

Input and output in shell script

Group G(Prajna C, Sandhya Lokesh, Pooja Kadam)

Command we use is **read**

Syntax : read <input variable>

Ex program:

```
read input1
```

```
echo "$input1"
```

read with -p

Syntax : read -p PROMPT <input variable>

Ex program:

```
read -p "Lets take an input" input1  
echo "$input1"
```

read with -sp

Syntax : read -sp PROMPT <input variable>

Ex program:

```
read -sp "Lets take an input secretly" input1  
echo "$input1"
```

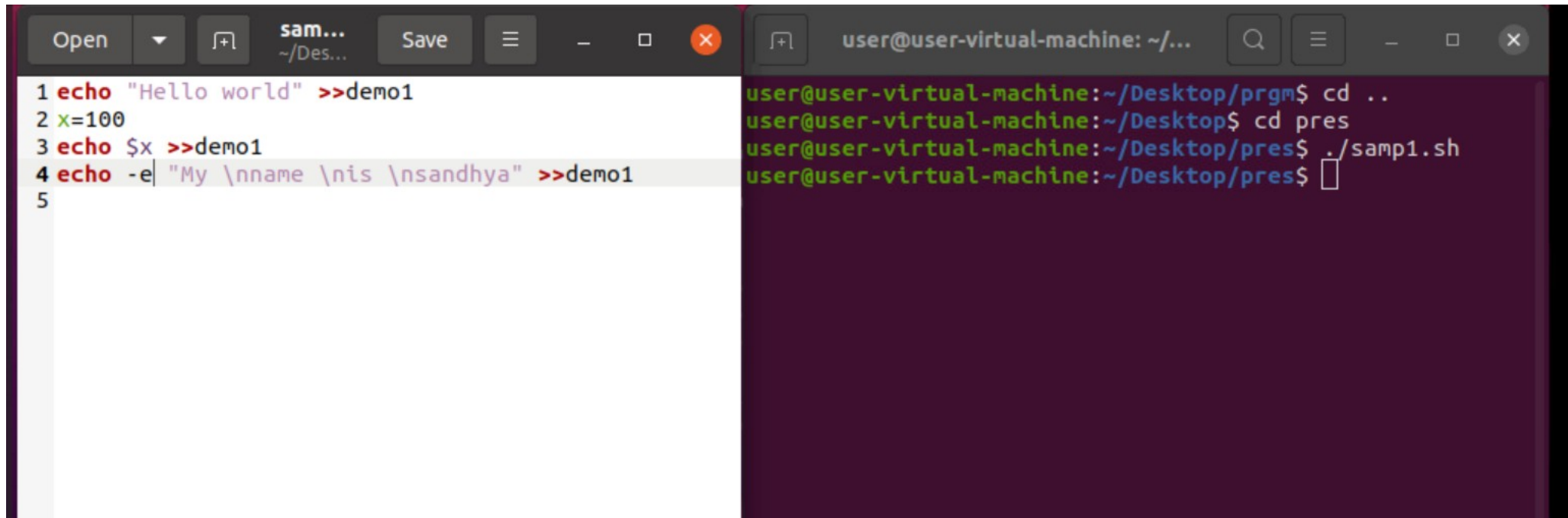
read with -a

Syntax : read -a <array_name>

Ex program:

```
read -ap "Lets take input as an array" input1
for var in $input1
do
    Echo "$var"
done
```

Printing the output on the file

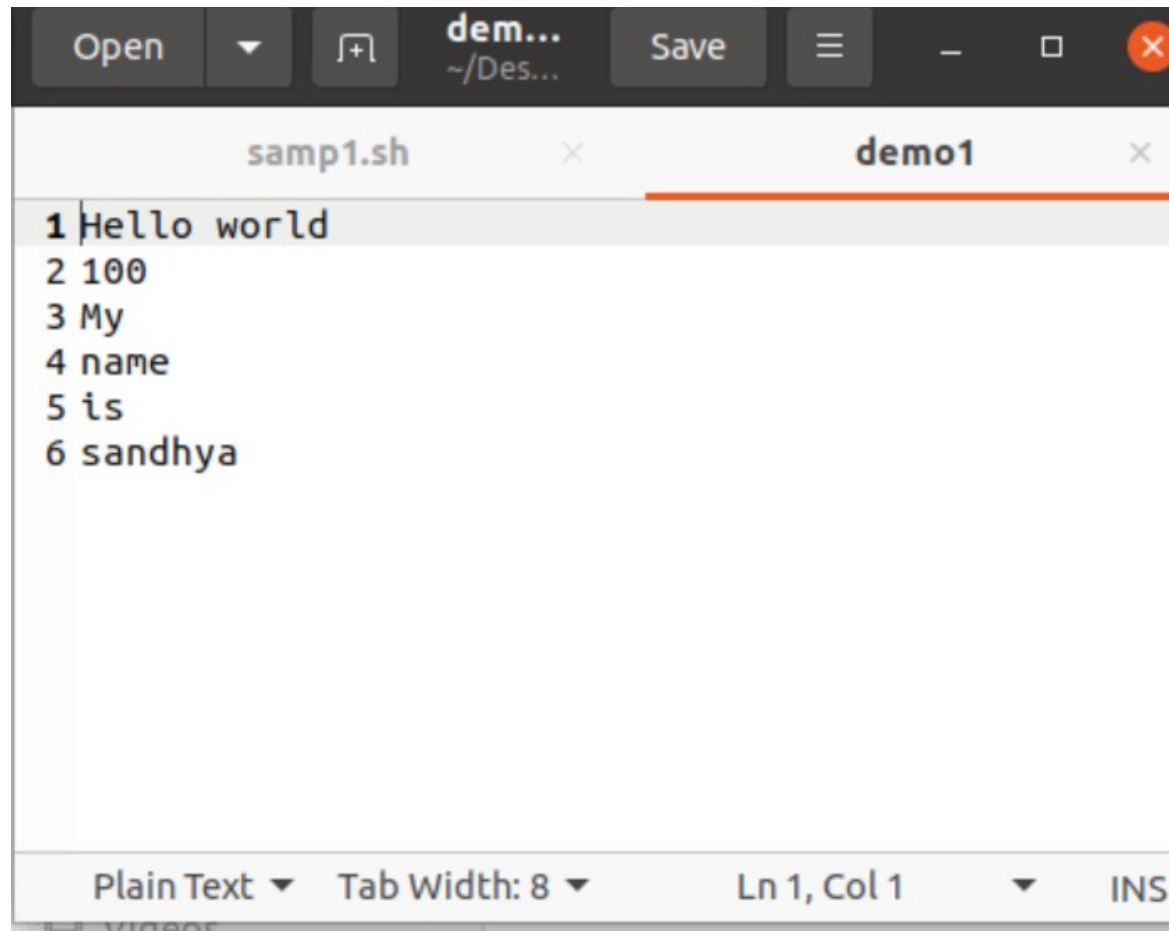


The image shows two windows side-by-side. The left window is a code editor with a dark theme. It has a menu bar with 'Open', a dropdown arrow, a file icon, a tab labeled 'sam...' with a path '~ / Des...', a 'Save' button, and a hamburger menu. The editor contains five lines of shell commands:
1 `echo "Hello world" >>demo1`
2 `x=100`
3 `echo $x >>demo1`
4 `echo -e "My \nname \nis \nsandhya" >>demo1`
5
The right window is a terminal with a dark purple background. The prompt is `user@user-virtual-machine: ~/...`. The terminal shows a sequence of commands:
`user@user-virtual-machine:~/Desktop/prgm$ cd ..`
`user@user-virtual-machine:~/Desktop$ cd pres`
`user@user-virtual-machine:~/Desktop/pres$./samp1.sh`
`user@user-virtual-machine:~/Desktop/pres$` followed by a cursor.

```
1 echo "Hello world" >>demo1
2 x=100
3 echo $x >>demo1
4 echo -e "My \nname \nis \nsandhya" >>demo1
5

user@user-virtual-machine:~/Desktop/prgm$ cd ..
user@user-virtual-machine:~/Desktop$ cd pres
user@user-virtual-machine:~/Desktop/pres$ ./samp1.sh
user@user-virtual-machine:~/Desktop/pres$
```

Output of Samp1.sh

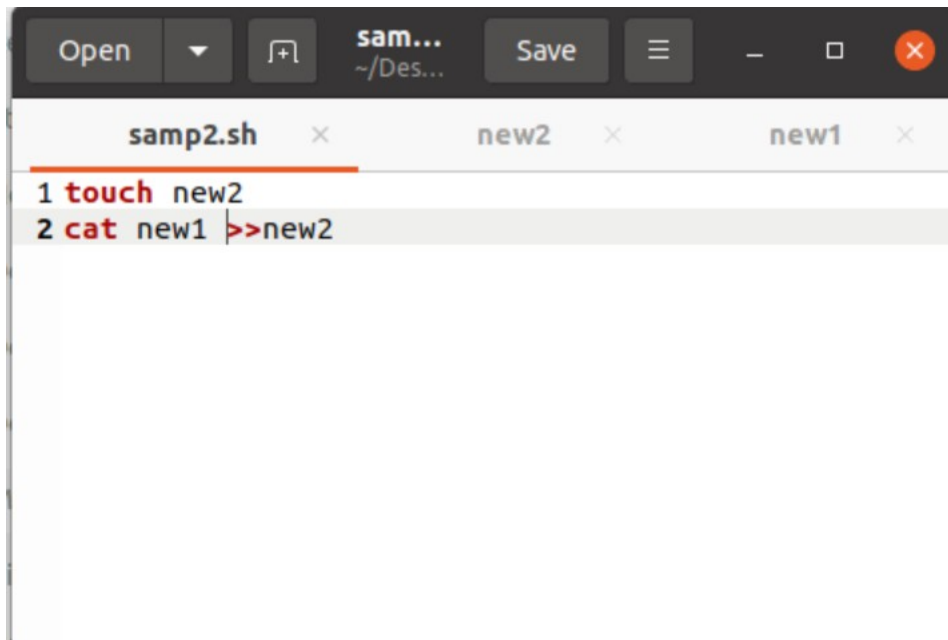


The image shows a screenshot of a code editor window. The window has a dark-themed title bar with buttons for 'Open', a dropdown menu, a 'dem...' button with a path '~ /Des...', a 'Save' button, and window control icons (minimize, maximize, close). Below the title bar, there are two tabs: 'samp1.sh' and 'demo1'. The 'demo1' tab is active and highlighted with an orange underline. The editor area shows the following text:

```
1 Hello world
2 100
3 My
4 name
5 is
6 sandhya
```

At the bottom of the editor, there is a status bar with the following information: 'Plain Text' (with a dropdown arrow), 'Tab Width: 8' (with a dropdown arrow), 'Ln 1, Col 1' (with a dropdown arrow), and 'INS'.

Samp2.sh



A screenshot of a terminal window with a dark theme. The title bar shows 'sam...' and the path '~/Des...'. The window has three tabs: 'samp2.sh', 'new2', and 'new1'. The 'samp2.sh' tab is active and contains two lines of code: '1 touch new2' and '2 cat new1 |>>new2'.

```
1 touch new2
2 cat new1 |>>new2
```

new1.txt

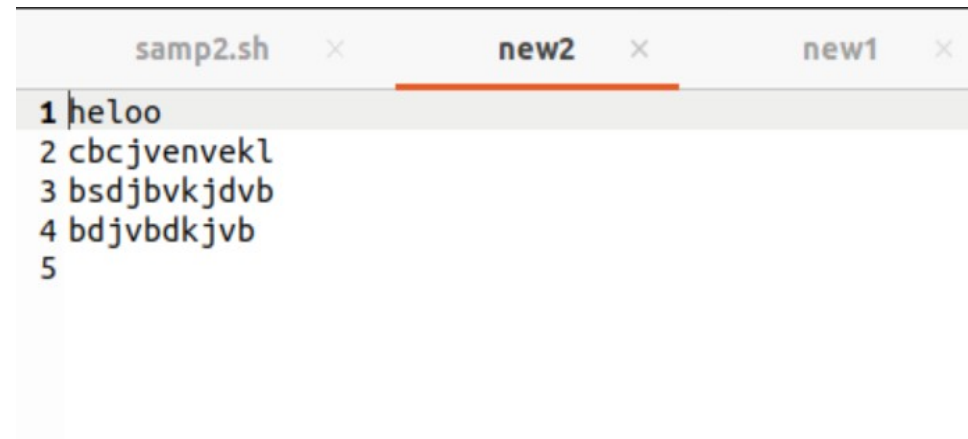


A screenshot of a terminal window with a light theme. The window has three tabs: 'samp2.sh', 'new2', and 'new1'. The 'new1' tab is active and contains five lines of text: '1 heloo', '2 cbcjvenvekl', '3 bsdjbvkjdvb', '4 bdjvbdkjvb', and '5'.

```
1 heloo
2 cbcjvenvekl
3 bsdjbvkjdvb
4 bdjvbdkjvb
5
```


new2.txt

```
user@user-virtual-machine:~/Desktop/pres$ ./samp2.sh
user@user-virtual-machine:~/Desktop/pres$
```



The screenshot shows a terminal window with three tabs: 'samp2.sh', 'new2', and 'new1'. The 'new2' tab is active and highlighted with an orange underline. The content of the 'new2' tab is as follows:

```
1 heloo
2 cbcjvenvekl
3 bsdjbvkjdvb
4 bdjvbdkjvb
5
```

Outputs on console

Using echo

Echo "string"

Ex:echo -e "hello \tbatch 86 \ti \tam \tinux" -

-e = Use dto detect \ and variable.

\t- = Used to add tab.

\n = Used to print output on new slide.

\v = It gives vertical tab

\b = It is used to remove spaces.

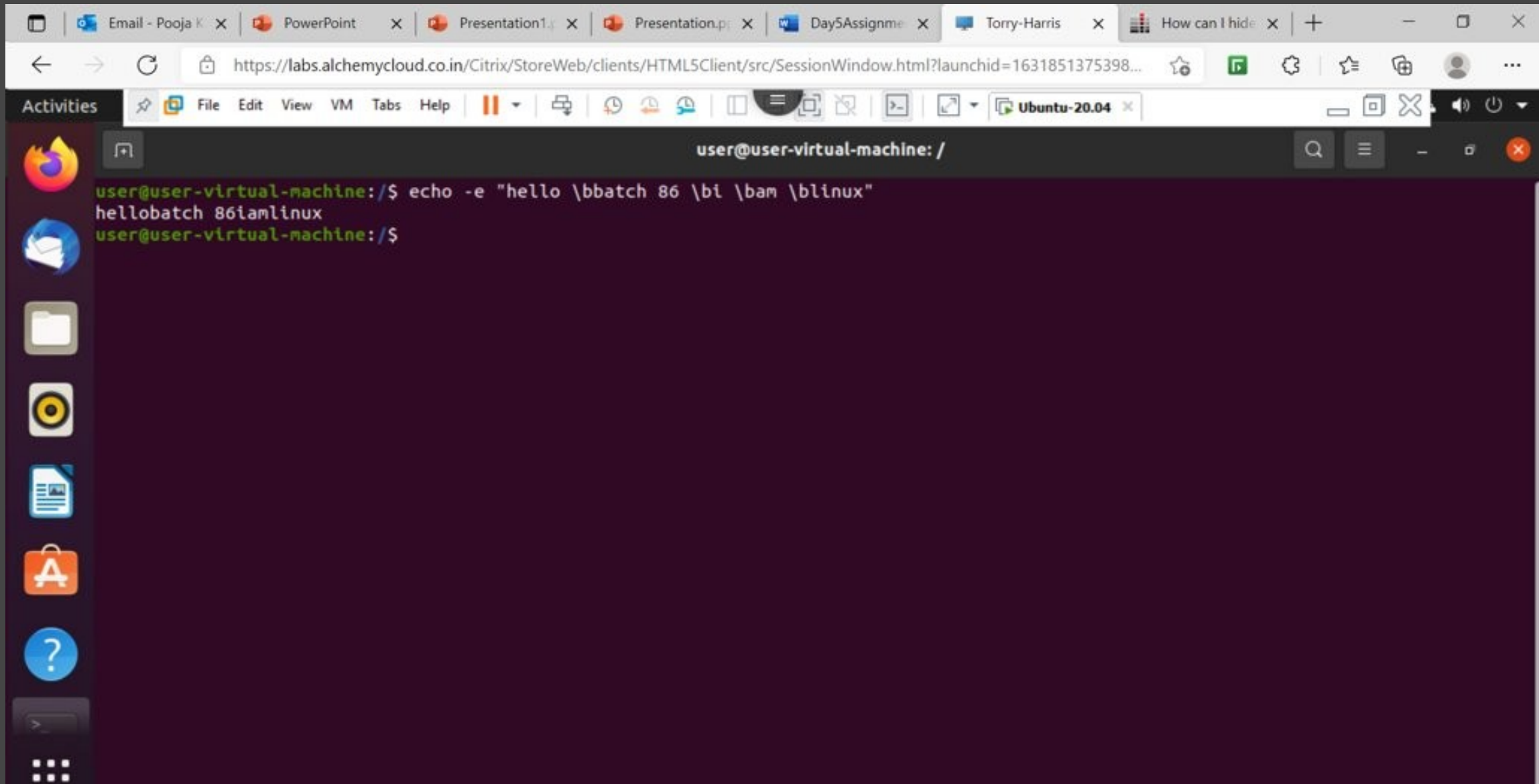
Using printf

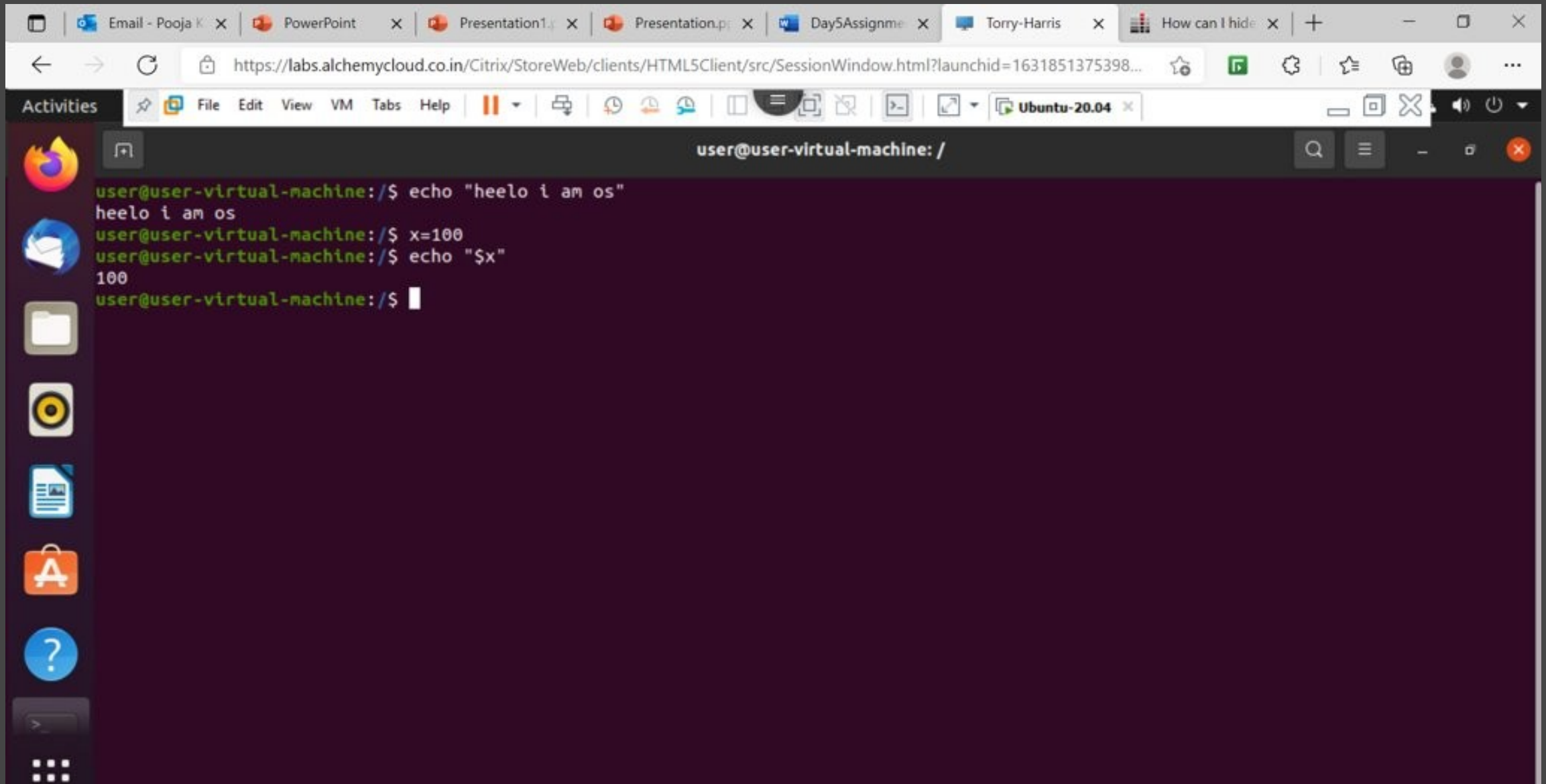
Printf "string"

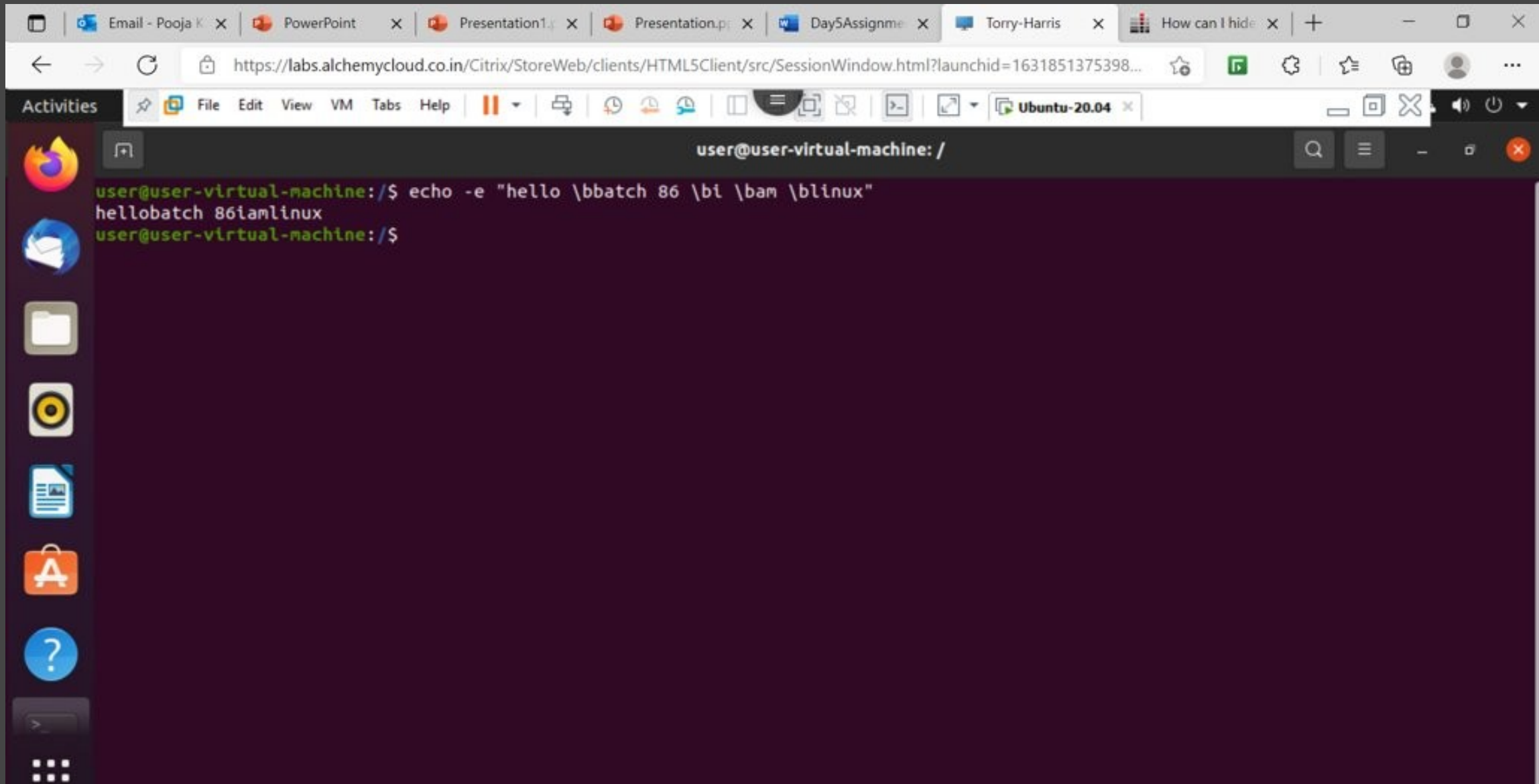
Ex:Printf "%d \n" 2.55 0xff 0324 3.5

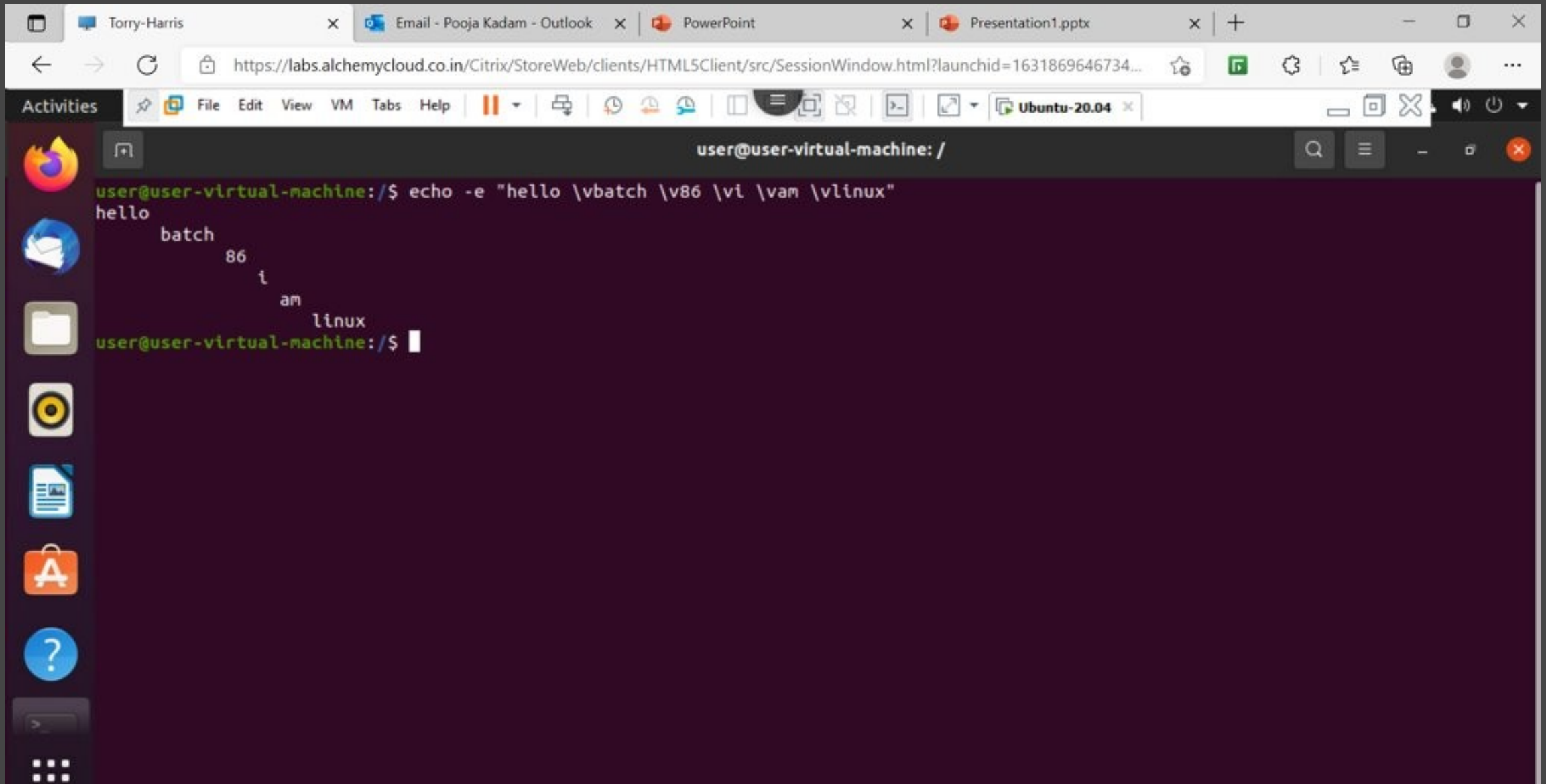
%d = It is used to print integer value.

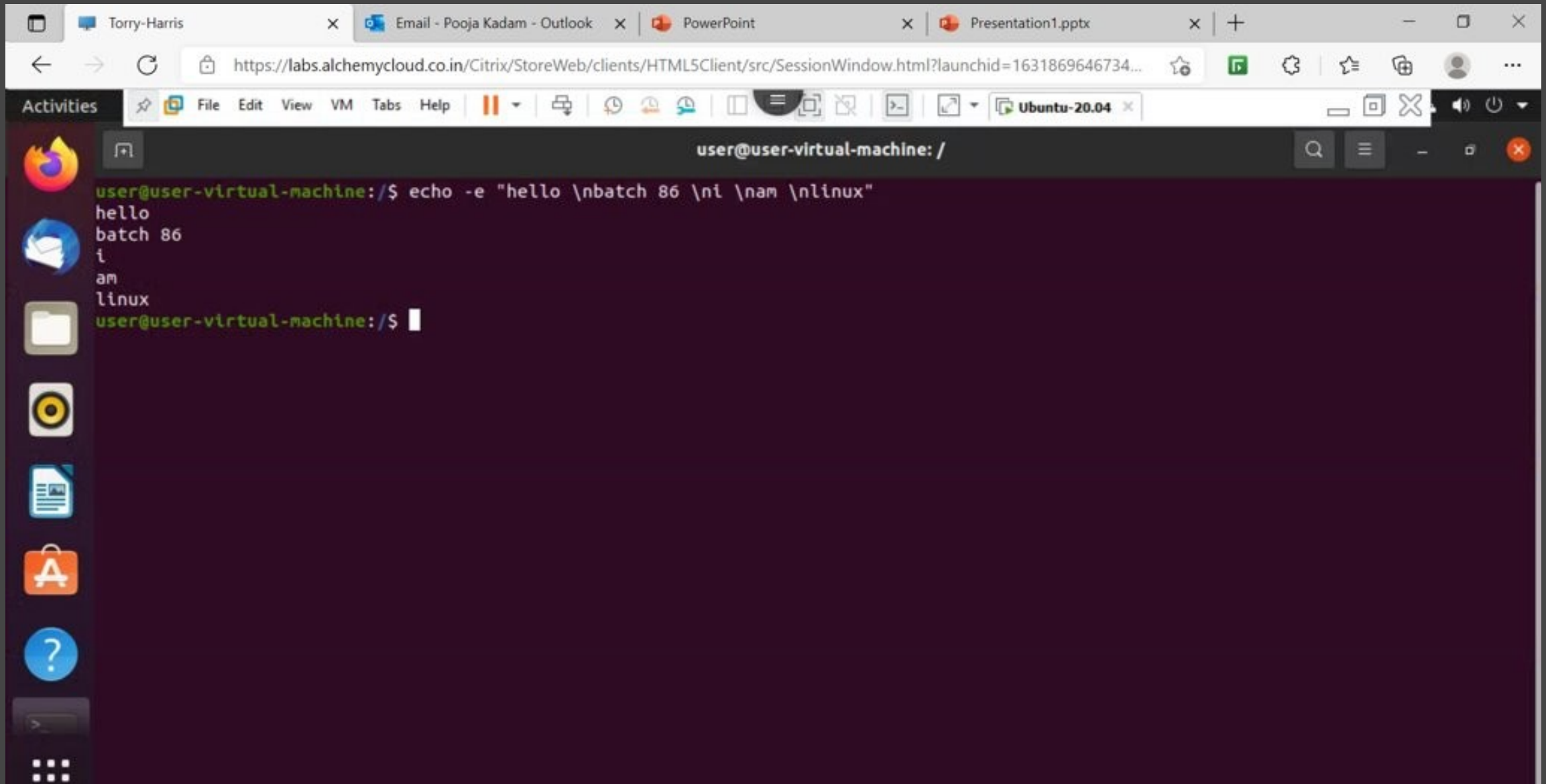
%f = it is used to print float value.

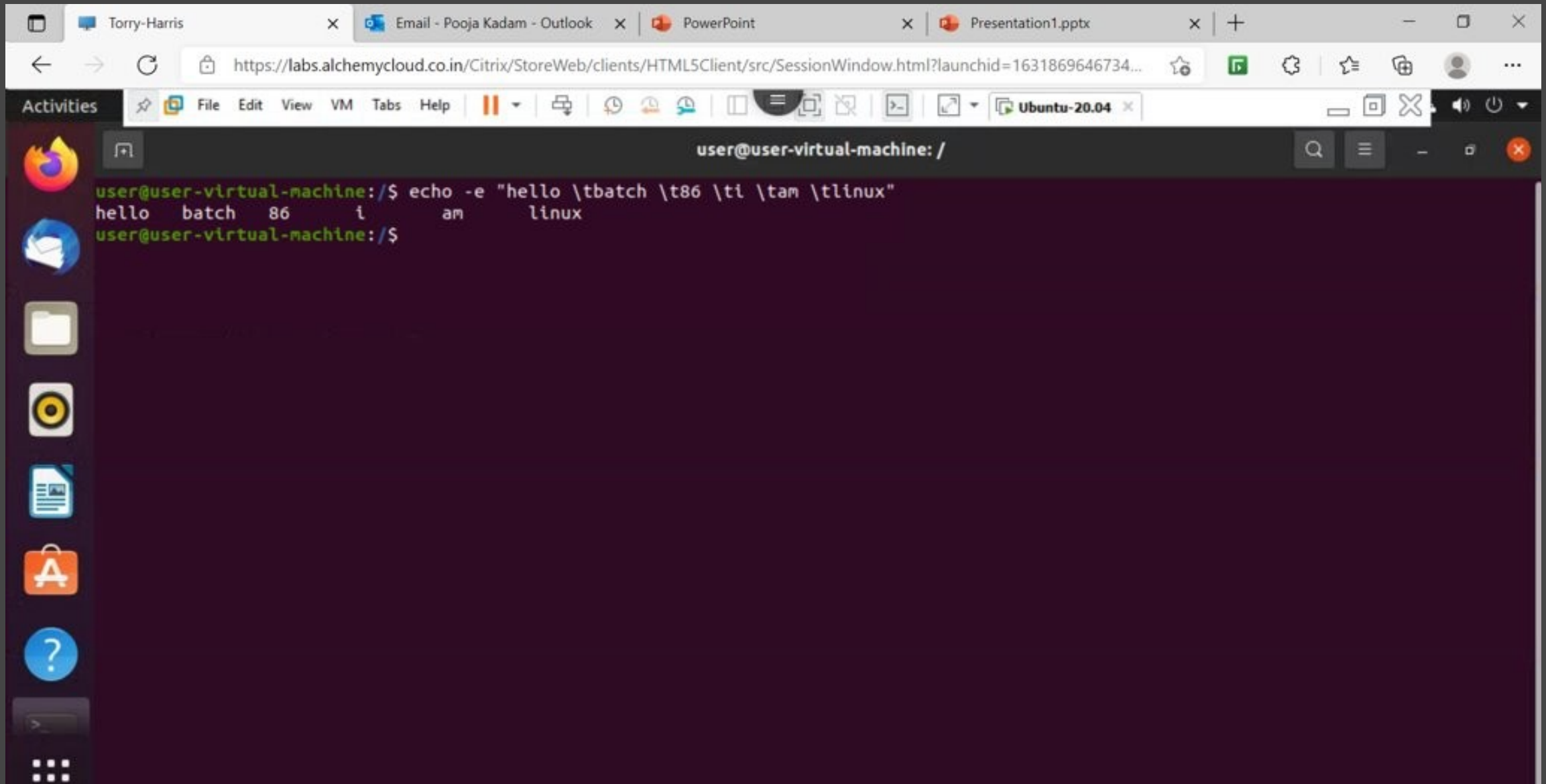












Activities

user data.sh

Trash data1.sh

demo append

Demo1 program1.txt

mybash.sh data2.sh

the
os
user@user-virtual-machine:~\$ printf "hello"
hellouser@user-virtual-machine:~\$ printf "hello"
hellouser@user-virtual-machine:~\$ printf "%d \n" 2.33 ofd 04343 346.6
bash: printf: 2.33: invalid number
2
bash: printf: ofd: invalid number
0
2275
bash: printf: 346.6: invalid number
346
user@user-virtual-machine:~\$ printf "%f \n" 2.33 ofd 04343 346.6
2.330000
bash: printf: ofd: invalid number
0.000000
4343.000000
346.600000
user@user-virtual-machine:~\$ printf "%c \n" 2.33 ofd 04343 346.6
2
o
o
3
user@user-virtual-machine:~\$