Navigating the File System

The Linux file system is structured in a hierarchical manner, resembling an upside-down tree. At the root of this tree lies the root directory, denoted by a single forward slash (`/`). From this root, all other files and directories branch out, forming a vast network of interconnected paths.

Each of these directories serves a specific purpose, housing various system files, user data, and configuration settings. Let's explore some of the most important ones:

1. `/bin` : Contains essential user binaries and commands.
2. `/etc` : Stores system-wide configuration files.
3. `/home` : Houses personal directories for individual users.
4. `/usr` : Holds user programs, libraries, and documentation.
5. `/var` : Stores variable data, such as logs and temporary files.

- To check the current working directory. We will use the pwd or print working directory command.

```
pwd
```

- To list the contents of the the current directory. We will use the ls or list command.

```
ls
```

- We can use various flags with the ls command also. The first one is the -l flag. It shows all the data in a nice list format with additional details.

```
ls -l
```

- Next we can also view the hidden files and folders with the -a flag.

```
ls -a
```

- Combining all of these with the -h flag which stands for human-readable gives us a nice and comprehensive output.

```
ls -alh
```

- Suppose we want to view the subdirectories inside a large directory or folder. How can we do that ? For that, we have a special -R flag

```
ls -lR Downloads/
```

Now we know how to view our current directory and list its contents. But what if we have to move into one of the directory. Let's say Desktop. How can we do that ?

Okay. So this is possible with the help of the cd command, where cd stands for change directory.

```
cd Desktop
```

- Now if we want to go back to our previous directory where we came from. We can again use the cd command with it but using a little shortcut.

```
cd ..
```

Next, suppose you went to far with your cd commands. You entered one folder, then another, then another and now to go back to the home folder, you have two choice. One is to do a cd.. or to use cd /home/kali. There is one shortcut option for this also. that is

```
cd ~
```

- Sometimes if you are not sure what does a command do. Just use the whatis command and it will tell you what it does

```
whatis ls
```

- Now, if you want to make a directory in Linux. You can easily do this by using the mkdir command.

```
mkdir Test
```

- Same if we want to delete the directory

```
rmdir Test
```

- Now linux has this amazing utility called man pages which is like the manual of the linux commands. If you are stuck on a command where you are confused which flag to

choose with it. Try the man command on it or the --help command. Both will display all the flags available and their respective information with it.

```
man ls

ls --help
```

Next, there is a command called echo. What does it do ? as the name suggest it just echo out what we ask it to do.

```
echo "Hello World!!!"
```

- We can echo out the same result in a text file by using a direction or greater than symbol with it.

```
echo "Hello World!!!" > hello.txt
```

- At last, we if want to copy a directory or file to another directory on the filesystem. We can use the cp command.

```
cp hello.txt /Test
```

- Same goes for moving the file from one place to another. We will use the mv command.

```
mv hello.txt /Desktop
```

- We can delete the files using the rm command.

```
rm hello.txt
```

In linux, there is a concept of Relative and Absoute path. Let's see that also briefly

An absolute path is the complete path from the root directory (/). It specifies the exact location of a file or directory from the topmost root. While A relative path is a path relative to the current working directory. It does not start from the root.
