

Navigating the filesystem structure

Topics covered: the cd command, mkdir, locate, find, default directory structure.

Exercise 1: Basic Navigation

Objective: Learn how to navigate the filesystem using common commands.

1. Open your terminal.
2. Check your current working directory using:

```
pwd
```

3. List the contents of your current directory:

```
ls
```

4. Change to the root directory:

```
cd /
```

5. Move to your home directory:

```
cd ~
```

6. Navigate to the /usr/bin directory:

```
cd /usr/bin
```

7. Confirm your current directory again:

```
pwd
```

Exercise 2: Using Relative and Absolute Paths

Objective: Understand the difference between absolute and relative paths.

1. From your home directory, create a new directory called "exercise_dir":

```
mkdir exercise_dir
```

2. Use a relative path to navigate into "exercise_dir":

```
cd exercise_dir
```

3. Create a subdirectory called "sub_dir" using a relative path:

```
mkdir sub_dir
```

4. Use an absolute path to navigate back to your home directory:

```
cd /home/$USER
```

question: is there another variable that holds you username?

5. Now navigate to "exercise_dir" using an absolute path:

```
cd /home/$USER/exercise_dir
```

Exercise 3: Navigating Up the Filesystem

Objective: Learn how to move up the directory tree.

1. From "exercise_dir", move up one directory level:

```
cd ..
```

2. Confirm that you are back in your home directory:

```
pwd
```

3. Navigate to the parent directory of your home directory using ...:

```
cd ..
```

4. List the contents of the parent directory to see where you are:

```
ls
```

Exercise 4: Using Tab Completion

Objective: Utilize tab completion to speed up navigation.

1. Starting from your home directory, type `cd /u` and press the Tab key.

- What happens? (It should auto-complete to `/usr` if it exists.)

2. Which directories do you see under `/usr` if you use the Tab key?

Exercise 5: Finding Your Way

Objective: Use commands to find your way around and locate files.

1. Use the `find` command to search for a file named "bash" starting from the root directory:

```
find / -name "bash" 2>/dev/null
```

2. Use the `locate` command to find a file named "passwd" (ensure the `mlocate` package is installed and updated):

```
locate passwd
```

3. Use the `which` command to find the executable path of the `ls` command:

```
which ls
```

Exercise 6: Understanding Special Directories

Objective: Learn about special directories like `.` and `...`

1. Create a new directory called "test_dir" in your home directory:

```
mkdir ~/test_dir
```

2. Navigate to "test_dir" using a relative path:

```
cd test_dir
```

3. Create a file named "myfile.txt" inside "test_dir":

```
touch myfile.txt
```

4. Use ls to display the contents of the current directory using .:

```
ls .
```

5. Move up one directory level and list the contents using ..:

```
cd ..
```

```
ls ..
```

Exercise 7 filesystem Hierarchy.

1. What sort of files can you expect in the following directories:

- a. /var
- b. /boot
- c. /usr/bin
- d. /bin
- e. /tmp
- f. /home
- g. /etc
- h. /proc

2. Have a look at /sbin and /usr/sbin. Are these the same binaries or not?
And what about /lib and /usr/lib and /bin and /usr/bin?

3. What option can you use to create a directory structure like this with a single command:
/home/rocky/scripts/administration/practice

4. Find out about the usage of the `file` command.