EXPERIMENT -2

2. Working on basic Linux Shell Commands: ls, mkdir, rmdir, cd, cat, banner, touch, file, wc,sort, cut, grep, dd, dfspace, du, ulimit.

1. **ls**: Lists files and directories in the current directory.

```
>> ls
```
  Output:

  file1.txt
  file2.txt
  directory1

2. **mkdir**: Creates a new directory.

>> mkdir new_directory

  Output: (No output if successful)

3. **rmdir**: Removes an empty directory.

  >> rmdir old_directory

  Output: (No output if successful)

4. **cd**: Changes the current directory.

  >>cd new_directory

  Output: *(No output if successful)*

5. **`cat`**: Concatenates and displays file contents.

  >> cat file1.txt

 Output:
  This is the content of file1.txt.

6. **banner**: Creates a large ASCII banner of the text (Note: not available on all distributions; on some, `figlet` might be used instead).

  >> banner Hello

Output:

```
   _  _  _  _
  | | | | | |
  | | | |___| | |___
  | __| |/ _ \| |/ _ \
  | | | | __/ | | () |
   \__._|\___|_|_|\___/
```

7. **touch**: Creates a new empty file or updates the timestamp of an existing file.

>> touch newfile.txt
```
   **Output:** *(No output if successful)*

8. **`file`**: Determines the type of a file.

>>  file file1.txt
```
   Output:
   file1.txt: ASCII text

9. **`wc`**: Counts lines, words, and characters in a file.

 >> wc file1.txt
  Output:
   10 50 200 file1.txt

   (Where 10 is the number of lines, 50 is the number of words, and 200 is the number of characters)

10. **`sort`**: Sorts the contents of a file.

   >>sort file1.txt
   Output:
   apple
   banana
   cherry
   ```

11. **`cut`**: Removes sections from each line of files.

   >> cut -d',' -f1 file.csv
   **Output:**
   Name1
   Name2

Name3
    ```

12. **`grep`**: Searches for patterns within files.
    >>   grep 'pattern' file1.txt

    Output:
    ```
    This line contains the pattern.
    ```

13. **`dd`**: Converts and copies files.
>>   dd if=inputfile of=outputfile bs=1M
    ```
    **Output:**
    1024+0 records in
    1024+0 records out
    ```

14. **`df`**: Reports file system disk space usage.

    >> df -h
    ```
    **Output:**
    Filesystem     Size  Used Avail Use% Mounted on
    /dev/sda1      50G   20G   28G  43% /
    ```

15. **`du`**: Estimates file space usage.
>>   du -h file1.txt
    Output:
    4.0K   file1.txt
    ```

16. **`ulimit`**: Displays or sets user process resource limits.
    >> ulimit -a
    **Output:**
    core file size          (blocks, -c) 0
    data seg size           (kbytes, -d) 8192
    max nice                   (-e) 40
    open files                 (-n) 1024
    stack size              (kbytes, -s) 8192
    ```