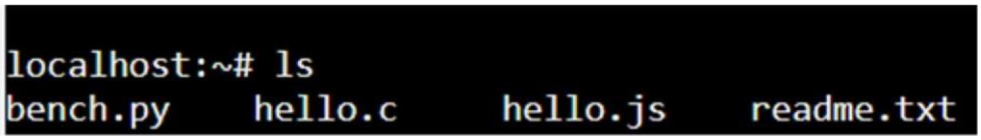
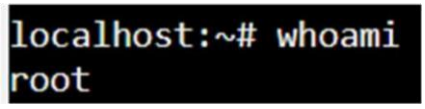
Experiment 01:

1. ls – It lists the contents of the directory.

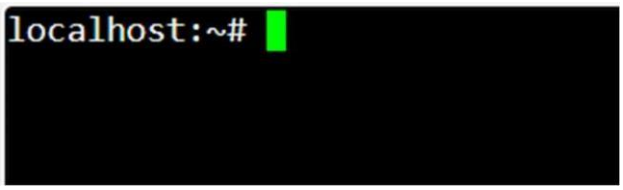


2. pwd – It prints the current working directory

3. whoami – It displays the current user name



4. clear – It clears the terminal screen.



5. mkdir – It creates a new directory

```
localhost:~# mkdir data
localhost:~# ls
bench.py    data        hello.c     hello.js    readme.txt
```

6. cd – It changes the current working directory

```
localhost:~# cd data
localhost:~/data#
```

7. cd .. – It moves back to the parent directory.

```
localhost:~/data# cd ..
localhost:~#
```

8. rmdir – It deletes the directory.

```
localhost:~# rmdir data
localhost:~# ls
bench.py    hello.c     hello.js    readme.txt
```

9. cat – It displays the contents of the files

```
localhost:~# cat hello.c
#include <stdio.h>
int main(void)
{
        printf("hello world\n");
        return 0;
}
```

10. cat > hello.txt – It allows us to create a file named hello.txt and allows to write content in it. Press Ctrl + D to save and exit.

```
localhost:~# cat > hello.txt
This is my 1st OS lab practical.
localhost:~# ls
bench.py    hello.c     hello.js    hello.txt   readme.txt
localhost:~# cat hello.txt
This is my 1st OS lab practical.
```

11. cat >> hello.txt – It appends the content to the hello.txt file.

```
localhost:~# cat >> hello.txt
OS stands for Operating Systems...
localhost:~# ls
bench.py    hello.c     hello.js    hello.txt   readme.txt
```

12. cp hello.txt bye.txt – It copies hello.txt file to a new file named bye.txt. Here, hello.txt acts as source file and bye.txt as a destination file.

```
localhost:~# cp hello.txt bye.txt
localhost:~# ls
bench.py    bye.txt     hello.c     hello.js    hello.txt   readme.txt
localhost:~# cat bye.txt
This is my 1st OS lab practical.
OS stands for Operating Systems...
```

13. mv bye.txt file.txt – It renames or moves bye.txt file to file.txt file. Here, bye.txt is source file and file.txt is destination file.

```
localhost:~# mv bye.txt file.txt
localhost:~# ls
bench.py    file.txt    hello.c     hello.js    hello.txt
localhost:~# cat file.txt
This is my 1st OS Lab practical
OS stands for Operating Systems...
```

14. rm file.txt – It deletes the file ‘file.txt’.

```
localhost:~# rm file.txt
localhost:~# ls
bench.py    hello.c     hello.js    hello.txt   readme.txt
localhost:~#
```

OUES. Create a directory with name animal, now add 2 more directories an animal namely mammal and reptile. In mammal directory create files cow.txt and lizard.txt with 2 line of text in both the files. Now use mv command to move lizard.txt from mammal to reptile directory.

1. Create the animal directory

```
localhost:~# mkdir animal
```

2. Navigate to the animal directory

```
localhost:~# cd animal
```

3. Create mammal and reptile directories inside animal

```
localhost:~/animal# mkdir mammal reptile
```

4. Create cow.txt and lizard.txt files in mammal
   a. Create cow.txt:

```
localhost:~/animal# cat> mammal/cow.txt
cow is a mammal
it provides milk
```

   b. Create lizard.txt:

```
localhost:~/animal# cat> mammal/lizard.txt
lizards are reptile
they are scary
```

5. List contents of mammal directory to verify files

```
localhost:~/animal# ls mammal
cow.txt     lizard.txt
```

6. Move lizard.txt from mammal to reptile

```
localhost:~/animal# mv mammal/lizard.txt reptile/
```

7. Verify the move using ls
   a. For mammal directory:

   b. For reptile directory:

```
localhost:~/animal# ls mammal
cow.txt
localhost:~/animal# ls reptile
lizard.txt
```