LINUX

MANIPULATING FILES AND DIRECTORIES

Basic command for manipulation are :

- touch
- cp
- mv
- mkdir
- rm

Touch :- It is used to create,change and modify timestamp of a file .

To create a file we write touch filename.extension

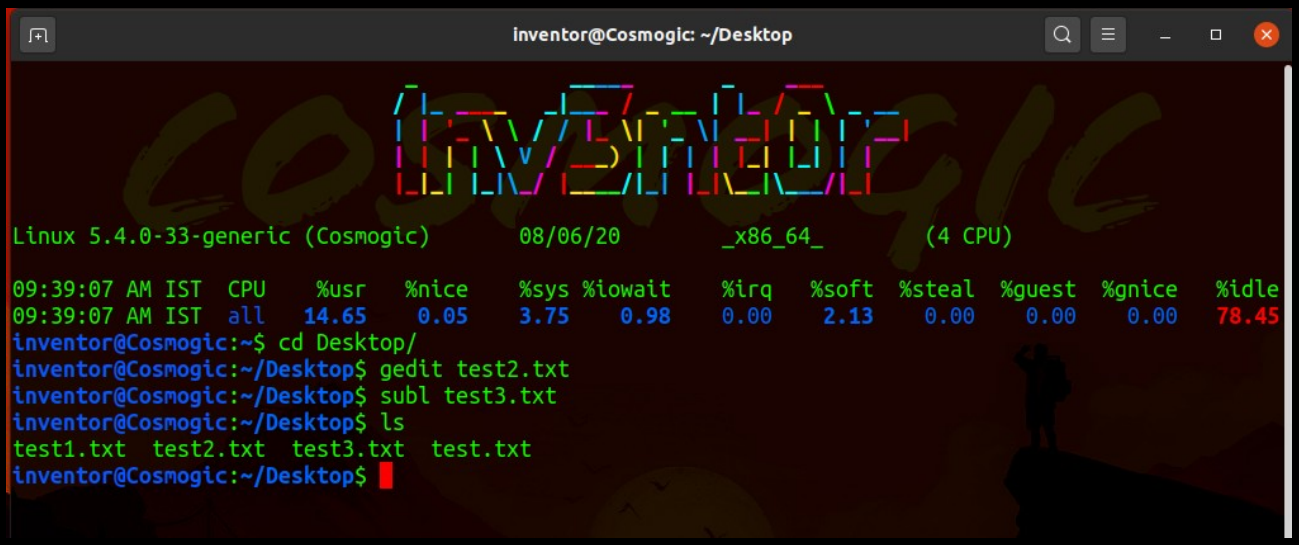
we can also create file by using > like

> filename.extension

we can get the same result by using gedit or subl

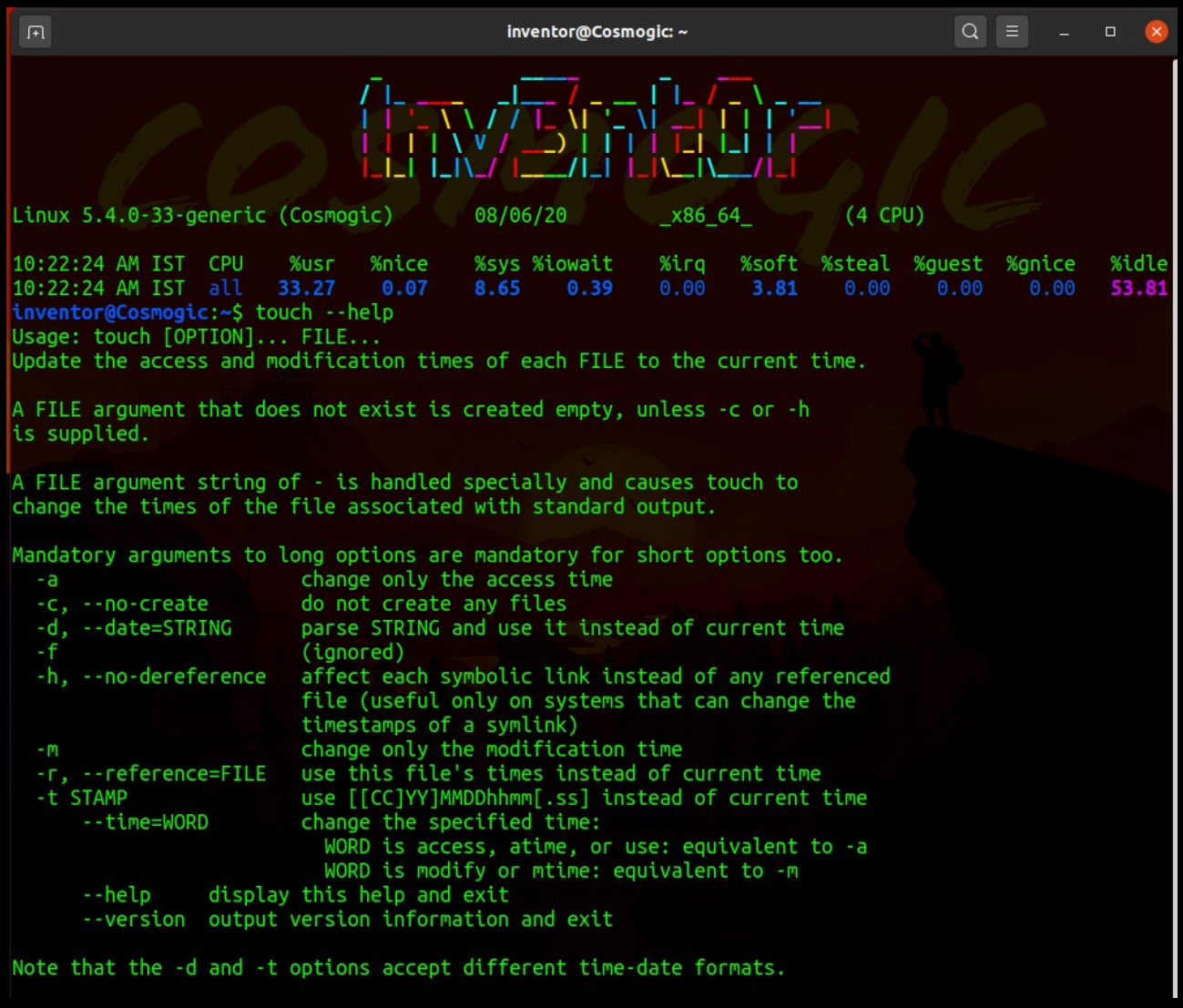
gedit filename.extension (gedit is preinstalled editor in ubuntu)

subl filename.extension (if you have installed sublime)



Gedit and subl will create file and we will be relocated to gedit or sublime we can view those file after we have saved them .

Different options of touch are :



we can create multiple file by
touch file1 file2 file3

we can write touch -a filename to update the last access and modified time of file

we might end creating file with same name so to avoid that we use touch -c filename

to change timestamp of one file with one another we write
touch -r file1 file2
this will change timestamp of file2 with file1

to create a file with specific timestamp we write
touch -t time (in YYMMDDHH.SS format) filename

CP :- it stands for copy and it's used to a copy file or a group of files to another location .

To copy files to a folder
cp file1 file2 file3 destination(folder)

option available with cp command are

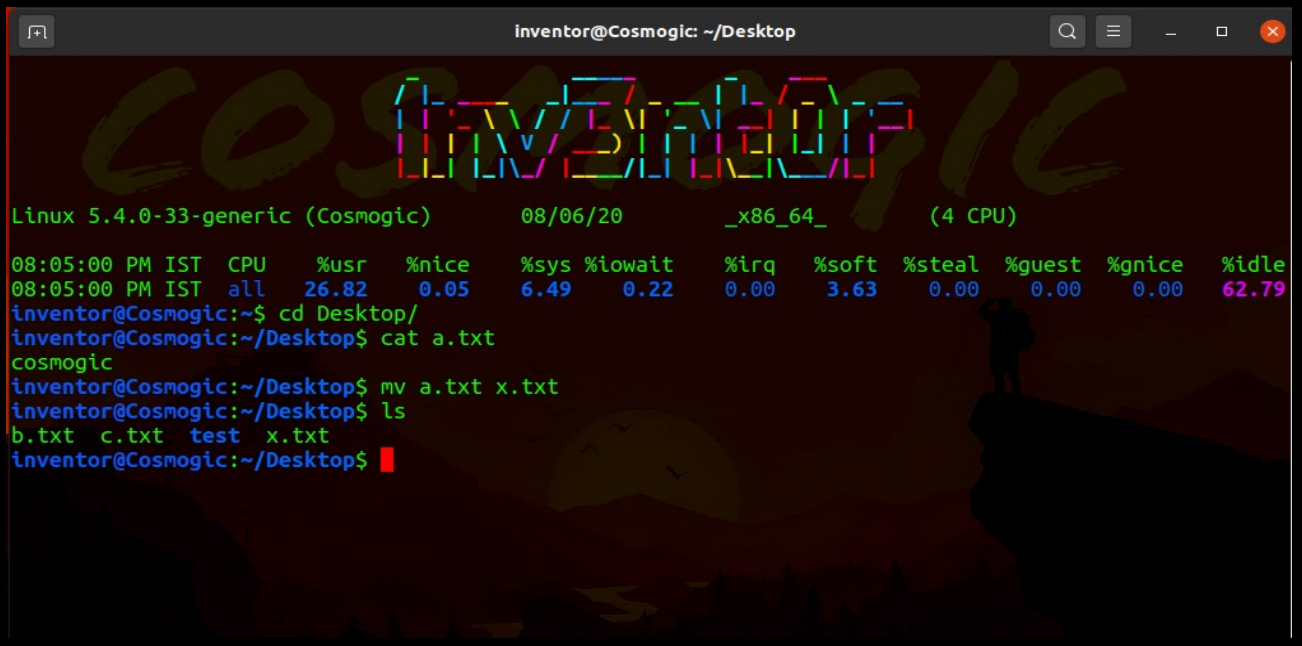
```
Mandatory arguments to long options are mandatory for short options too.
  -a, --archive                same as -dR --preserve=all
      --attributes-only        don't copy the file data, just the attributes
      --backup[=CONTROL]       make a backup of each existing destination file
  -b                           like --backup but does not accept an argument
      --copy-contents          copy contents of special files when recursive
  -d                           same as --no-dereference --preserve=links
  -f, --force                  if an existing destination file cannot be
                                 opened, remove it and try again (this option
                                 is ignored when the -n option is also used)
  -i, --interactive            prompt before overwrite (overrides a previous -n
                                  option)
  -H                           follow command-line symbolic links in SOURCE
  -l, --link                   hard link files instead of copying
  -L, --dereference            always follow symbolic links in SOURCE
  -n, --no-clobber             do not overwrite an existing file (overrides
                                 a previous -i option)
  -P, --no-dereference         never follow symbolic links in SOURCE
  -p                           same as --preserve=mode,ownership,timestamps
      --preserve[=ATTR_LIST]   preserve the specified attributes (default:
                                 mode,ownership,timestamps), if possible
                                 additional attributes: context, links, xattr,
                                 all
      --no-preserve=ATTR_LIST  don't preserve the specified attributes
      --parents                use full source file name under DIRECTORY
  -R, -r, --recursive          copy directories recursively
      --reflink[=WHEN]         control clone/CoW copies. See below
      --remove-destination     remove each existing destination file before
                                 attempting to open it (contrast with --force)
      --sparse=WHEN            control creation of sparse files. See below
      --strip-trailing-slashes  remove any trailing slashes from each SOURCE
                                 argument
  -s, --symbolic-link          make symbolic links instead of copying
  -S, --suffix=SUFFIX          override the usual backup suffix
  -t, --target-directory=DIRECTORY  copy all SOURCE arguments into DIRECTORY
  -T, --no-target-directory    treat DEST as a normal file
  -u, --update                 copy only when the SOURCE file is newer
                                 than the destination file or when the
                                 destination file is missing
  -v, --verbose                explain what is being done
  -x, --one-file-system        stay on this file system
  -Z                           set SELinux security context of destination
                                 file to default type
      --context[=CTX]          like -Z, or if CTX is specified then set the
                                 SELinux or SMACK security context to CTX
```

to copy all the files of one directory to another
cp -R Source Destination (this will copy all file and sub directories recursively)

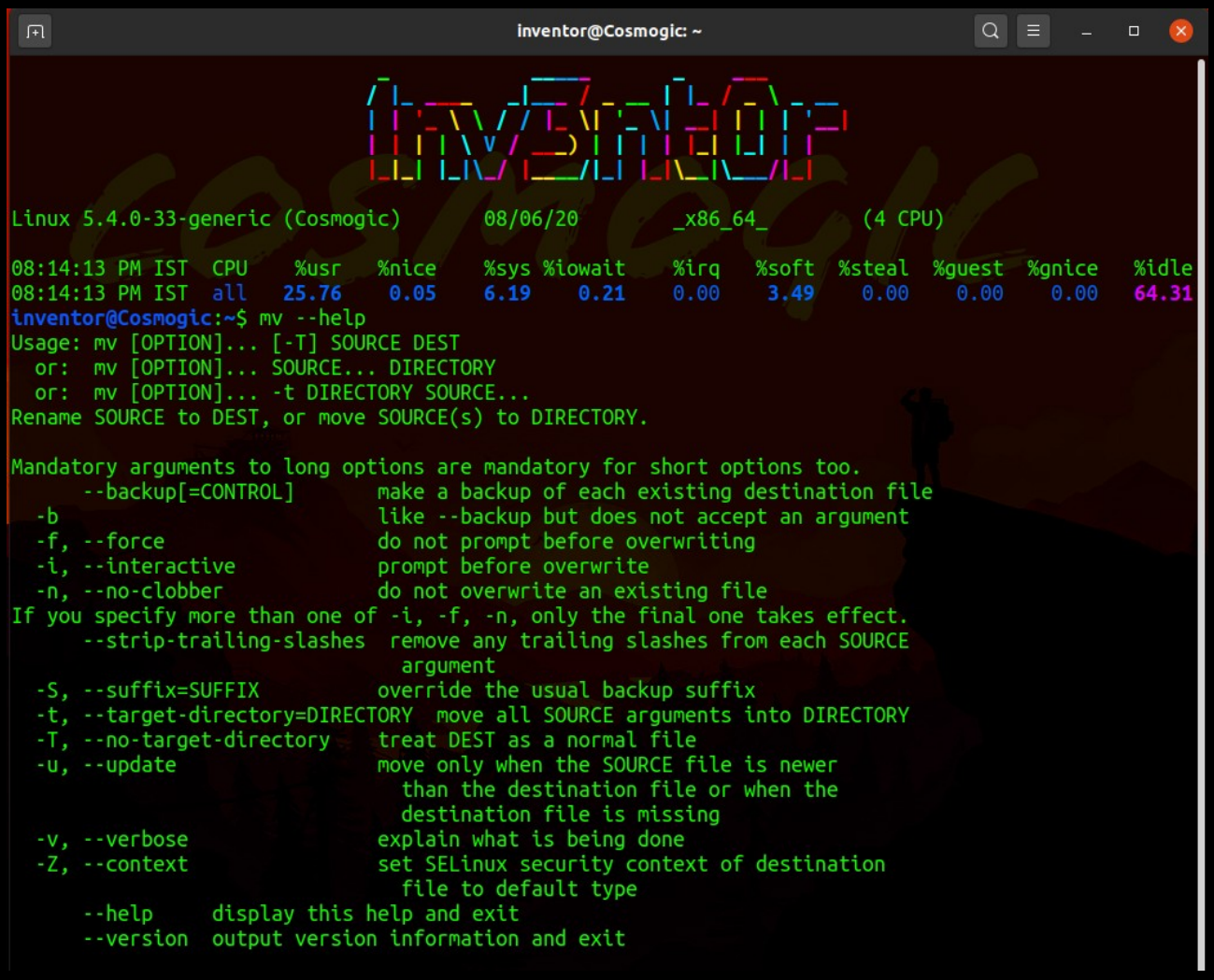
cp -i a.txt b.txt this is an intractive method of copying file . It will ask for permission to overwrite file

if we write y it will overwrite the file and we can write n to cancel the action
if we want to copy all file with same extension we can use cp *.txt location(folder)

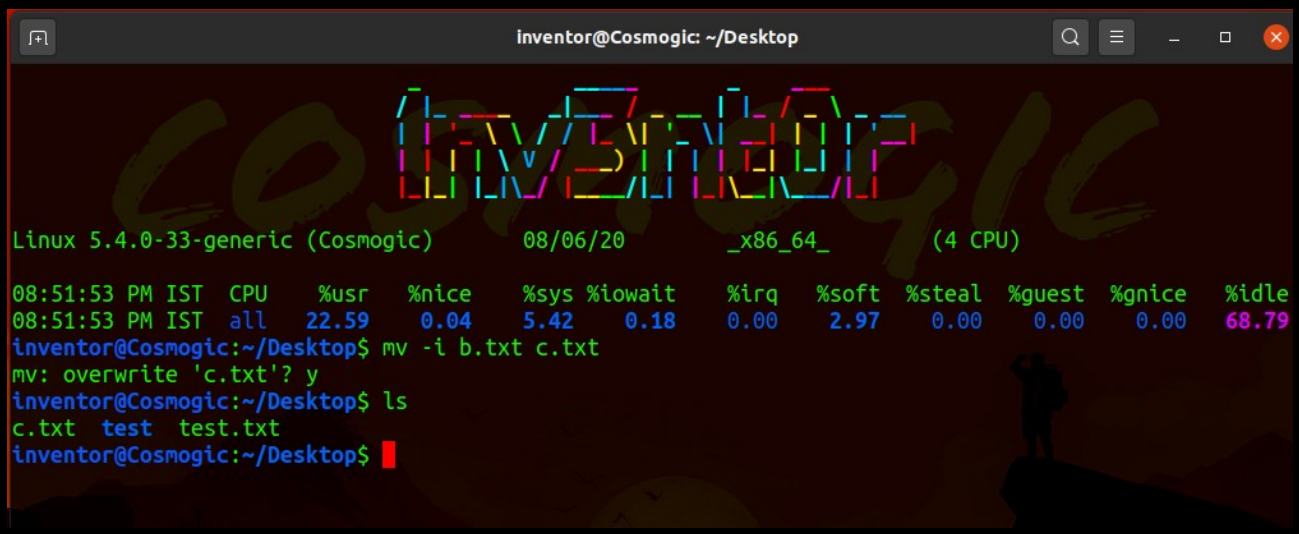
MV :- mv stands for move and as its name suggest it is used to move file or directories from one location to another location . We can also use it to rename files .



Options available with mv command are



we can use mv -i a.txt b.txt this will ask for confirmation before taking action



MKDIR :- This command is used to create directories (folders) .

mkdir directory name

we can also create multiple directories

mkdir dir1 dir2 dir3

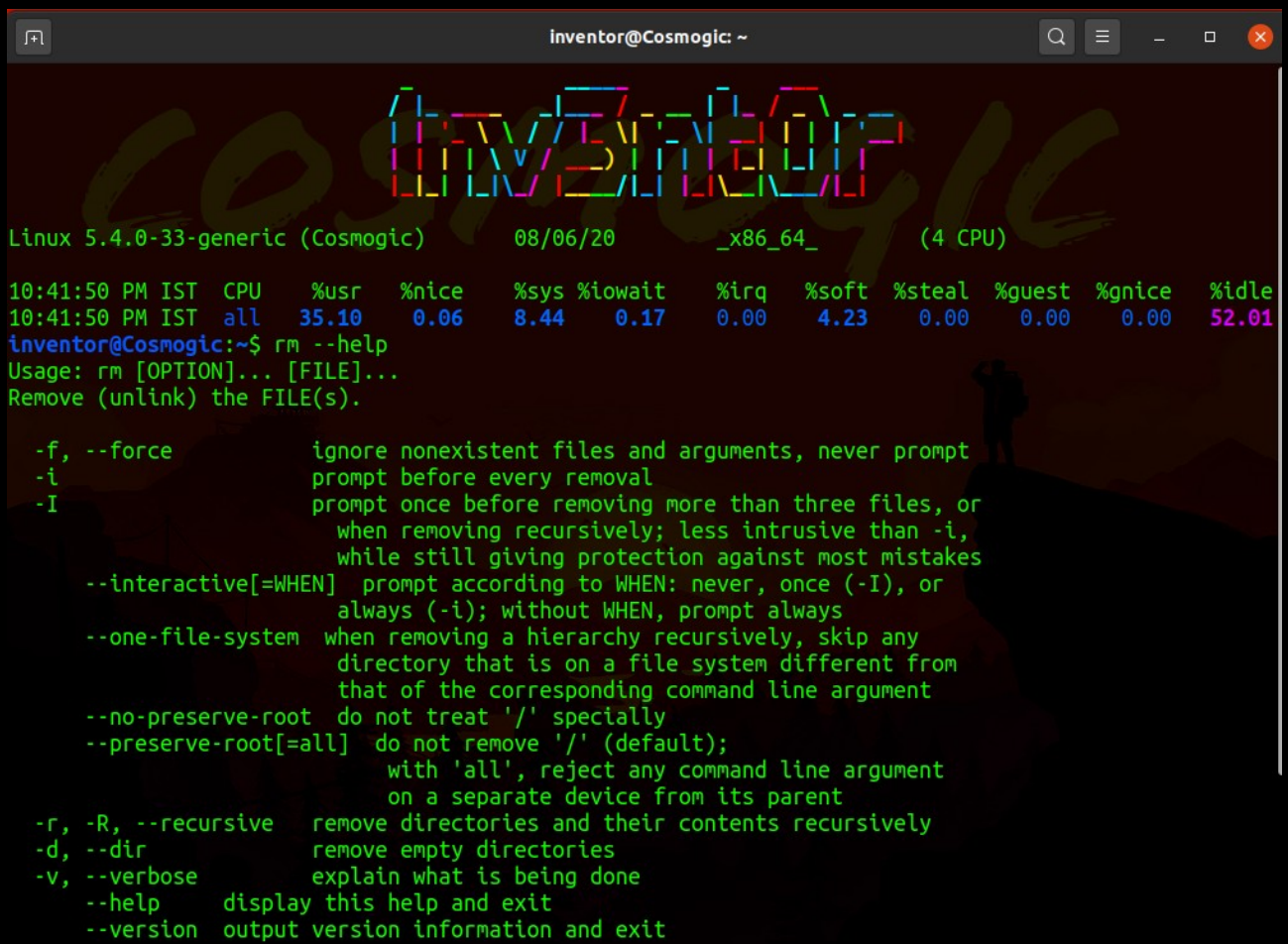
options of mkdir command are

mkdir -v dir1 dir2

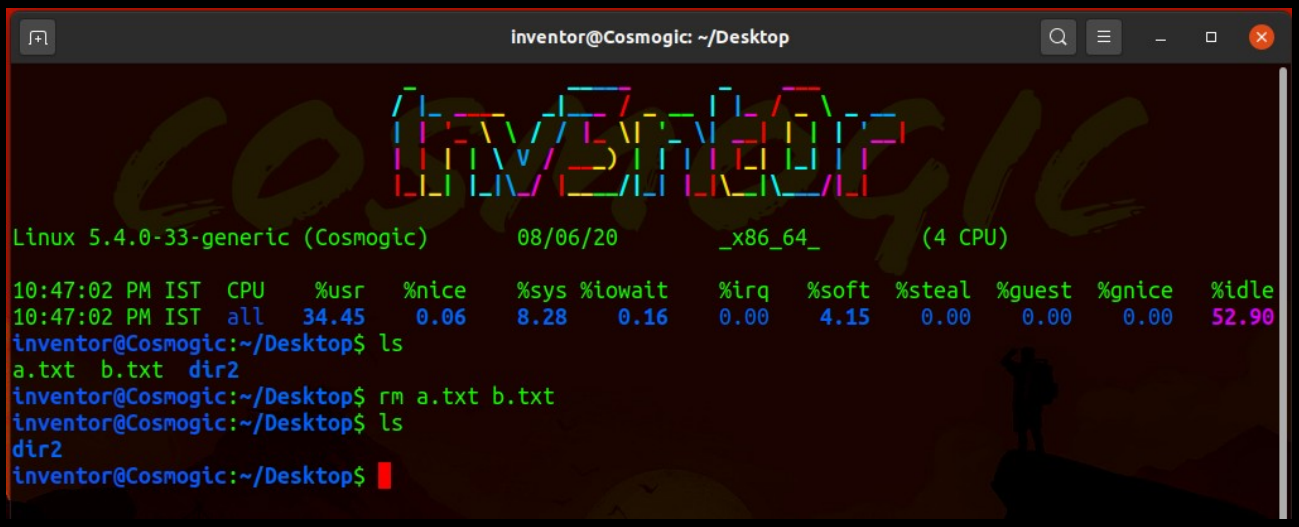
this will show a message after directory is been created

RM :- rm stands for remove and as the name suggest it remove files and directories . By default it does not remove directories (we use rm -rf)

options of rm command



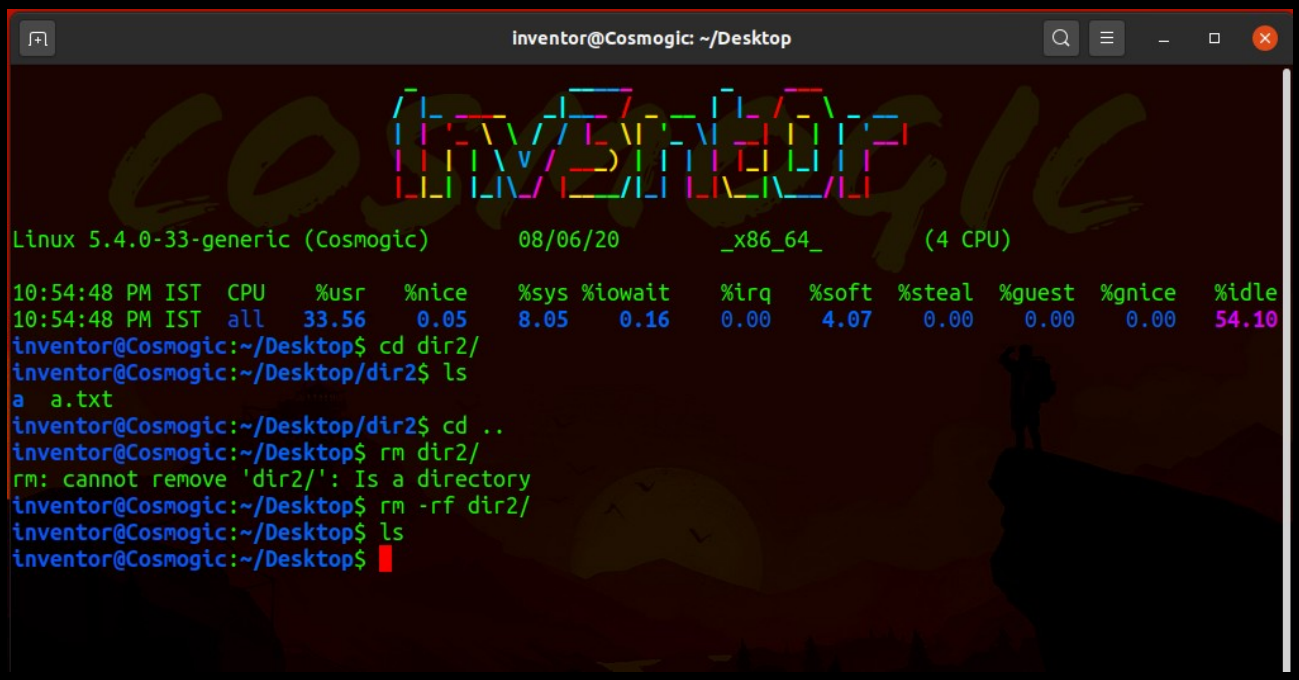
we can also remove more than one file



rm -f: forcefully remove file even file is write protected

rm -r :recursive deletion,this perform a tree walk and remove every thing in a directory

they both are used in combination to remove directory



ln command is used to create links .
ln -s source destination
this is used to create soft links .

Source code of commands
Touch
https://github.com/coreutils/coreutils/blob/master/src/touch.c
CP
https://github.com/coreutils/coreutils/blob/master/src/cp.c
MV
https://github.com/coreutils/coreutils/blob/master/src/mv.c
MKDIR
https://github.com/coreutils/coreutils/blob/master/src/mkdir.c
RM
https://github.com/coreutils/coreutils/blob/master/src/rm.c
LN
https://github.com/coreutils/coreutils/blob/master/src/ln.c

Thank you
- 1nv3ntor