CLI RESEARCH

Question -1

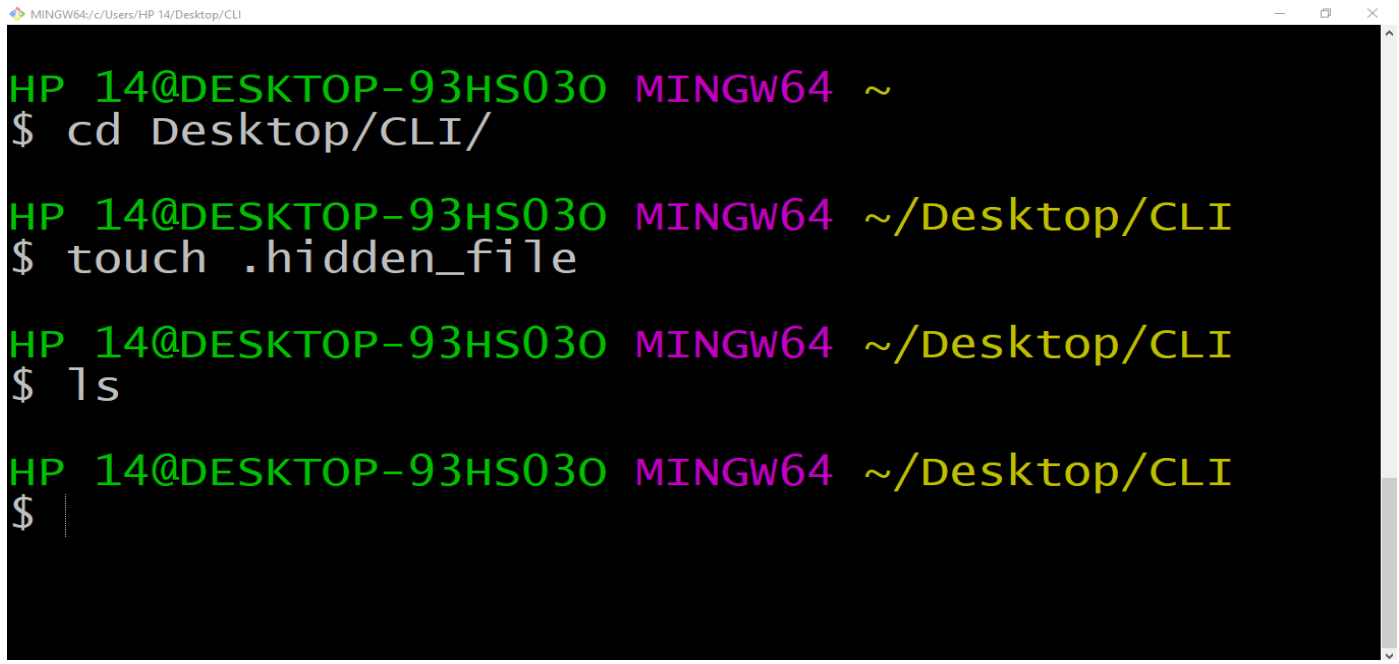
''How to create a hidden file and how to display it
using 'ls' command?''

CLI RESEARCH

ANSWER

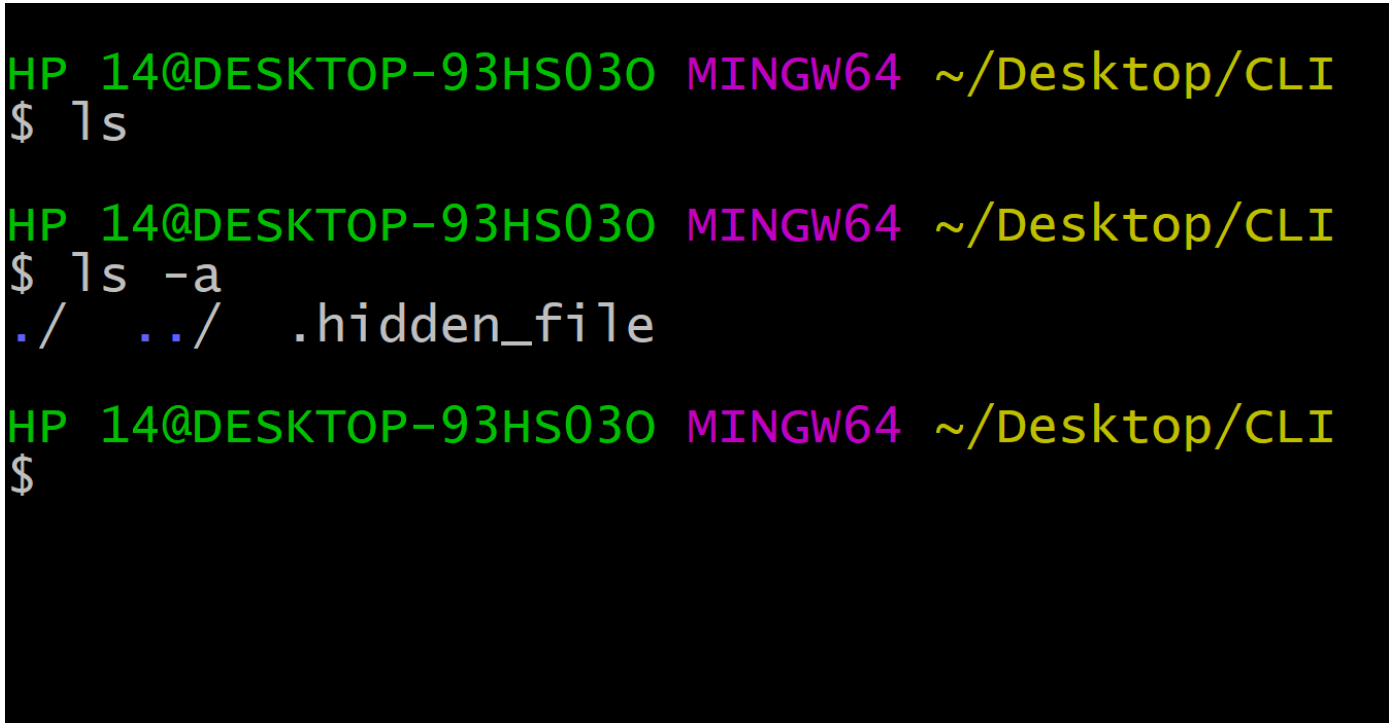
Files starting with a dot are by default hidden.

As you can see in the example below, we are in CLI directory which is located in Desktop. We created a file starting with a dot(.) and ls command couldn't list it.



CLI RESEARCH

In these cases, we have another commanda that we can use alongside with ls; -a command. With these commands we can display all the files alongside with the hidden ones. In the example below'you can see that we managed to list the hidden folders using -a command.



CLI RESEARCH

The -a listing will give us our solution but if we want files to be properly hidden, we have to change folder permissions using chmod.

Chmod= It's name is an abbreviation for 'change mode'. In Unix and Unix-like operating systems, **chmod** is the command and system call which is used to change the access permissions of file system objects (files and directories).

CLI RESEARCH

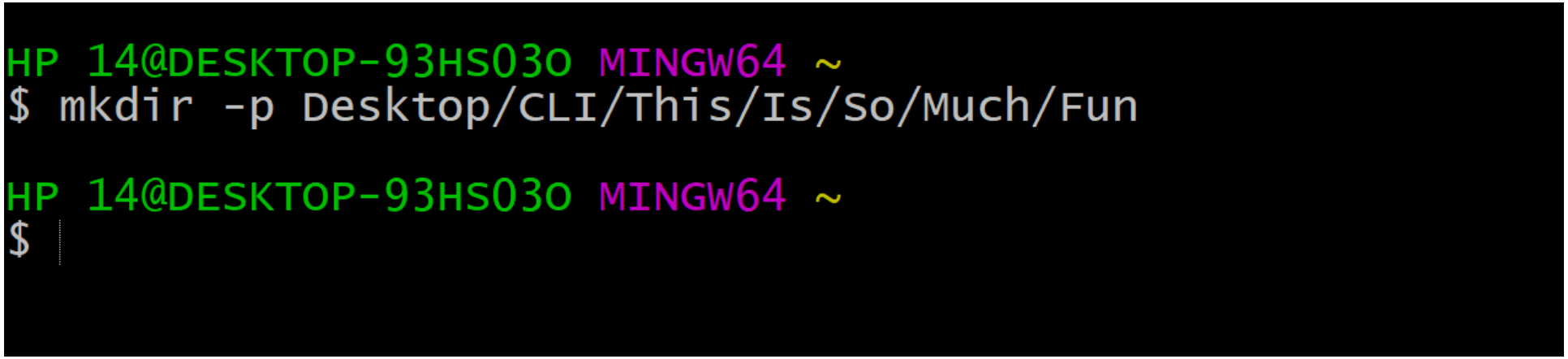
Question – 2

"How to create multiple nested directories with one mkdir commad?"

CLI RESEARCH

ANSWER

First of all if we want to create a new directory, 'mkdir' command is inevitable. So, each time we want to create a new directory we are going to need it. By adding it -p flag, we can create multiple nested directories. Like this:



CLI RESEARCH

Question – 3

Execute following commands terminal:

```
echo "test" > fun

echo "another test" >> fun

wc -c fun
```

Verify number of characters in

CLI RESEARCH

ANSWER

So, when we execute the given commands the result would be just like there below;

```
HP 14@DESKTOP-93HS03O MINGW64 ~/Desktop
$ echo "test" > fun

HP 14@DESKTOP-93HS03O MINGW64 ~/Desktop
$ echo "another test" >> fun

HP 14@DESKTOP-93HS03O MINGW64 ~/Desktop
$ wc -c fun
18 fun

HP 14@DESKTOP-93HS03O MINGW64 ~/Desktop
$
```

CLI RESEARCH

Command wc is used to count the number of lines, words, and bytes in the files specified by the *File* parameter.

So without using the -c flag, the output would be like this;

```
HP 14@DESKTOP-93HS03O MINGW64 ~/Desktop
$ wc -c fun
18 fun

HP 14@DESKTOP-93HS03O MINGW64 ~/Desktop
$ wc fun
 2  3 18 fun
```

The first result indicates the number of lines (which is 2 in this case)

Second result is the number of words in my folder (test - another - test) and **the third one** screens the bytes of my file.

To comprehend it properly, let just give another similar example. In this example we inserted another string to another file.

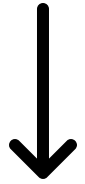
```
HP 14@DESKTOP-93HS03O MINGW64 ~
$ cd Desktop/

HP 14@DESKTOP-93HS03O MINGW64 ~/Desktop
$ echo "i-m still trying to understand this" > newfile

HP 14@DESKTOP-93HS03O MINGW64 ~/Desktop
$ wc newfile
 1  6 36 newfile

HP 14@DESKTOP-93HS03O MINGW64 ~/Desktop
$
```

Here in our new file, we have just 1 line 6 words (i-m counted as 1 word naturally) and 36 bytes. In order to be completely sure, we took a look at the file's properties.



CLI RESEARCH

For variate wc results, you can use the following flags;

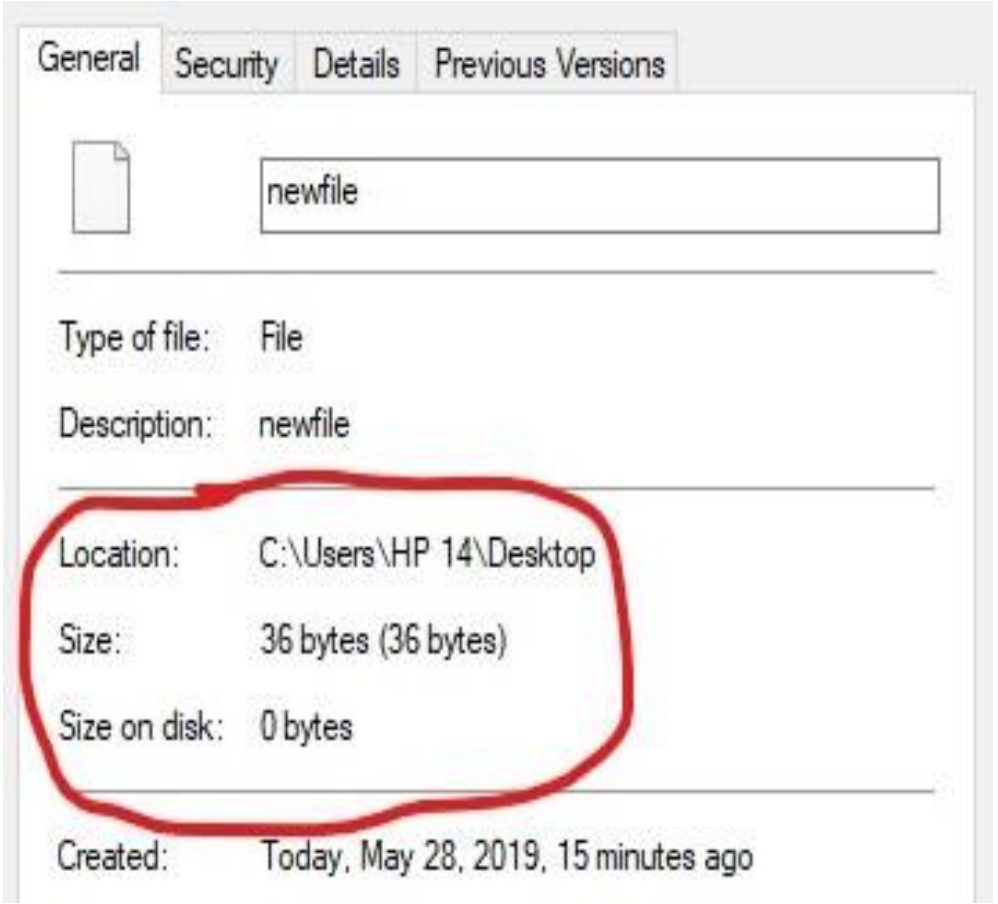
wc -l : Prints the number of lines in a file.

wc -w : prints the number of words in a file.

wc -m : prints the count of characters from a file.

wc -L : prints only the length of the longest line in a file.

wc -c :
Displays the count of bytes in a file



CLI RESEARCH

QUESTION - 4

Research how to append something in the file *WITHOUT* a newline character.

E.g.

```
echo "test" > fun

echo "another test" >> fun

cat fun
```

Output of "cat fun" is

```
test
```

CLI RESEARCH

Answer

Normally, when we append something in the file, terminal will print them in a new line. Like below here:

```
HP 14@DESKTOP-93HS03O MINGW64 ~/Desktop
$ cat fun
test
another test
```

So by adding -n parameter we can actually prevent this situation.

By adding -n parameter to echo command and starting with a space to our new input, we can actually continue to write our content on the same line.

```
HP 14@DESKTOP-93HS03O MINGW64 ~/Desktop
$ echo -n " test another test" >> fun

HP 14@DESKTOP-93HS03O MINGW64 ~/Desktop
$ cat fun
test
another test test another test
HP 14@DESKTOP-93HS03O MINGW64 ~/Desktop
$
```

Resources

https://www.tecmint.com/

https://stackoverflow.com/

https://stackexchange.com/