Kenzh_hw4

April 4, 2020

[2]: ```
cd
```

[3]: ```
pwd
```

```
/home/jupyter-kkenzh
```

[4]: ```
ls -a
```

```
.        Aruzhan        hw.ipynb              .lesshst           .octave_hist
..       .bash_history  .ipynb_checkpoints    lessoninfo.ipynb   octave-workspace
1.txt    .bash_logout   .ipython              lifeisgood         .profile
2.jpg    .bashrc        .jupyter              .local             shared
3.png    .cache         Kenzh.ipynb           Loop               untitled.dio
```

[5]: ```
la -l
```

```
total 268
-rw--w-r-- 1 jupyter-kkenzh jupyter-kkenzh      0 Apr  2 18:04 1.txt
-rw-r--r-- 1 jupyter-kkenzh jupyter-kkenzh      0 Apr  2 18:04 2.jpg
-rw-r--r-- 1 jupyter-kkenzh jupyter-kkenzh      0 Apr  2 18:04 3.png
drwxr-xr-x 4 jupyter-kkenzh jupyter-kkenzh   4096 Mar 28 01:27 Aruzhan
-rw------- 1 jupyter-kkenzh jupyter-kkenzh  12739 Apr  2 19:00 .bash_history
-rw-r--r-- 1 jupyter-kkenzh jupyter-kkenzh    220 Apr  4  2018 .bash_logout
-rw-r--r-- 1 jupyter-kkenzh jupyter-kkenzh   3771 Apr  4  2018 .bashrc
drwxr-xr-x 3 jupyter-kkenzh jupyter-kkenzh   4096 Mar 19 18:47 .cache
-rw-r--r-- 1 jupyter-kkenzh jupyter-kkenzh  62752 Apr  2 18:49 hw.ipynb
drwxr-xr-x 2 jupyter-kkenzh jupyter-kkenzh   4096 Apr  2 17:23
.ipynb_checkpoints
drwxr-xr-x 6 jupyter-kkenzh jupyter-kkenzh   4096 Mar 19 18:50 .ipython
drwxr-xr-x 3 jupyter-kkenzh jupyter-kkenzh   4096 Mar 21 12:13 .jupyter
-rw-r--r-- 1 jupyter-kkenzh jupyter-kkenzh 110963 Mar 28 13:58 Kenzh.ipynb
-rw------- 1 jupyter-kkenzh jupyter-kkenzh     48 Mar 28 13:10 .lesshst
-rw-r--r-- 1 jupyter-kkenzh jupyter-kkenzh   1459 Apr  2 18:49 lessoninfo.ipynb
drwxr-xr-x 4 jupyter-kkenzh jupyter-kkenzh   4096 Mar 21 13:57 lifeisgood
drwxr-xr-x 3 jupyter-kkenzh jupyter-kkenzh   4096 Mar 19 18:47 .local
drwxr-xr-x 2 jupyter-kkenzh jupyter-kkenzh   4096 Mar 28 13:05 Loop
-rw------- 1 jupyter-kkenzh jupyter-kkenzh    380 Mar 19 19:19 .octave_hist
-rw-r--r-- 1 jupyter-kkenzh jupyter-kkenzh     54 Mar 19 19:19 octave-workspace
```

```
-rw-r--r-- 1 jupyter-kkenzh jupyter-kkenzh    807 Apr  4  2018 .profile
lrwxrwxrwx 1 jupyter-kkenzh jupyter-kkenzh     16 Mar 10 11:57 shared ->
/srv/data/shared
-rw-r--r-- 1 jupyter-kkenzh jupyter-kkenzh  15496 Apr  2 17:25 untitled.dio
```

[6]: `touch star.mp3`

[7]: `touch start.png`

[8]: `touch 14.png`

[10]: `ls star*.png`

```
start.png
```

[12]: `ls sta*.*`

```
star.mp3   start.png
```

[13]: `ls *.[mp3][png]`

```
ls: cannot access '*.[mp3][png]': No such file or directory
```

[14]: `touch start.jpg`

[15]: `ls *.[jp][pn]g`

```
14.png  2.jpg  3.png  start.jpg  start.png
```

[16]: `ls [1-3].png`

```
3.png
```

[19]: `ls [1-3].[jp][pn]g`

```
2.jpg  3.png
```

[22]: `cd`

[23]: `ls -l star.mp3`

```
-rw-r--r-- 1 jupyter-kkenzh jupyter-kkenzh 0 Apr  4 03:25 star.mp3
```

[25]: `chmod u+x star.mp3`

[26]: `ls -l star.mp3`

```
      -rwxr--r-- 1 jupyter-kkenzh jupyter-kkenzh 0 Apr  4 03:25 star.mp3
```

[29]: `chmod ugo-r star.mp3`

[30]: `ls -l star.mp3`

```
      --wx------ 1 jupyter-kkenzh jupyter-kkenzh 0 Apr  4 03:25 star.mp3
```

[31]: `chmod ugo+wrx star.mp3`

[33]: `ls -l star.mp3`

```
      -rwxrwxrwx 1 jupyter-kkenzh jupyter-kkenzh 0 Apr  4 03:25 star.mp3
```

[34]: `man chmod`

```
      CHMOD(1)                      User Commands                      CHMOD(1)

      NAME
             chmod - change file mode bits

      SYNOPSIS
             chmod [OPTION]… MODE[,MODE]… FILE…
             chmod [OPTION]… OCTAL-MODE FILE…
             chmod [OPTION]… --reference=RFILE FILE…

      DESCRIPTION
             This manual page documents the GNU version of chmod.  chmod changes the
             file mode bits of each given file  according  to  mode,  which  can  be
             either a symbolic representation of changes to make, or an octal number
             representing the bit pattern for the new mode bits.

             The format of a symbolic mode is  [ugoa…][[-+=][perms…]…],  where
             perms  is  either zero or more letters from the set rwxXst, or a single
             letter from the set ugo.  Multiple symbolic modes can be  given,  sepa-
             rated by commas.

             A  combination  of the letters ugoa controls which users' access to the
             file will be changed: the user who owns it  (u),  other  users  in  the
             file's group (g), other users not in the file's group (o), or all users
             (a).  If none of these are given, the effect is as if (a)  were  given,
             but bits that are set in the umask are not affected.

             The  operator  +  causes the selected file mode bits to be added to the
             existing file mode bits of each file; - causes them to be removed;  and
             =  causes  them  to  be added and causes unmentioned bits to be removed
             except that a directory's unmentioned set user and group  ID  bits  are
             not affected.
```

The letters rwxXst select file mode bits for the affected users: read
(r), write (w), execute (or search for directories) (x), execute/search
only if the file is a directory or already has execute permission for
some user (X), set user or group ID on execution (s), restricted dele-
tion flag or sticky bit (t). Instead of one or more of these letters,
you can specify exactly one of the letters ugo: the permissions granted
to the user who owns the file (u), the permissions granted to other
users who are members of the file's group (g), and the permissions
granted to users that are in neither of the two preceding categories
(o).

A numeric mode is from one to four octal digits (0-7), derived by
adding up the bits with values 4, 2, and 1. Omitted digits are assumed
to be leading zeros. The first digit selects the set user ID (4) and
set group ID (2) and restricted deletion or sticky (1) attributes. The
second digit selects permissions for the user who owns the file: read
(4), write (2), and execute (1); the third selects permissions for
other users in the file's group, with the same values; and the fourth
for other users not in the file's group, with the same values.

chmod never changes the permissions of symbolic links; the chmod system
call cannot change their permissions. This is not a problem since the
permissions of symbolic links are never used. However, for each sym-
bolic link listed on the command line, chmod changes the permissions of
the pointed-to file. In contrast, chmod ignores symbolic links encoun-
tered during recursive directory traversals.

SETUID AND SETGID BITS
       chmod clears the set-group-ID bit of a regular file if the file's group
       ID does not match the user's effective group ID or one of the user's
       supplementary group IDs, unless the user has appropriate privileges.
       Additional restrictions may cause the set-user-ID and set-group-ID bits
       of MODE or RFILE to be ignored. This behavior depends on the policy
       and functionality of the underlying chmod system call. When in doubt,
       check the underlying system behavior.

       chmod preserves a directory's set-user-ID and set-group-ID bits unless
       you explicitly specify otherwise. You can set or clear the bits with
       symbolic modes like u+s and g-s, and you can set (but not clear) the
       bits with a numeric mode.

RESTRICTED DELETION FLAG OR STICKY BIT
       The restricted deletion flag or sticky bit is a single bit, whose
       interpretation depends on the file type. For directories, it prevents
       unprivileged users from removing or renaming a file in the directory
       unless they own the file or the directory; this is called the
       restricted deletion flag for the directory, and is commonly found on

world-writable directories like /tmp. For regular files on some older
systems, the bit saves the program's text image on the swap device so
it will load more quickly when run; this is called the sticky bit.

OPTIONS
Change the mode of each FILE to MODE.   With --reference, change the
mode of each FILE to that of RFILE.

-c, --changes
        like verbose but report only when a change is made

-f, --silent, --quiet
        suppress most error messages

-v, --verbose
        output a diagnostic for every file processed

--no-preserve-root
        do not treat '/' specially (the default)

--preserve-root
        fail to operate recursively on '/'

--reference=RFILE
        use RFILE's mode instead of MODE values

-R, --recursive
        change files and directories recursively

--help display this help and exit

--version
        output version information and exit

Each   MODE   is   of   the   form
'[ugoa]*([-+=]([rwxXst]*|[ugo]))+|[-+=][0-7]+'.

AUTHOR
Written by David MacKenzie and Jim Meyering.

REPORTING BUGS
GNU coreutils online help: <http://www.gnu.org/software/coreutils/>
Report chmod translation bugs to <http://translationproject.org/team/>

COPYRIGHT
Copyright © 2017 Free Software Foundation, Inc.   License GPLv3+: GNU
GPL version 3 or later <http://gnu.org/licenses/gpl.html>.
This is free software: you are free to change and redistribute it.

```
       There is NO WARRANTY, to the extent permitted by law.

SEE ALSO
       chmod(2)

       Full documentation at: <http://www.gnu.org/software/coreutils/chmod>
       or available locally via: info '(coreutils) chmod invocation'

GNU coreutils 8.28              January 2018                      CHMOD(1)
```

[ ]: