

## **1. Perform different file handling commands**

### **a. Listing all files**

```
try@try:~/testing/programs$ ls
employee example.java file.txt new_file report.pl report.txt rev.sh revstr.sh
```

### **b. Creating a file**

```
try@try:~/testing$ vi test
try@try:~/testing$ cat test
This is a unix file...I created it using vi text editor.....
Now I will save it using :wq command.
```

### **c. Renaming or moving a file**

```
try@try:~/testing/programs$ mv file.txt new.txt
try@try:~/testing/programs$ ls
employee example.java new_file new.txt report.pl report.txt rev.sh revstr.sh
```

### **d. Copying a file**

```
try@try:~/testing/programs$ cp employee ../
try@try:~/testing/programs$ cd ..
try@try:~/testing$ ls
a.txt employee example.java new next num.py programs round.exe test
```

### **e. Displaying a file**

```
try@try:~/testing$ cat employee
ajay manager account 50000
sunil clerk account 10000
varun manager sales 60000
tarun peon sales 5000
sunil director purchase 90000
```

### **f. Deleting a file**

```
try@try:~/testing$ rm employee
try@try:~/testing$ ls
a.txt example.java new next num.py programs round.exe test
```

## 2. Change file permissions and ownership

```
try@try:~/testing/new$ ls -l
```

```
total 0
```

```
-rw-rw-r-- 1 try try 0 Nov 29 17:57 example.java
```

```
try@try:~/testing/new$ sudo chmod 777 example.java
```

```
[sudo] password for try:
```

```
try@try:~/testing/new$ sudo chown root example.java
```

```
try@try:~/testing/new$ ls -l
```

```
total 0
```

```
-rwxrwxrwx 1 root try 0 Nov 29 17:57 example.java
```

### **3. Copying and moving files to different folders using relative and absolute path**

#### **a. Copying a file using relative path**

```
try@try:~/testing$ cp ./example.java ./new
```

```
try@try:~/testing$ cd new
```

```
try@try:~/testing/new$ ls
```

**example.java**

#### **b. Copying a file using absolute path**

```
try@try:~/testing/new$ cp ./example.java /home/try/testing/programs/
```

```
try@try:~/testing/new$ cd ..
```

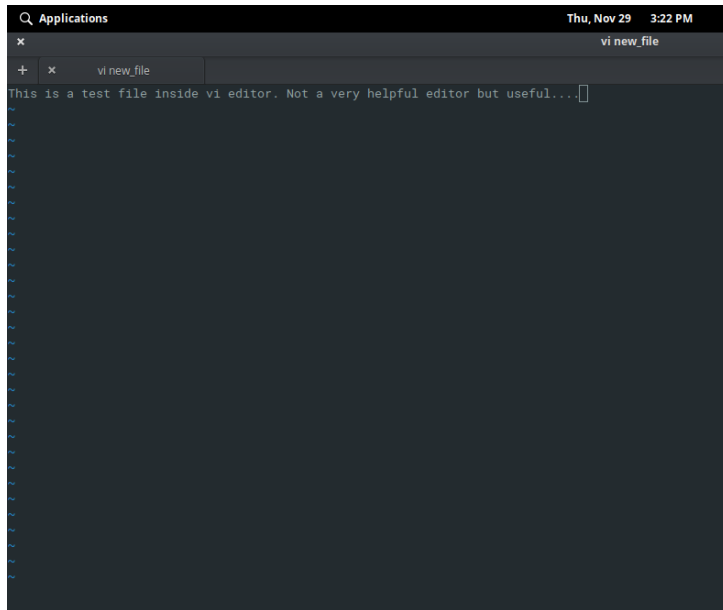
```
try@try:~/testing$ cd programs/
```

```
try@try:~/testing/programs$ ls
```

**employee example.java file.txt new\_file report.pl report.txt rev.sh revstr.sh**

## 4. Using vi editor

try@try:~vi newfile



try@try:~cat newfile

This is a test file inside vi editor. Not a very helpful editor but useful...

## 5. Handling files using awk and perl

a. Awk command to handle files

```
try@try:~/testing/programs$ awk '{print}' employee
```

```
ajay manager account 50000
```

```
sunil clerk account 10000
```

```
varun manager sales 60000
```

```
tarun peon sales 5000
```

```
sunil director purchase 90000'
```

```
try@try:~/testing/programs$ awk '{print $1,$4}' employee
```

```
ajay 50000
```

```
sunil 10000
```

```
varun 60000
```

```
tarun 5000
```

```
sunil 90000
```

```
try@try:~/testing/programs$ awk '/manager/{print}' employee
```

```
ajay manager account 50000
```

```
varun manager sales 60000
```

## b. Perl language to handle a file

```
use strict;
```

```
use warnings;
```

```
open(my $f, '>', 'report.txt');
```

```
print $f "My first perl program to handle files\n";
```

```
close $f;
```

```
print "done\n";
```

```
try@try:~/testing/programs$ ls
```

```
employee example.java file.txt new_file report.pl rev.sh revstr.sh
```

```
try@try:~/testing/programs$ perl report.pl
```

```
done
```

```
try@try:~/testing/programs$ ls
```

```
employee example.java file.txt new_file report.pl report.txt rev.sh revstr.sh
```

## 6. Create a shell program to reverse a number

```
echo "Enter a number: "  
read num  
  
rev=0  
mod=0  
n=$num  
  
while [ $num -gt 0 ]  
do  
    mod=`expr $num % 10`  
    temp=`expr $rev \* 10`  
    rev=`expr $temp + $mod`  
    num=`expr $num / 10`  
done  
  
echo "The reverse of $n is $rev"  
  
try@try:~/testing/programs$ ./rev.sh  
Enter a number:  
123456  
The reverse of 123456 is 654321
```

## 7. Create a shell program to reverse a string

```
echo "Enter a string: "
```

```
read str
```

```
len=${#str}
```

```
for (( i=len-1; i>=0; i--))
```

```
do
```

```
    reverse="$reverse${str:$i:1}"
```

```
done
```

```
echo "$reverse"
```

```
try@try:~/testing/programs$ ./revstr.sh
```

**Enter a string:**

**aditya**

**aytida**

## 8. Create a shell program to update a file

```
try@try:~/testing$ vi tests
```

```
try@try:~/testing$ cat tests
```

**UNIX operating system files**

**foo is nice**

**foo is good**

```
try@try:~/testing$ sed -i 's/files/file/g' tests
```

```
try@try:~/testing$ sed -i 's/foos/boo/' tests
```

```
try@try:~/testing$ cat tests
```

**UNIX operating system file**

**boo is nice**

**boo is good**



## 9. Create new user and groups

### a. Creating new user

```
try@try:~/testing$ sudo useradd adrian
try@try:~/testing$ sudo tail /etc/passwd
speech-dispatcher:x:113:29:Speech Dispatcher,,,:/var/run/speech-dispatcher:/bin/false
nm-openvpn:x:114:119:NetworkManager OpenVPN,,,:/var/lib/openvpn/chroot:/usr/sbin/nologin
saned:x:115:121::/var/lib/saned:/usr/sbin/nologin
pulse:x:116:122:PulseAudio daemon,,,:/var/run/pulse:/usr/sbin/nologin
avahi:x:117:124:Avahi mDNS daemon,,,:/var/run/avahi-daemon:/usr/sbin/nologin
colord:x:118:125:colord colour management daemon,,,:/var/lib/colord:/usr/sbin/nologin
hplip:x:119:7:HPLIP system user,,,:/var/run/hplip:/bin/false
geoclue:x:120:126::/var/lib/geoclue:/usr/sbin/nologin
try:x:1000:1000:try,,,:/home/try:/bin/bash
adrian:x:1001:1001::/home/adrian:/bin/sh
```

### b. Creating new group

```
try@try:~/testing$ sudo groupadd -g 5000 alphonse
try@try:~/testing$ sudo tail /etc/group
saned:x:121:
pulse:x:122:
pulse-access:x:123:
avahi:x:124:
colord:x:125:
geoclue:x:126:
try:x:1000:
sambashare:x:127:try
adrian:x:1001:
alphonse:x:5000:
```

## 10. Display partition information and system information

```
try@try:~/testing$ sudo lsblk -o NAME,FSTYPE,SIZE,MOUNTPOINT,LABEL
```

NAME	FSTYPE	SIZE	MOUNTPOINT	LABEL
------	--------	------	------------	-------

sda		8G		
└─sda1	ext4	8G	/	
sr0		1024M		

```
try@try:~/testing$ sudo fdisk -l
```

**Disk /dev/sda: 8 GiB, 8589934592 bytes, 16777216 sectors**

**Units: sectors of 1 \* 512 = 512 bytes**

**Sector size (logical/physical): 512 bytes / 512 bytes**

**I/O size (minimum/optimal): 512 bytes / 512 bytes**

**Disklabel type: dos**

**Disk identifier: 0xabb73fc1**

Device	Boot	Start	End	Sectors	Size	Id	Type
--------	------	-------	-----	---------	------	----	------

/dev/sda1	*	2048	16775167	16773120	8G	83	Linux
-----------	---	------	----------	----------	----	----	-------