john@john-VirtualBox:~/My-Bash-Script$ script\_name="My First Bash Script"

john@john-VirtualBox:~/My-Bash-Script$ echo "My First Bash Script"

My First Bash Script

john@john-VirtualBox:~/My-Bash-Script$ echo "$script\_name"

My First Bash Script

john@john-VirtualBox:~/My-Bash-Script$ greeting="What's up?"

john@john-VirtualBox:~/My-Bash-Script$ echo "$greeting"

What's up?

john@john-VirtualBox:~/My-Bash-Script$ echo -n "What is your name?: "

What is your name?: john@john-VirtualBox:~/My-Bash-Script$ read user\_name

john@john-VirtualBox:~/My-Bash-Script$ echo "Hello, $user\_name! Glad you're here."

Hello, ! Glad you're here.

john@john-VirtualBox:~/My-Bash-Script$ echo -n "What's your favorite color? "

What's your favorite color? john@john-VirtualBox:~/My-Bash-Script$ read favorite\_color

john@john-VirtualBox:~/My-Bash-Script$ echo "$user\_name, your favorite color is $favorite\_color."

, your favorite color is .

john@john-VirtualBox:~/My-Bash-Script$ echo "$user\_name, your favorite color is $favorite\_color."

, your favorite color is .

john@john-VirtualBox:~/My-Bash-Script$ echo "Blue is a great choice!"

Blue is a great choice!

john@john-VirtualBox:~/My-Bash-Script$ elif [ "$favorite\_color" == "green" ]; then

bash: syntax error near unexpected token `elif'

john@john-VirtualBox:~/My-Bash-Script$ elif [ "$favorite\_color" == "green" ]; then echo "Green is the color of nature."

bash: syntax error near unexpected token `elif'

john@john-VirtualBox:~/My-Bash-Script$ elif [ "$favorite\_color" == "green" ]; then echo "Green is the color of nature." else echo "That's an interesting color choice!" fi

bash: syntax error near unexpected token `elif'

john@john-VirtualBox:~/My-Bash-Script$ ^C

john@john-VirtualBox:~/My-Bash-Script$ elif

bash: syntax error near unexpected token `elif'

john@john-VirtualBox:~/My-Bash-Script$ echo "Counting to 5 with a for loop:"

Counting to 5 with a for loop:

john@john-VirtualBox:~/My-Bash-Script$ for i in 1 2 3 4 5; do

> echo "Number: $i" done

> done

Number: 1 done

Number: 2 done

Number: 3 done

Number: 4 done

Number: 5 done

john@john-VirtualBox:~/My-Bash-Script$ echo "Counting down from 3 with a while loop:"

Counting down from 3 with a while loop:

john@john-VirtualBox:~/My-Bash-Script$ count=3

while [ $count -gt 0 ]; do echo "Count: $count"

((count--))

> done

Count: 3

Count: 2

Count: 1

john@john-VirtualBox:~/My-Bash-Script$ echo "Blast off!"

Blast off!

john@john-VirtualBox:~/My-Bash-Script$ fruits=("Apple" "Banana" "Cherry" "Strawberry")

john@john-VirtualBox:~/My-Bash-Script$ echo "My favorite fruits are: ${fruits[@]}"

My favorite fruits are: Apple Banana Cherry Strawberry

john@john-VirtualBox:~/My-Bash-Script$ echo "The first fruit is: ${fruits[0]}"

The first fruit is: Apple

john@john-VirtualBox:~/My-Bash-Script$ num1=10

num2=5

john@john-VirtualBox:~/My-Bash-Script$ sum=$((num1 + num2))

john@john-VirtualBox:~/My-Bash-Script$ echo "Sum of $num1 and $num2 is: $sum"

Sum of 10 and 5 is: 15

john@john-VirtualBox:~/My-Bash-Script$ difference=$((num1 - num2))

john@john-VirtualBox:~/My-Bash-Script$ ))

echo "Difference of $num1 and $num2 is: $difference"

bash: syntax error near unexpected token `)'

Difference of 10 and 5 is: 5

john@john-VirtualBox:~/My-Bash-Script$ product=$((num1 \* num2))

john@john-VirtualBox:~/My-Bash-Script$ echo "Product of $num1 and $num2 is: $product"

Product of 10 and 5 is: 50

john@john-VirtualBox:~/My-Bash-Script$ quotient=$((num1 / num2))

john@john-VirtualBox:~/My-Bash-Script$ echo "Quotient of $num1 and $num2 is: $quotient"

Quotient of 10 and 5 is: 2

john@john-VirtualBox:~/My-Bash-Script$ remainder=$((num1 % 3))

john@john-VirtualBox:~/My-Bash-Script$ echo "Remainder of $num1 divided by 3 is: $remainder"

Remainder of 10 divided by 3 is: 1

john@john-VirtualBox:~/My-Bash-Script$ data\_file="data.txt"

john@john-VirtualBox:~/My-Bash-Script$ echo "This is the first line." > "$data\_file"

john@john-VirtualBox:~/My-Bash-Script$ echo "This is the first line." > "$data\_file"

john@john-VirtualBox:~/My-Bash-Script$ echo "This is the second line." >> "$data\_file"

john@john-VirtualBox:~/My-Bash-Script$ echo "And a third line." >> "$data\_file"

john@john-VirtualBox:~/My-Bash-Script$ echo "Content of '$data\_file':"

Content of 'data.txt':

john@john-VirtualBox:~/My-Bash-Script$ cat "$data\_file"

This is the first line.

This is the second line.

And a third line.

john@john-VirtualBox:~/My-Bash-Script$ if [ -s "$data\_file" ]; then

> echo "'$data\_file' is not empty."

> else

echo "'$data\_file' is empty."

> fi

'data.txt' is not empty.

john@john-VirtualBox:~/My-Bash-Script$ rm "$data\_file"

john@john-VirtualBox:~/My-Bash-Script$ echo "Removed '$data\_file'."

Removed 'data.txt'.

john@john-VirtualBox:~/My-Bash-Script$ echo "Processing .txt files:"

Processing .txt files:

john@john-VirtualBox:~/My-Bash-Script$ for file in \*.txt; do

> echo " Found: $file"

> echo "Added line to $file" >> "$file"

> done

Found: \*.txt

john@john-VirtualBox:~/My-Bash-Script$ PS3="Choose an option: "

john@john-VirtualBox:~/My-Bash-Script$ options=("Option 1" "Option 2" "Exit")

john@john-VirtualBox:~/My-Bash-Script$ case "$choice" in

> "Option 1")

> echo "You chose Option 2."

> ;;

"Exit")

> echo "Exiting script."

> break

> ;;

> \*)

> echo "Invalid option: $REPLY" ;;

> esac

Invalid option:

john@john-VirtualBox:~/My-Bash-Script$ #!/bin/bash

for i in {1..50}; do

echo "$i"

done

1

2

3

4

5

6

7

8

9

10

11

12

13

14

15

16

17

18

19

20

21

22

23

24

25

26

27

28

29

30

31

32

33

34

35

36

37

38

39

40

41

42

43

44

45

46

47

48

49

50

john@john-VirtualBox:~/My-Bash-Script$ chmod +x count.sh

chmod: cannot access 'count.sh': No such file or directory

john@john-VirtualBox:~/My-Bash-Script$ chmod +x My-Bash-Script

chmod: cannot access 'My-Bash-Script': No such file or directory

john@john-VirtualBox:~/My-Bash-Script$

john@john-VirtualBox:~/My-Bash-Script$

john@john-VirtualBox:~/My-Bash-Script$ chmod +x My-Bash-Script

chmod: cannot access 'My-Bash-Script': No such file or directory

john@john-VirtualBox:~/My-Bash-Script$ ls

'name=My-Bash-Script' README.md '\*.txt'

john@john-VirtualBox:~/My-Bash-Script$ chmod +x My-Bash-Script

chmod: cannot access 'My-Bash-Script': No such file or directory

john@john-VirtualBox:~/My-Bash-Script$