

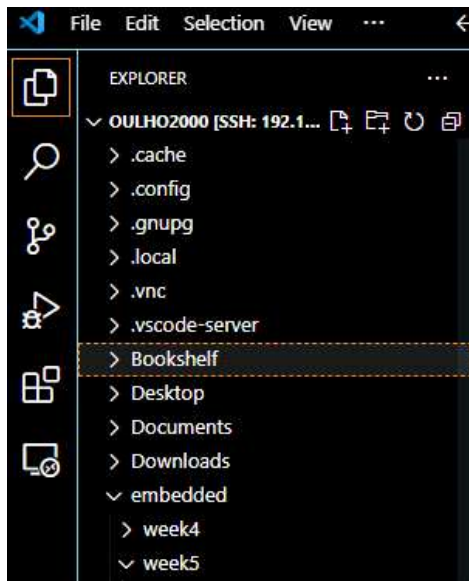
임베디드 응용 및 실습

학과 : 정보통신공학과

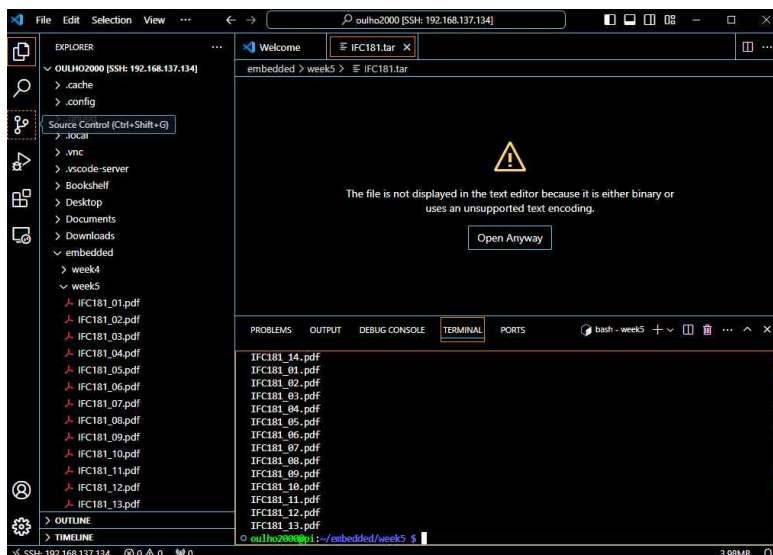
학번 : 2022161094

이름 : 이율호

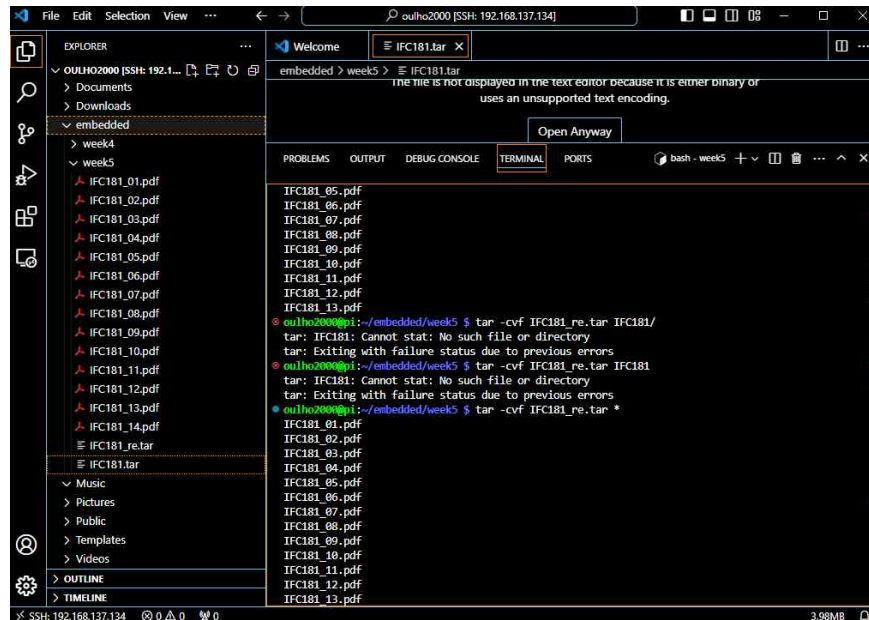
1. IFC181.tar 파일을 받아 ~/embedded/week5 로 옮긴다.



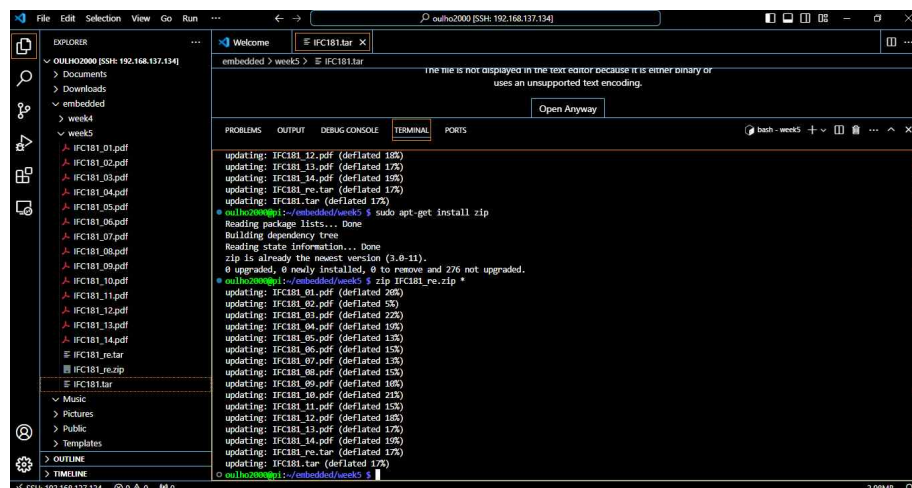
2. tar 명령으로 압축을 해제한다.



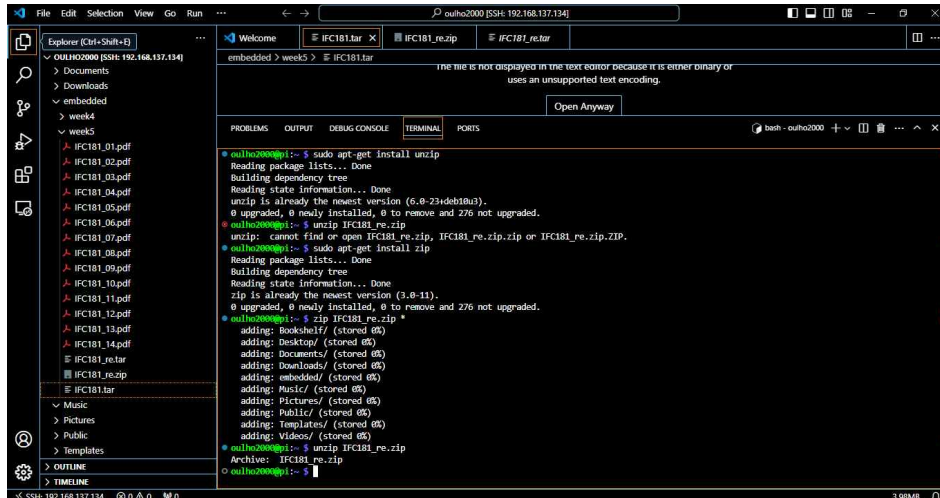
3. 2에서 압축해제된 파일들을 IFC181_re.tar로 압축한다.



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



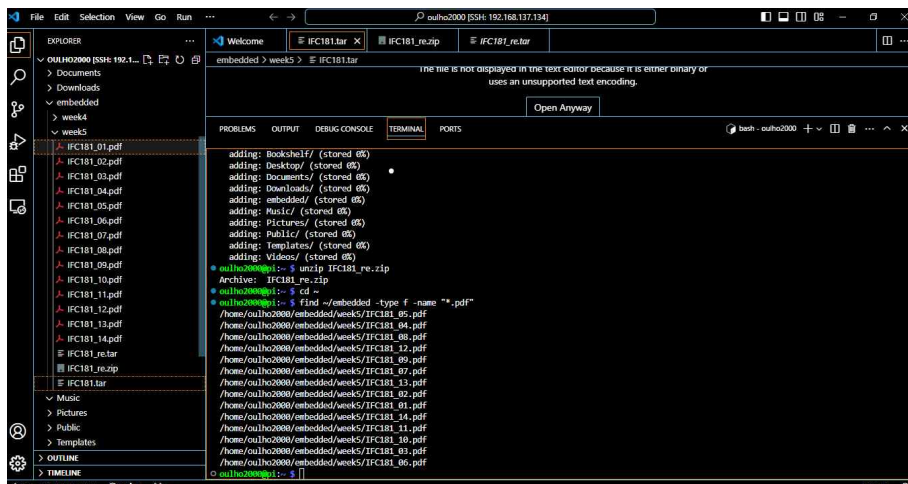
5. 4에서 나온 IFC181_re.zip 파일을 unzip 명령을 통해 압축해제 한다.



```
oalha2000@oalha2000 [SSH: 192.168.137.134]
embedded > weeks > IFC181.tar
The file is not displayed in the text editor because it is either binary or
uses an unsupported text encoding.
[Open Anyway]

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
oalha2000@oalha2000:~$ sudo apt-get install unzip
Reading package lists... Done
Building dependency tree
Reading state information... Done
unzip is already the newest version (6.0-23+deb10u1).
0 upgraded, 0 newly installed, 0 to remove and 276 not upgraded.
oalha2000@oalha2000:~$ unzip IFC181_re.zip
unzip: cannot find or open IFC181_re.zip, IFC181_re.zip.zip or IFC181_re.zip.ZIP.
oalha2000@oalha2000:~$ sudo apt-get install zip
Reading package lists... Done
Building dependency tree
Reading state information... Done
zip is already the newest version (3.0-11).
0 upgraded, 0 newly installed, 0 to remove and 276 not upgraded.
oalha2000@oalha2000:~$ zip IFC181_re.zip *
adding: Bookshelf/ (stored 0%)
adding: Desktop/ (stored 0%)
adding: Documents/ (stored 0%)
adding: Downloads/ (stored 0%)
adding: embedded/ (stored 0%)
adding: Music/ (stored 0%)
adding: Pictures/ (stored 0%)
adding: Public/ (stored 0%)
adding: Templates/ (stored 0%)
adding: Videos/ (stored 0%)
oalha2000@oalha2000:~$ unzip IFC181_re.zip
Archive:  IFC181_re.zip
oalha2000@oalha2000:~$
```

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



```
oalha2000 [SSH: 192.168.137.134]
embedded > weeks > IFC181.tar
The file is not displayed in the text editor because it is either binary or
uses an unsupported text encoding.
[Open Anyway]

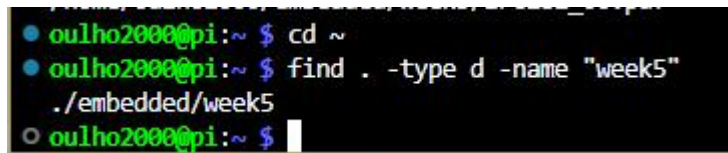
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
oalha2000@oalha2000:~$ find ~/embedded -type f -name "*.pdf"
/home/oalha2000/embedded/weeks/IFC181_06.pdf
/home/oalha2000/embedded/weeks/IFC181_04.pdf
/home/oalha2000/embedded/weeks/IFC181_08.pdf
/home/oalha2000/embedded/weeks/IFC181_12.pdf
/home/oalha2000/embedded/weeks/IFC181_09.pdf
/home/oalha2000/embedded/weeks/IFC181_07.pdf
/home/oalha2000/embedded/weeks/IFC181_13.pdf
/home/oalha2000/embedded/weeks/IFC181_02.pdf
/home/oalha2000/embedded/weeks/IFC181_01.pdf
/home/oalha2000/embedded/weeks/IFC181_14.pdf
/home/oalha2000/embedded/weeks/IFC181_11.pdf
/home/oalha2000/embedded/weeks/IFC181_10.pdf
/home/oalha2000/embedded/weeks/IFC181_03.pdf
oalha2000@oalha2000:~$
```

7. 아래의 명령 결과가 동일한지 아닌지 결과를 보이고 동작 결과를 설명하시오

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

위 명령어들은 파일 검색 경로가 다르기 때문에 결과가 동일하지 않으며, 각 명령어의 동작 방식에 따라 검색 범위가 달라진다.

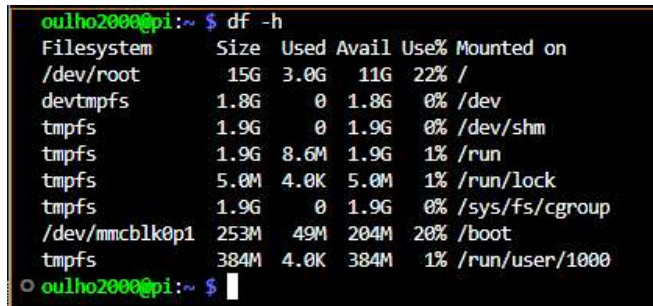
8. ~ (home) 디렉토리로 이동하고, week5 폴더가 있는지 검색하려고 한다.
적절한 명령을 수행하여 week5 폴더를 검색하고 결과를 보이시오.



```
oulho2000@pi:~ $ cd ~
oulho2000@pi:~ $ find . -type d -name "week5"
./embedded/week5
oulho2000@pi:~ $
```

9. 아래 명령을 수행한 결과를 보이시오.

df -h



```
oulho2000@pi:~ $ df -h
Filesystem      Size  Used Avail Use% Mounted on
/dev/root        15G   3.0G   11G   22% /
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
oulho2000@pi:~ $
```

10. 아래 명령을 수행한 결과를 보이시오.

```
cd ~
```

```
cd embedded
```

```
du -h
```

```
oulho2000@pi:~ $ cd ~
oulho2000@pi:~ $ cd embedded
oulho2000@pi:~/embedded $ du -h
4.0K    ./week4
30M     ./week5
30M     .
```

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

마지막 줄이 embedded 폴더의 총 사용량을 나타낸다

: 30MB

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

1) df .

2) 이 폴더에 temp_file 파일 생성하고 파일에 1을 기록

3) stat temp_file 명령을 통해 파일의 크기를 확인

4) df .

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

1)

```
oulho2000@pi:~/embedded $ df
Filesystem      1K-blocks    Used Available Use% Mounted on
/dev/root       14986204 3125764  11198932  22% /
devtmpfs        1800568      0    1800568   0% /dev
tmpfs           1965432      0    1965432   0% /dev/shm
tmpfs           1965432    8728    1956704   1% /run
tmpfs            5120         4        5116   1% /run/lock
tmpfs           1965432      0    1965432   0% /sys/fs/cgroup
/dev/mmcblk0p1   258095    49323    208772  20% /boot
tmpfs           393084         4    393080   1% /run/user/1000
```

2)

```
oulho2000@pi:~/embedded $ cd ~/embedded
oulho2000@pi:~/embedded $ echo "1" > temp_file
oulho2000@pi:~/embedded $ cat temp_file
1
```

3)

```
oulho2000@pi:~/embedded $ cd ~/embedded
oulho2000@pi:~/embedded $ stat temp_file
  File: temp_file
  Size: 2          Blocks: 8          IO Block: 4096   regular file
Device: b302h/45826d  Inode: 263642      Links: 1
Access: (0644/-rw-r--r--)  Uid: ( 1000/oulho2000)   Gid: ( 1000/oulho2000)
Access: 2024-10-08 09:47:21.797134217 +0100
Modify: 2024-10-08 09:47:21.797134217 +0100
Change: 2024-10-08 09:47:21.797134217 +0100
 Birth: -
```

4)

```
oulho2000@pi:~/embedded $ df
Filesystem      1K-blocks    Used Available Use% Mounted on
/dev/root        14986204 3125808  11198888  22% /
devtmpfs         1800568      0    1800568   0% /dev
tmpfs            1965432      0    1965432   0% /dev/shm
tmpfs            1965432    8736    1956696   1% /run
tmpfs             5120         4       5116   1% /run/lock
tmpfs            1965432      0    1965432   0% /sys/fs/cgroup
/dev/mmcblk0p1   258095    49323    208772  20% /boot
tmpfs            393084      4    393080   1% /run/user/1000
```

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

df는 파일 시스템에서 블록 단위로 사용량을 표시하고 stat는 파일의 실제 크기를 바이트 단위로 보여주기 때문이다.

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

```
oulho2000@pi:~/embedded $ uname -m
armv7l
```

따라서 현재 사용하는 라즈비언 OS는 32비트 시스템이다.