

임베디드응용및실습

-리눅스 간단 명령 수행하기

학과	정보통신공학과
학번	2020161010
이름	김기훈

<파일 압축 해제>

1. IFC181.tar 파일을 받아 ~/embedded/week5 로 옮긴다.

```
• kihon@pi:~ $ mkdir embedded
• kihon@pi:~ $ cd embedded
• kihon@pi:~/embedded $ mkdir week5
```

```
• kihon@pi:~/embedded $ cd ~
```

```
• kihon@pi:~ $ mv IFC181.tar ~/embedded/week5
```

▼ embedded / week5

≡ IFC181.tar

2. tar 명령으로 압축을 해제한다.

```
• kihon@pi:~ $ cd embedded/week5
• kihon@pi:~/embedded/week5 $ tar -xvf IFC181.tar
IFC181_14.pdf
IFC181_01.pdf
IFC181_02.pdf
IFC181_03.pdf
IFC181_04.pdf
.....
```

Open file in editor (ctrl + click)

```
IFC181_07.pdf
IFC181_08.pdf
IFC181_09.pdf
IFC181_10.pdf
IFC181_11.pdf
IFC181_12.pdf
IFC181_13.pdf
```

▼ embedded / week5

```
🔥 IFC181_01.pdf
🔥 IFC181_02.pdf
🔥 IFC181_03.pdf
🔥 IFC181_04.pdf
🔥 IFC181_05.pdf
🔥 IFC181_06.pdf
🔥 IFC181_07.pdf
🔥 IFC181_08.pdf
🔥 IFC181_09.pdf
🔥 IFC181_10.pdf
🔥 IFC181_11.pdf
🔥 IFC181_12.pdf
🔥 IFC181_13.pdf
🔥 IFC181_14.pdf
≡ IFC181.tar
```

3. 2에서 압축해제된 파일들을 IFC181_re.tar로 압축한다.

```
● kihon@pi:~/embedded/week5 $ tar -cvf IFC181_re.tar IFC181*.pdf
IFC181_01.pdf
IFC181_02.pdf
IFC181_03.pdf
IFC181_04.pdf
IFC181_05.pdf
IFC181_06.pdf
IFC181_07.pdf
IFC181_08.pdf
IFC181_09.pdf
IFC181_10.pdf
IFC181_11.pdf
IFC181_12.pdf
IFC181_13.pdf
IFC181_14.pdf
```

▼ embedded/week5

```
└─ IFC181_01.pdf
└─ IFC181_02.pdf
└─ IFC181_03.pdf
└─ IFC181_04.pdf
└─ IFC181_05.pdf
└─ IFC181_06.pdf
└─ IFC181_07.pdf
└─ IFC181_08.pdf
└─ IFC181_09.pdf
└─ IFC181_10.pdf
└─ IFC181_11.pdf
└─ IFC181_12.pdf
└─ IFC181_13.pdf
└─ IFC181_14.pdf
└─ IFC181_re.tar
```

4. 2에서 압축해제된 파일들을 zip명령으로 IFC181_re.zip으로 압축한다.

- 이때 zip 명령을 수행할 수 있도록 zip을 설치한다.

: sudo apt-get install zip

```
● kihon@pi:~/embedded/week5 $ zip IFC181_re.zip IFC181*.pdf
  adding: IFC181_01.pdf (deflated 20%)
  adding: IFC181_02.pdf (deflated 5%)
  adding: IFC181_03.pdf (deflated 22%)
  adding: IFC181_04.pdf (deflated 19%)
  adding: IFC181_05.pdf (deflated 13%)
  adding: IFC181_06.pdf (deflated 15%)
  adding: IFC181_07.pdf (deflated 13%)
  adding: IFC181_08.pdf (deflated 15%)
  adding: IFC181_09.pdf (deflated 10%)
  adding: IFC181_10.pdf (deflated 21%)
  adding: IFC181_11.pdf (deflated 15%)
  adding: IFC181_12.pdf (deflated 18%)
  adding: IFC181_13.pdf (deflated 17%)
  adding: IFC181_14.pdf (deflated 19%)
```

■ IFC181_re.zip

5. 4에서 나온 IFC181_re.zip 파일을 unzip 명령을 통해 압축해제 한다.

```
● kihon@pi:~/embedded/week5 $ unzip IFC181_re.zip
Archive: IFC181_re.zip
replace IFC181_01.pdf? [y]es, [n]o, [A]ll, [N]one, [r]ename: A
  inflating: IFC181_01.pdf
  inflating: IFC181_02.pdf
  inflating: IFC181_03.pdf
  inflating: IFC181_04.pdf
  inflating: IFC181_05.pdf
  inflating: IFC181_06.pdf
  inflating: IFC181_07.pdf
  inflating: IFC181_08.pdf
  inflating: IFC181_09.pdf
  inflating: IFC181_10.pdf
  inflating: IFC181_11.pdf
  inflating: IFC181_12.pdf
  inflating: IFC181_13.pdf
  inflating: IFC181_14.pdf
```

<find 명령어 연습>

6. ~ (home) 디렉토리로 이동하고 find 명령을 통해 embedded폴더에서(하위폴더 포함) .pdf 파일을 찾는 명령을 수행해본다.

```
● kihon@pi:~ $ find ~/embedded -name "*.pdf"
/home/kihon/embedded/week5/IFC181_05.pdf
/home/kihon/embedded/week5/IFC181_04.pdf
/home/kihon/embedded/week5/IFC181_08.pdf
/home/kihon/embedded/week5/IFC181_12.pdf
/home/kihon/embedded/week5/IFC181_09.pdf
/home/kihon/embedded/week5/IFC181_07.pdf
/home/kihon/embedded/week5/IFC181_13.pdf
/home/kihon/embedded/week5/IFC181_02.pdf
/home/kihon/embedded/week5/IFC181_01.pdf
/home/kihon/embedded/week5/IFC181_14.pdf
/home/kihon/embedded/week5/IFC181_11.pdf
/home/kihon/embedded/week5/IFC181_10.pdf
/home/kihon/embedded/week5/IFC181_03.pdf
/home/kihon/embedded/week5/IFC181_06.pdf
```

7. 아래의 명령 결과가 동일한지 아닌지 결과를 보이고 동작 결과를 설명하시오
동작결과 설명: 1,2,3은 find 다음에 .을 써서 파일경로가 ./형태로 시작했지만
4,5는 find 다음에 ~를 써서 /home/kihon이 붙었다. 아래는 결과사진 첨부입니다.

innosm@innosm:~ \$find . -name *.pdf

```
● kihon@pi:~ $ find . -name *.pdf
./Bookshelf/BeginnersGuide-4thEd-Eng_v2.pdf
./embedded/week5/IFC181_05.pdf
./embedded/week5/IFC181_04.pdf
./embedded/week5/IFC181_08.pdf
./embedded/week5/IFC181_12.pdf
./embedded/week5/IFC181_09.pdf
./embedded/week5/IFC181_07.pdf
./embedded/week5/IFC181_13.pdf
./embedded/week5/IFC181_02.pdf
./embedded/week5/IFC181_01.pdf
./embedded/week5/IFC181_14.pdf
./embedded/week5/IFC181_11.pdf
./embedded/week5/IFC181_10.pdf
./embedded/week5/IFC181_03.pdf
./embedded/week5/IFC181_06.pdf
```


innosm@innosm:~ \$find ./ -name *.pdf

```
● kihon@pi:~ $ find ./ -name *.pdf
./Bookshelf/BeginnersGuide-4thEd-Eng_v2.pdf
./embedded/week5/IFC181_05.pdf
./embedded/week5/IFC181_04.pdf
./embedded/week5/IFC181_08.pdf
./embedded/week5/IFC181_12.pdf
./embedded/week5/IFC181_09.pdf
./embedded/week5/IFC181_07.pdf
./embedded/week5/IFC181_13.pdf
./embedded/week5/IFC181_02.pdf
./embedded/week5/IFC181_01.pdf
./embedded/week5/IFC181_14.pdf
./embedded/week5/IFC181_11.pdf
./embedded/week5/IFC181_10.pdf
./embedded/week5/IFC181_03.pdf
./embedded/week5/IFC181_06.pdf
```

innosm@innosm:~ \$find -name *.pdf

```
● kihon@pi:~ $ find -name *.pdf
./Bookshelf/BeginnersGuide-4thEd-Eng_v2.pdf
./embedded/week5/IFC181_05.pdf
./embedded/week5/IFC181_04.pdf
./embedded/week5/IFC181_08.pdf
./embedded/week5/IFC181_12.pdf
./embedded/week5/IFC181_09.pdf
./embedded/week5/IFC181_07.pdf
./embedded/week5/IFC181_13.pdf
./embedded/week5/IFC181_02.pdf
./embedded/week5/IFC181_01.pdf
./embedded/week5/IFC181_14.pdf
./embedded/week5/IFC181_11.pdf
./embedded/week5/IFC181_10.pdf
./embedded/week5/IFC181_03.pdf
./embedded/week5/IFC181_06.pdf
```

innosm@innosm:~ \$find ~ -name *.pdf

```
● kihon@pi:~ $ find ~ -name *.pdf
/home/kihon/Bookshelf/BeginnersGuide-4thEd-Eng_v2.pdf
/home/kihon/embedded/week5/IFC181_05.pdf
/home/kihon/embedded/week5/IFC181_04.pdf
/home/kihon/embedded/week5/IFC181_08.pdf
/home/kihon/embedded/week5/IFC181_12.pdf
/home/kihon/embedded/week5/IFC181_09.pdf
/home/kihon/embedded/week5/IFC181_07.pdf
/home/kihon/embedded/week5/IFC181_13.pdf
/home/kihon/embedded/week5/IFC181_02.pdf
/home/kihon/embedded/week5/IFC181_01.pdf
/home/kihon/embedded/week5/IFC181_14.pdf
/home/kihon/embedded/week5/IFC181_11.pdf
/home/kihon/embedded/week5/IFC181_10.pdf
/home/kihon/embedded/week5/IFC181_03.pdf
/home/kihon/embedded/week5/IFC181_06.pdf
```

innosm@innosm:~ \$find /home/innosm -name *.pdf

```
● kihon@pi:~ $ find /home/kihon -name *.pdf
/home/kihon/Bookshelf/BeginnersGuide-4thEd-Eng_v2.pdf
/home/kihon/embedded/week5/IFC181_05.pdf
/home/kihon/embedded/week5/IFC181_04.pdf
/home/kihon/embedded/week5/IFC181_08.pdf
/home/kihon/embedded/week5/IFC181_12.pdf
/home/kihon/embedded/week5/IFC181_09.pdf
/home/kihon/embedded/week5/IFC181_07.pdf
/home/kihon/embedded/week5/IFC181_13.pdf
/home/kihon/embedded/week5/IFC181_02.pdf
/home/kihon/embedded/week5/IFC181_01.pdf
/home/kihon/embedded/week5/IFC181_14.pdf
/home/kihon/embedded/week5/IFC181_11.pdf
/home/kihon/embedded/week5/IFC181_10.pdf
/home/kihon/embedded/week5/IFC181_03.pdf
/home/kihon/embedded/week5/IFC181_06.pdf
```

8. ~ (home) 디렉토리로 이동하고, week5 폴더가 있는지 검색하려고 한다.
적절한 명령을 수행하여 week5 폴더를 검색하고 결과를 보이시오.

```
● kihon@pi:~ $ find ./ -name week5
./embedded/week5
```

<파일 용량 확인>

9. 아래 명령을 수행한 결과를 보이시오.

df -h

```
● kihon@pi:~ $ df -h
Filesystem      Size  Used Avail Use% Mounted on
/dev/root        15G   3.5G   11G   26% /
devtmpfs         1.8G     0   1.8G    0% /dev
tmpfs            1.9G     0   1.9G    0% /dev/shm
tmpfs            1.9G   8.6M   1.9G    1% /run
tmpfs            5.0M   4.0K   5.0M    1% /run/lock
tmpfs            1.9G     0   1.9G    0% /sys/fs/cgroup
/dev/mmcblk0p1  253M   49M  204M   20% /boot
tmpfs            384M   4.0K   384M    1% /run/user/1000
```

10. 아래 명령을 수행한 결과를 보이시오.

cd ~

cd embedded

du -h

```
● kihon@pi:~ $ cd ~
● kihon@pi:~ $ cd embedded
● kihon@pi:~/embedded $ du -h
16M   ./week5
16M   .
```

11. 10번에서 현재 embedded 폴더의 총 사용량은(하위폴더 포함, 단위 표시)?

답: 16M

12. cd embedded/week5 를 수행하여 위치를 이동하고, 아래 명령을 차례로 수행하시오

1) df .

```
● kihon@pi:~/embedded/week5 $ df .  
Filesystem      1K-blocks    Used Available Use% Mounted on  
/dev/root        14986204 3641088  10683608  26% /
```

2) 이 폴더에 temp_file 파일 생성하고 파일에 1을 기록

```
● kihon@pi:~/embedded/week5 $ touch temp_file  
● kihon@pi:~/embedded/week5 $ echo 1 > temp_file
```

3) stat temp_file 명령을 통해 파일의 크기를 확인

```
● kihon@pi:~/embedded/week5 $ stat temp_file  
File: temp_file  
Size: 2          Blocks: 8          IO Block: 4096   일반 파일  
Device: b302h/45826d Inode: 264826       Links: 1  
Access: (0644/-rw-r--r--)  Uid: ( 1000/   kihon)   Gid: ( 1000/   kihon)  
Access: 2024-10-07 22:30:27.112138858 +0900  
Modify: 2024-10-07 22:30:29.272094990 +0900  
Change: 2024-10-07 22:30:29.272094990 +0900  
Birth: -
```

4) df .

```
● kihon@pi:~/embedded/week5 $ df .  
Filesystem      1K-blocks    Used Available Use% Mounted on  
/dev/root        14986204 3641092  10683604  26% /
```

1)과 4)에서 줄어든 용량과 3)에서 확인한 용량이 다르다면 그 이유는?

파일의 실제 크기는 2바이트지만 파일을 블록 단위로 저장된다. 1블록은 4k 바이트이다. 따라서 1, 4와 3에서 확인한 용량에서 차이가 나는 것이다.

<cpu architecture>

13. 현재 사용하는 라즈비언 OS가 몇비트 시스템인지 확인하고 결과를 첨부하시오.

```
● kihon@pi:~ $ uname -m  
armv7l
```

32비트 시스템이다.