20CO24
Khan Arshad Abdulla

Experiment No: 03

Aim: Implement any one basic commands of linux like ls, cp, mv and others using kernel APIs.

Theory:

What Are Commands?

The Linux command is a utility of the Linux operating system. All basic and advanced tasks can be done by executing commands. The commands are executed on the Linux terminal. The terminal is a command-line interface to interact with the system, which is similar to the command prompt in the Windows OS. Commands in Linux are case-sensitive.

Examples:

1. ls command (list):

We use ls command to view the contents of a directory. By default, this command will display the contents of your current working directory. e.g.: -

```
arshad@arshad-virtual-machine:~$ ls
beef    Documents  Music    Public  Templates  Videos
Desktop  Downloads  Pictures  src    twint
```

ls command is showing the list in currently working directory **ls -R** will list all the files in the sub-directories as well **ls -a** will show the hidden files **ls -al** will list the files and directories with detailed information like the permissions, size, owner, etc.

2. cp command(copy paste):

we use the cp command to copy files from the current directory to a different directory. We can use cp command by just typing cp <filename.ext> <filename.ext1> it will copy and paste the file

e.g.:-

```
arshad@arshad-virtual-machine:~/Pictures$ cp index.png index.png1
 arshad@arshad-virtual-
 machine:~/Pictures$ls index.png
 index.png1
```

3. cat command(concatenate):

It is used to list the contents of a file on the standard output. To run this command, type cat followed by the file’s name.

20CO24
Khan Arshad Abdulla

e.g. :- arshad@arshad-virtual-machine:~/Documents/Example$ cat 20co24
I'm Khan Arshad and my roll no is 20co24 cat
> filename creates a new file cat
filename1 filename2>filename3 joins two files (1 and 2) and
stores the output of them in a new file (3)
to convert a file to upper or lower case use, cat filename | tr a-z AZ >output.txt

Implementation:

In this experiment we are using implementing cat command using kernel APIs in java language. As java reqires some packages that we need to install. before we start we have to install jdk in our linux with the help of following command sudo apt install default-jdk after installing we can verify thr version with following command java --version .

Classes:

- BufferedReader: Java BufferedReader class is used to read the text from a character-based input stream. It can be used to read data line by line by readLine() method. It makes the performance fast. It inherits Reader class.

- FileNotFoundException: FileNotFoundException is another exception class available in the java.io package. The exception occurs when we try to access that file which is not available in the system. It is a checked exception because it occurs at run time, not compile-time.

- FileReader: Java FileReader class is used to read data from the file. It returns data in byte format like FileInputStream class. It is characteroriented class which is used for file handling in java.

- IOException: The ioException() is a method of Java Scanner class which is used to get the IOException last thrown by this Scanner's readable. It returns null if no such exception exists. The ioException() method returns the last exception thrown by this scanner's readable.

- Try-Catch: Java try block is used to enclose the code that might throw an exception. It must be used within the method. If an exception occurs at the particular statement in the try block, the rest of the block code will not execute. So, it is recommended not to keep the code in try block that will not throw an exception.

Program:

```
import java.io.BufferedReader;

import java.io.FileNotFoundException;
```

20CO24
Khan Arshad Abdulla

```
import java.io.FileReader; import

java.io.IOException;

  public class Catcommand {     public static void

main(String[] args) {        if(args.length==1){

try {

                FileReader fileReader = new FileReader(args[0]);

                BufferedReader in = new BufferedReader(fileReader);

String line;                 while((line = in.readLine())!= null){

                    System.out.println(line);

                }

            } catch (FileNotFoundException ex) {

                System.out.println(args[0]+", file not found.");

            }           catch (IOException ex) {

                System.out.println(args[0]+", input/output error.");

            }

        }

    }

}
```

Output:

```
              ┌──(kali㉿kali)-[~]
└─$ java Catcommand /etc/os-release
Picked up _JAVA_OPTIONS: -Dawt.useSystemAAFontSettings=on -Dswing.aatext=true
PRETTY_NAME="Kali GNU/Linux Rolling"
NAME="Kali GNU/Linux"
ID=kali
VERSION="2021.4"
VERSION_ID="2021.4"
VERSION_CODENAME="kali-rolling"
```

20CO24
Khan Arshad Abdulla

```
ID_LIKE=debian

ANSI_COLOR="1;31"

HOME_URL="https://www.kali.org/"

SUPPORT_URL="https://forums.kali.org/"

BUG_REPORT_URL=https://bugs.kali.org/
```

If user gives wrong command: `java Catcommand /etc/os-releas`

```
Picked up _JAVA_OPTIONS: -Dawt.useSystemAAFontSettings=on -Dswing.aatext=true

/etc/os-releas, file not found.
```

Conclusion:

In this particular experiment we have successfully implemented one basic command of linux i.e., cat command using kernels APIs, we have used java language and different classes like BufferedReader, FileNotFoundException, FileReader, IOException to implement the program.