# Practice questions for shell scripting

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

   → SCRIPT

```
#!/usr/bin/bash

echo " HEllo , $LOGNAME "
echo "Current date is `date` "
echo " User is `whoami`"
echo " Current directory is `pwd`"
~
~
```

   →output

```
[root@localhost practice_questions]# ./question1.sh
 HEllo , root
Current date is Wed Sep  7 01:15:58 IST 2022
 User is root
 Current directory is /root/Shell_scripting/practice_questions
[root@localhost practice_questions]#
```

2. Write a script to print the reverse of the input number
   - Example :- script_name 1234 Output:- 4321
   - The script should take only one parameter as an input
   - If the number is less than zero scripts should say “Input is less than 0, retry with a different number.”
   - If no parameter is passed while running the script it should prompt "ERROR: Retry with one parameter."

→ Script

```
#!/usr/bin/bash

if [ $# -eq 1 ]; then
    if [ $1 -gt 0 ]; then
        num=$1
        revNum=0
        while [ $num -ne 0 ]
        do
            testnum=$(( $num %  10 ))
            revNum=$(( $revNum * 10 + $testnum ))
            num=$(( $num / 10 ))
        done
        echo "Reverse Number:  $revNum of $1"
    else
        echo "Input is less than 0, retry with a different number."
    fi
else
    echo "ERROR: Retry with one parameter."
fi
~
~
~
```

→ Output

```
[root@localhost practice_questions]# ./reverse.sh 123456
Reverse Number:  654321 of 123456
[root@localhost practice_questions]# ./reverse.sh
ERROR: Retry with one parameter.
[root@localhost practice_questions]# ./reverse.sh 123 456 4
ERROR: Retry with one parameter.
[root@localhost practice_questions]#
```

**3.** 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 an ls command against the file or directory with the long listing option. 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 an ls command against the file or directory with the long listing option.

**→ Script**

```
#!/bin/bash
echo "Enter the file path"
read FILE
if [ -f "$FILE" ]
  then
    echo "$FILE is a reguler file"
elif [ -d "$FILE" ]
  then
    echo "$FILE is a directory"
else
    echo "$FILE is another type of file"
fi
ls -l $FILE
~
~
```

**→Output**

```
[root@localhost Shell_SCripting]# ./question3.sh
"Enter the file path"
/root
"/root is another type of file"
total 4
-rw-------. 1 root root 1056 Sep  8 18:23 anaconda-ks.cfg
drwxr-xr-x. 2 root root    6 Sep  8 20:38 Desktop
drwxr-xr-x. 2 root root    6 Sep  8 20:38 Documents
drwxr-xr-x. 2 root root   44 Sep  9 00:46 Downloads
drwxr-xr-x. 2 root root    6 Sep  8 20:38 Music
drwxr-xr-x. 2 root root    6 Sep  8 20:38 Pictures
drwxr-xr-x. 2 root root    6 Sep  8 20:38 Public
drwxr-xr-x. 2 root root   26 Sep  9 20:26 Shell_SCripting
drwxr-xr-x. 2 root root    6 Sep  8 20:38 Templates
drwxr-xr-x. 2 root root    6 Sep  8 20:38 Videos
[root@localhost Shell_SCripting]#
```

4. Write a shell script that consists of a function that displays the number of files in the present working directory. Name this function “file_count” and call it in your script. If you use a variable in your function, remember to make it a local variable.

**→Script**

```
#!/bin/bash
function file_count()
 {
   local NUMBER_OF_FILE=$(ls -l | wc -l)
    echo "$NUMBER_OF_FILE"
 }
file_count
~
~
~
~
~
```

**→ Output**

```
[root@localhost Shell_SCripting]# ./question4.sh
3
[root@localhost Shell_SCripting]# S
```

5. Write a script that takes two numbers from the user and asks choice for addition, subtraction, multiplication & division of two numbers and returns the output of it

**→Script**

```
#!/usr/bin/bash
echo "1) Addition "
echo "2) Multiplication "
echo "3) Division "
echo "4) Substraction"
read -p  "Enter your choice : " choice
if `[ $choice -eq 1 ]`
then
        read -p "Enter first number " num1
        read -p "Enter second number " num2
        exp=$(($num1 + $num2))
        echo "Addition is $exp "
elif `[ $choice -eq 2 ]`
then
        read -p "Enter first number " num1
        read -p "Enter second number " num2
        exp=$(($num1 * $num2 ))
        echo "Mutliplication of two number is $exp "
elif `[ $choice -eq 3 ]`
then
        read -p "Enter first number " num1
        read -p "Enter second number " num2
        exp=$(($num1 / $num2 ))
                                                        1,1
```

```
elif `[ $choice -eq 4 ]`
then
        read -p "Enter first number " num1
        read -p "Enter second number " num2
        exp=$(($num1 * $num2 ))
        echo "Substraction of two number is $exp "
else
        echo " ENTER Valid Option"
fi
```

→**Output**

```
[root@localhost Shell_SCripting]# ./question5.sh
1) Addition
2) Multiplication
3) Division
4) Substraction
Enter your choice : 1
Enter first number 10
Enter second number 5
Addition is 15
[root@localhost Shell_SCripting]#
```

6. Write a script to take input as a number & check if it is an even or odd number

→**Script**

```
#!/usr/bin/bash

read -p "Enter a number: " number
if [ $((number%2)) -eq 0 ]
then
  echo "Number is even."
else
  echo "Number is odd."
fi
~
~
~
~
```

→**Output**

```
[root@localhost Shell_SCripting]# vim question6.sh
[root@localhost Shell_SCripting]# ./question6.sh
Enter a number: 6
Number is even.
[root@localhost Shell_SCripting]# ./question6.sh
Enter a number: 7
Number is odd.
[root@localhost Shell_SCripting]#
```

**7.** Write a script that takes two numbers from the user and swaps its

**→Script**

```
#!/usr/bin/bash

read -p "Enter first number " num1
read -p "Enter second number " num2

echo "Before Swapping"
echo "First number: $num1"
echo "Second number: $num2"
temp=$num1
num1=$num2
num2=$temp
echo "After Swapping"
echo "First number: $num1"
echo "Second number: $num2"
~
~
~
```

**→Output**

```
[root@localhost Shell_SCripting]# ./question7.sh
Enter first number 9
Enter second number 27
Before Swapping
First number: 9
Second number: 27
After Swapping
First number: 27
Second number: 9
[root@localhost Shell_SCripting]#
```