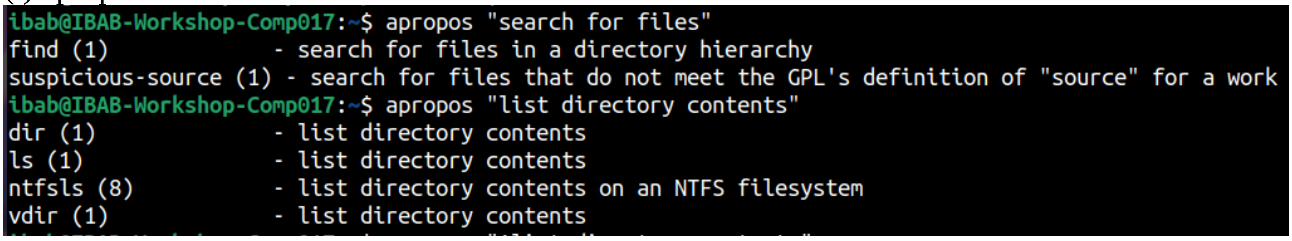
Lab 7

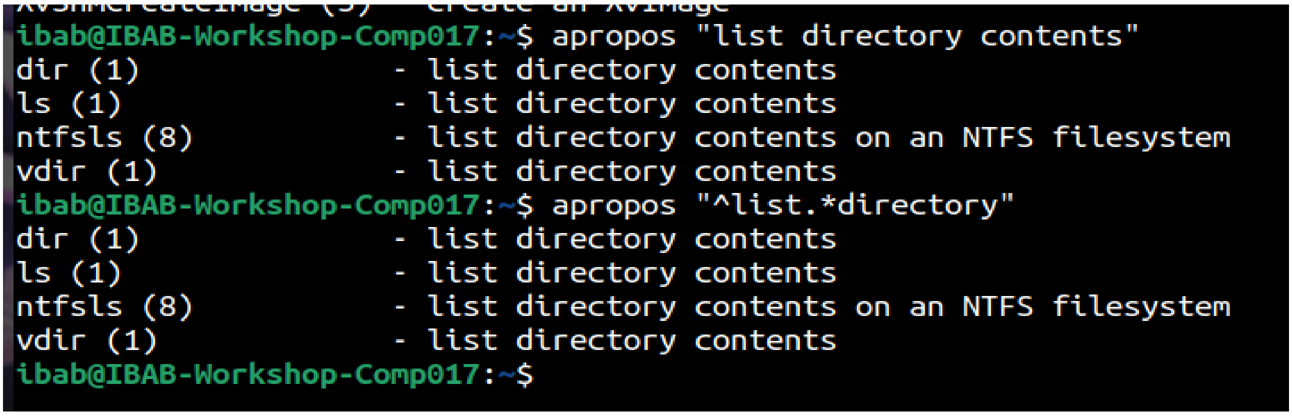
(1) The apropos command allows you to search the system’s manual pages for commands that match a specific task or keyword. This is particularly useful when you need to find a command that performs a specific task but you don’t know the name of the command. Try the following and display the output in your lab document.

(i) apropos "search for files"



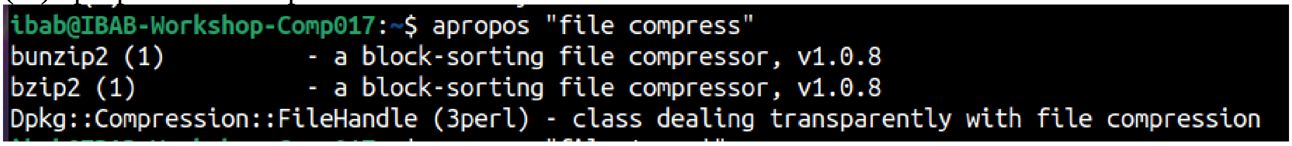
```
ibab@IBAB-Workshop-Comp017:~$ apropos "search for files"
find (1)             - search for files in a directory hierarchy
suspicious-source (1) - search for files that do not meet the GPL's definition of "source" for a work
ibab@IBAB-Workshop-Comp017:~$ apropos "list directory contents"
dir (1)              - list directory contents
ls (1)               - list directory contents
ntfsls (8)           - list directory contents on an NTFS filesystem
vdir (1)             - list directory contents
```

(ii) apropos "list directory contents". Another way to get the same output is apropos "^list.*directory" (note the use of wildcards here). The ^ denotes that there is no other character before the word ‘list’ in the search.



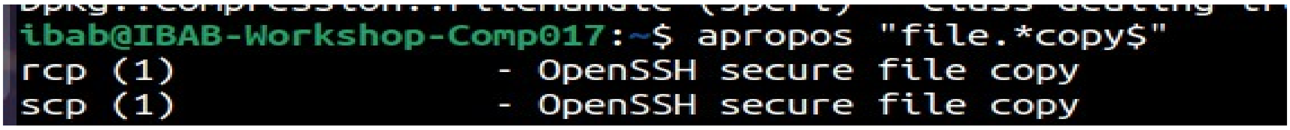
```
ibab@IBAB-Workshop-Comp017:~$ apropos "list directory contents"
dir (1)              - list directory contents
ls (1)               - list directory contents
ntfsls (8)           - list directory contents on an NTFS filesystem
vdir (1)             - list directory contents
ibab@IBAB-Workshop-Comp017:~$ apropos "^list.*directory"
dir (1)              - list directory contents
ls (1)               - list directory contents
ntfsls (8)           - list directory contents on an NTFS filesystem
vdir (1)             - list directory contents
ibab@IBAB-Workshop-Comp017:~$
```

(iii) apropos "file compress"



```
ibab@IBAB-Workshop-Comp017:~$ apropos "file compress"
bunzip2 (1)          - a block-sorting file compressor, v1.0.8
bzip2 (1)            - a block-sorting file compressor, v1.0.8
Dpkg::Compression::FileHandle (3perl) - class dealing transparently with file compression
```

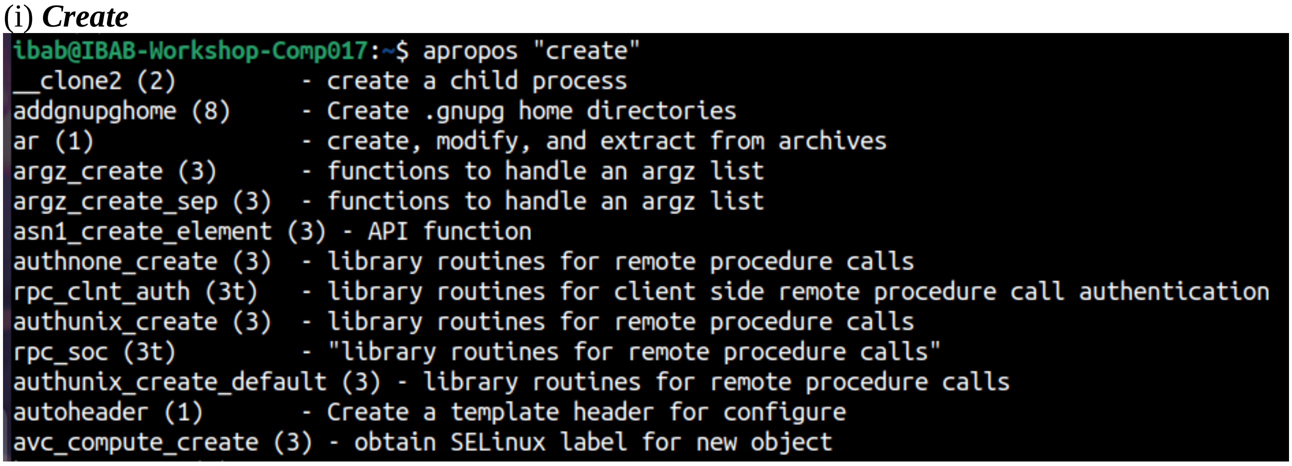
(iv) apropos "file.*copy$" (The $ denotes that there is no character to be searched after the word ‘copy’.
Come up with 3 more keywords/phrases that you can try for this command and got meaningful output (show the screenshots of all your outputs)



```
ibab@IBAB-Workshop-Comp017:~$ apropos "file.*copy$"
rcp (1)              - OpenSSH secure file copy
scp (1)              - OpenSSH secure file copy
```

3 more keywords /Phrases are

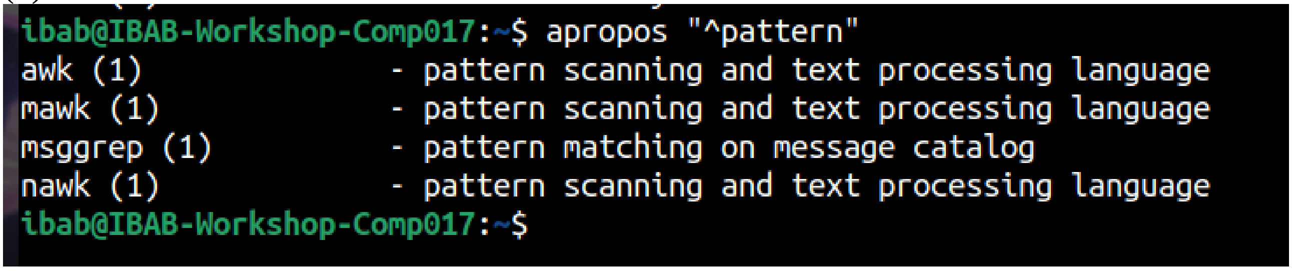
(i) ***Create***



```
ibab@IBAB-Workshop-Comp017:~$ apropos "create"
__clone2 (2)          - create a child process
addgnupghome (8)      - Create .gnupg home directories
ar (1)                - create, modify, and extract from archives
argz_create (3)       - functions to handle an argz list
argz_create_sep (3)   - functions to handle an argz list
asn1_create_element (3) - API function
authnone_create (3)   - library routines for remote procedure calls
rpc_clnt_auth (3t)    - library routines for client side remote procedure call authentication
authunix_create (3)   - library routines for remote procedure calls
rpc_soc (3t)          - "library routines for remote procedure calls"
authunix_create_default (3) - library routines for remote procedure calls
autoheader (1)        - Create a template header for configure
avc_compute_create (3) - obtain SELinux label for new object
```

.

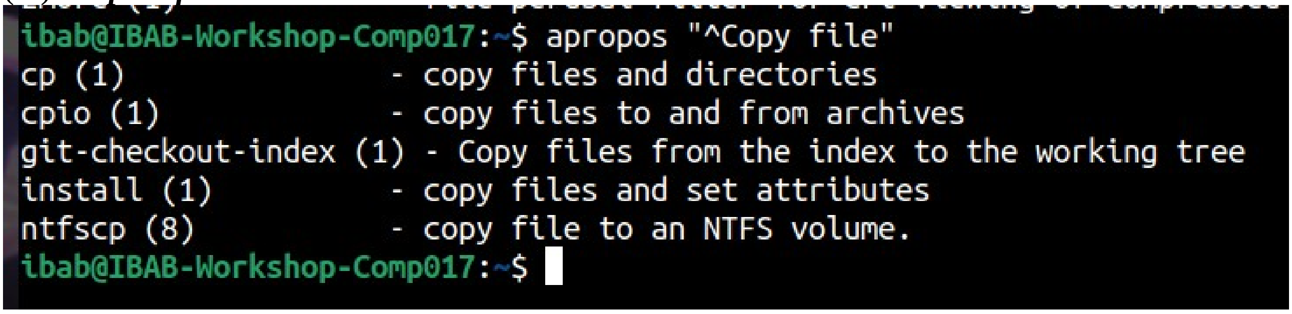
(ii) ***Pattern***



```
ibab@IBAB-Workshop-Comp017:~$ apropos "^pattern"
awk (1)              - pattern scanning and text processing language
mawk (1)             - pattern scanning and text processing language
msggrep (1)          - pattern matching on message catalog
nawk (1)             - pattern scanning and text processing language
ibab@IBAB-Workshop-Comp017:~$
```

.

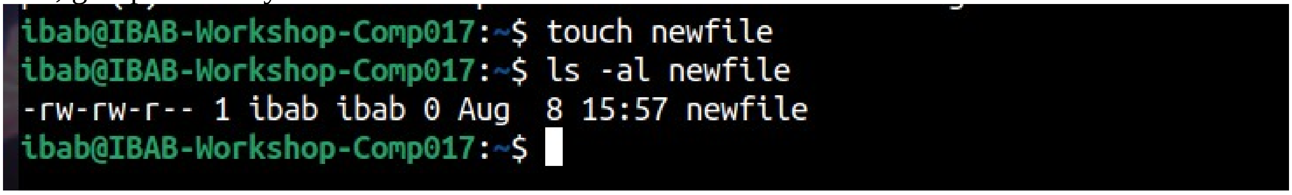
(iii) ***Output of Characters***



```
ibab@IBAB-Workshop-Comp017:~$ apropos "^Copy file"
cp (1)               - copy files and directories
cpio (1)             - copy files to and from archives
git-checkout-index (1) - Copy files from the index to the working tree
install (1)          - copy files and set attributes
ntfscp (8)           - copy file to an NTFS volume.
ibab@IBAB-Workshop-Comp017:~$
```

.

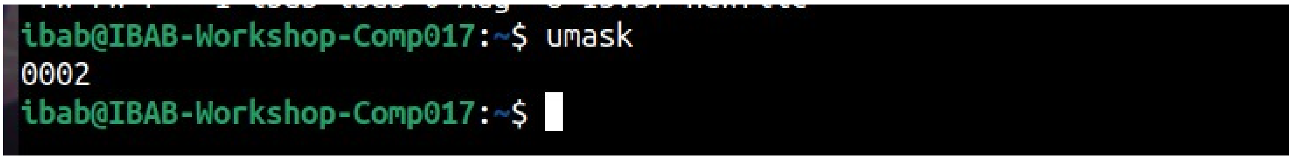
(2) File permisisons. Try the following exercises and explain the output in your work.
(i) touch newfile followed by ls -al newfile. Describe the permissions set for the user, group and everyone else.



```
ibab@IBAB-Workshop-Comp017:~$ touch newfile
ibab@IBAB-Workshop-Comp017:~$ ls -al newfile
-rw-rw-r-- 1 ibab ibab 0 Aug  8 15:57 newfile
ibab@IBAB-Workshop-Comp017:~$ 
```

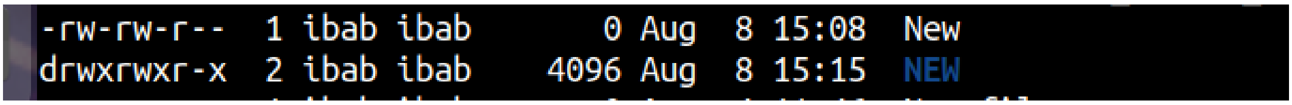
The permisssions for -
-User: rw- which means that the user can only read and write the file ,the user cannot execute it
- Group: rw- which means that the users of a group can only read and write the file ,the people in the group cannot execute it
- Everyone else: everyone else outside the group and user can only read the file they can neither write nor exceute the file

(ii) Print the output of umask command. Explain what is the actual level of default permission based on this output for a file, and for a directory



```
ibab@IBAB-Workshop-Comp017:~$ umask
0002
ibab@IBAB-Workshop-Comp017:~$ 
```

The umask command gives the octal code wherein the 1st digit is a sticky bit and the next 3 are for file permissions.

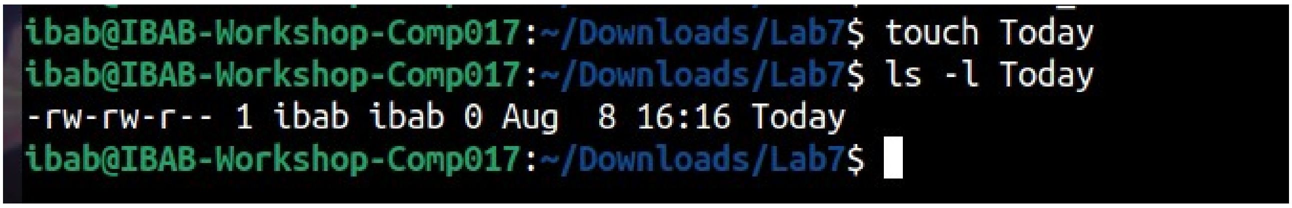


```
-rw-rw-r--   1 ibab ibab       0 Aug   8 15:08   New
drwxrwxr-x   2 ibab ibab    4096 Aug   8 15:15   NEW
```

Here **New** is a new file which was created the default file permission for a file is rw-rw-r--
that means the **User** can read,write but cannot execute the file. **Group** can read,write but cannot execute the file and for everyone else outside the group can only read the file they can neither write nor execute the file.

The **NEW** directory which was created have the default permission as **User** can read,write and execute the directory . **Group** can read,write and execute the directory and for everyone else outside the group can only read the directory and execute it they cannot write .

(iii) Create a new file in your Lab7 directory and print and explain the current permissions for the file. Write down the octal values for the user, group and everyone else

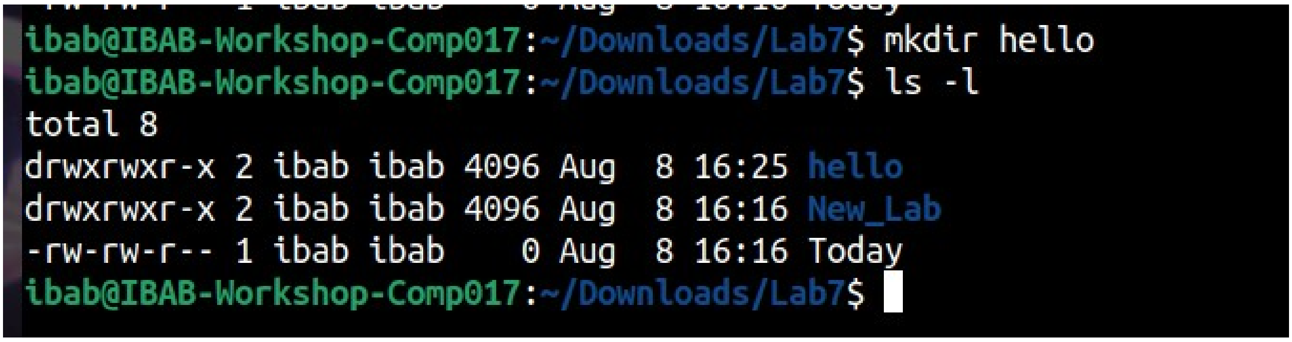


```
ibab@IBAB-Workshop-Comp017:~/Downloads/Lab7$ touch Today
ibab@IBAB-Workshop-Comp017:~/Downloads/Lab7$ ls -l Today
-rw-rw-r-- 1 ibab ibab 0 Aug  8 16:16 Today
ibab@IBAB-Workshop-Comp017:~/Downloads/Lab7$ 
```

The current permissions of the file are that the **User** can read,write but cannot execute the file. **Group** can read,write but cannot execute the file and for **everyone else** outside the group can only read the file they can neither write nor execute the file.

The Octal value for ***user*** is **6**
The Octal value for ***group*** is **6**
The Octal value for ***everyone else*** is **4**

The full possible permission octal value of a directory is **666**
And after subtracting the umask value that is **002** it gives the default value for directories i.e. **664**
(iv) Create a new directory in your Lab7 directory and print and and explain the current permissions for the file. Write down the octal values for the user, group and everyone else.



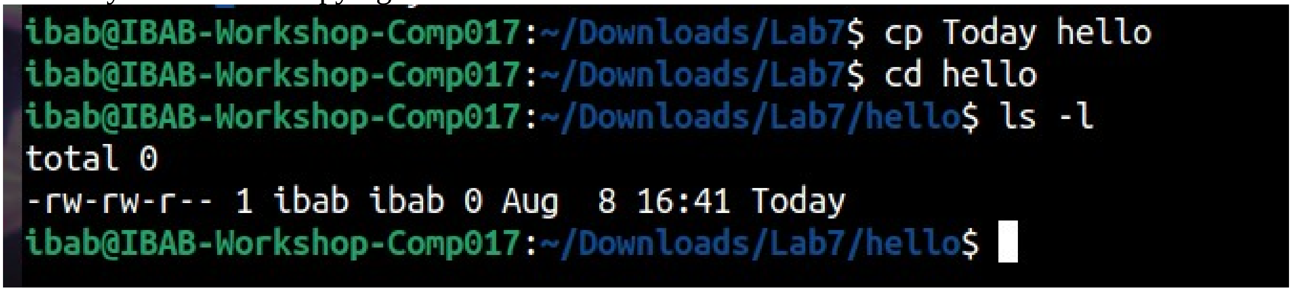
```
ibab@IBAB-Workshop-Comp017:~/Downloads/Lab7$ mkdir hello
ibab@IBAB-Workshop-Comp017:~/Downloads/Lab7$ ls -l
total 8
drwxrwxr-x 2 ibab ibab 4096 Aug  8 16:25 hello
drwxrwxr-x 2 ibab ibab 4096 Aug  8 16:16 New_Lab
-rw-rw-r-- 1 ibab ibab    0 Aug  8 16:16 Today
ibab@IBAB-Workshop-Comp017:~/Downloads/Lab7$ 
```

The new directory created was ***hello***
The permissions of the new directory are **drwxrwxr-x** meaning that the **User** can read,write and execute the directory. **Group** can read,write and execute the directory and **everyone else** outside the group can only read and execute the directory, they cannot write in the directory.

The Octal value for ***user*** is **7**
The Octal value for ***group*** is **7**
The Octal value for ***everyone else*** is **5**

The full possible permission octal value of a directory is **777**
And after subtracting the umask value that is **002** it gives the default value for directories i.e. **775**

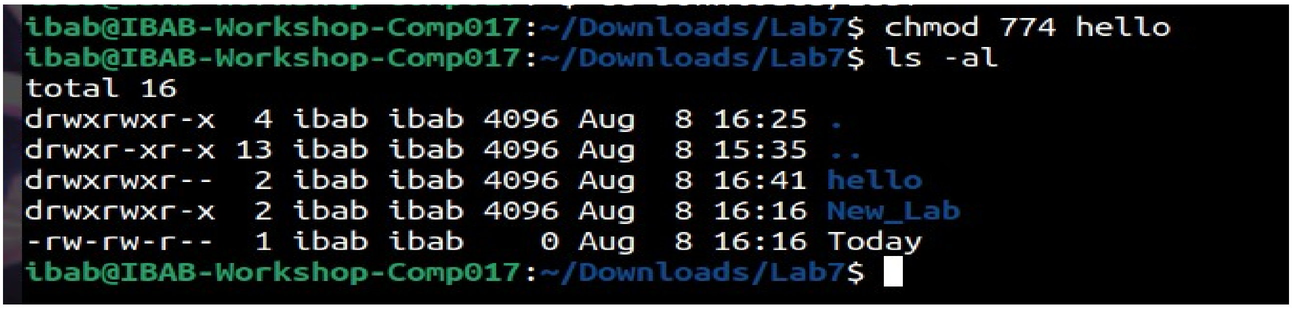
(v) Copy the new file from above to the new directory and list the permissions of the file inside the directory. Is it the same as the file outside the directory? What does this tell you about the copying action?



```
ibab@IBAB-Workshop-Comp017:~/Downloads/Lab7$ cp Today hello
ibab@IBAB-Workshop-Comp017:~/Downloads/Lab7$ cd hello
ibab@IBAB-Workshop-Comp017:~/Downloads/Lab7/hello$ ls -l
total 0
-rw-rw-r-- 1 ibab ibab 0 Aug  8 16:41 Today
ibab@IBAB-Workshop-Comp017:~/Downloads/Lab7/hello$ 
```

.
The current permissions of the file are that the **User** can read,write but cannot execute the file. **Group** can read,write but cannot execute the file and **everyone else** outside the group can only read the file they can neither write nor execute the file. It is the same as the permissions of the file outside the directory and as it was when it was created.
It tells that after copying a file to a new directory doesn’t change the file permissions of that fileand the permissions stay as they were when the file was created octal value being **664**

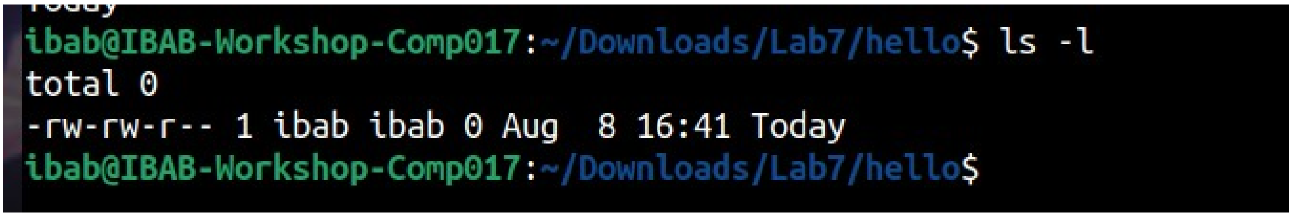
(vi) Change the permissions of the directory such that it is only readable, writable and executable by the user and the group, and only readable by everyone else. What happens to the permissions of the file inside the directory? Does it inherit the new permissions you just set? What are the new octal values for the directory permissions?



```
ibab@IBAB-Workshop-Comp017:~/Downloads/Lab7$ chmod 774 hello
ibab@IBAB-Workshop-Comp017:~/Downloads/Lab7$ ls -al
total 16
drwxrwxr-x  4 ibab ibab 4096 Aug  8 16:25 .
drwxr-xr-x 13 ibab ibab 4096 Aug  8 15:35 ..
drwxrwxr--  2 ibab ibab 4096 Aug  8 16:41 hello
drwxrwxr-x  2 ibab ibab 4096 Aug  8 16:16 New_Lab
-rw-rw-r--  1 ibab ibab    0 Aug  8 16:16 Today
ibab@IBAB-Workshop-Comp017:~/Downloads/Lab7$ 
```

The directory permissions have been changed to readable,writable and executable for user and group while only readble for everyone else outside the group.

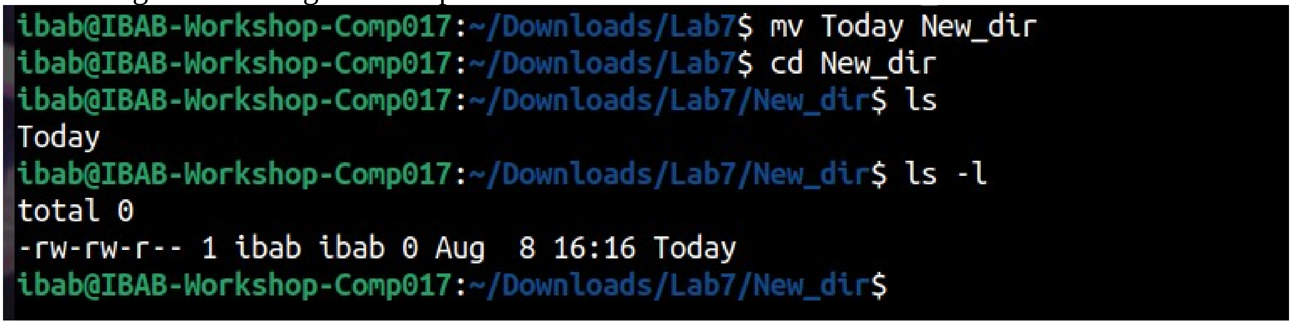
The new octal value of the directory is **774.**



```
ibab@IBAB-Workshop-Comp017:~/Downloads/Lab7/hello$ ls -l
total 0
-rw-rw-r-- 1 ibab ibab 0 Aug  8 16:41 Today
ibab@IBAB-Workshop-Comp017:~/Downloads/Lab7/hello$
```

No, the file permissions of the file inside the directory are unchanged while the permissions of the directory have changed from octal value **775** to octal value **774** . While the file permissions are the same as they were when created.

(vii) Move the above-created new file to the new directory under a different name. Does the moving action change the file permissions?



```
ibab@IBAB-Workshop-Comp017:~/Downloads/Lab7$ mv Today New_dir
ibab@IBAB-Workshop-Comp017:~/Downloads/Lab7$ cd New_dir
ibab@IBAB-Workshop-Comp017:~/Downloads/Lab7/New_dir$ ls
Today
ibab@IBAB-Workshop-Comp017:~/Downloads/Lab7/New_dir$ ls -l
total 0
-rw-rw-r-- 1 ibab ibab 0 Aug  8 16:16 Today
ibab@IBAB-Workshop-Comp017:~/Downloads/Lab7/New_dir$
```

.

Even after moving the file to a new directory, it doesnt change the permissions of the file and it still stays readable writable by user and group while only readable by everyone else.
And the octal value stays the same that is **664.**