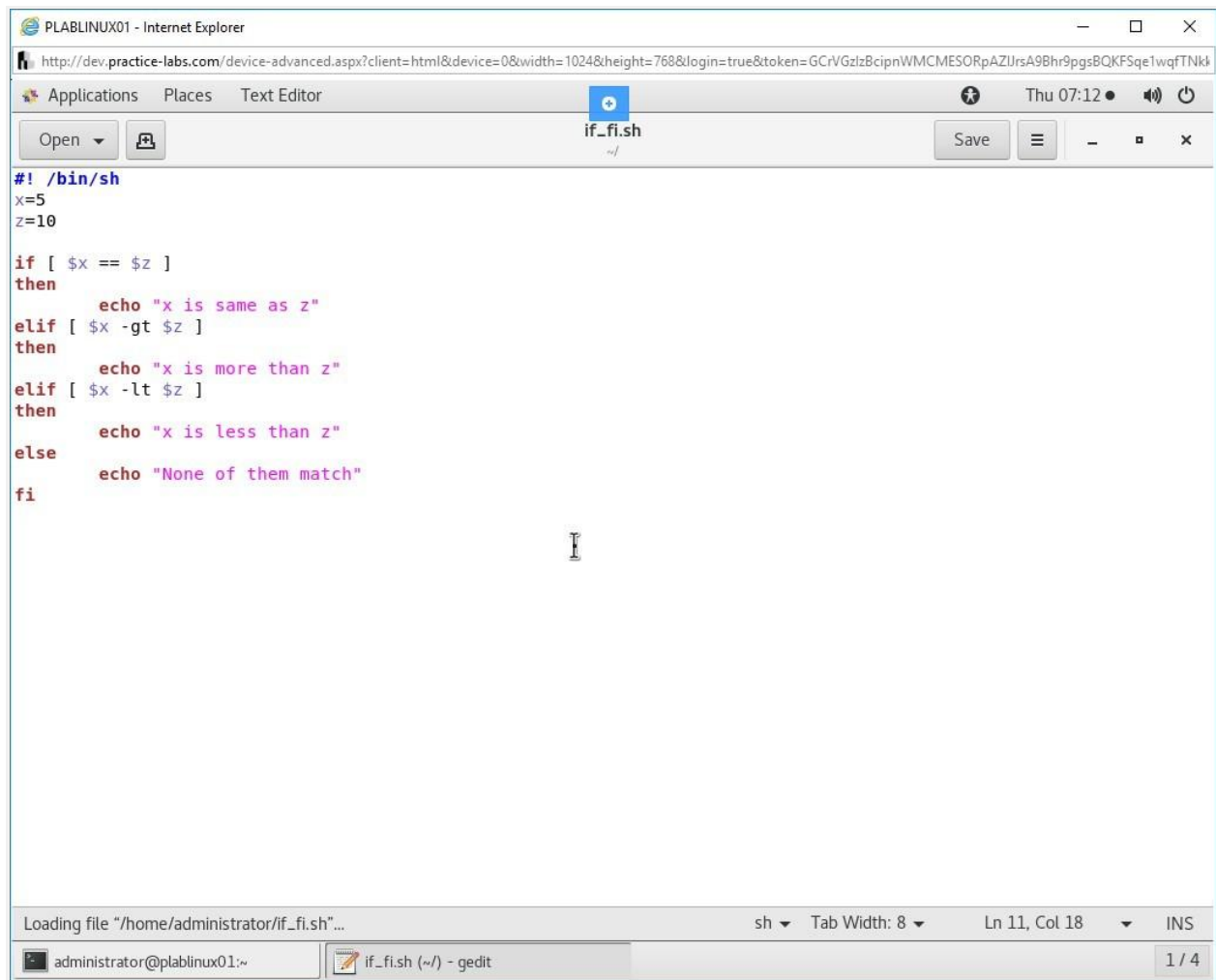


Figure 1.3 Screenshot of PLABINUX01: Entering the content in the file.

## Step 4

Click Save to save the file. Then, close the file.

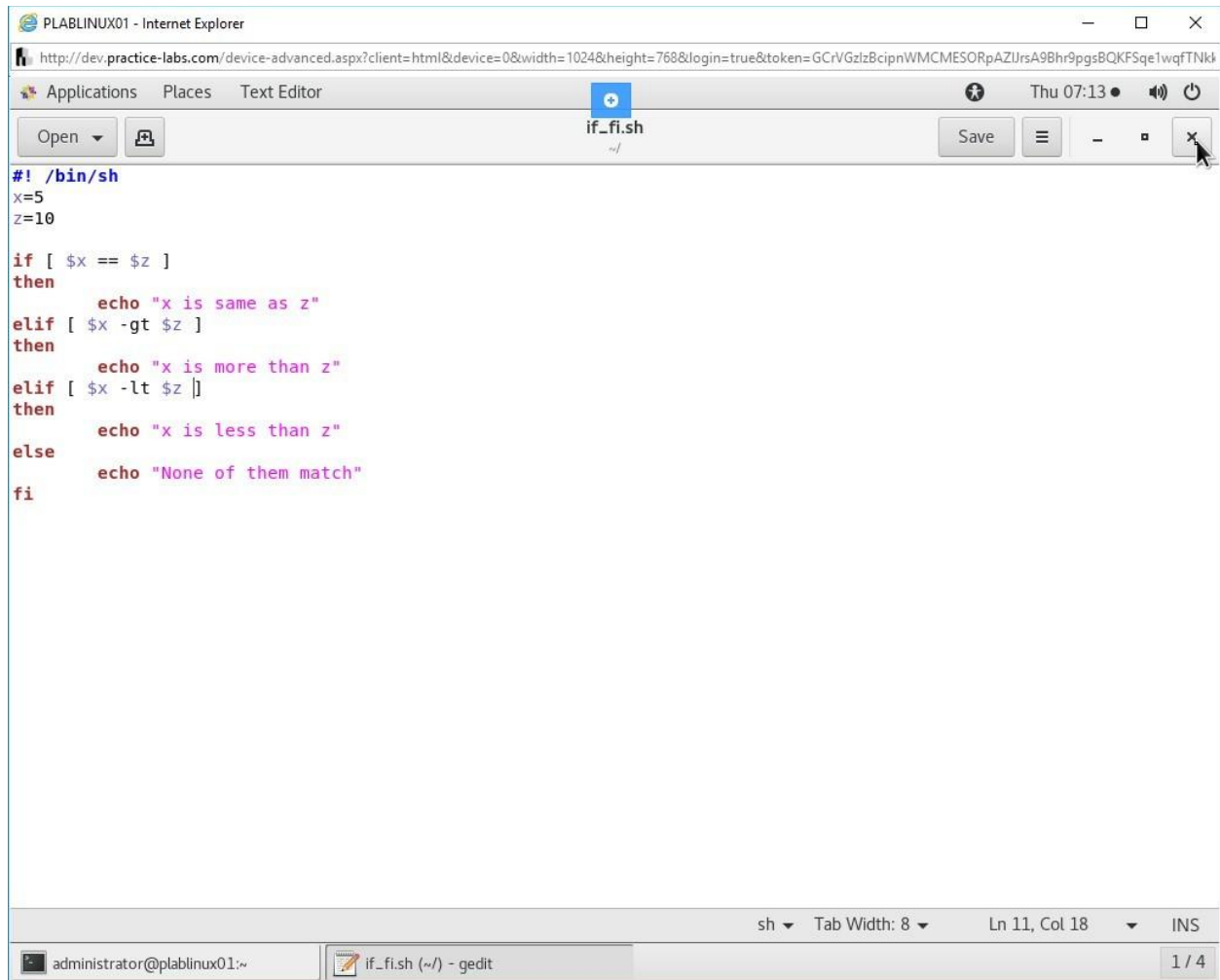


Figure 1.4 Screenshot of PLABLINUX01: Saving and closing the file.

## Step 5

Clear the screen by entering the following command:

```
clear
```

Press Enter. You need to make the shell script executable. Type the following command:

```
sudo chmod +x if-fi.sh
```

Press Enter.

When prompted, type the following password:

**Passw0rd**

Press Enter.

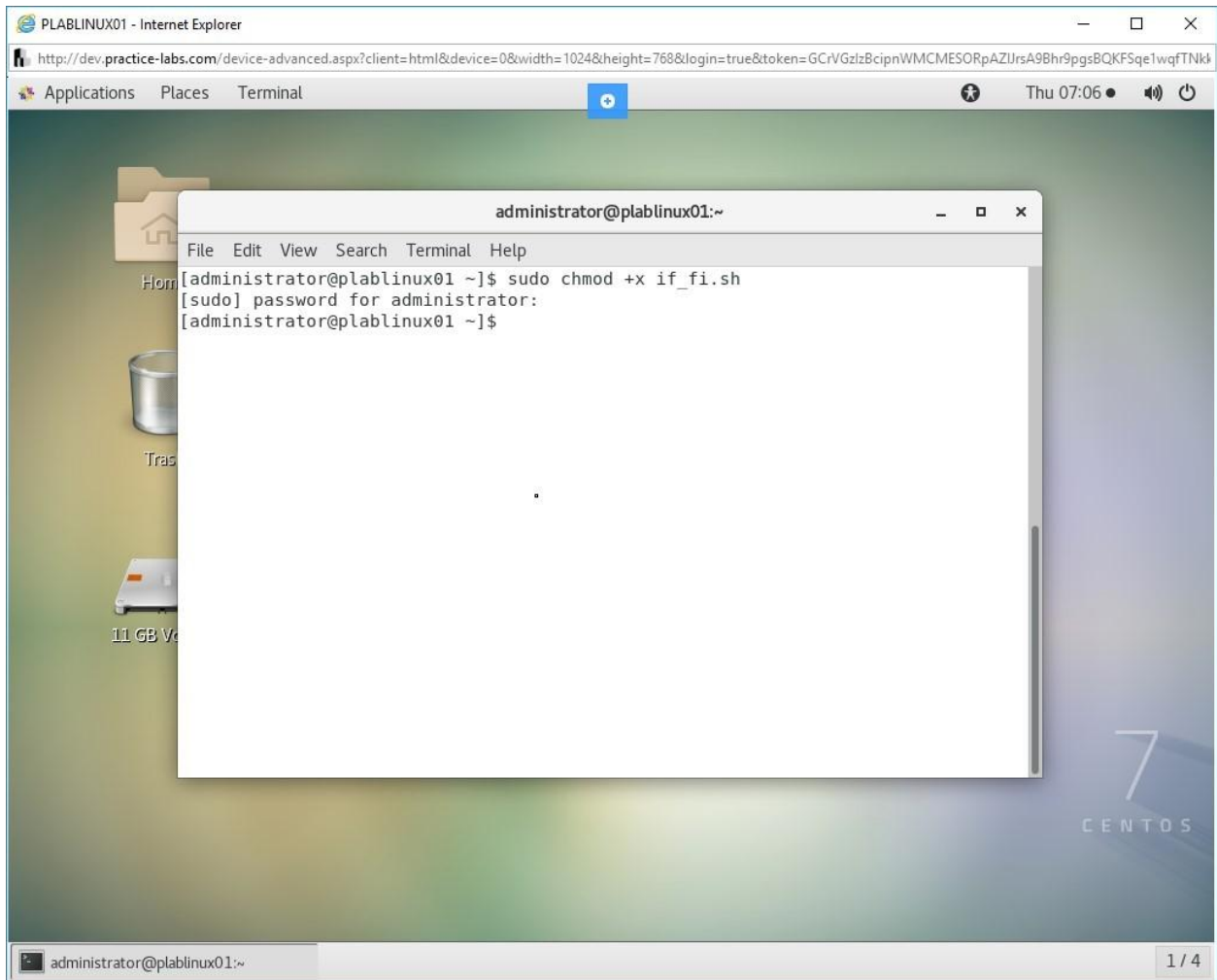


Figure 1.5 Screenshot of PLABLINUX01: Making the file executable.

## **Step 6**

Clear the screen by entering the following command:

`clear`



Press Enter. You will now execute the shell script. Type the following command:

```
./if-fi.sh
```

Press Enter. Note that the last condition is met in which x is less than z.

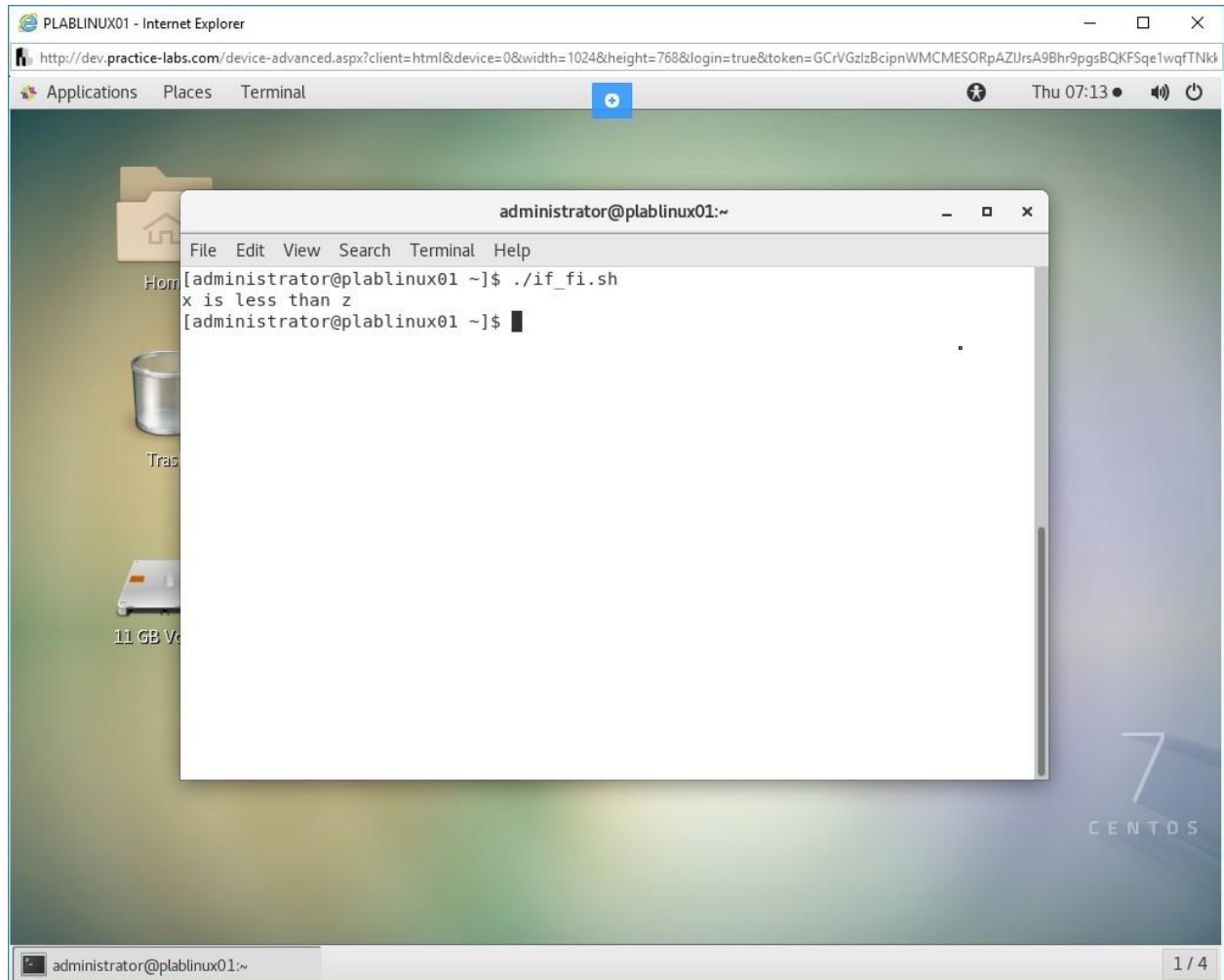


Figure 1.6 Screenshot of PLABLINUX01: Executing the shell script.

## Step 7

In a scenario, where you have multiple conditions, it is best to use case...esac statements. Create a new file with gedit and name is case.sh

Type the following:

```
#!/bin/sh
CAR="Honda"
case "$CAR" in
    "Ferrari") echo "Ferrari is quite expensive."
    ;;
    "Jaguar") echo "I like Jaguar."
    ;;
    "Honda") echo "Honda is the car of the year."
    ;;
esac
```

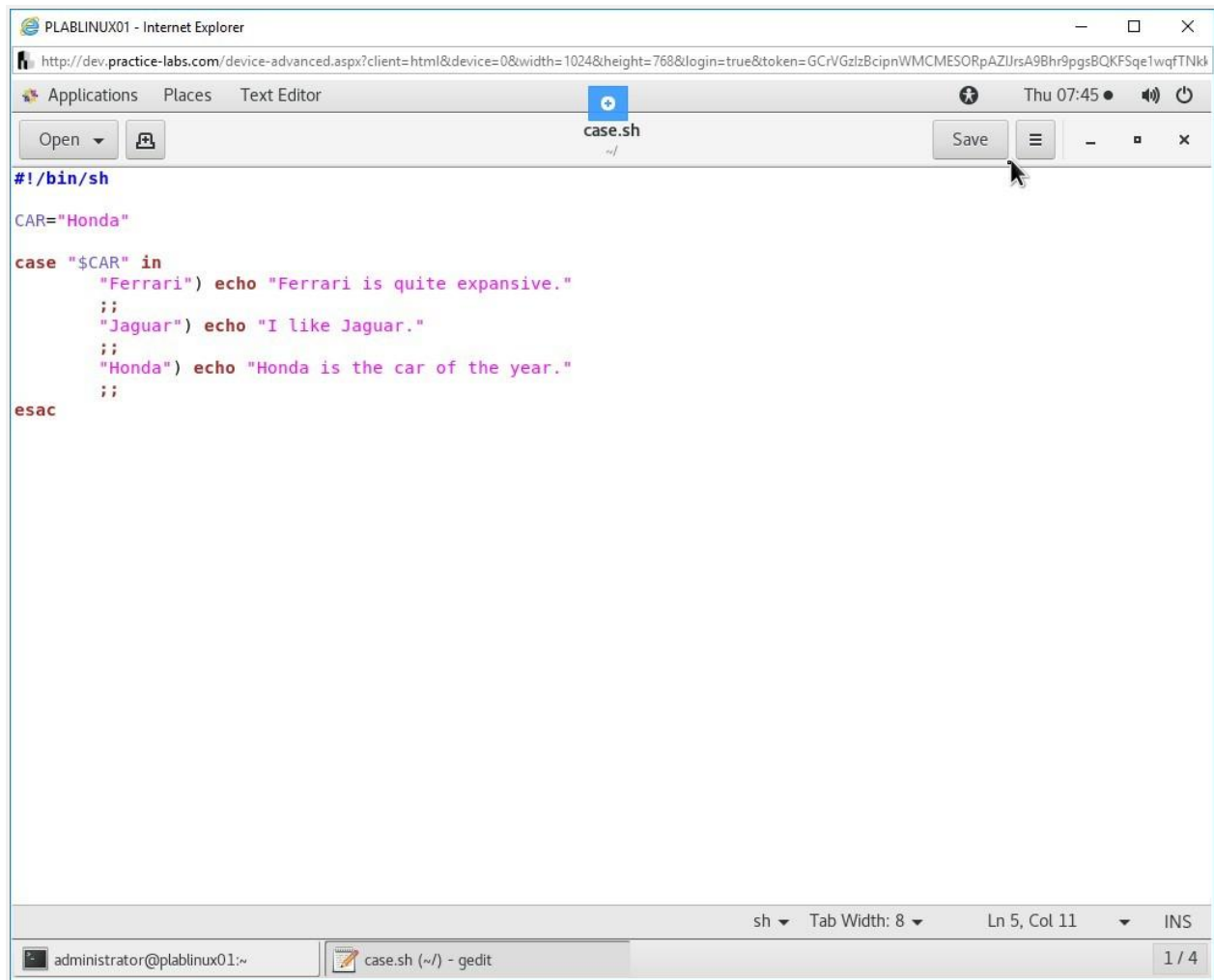


Figure 1.7 Screenshot of PLABLINUX01: Entering the content in a new file.

## Step 8

Click Save to save the file. Then, close the file.

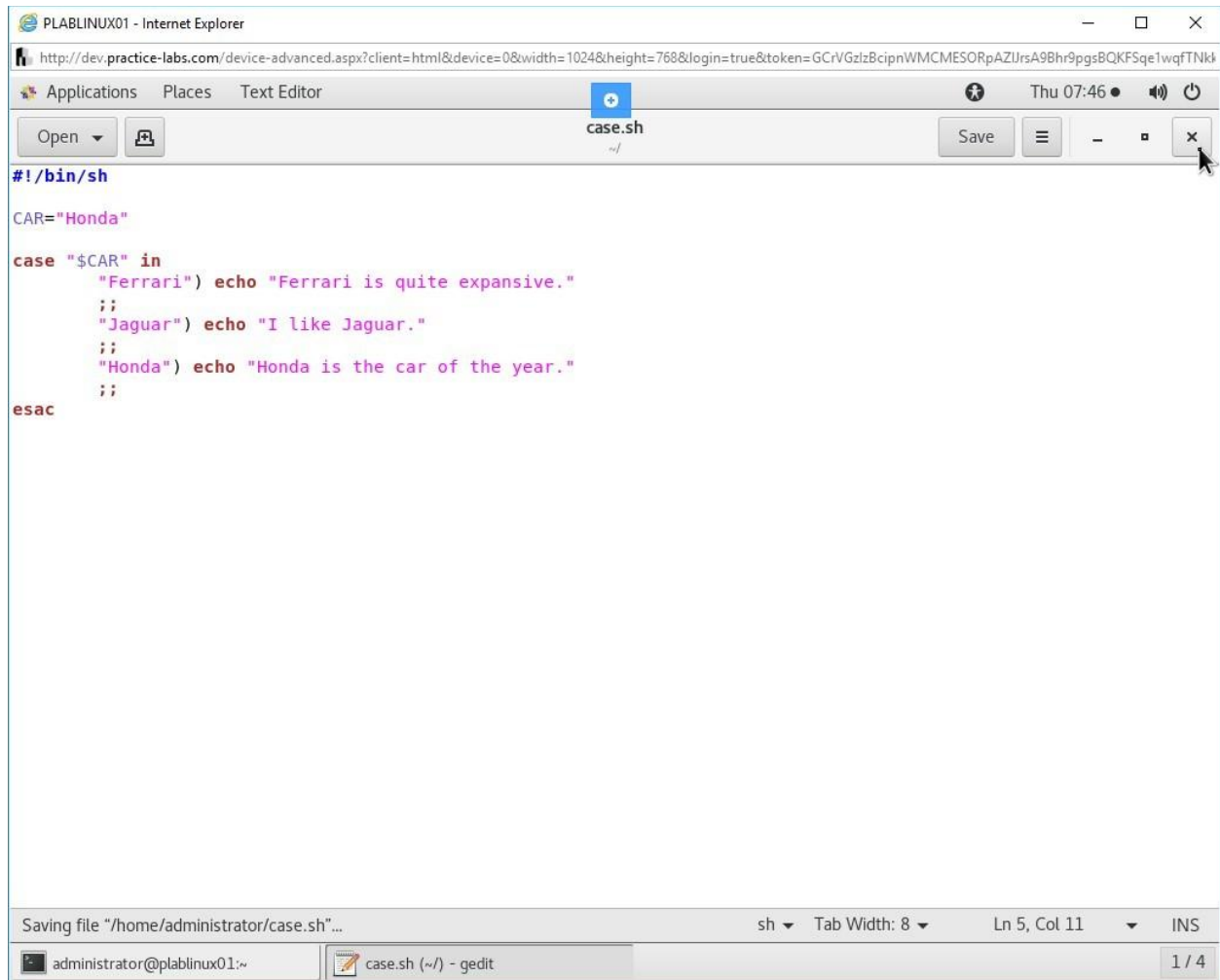


Figure 1.8 Screenshot of PLABLINUX01: Saving and closing the file.

## Step 9

Clear the screen by entering the following command:

```
clear
```

Press Enter. You need to make the shell script executable. Type the following command:

```
sudo chmod +x case.sh
```

Press Enter.

When prompted, type the following password:

**Passw0rd**

Press Enter.

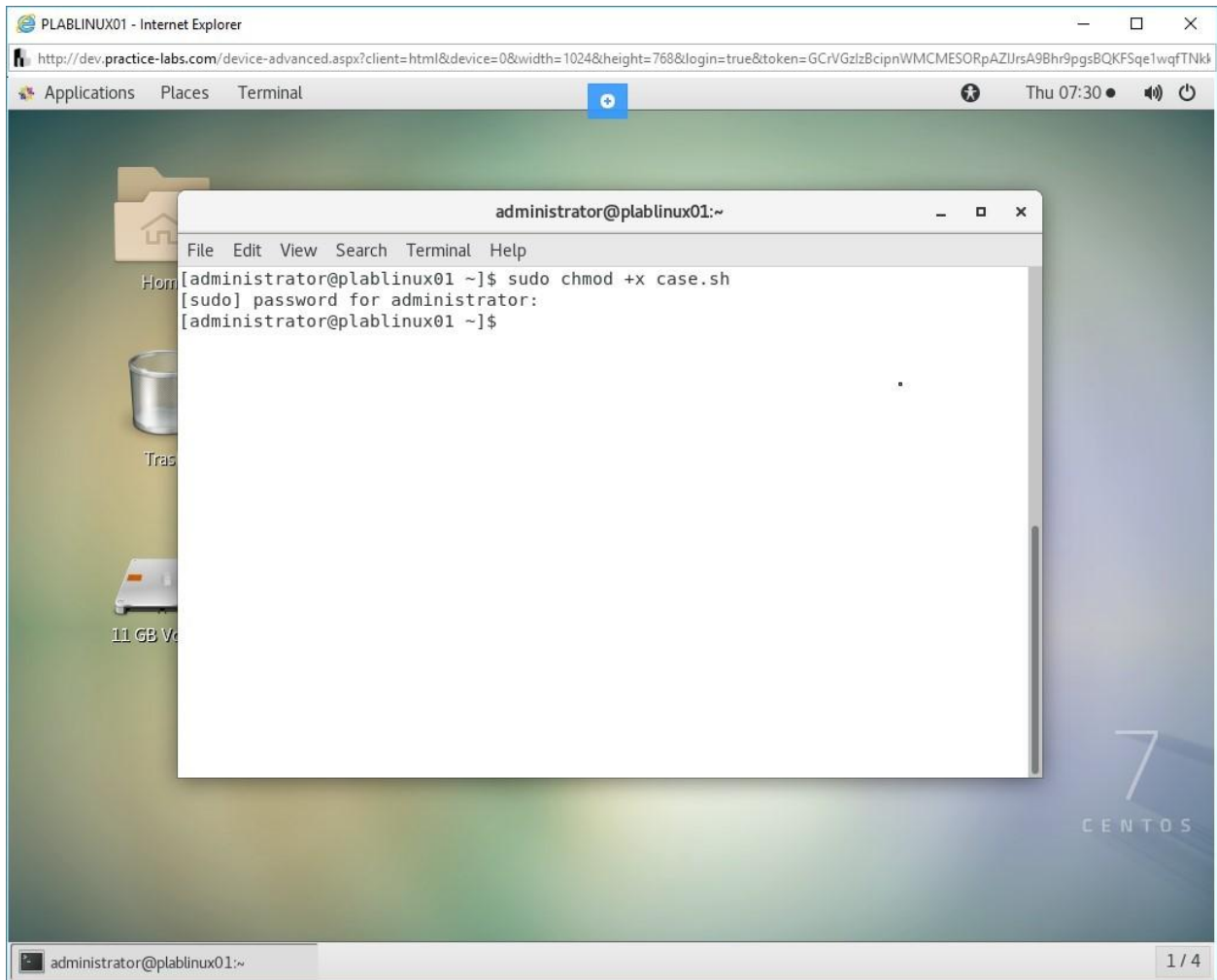


Figure 1.9 Screenshot of PLABLINUX01: Making the file executable.

## **Step 10**

Clear the screen by entering the following command:

`clear`

Press Enter. You will now execute the shell script. Type the following command:

```
./case.sh
```

Press Enter. Notice the answer. After evaluating two conditions, the third condition meets the criteria and is printed as the answer.

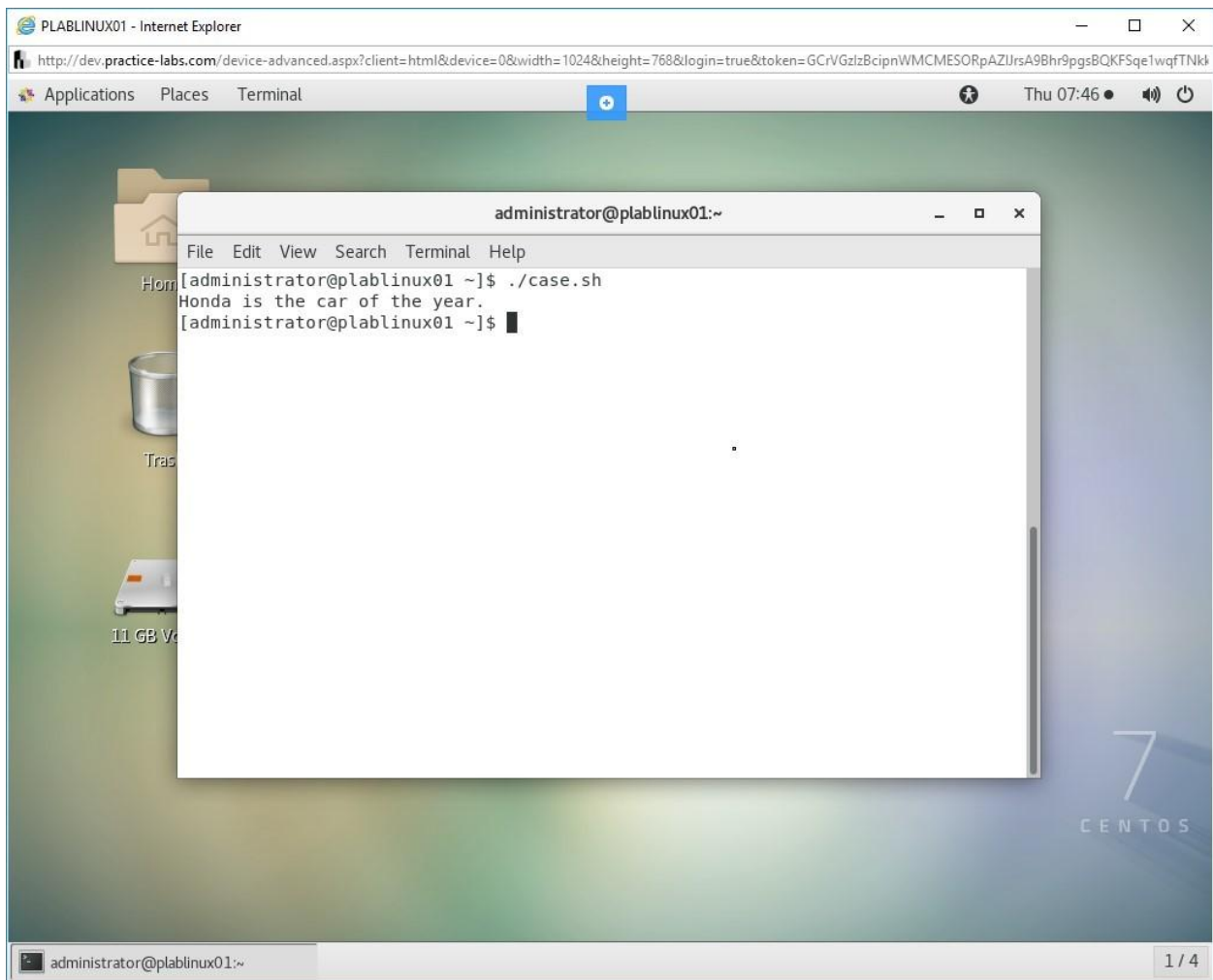


Figure 1.10 Screenshot of PLABLINUX01: Executing the shell script.

## Task 2 - Use While Loops

A while loop runs until a statement becomes true. It can nest other loops. There is no limit to nesting loops. The number is limited to your requirement.

In this task, you will learn to use while loop. To use the loop, perform the following steps:

### ***Step 1***

Clear the screen by entering the following command:

```
clear
```

Press Enter. Let's first create a new shell script with the following command:

```
gedit while.sh
```

Press Enter.

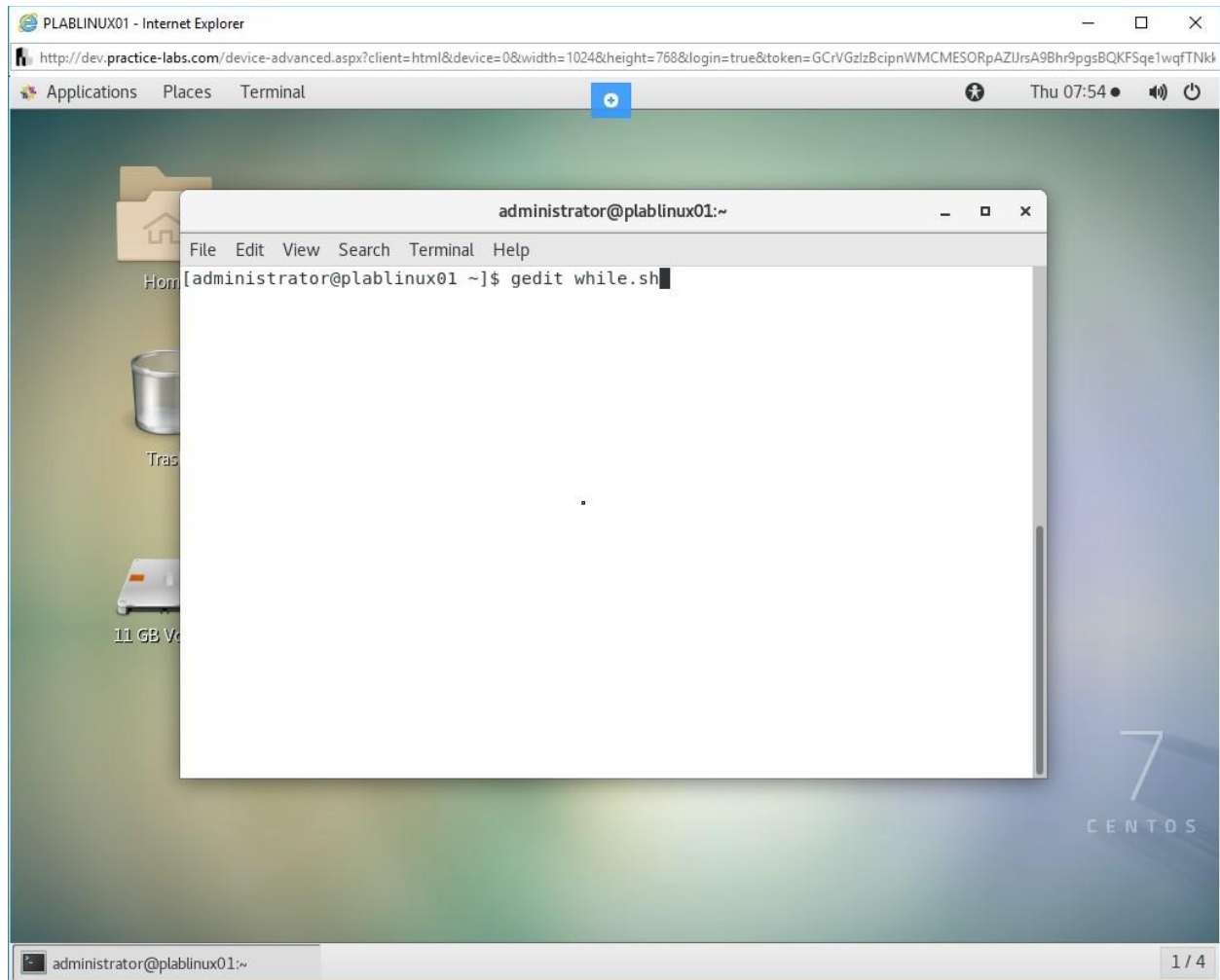
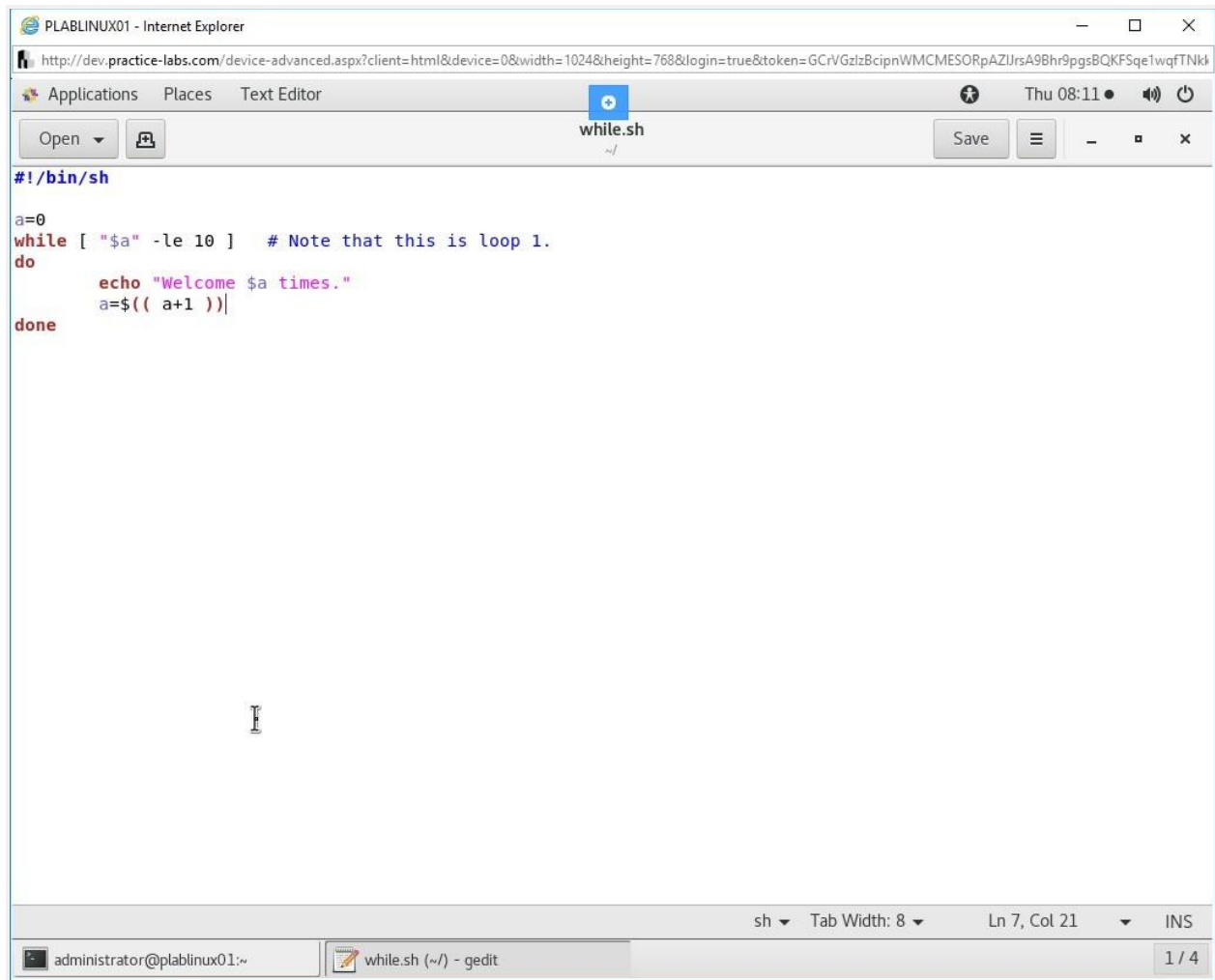


Figure 1.11 Screenshot of PLABLINUX01: Opening a new file in gedit.

## ***Step 2***

Notice a new file is created with the name while.sh. Type the following:

```
#!/bin/bash
a=0
while [ $a -le 10 ]
do
    echo "Welcome $a times."
    a=$(( a+1 ))
done
```



The screenshot shows a web browser window titled "PLABLINUX01 - Internet Explorer". The address bar contains a URL from "dev.practice-labs.com". The browser's menu bar includes "Applications", "Places", and "Text Editor". A toolbar at the top of the editor shows "Open", "while.sh", "Save", and window controls. The main text area contains a shell script with the following content:

```
#!/bin/sh

a=0
while [ "$a" -le 10 ] # Note that this is loop 1.
do
    echo "Welcome $a times."
    a=$(( a+1 ))
done
```

The status bar at the bottom indicates the shell is "sh", tab width is "8", and the cursor is at "Ln 7, Col 21". The terminal bar shows the user is "administrator@plablinux01:~" and the file "while.sh (~/) - gedit" is open.

Figure 1.12 Screenshot of PLABLINUX01: Entering the content in the file.

### Step 3

Click Save to save the file. Then, close the file.



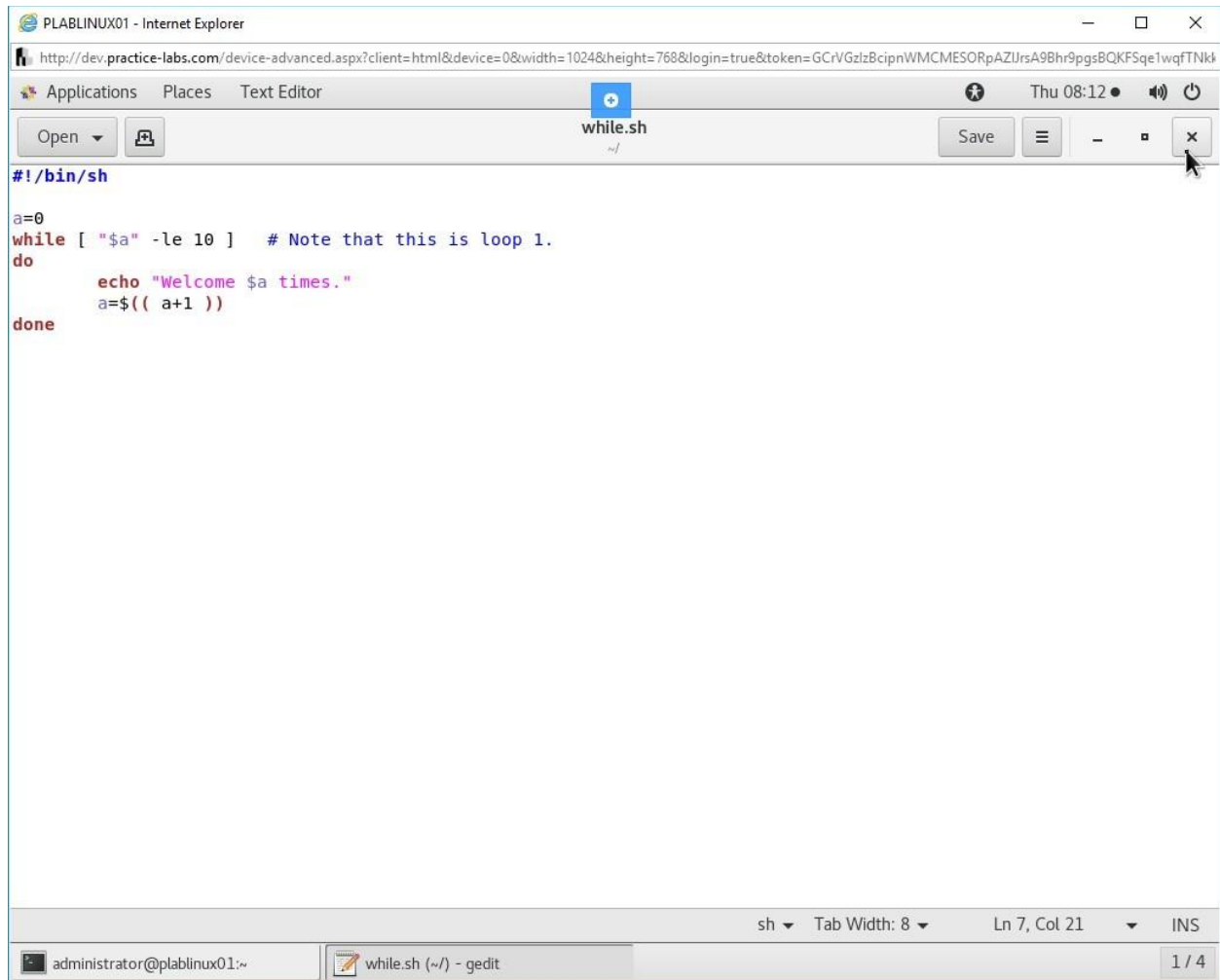


Figure 1.13 Screenshot of PLABINUX01: Saving and closing the file.

## Step 4

Clear the screen by entering the following command:

```
clear
```

Press Enter. You need to make the shell script executable. Type the following command:

```
sudo chmod +x while.sh
```

Press Enter.

When prompted, type the following password:

**Passw0rd**

Press Enter.

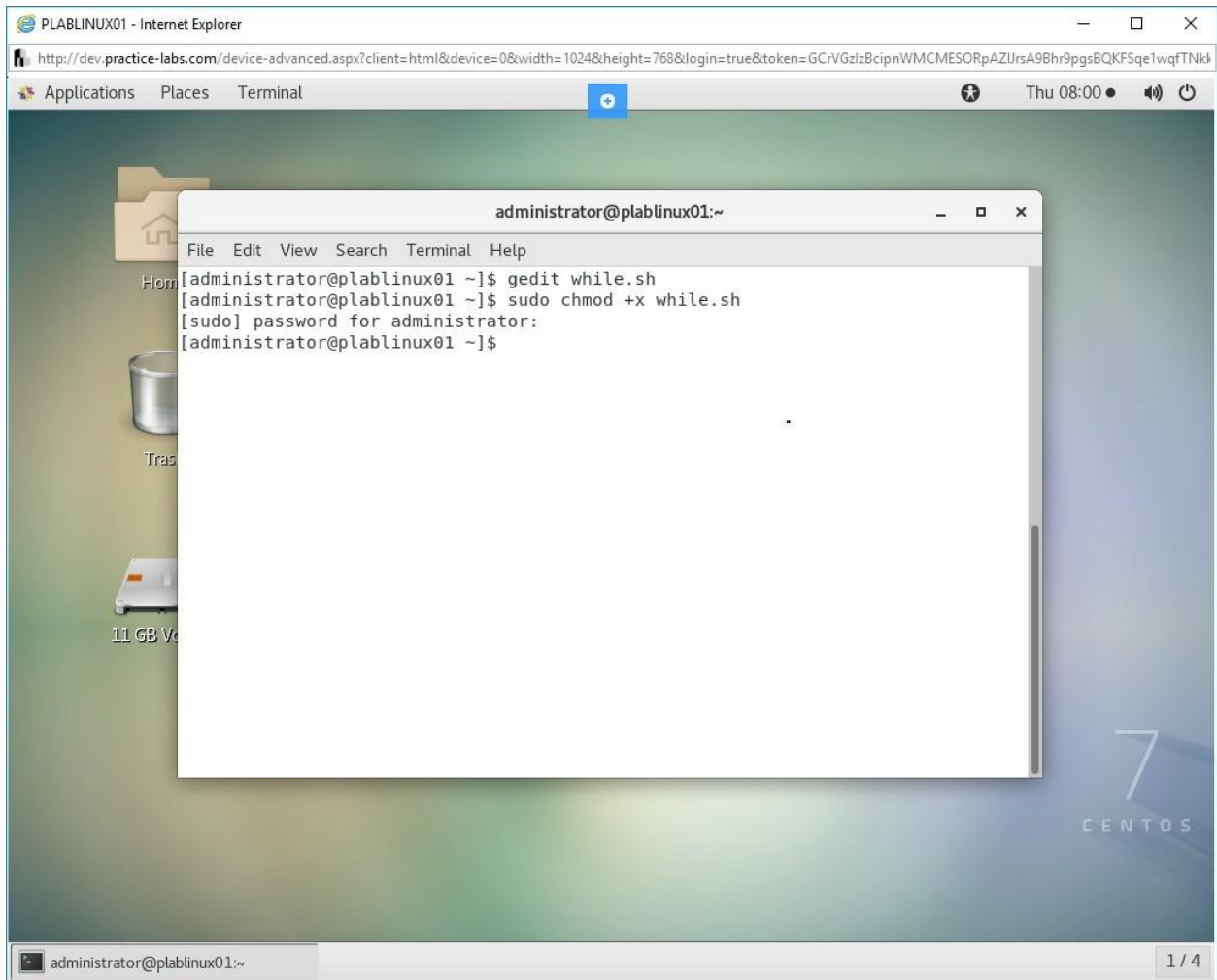


Figure 1.14 Screenshot of PLABLINUX01: Making the file executable.

## **Step 5**

Clear the screen by entering the following command:

`clear`

Press Enter. You will now execute the shell script. Type the following command:

```
./while.sh
```

Press Enter. Note that the last condition is met in which x is less than z.

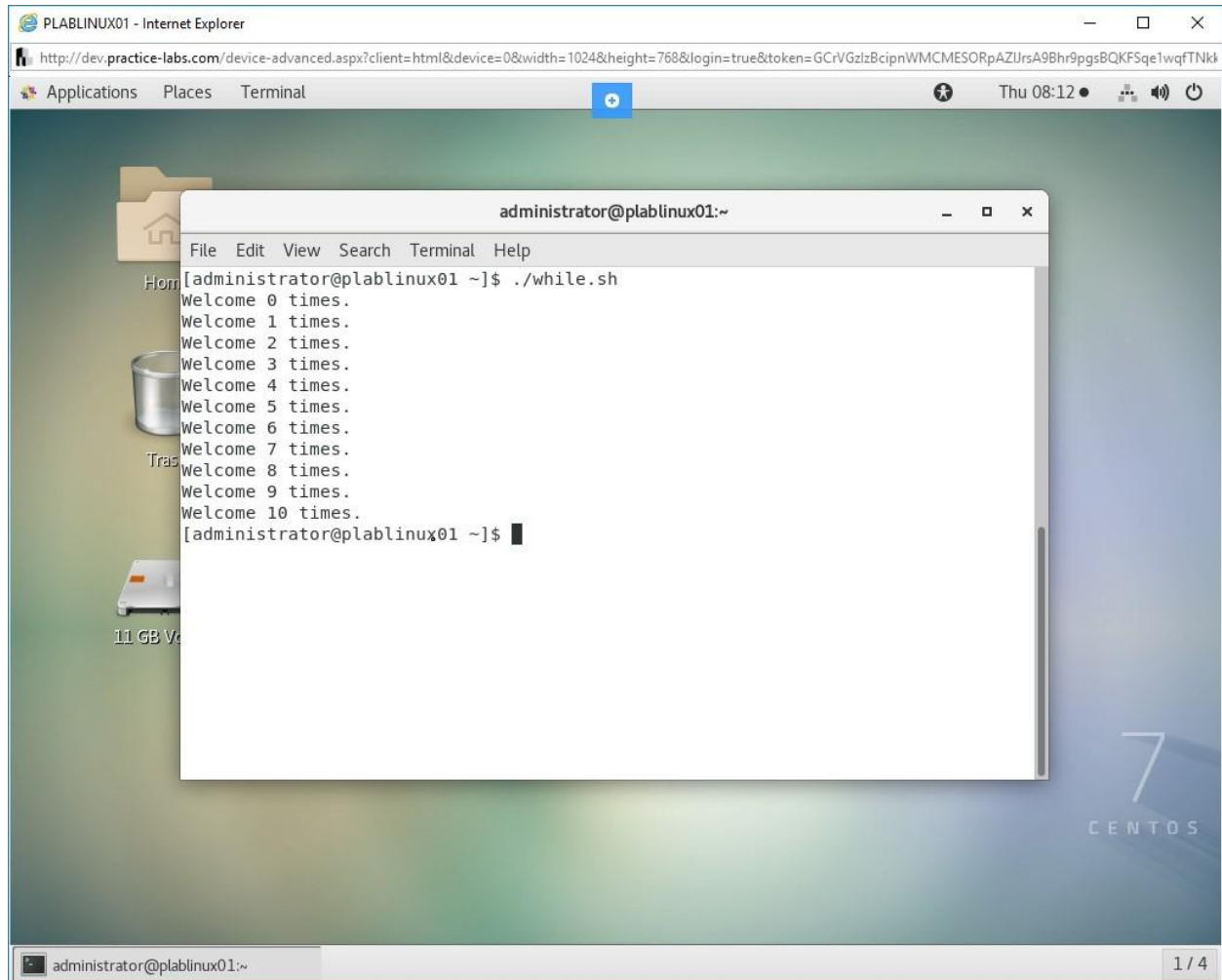


Figure 1.15 Screenshot of PLABLINUX01: Executing the shell script.

### Task 3 - Use Until Loops

An until loop runs until a statement becomes true.

In this task, you will learn to use until loop. To use the until loop, perform the following steps:

## Step 1

Clear the screen by entering the following command:

```
clear
```

Press Enter. Let's first create a new shell script with the following command:

```
gedit until.sh
```

Press Enter.

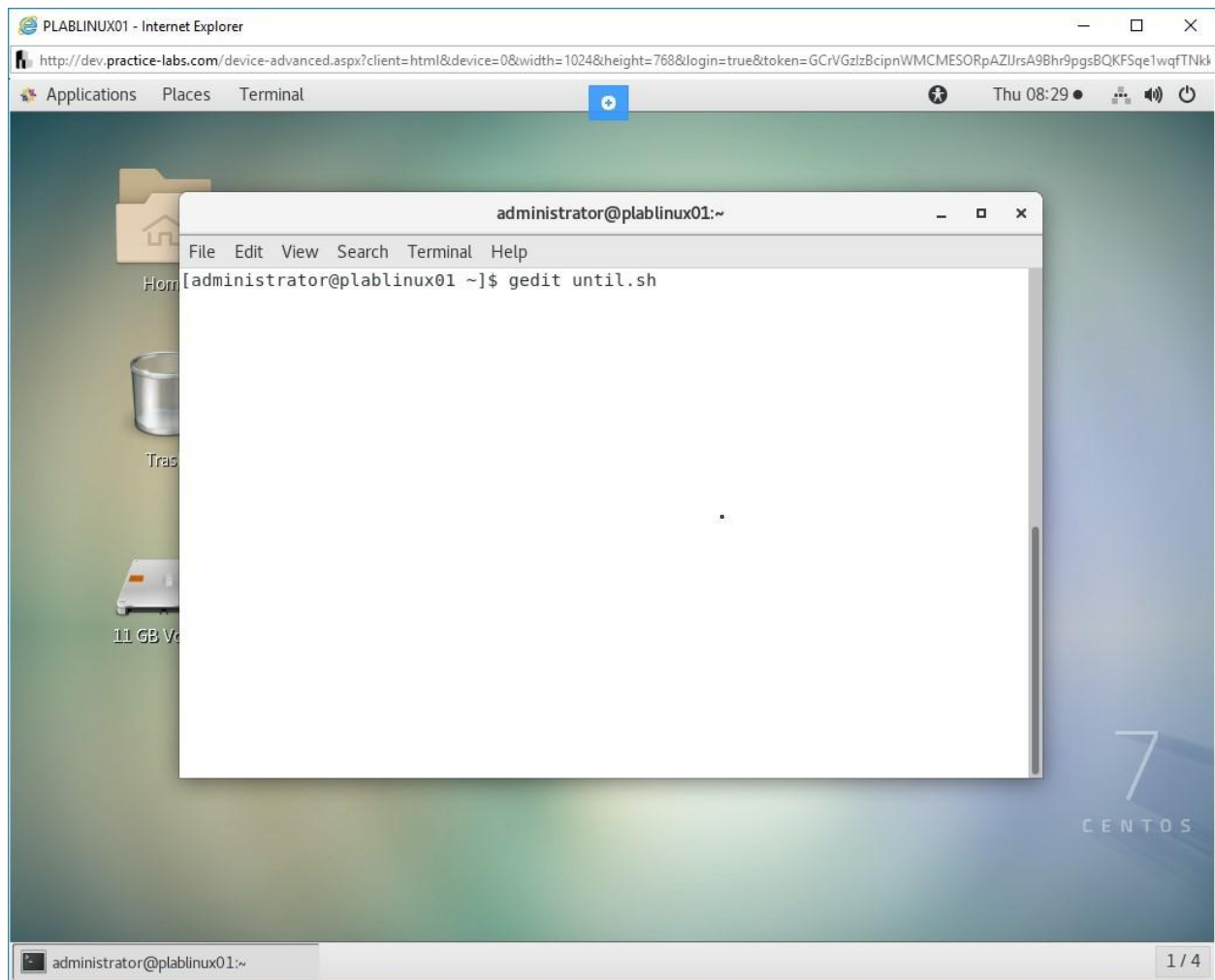


Figure 1.16 Screenshot of PLABLINUX01: Creating a new file in gedit.

## Step 2

Notice a new file is created with the name `until.sh`. Type the following:

```
#!/bin/sh
a=1
until [ $a -gt 6 ]
do
    echo $a
    a=$(( a+1 ))
done
```

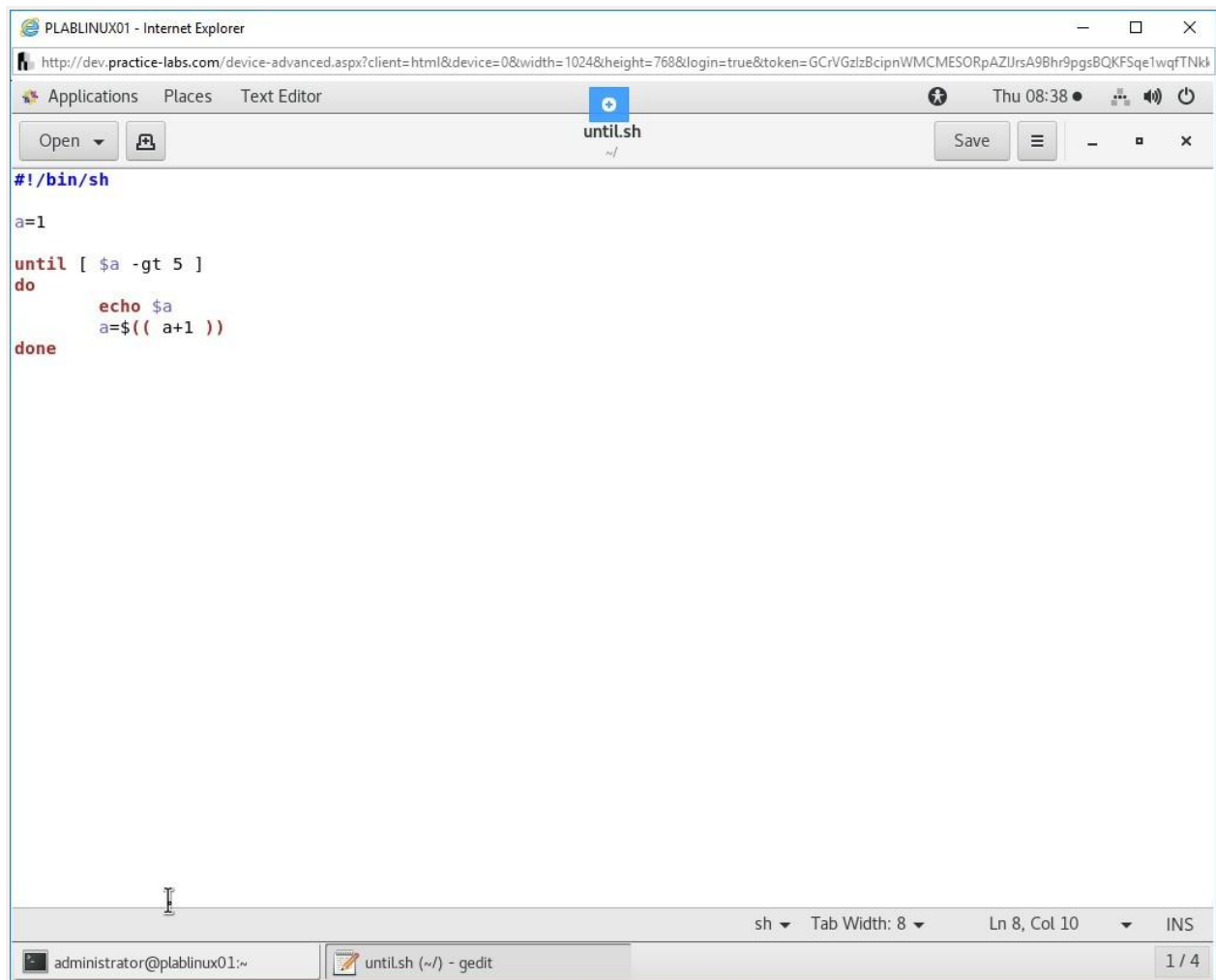


Figure 1.17 Screenshot of PLABLINUX01: Entering the content in the file.

## Step 3

Click Save to save the file. Then, close the file.

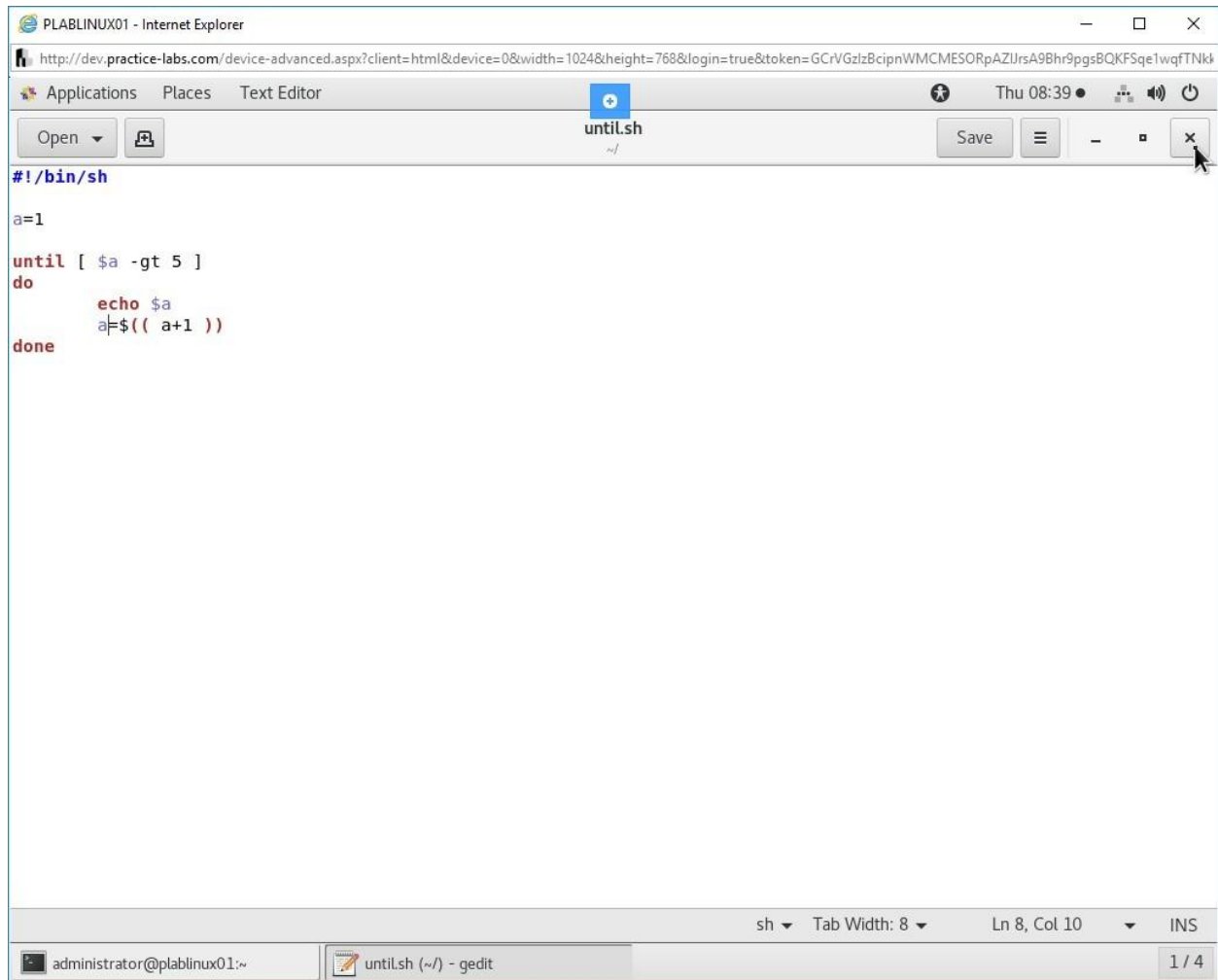


Figure 1.18 Screenshot of PLABLINUX01: Saving and closing the file.

## Step 4

Clear the screen by entering the following command:

```
clear
```

Press Enter. You need to make the shell script executable. Type the following command:

```
sudo chmod +x until.sh
```

Press Enter.

When prompted, type the following password:

**Passw0rd**

Press Enter.

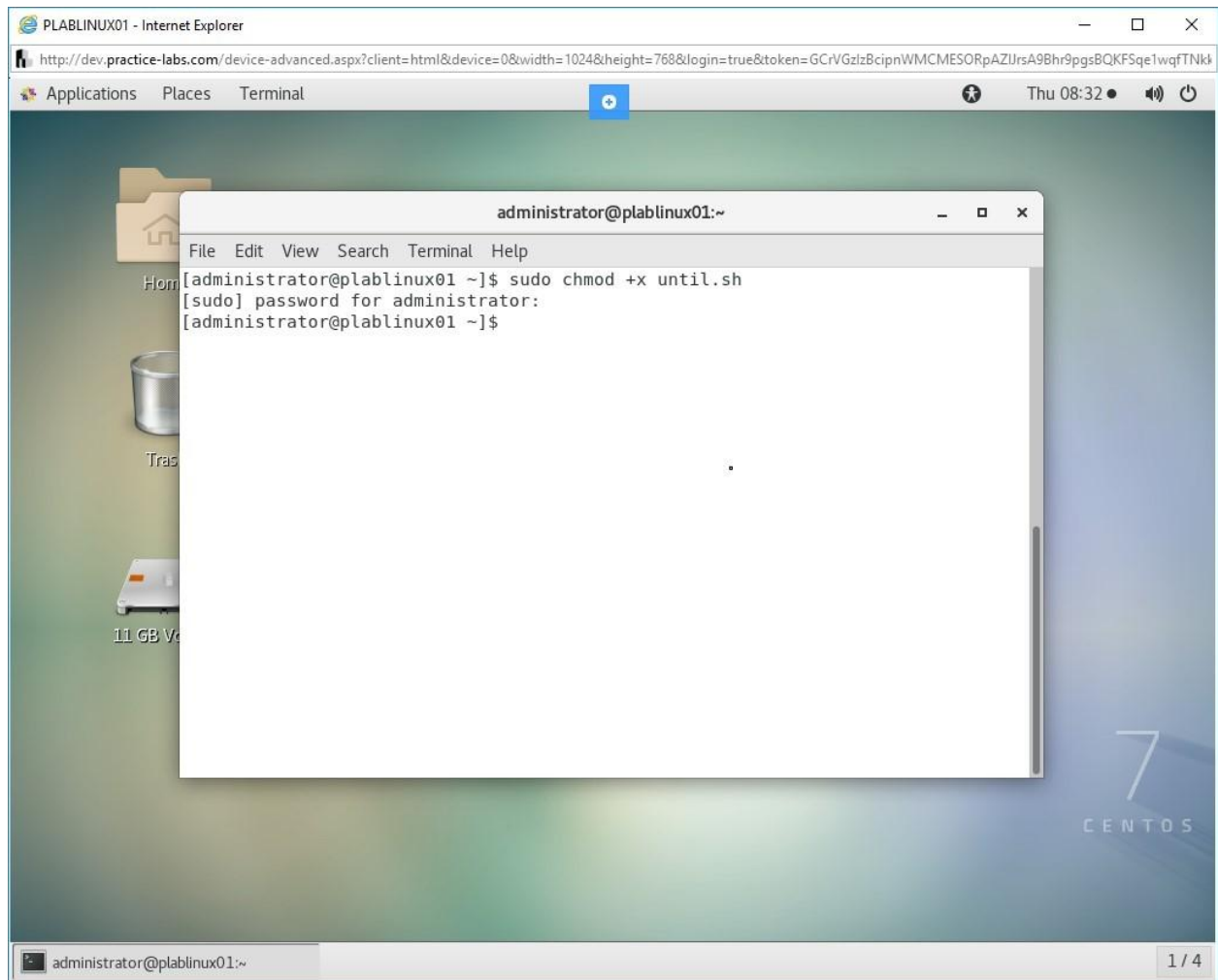


Figure 1.19 Screenshot of PLABLINUX01: Making the file executable.

## **Step 5**

Clear the screen by entering the following command:

`clear`

Press Enter. You will now execute the shell script. Type the following command:

```
./until.sh
```

Press Enter.

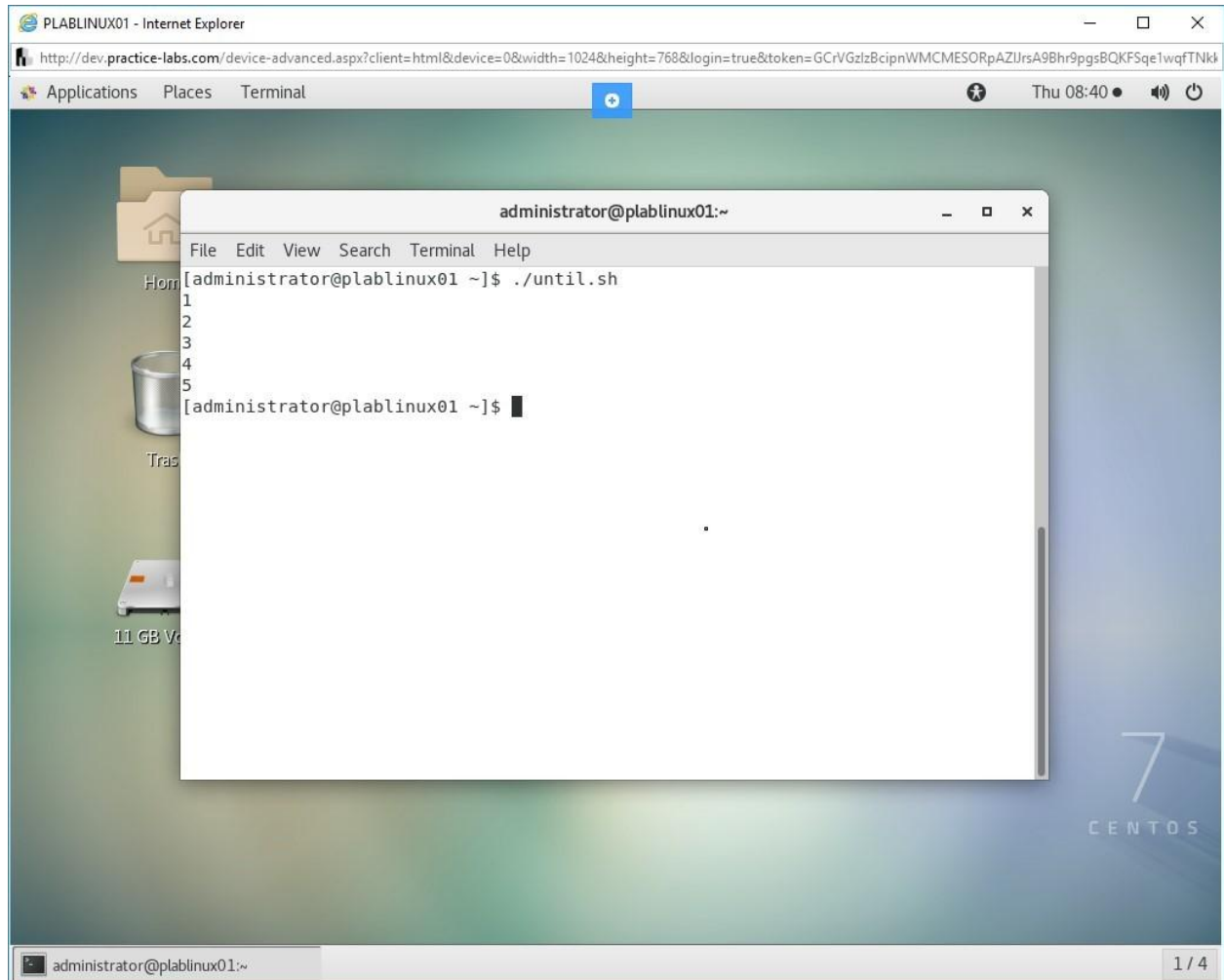


Figure 1.20 Screenshot of PLABLINUX01: Executing the shell script.

#### Task 4 - Use for Loops

The for loop runs through a list of values in a list until the time list values are exhausted.



In this task, you will learn to use for loop. To use the for loop, perform the following steps:

### ***Step 1***

Clear the screen by entering the following command:

```
clear
```

Press Enter. Let's first create a new shell script with the following command:

```
gedit for.sh
```

Press Enter.

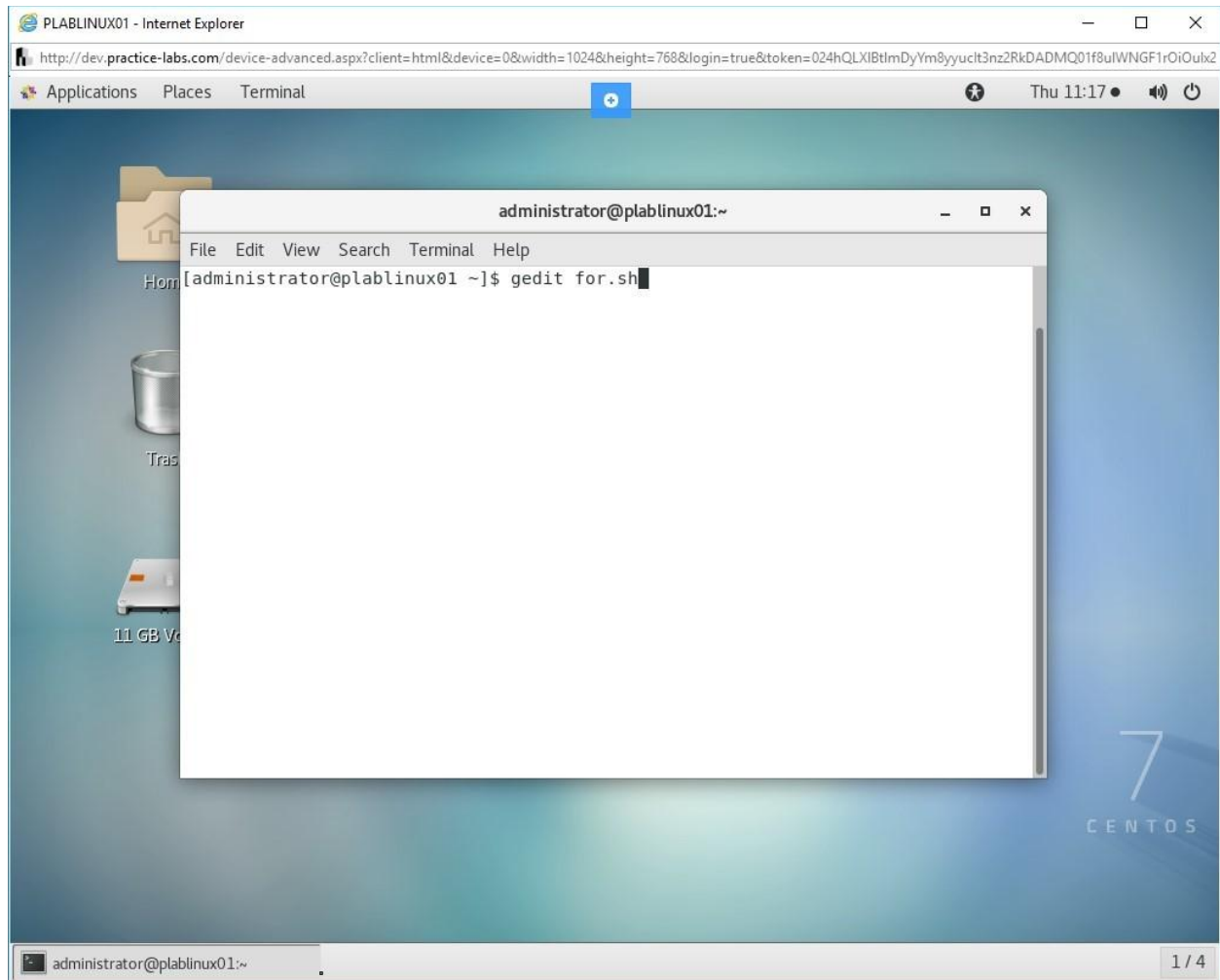


Figure 1.21 Screenshot of PLABLINUX01: Creating a new file in gedit.

## ***Step 2***

Notice a new file is created with the name for.sh. Type the following:

```
#!/bin/bash
echo "Files in the /etc directory are:"
for f in $(ls /etc/*)
do
    echo $f
done
```

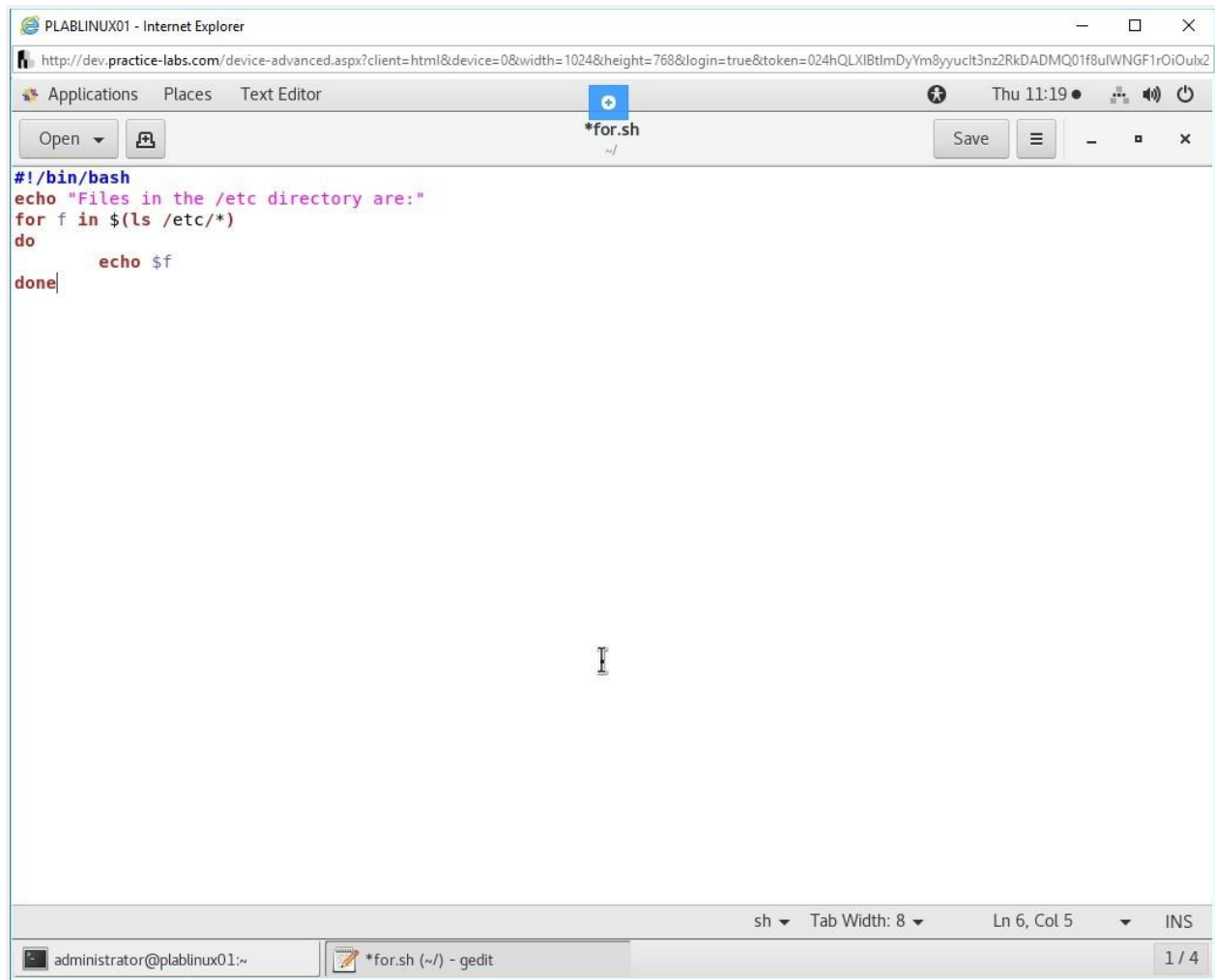


Figure 1.22 Screenshot of PLABLINUX01: Entering the content in the file.

### ***Step 3***

Click Save to save the file. Then, close the file.

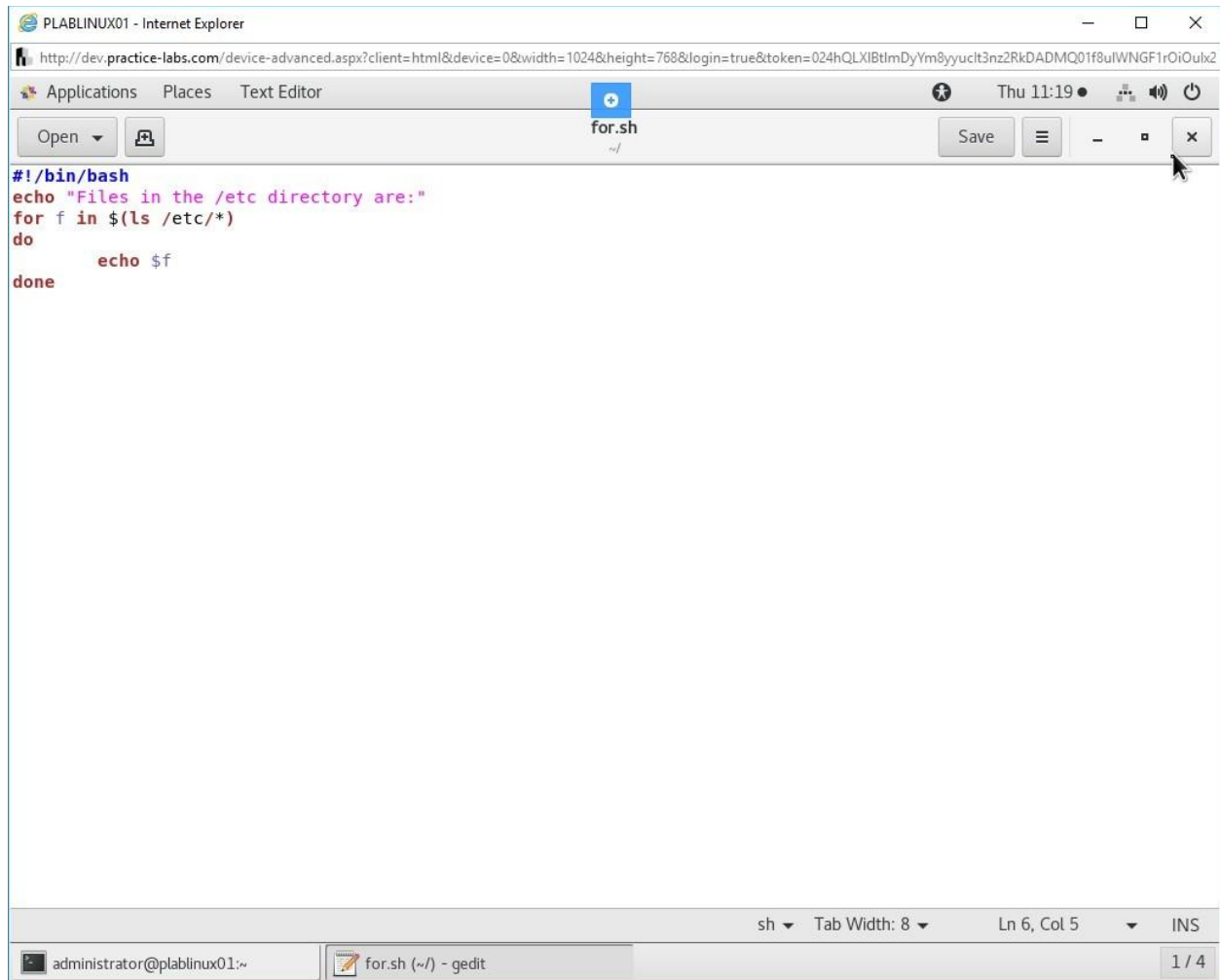


Figure 1.23 Screenshot of PLABLINUX01: Saving and closing the file.

## Step 4

Clear the screen by entering the following command:

```
clear
```

Press Enter. You need to make the shell script executable. Type the following command:

```
sudo chmod +x for.sh
```

Press Enter.

When prompted, type the following password:

**Passw0rd**

Press Enter.

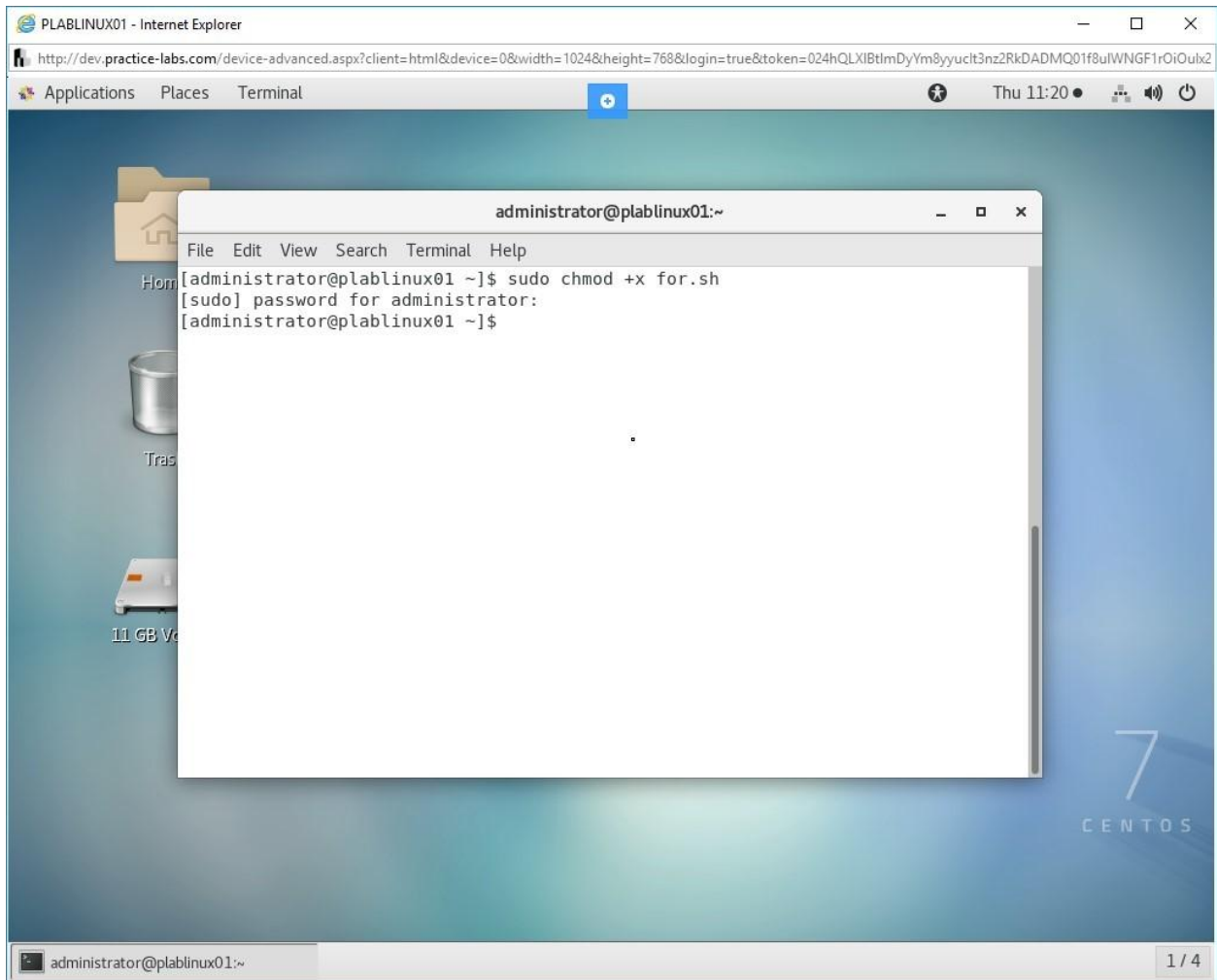


Figure 1.24 Screenshot of PLABLINUX01: Making the file executable.

## **Step 5**

Clear the screen by entering the following command:

`clear`

Press Enter. You will now execute the shell script. Type the following command:

```
./for.sh
```

Press Enter.

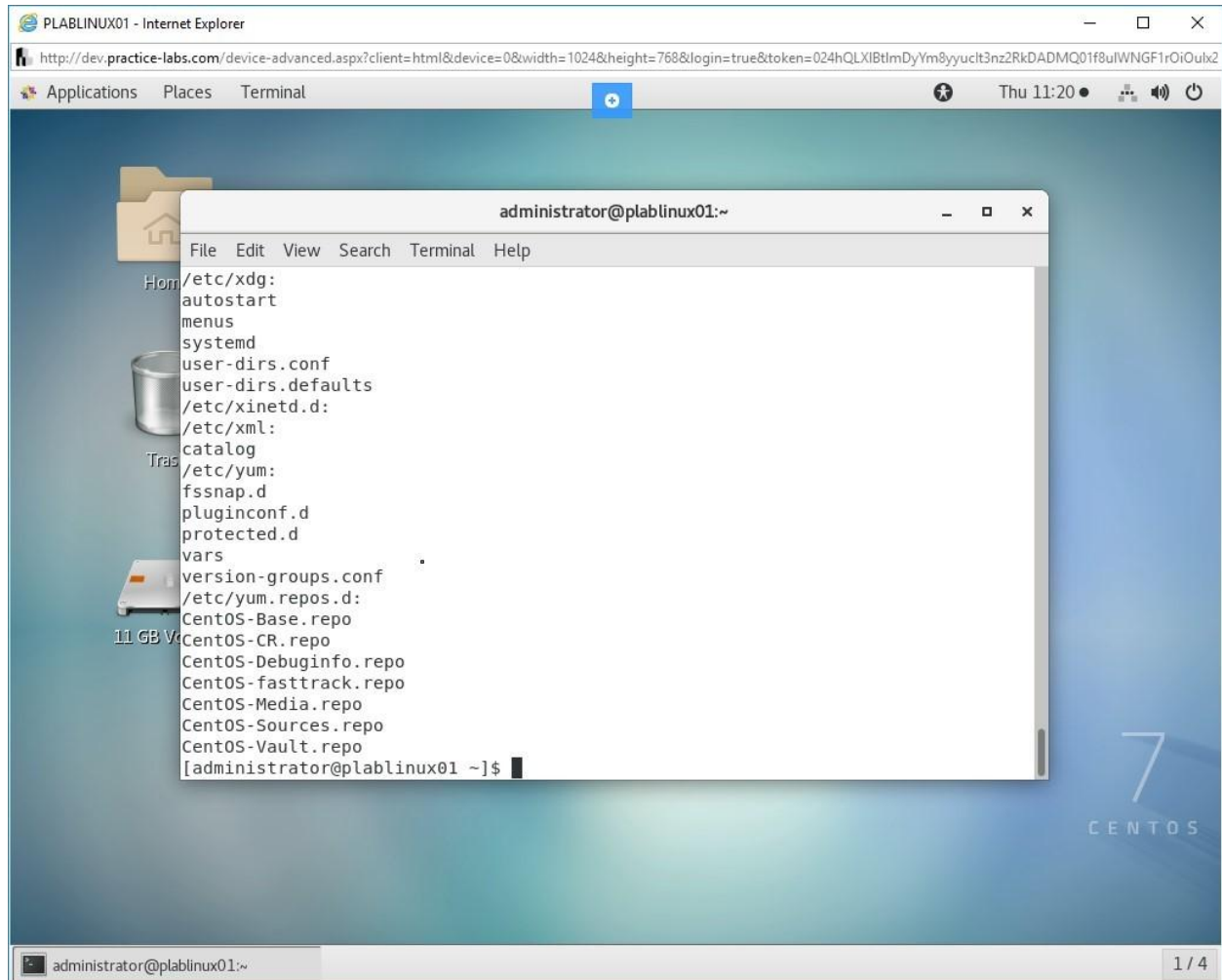


Figure 1.25 Screenshot of PLABLINUX01: Executing the shell script.

Keep all devices in their current state and proceed to the next exercise.

## Review

Well done, you have completed the Work with the Flow Control Constructs Practice Lab.

## **Summary**

You completed the following exercise:

- Exercise 1 - Work with the flow control constructs

You should now be able to:

- Use the if...else statement
- Use the while loop
- Use the until loop
- Use the for loop