1- Inside iot_logger, create logs/temperature.log and scripts/sensor_script.py.

```
ghannam@Ghannam:~/iot_logger$ touch logs/temperature.log scripts/sensor_script.py
ghannam@Ghannam:~/iot_logger$
```

2- Copy /etc/services into data and search for patterns like ssh or http.

```
ghannam@Ghannam:~/iot_logger$ cp /etc/services data
ghannam@Ghannam:~/iot_logger$ ls data
services
```

3- Use regex to find lines starting with t or containing numbers.

```
ghannam@Ghannam:~/iot_logger$ grep -E "^t|[0-9]" data/services
tcpmux          1/tcp                           # TCP port service multiplexer
echo            7/tcp
echo            7/udp
discard         9/tcp           sink null
discard         9/udp           sink null
systat          11/tcp          users
daytime         13/tcp
daytime         13/udp
netstat         15/tcp
qotd            17/tcp          quote
chargen         19/tcp          ttytst source
chargen         19/udp          ttytst source
ftp-data        20/tcp
ftp             21/tcp
fsp             21/udp          fspd
ssh             22/tcp                          # SSH Remote Login Protocol
telnet          23/tcp
smtp            25/tcp          mail
time            37/tcp          timserver
time            37/udp          timserver
whois           43/tcp          nicname
tacacs          49/tcp                          # Login Host Protocol (TACACS)
tacacs          49/udp
domain          53/tcp                          # Domain Name Server
domain          53/udp
bootps          67/udp
bootpc          68/udp
tftp            69/udp
gopher          70/tcp                          # Internet Gopher
finger          79/tcp
http            80/tcp          www             # WorldWideWeb HTTP
kerberos        88/tcp          kerberos5 krb5 kerberos-sec     # Kerberos v5
kerberos        88/udp          kerberos5 krb5 kerberos-sec     # Kerberos v5
iso-tsap        102/tcp         tsap            # part of ISODE
```

```
iso-tsap        102/tcp         tsap            # part of ISODE
acr-nema        104/tcp         dicom           # Digital Imag. & Comm. 300
pop3            110/tcp         pop-3           # POP version 3
sunrpc          111/tcp         portmapper      # RPC 4.0 portmapper
sunrpc          111/udp         portmapper
auth            113/tcp         authentication tap ident
nntp            119/tcp         readnews untp   # USENET News Transfer Protocol
ntp             123/udp                         # Network Time Protocol
epmap           135/tcp         loc-srv         # DCE endpoint resolution
netbios-ns      137/udp                         # NETBIOS Name Service
netbios-dgm     138/udp                         # NETBIOS Datagram Service
netbios-ssn     139/tcp                         # NETBIOS session service
imap2           143/tcp         imap            # Interim Mail Access P 2 and 4
snmp            161/tcp                         # Simple Net Mgmt Protocol
snmp            161/udp
snmp-trap       162/tcp         snmptrap        # Traps for SNMP
snmp-trap       162/udp         snmptrap
cmip-man        163/tcp                         # ISO mgmt over IP (CMOT)
cmip-man        163/udp
cmip-agent      164/tcp
cmip-agent      164/udp
mailq           174/tcp                 # Mailer transport queue for Zmailer
xdmcp           177/udp                 # X Display Manager Control Protocol
bgp             179/tcp                         # Border Gateway Protocol
smux            199/tcp                         # SNMP Unix Multiplexer
qmtp            209/tcp                         # Quick Mail Transfer Protocol
z3950           210/tcp         wais            # NISO Z39.50 database
ipx             213/udp                         # IPX [RFC1234]
ptp-event       319/udp
ptp-general     320/udp
pawserv         345/tcp                         # Perf Analysis Workbench
zserv           346/tcp                         # Zebra server
rpc2portmap     369/tcp
rpc2portmap     369/udp                         # Coda portmapper
codaauth2       370/tcp
codaauth2       370/udp                         # Coda authentication server
clearcase       371/udp         Clearcase
```

```
zebrasrv        2600/tcp                        # zebra service
zebra           2601/tcp                        # zebra vty
ripd            2602/tcp                        # ripd vty (zebra)
ripngd          2603/tcp                        # ripngd vty (zebra)
ospfd           2604/tcp                        # ospfd vty (zebra)
bgpd            2605/tcp                        # bgpd vty (zebra)
ospf6d          2606/tcp                        # ospf6d vty (zebra)
ospfapi         2607/tcp                        # OSPF-API
isisd           2608/tcp                        # ISISd vty (zebra)
fax             4557/tcp                        # FAX transmission service (old)
hylafax         4559/tcp                        # HylaFAX client-server protocol (new)
munin           4949/tcp        lrrd            # Munin
rplay           5555/udp                        # RPlay audio service
nrpe            5666/tcp                        # Nagios Remote Plugin Executor
nsca            5667/tcp                        # Nagios Agent - NSCA
canna           5680/tcp                        # cannaserver
syslog-tls      6514/tcp                        # Syslog over TLS [RFC5425]
sane-port       6566/tcp        sane saned      # SANE network scanner daemon
ircd            6667/tcp                        # Internet Relay Chat
zope-ftp        8021/tcp                        # zope management by ftp
tproxy          8081/tcp                        # Transparent Proxy
omniorb         8088/tcp                        # OmniORB
clc-build-daemon 8990/tcp                       # Common lisp build daemon
xinetd          9098/tcp
git             9418/tcp                        # Git Version Control System
zope            9673/tcp                        # zope server
webmin          10000/tcp
kamanda         10081/tcp                       # amanda backup services (Kerberos)
amandaidx       10082/tcp                       # amanda backup services
amidxtape       10083/tcp                       # amanda backup services
sgi-cmsd        17001/udp               # Cluster membership services daemon
sgi-crsd        17002/udp
sgi-gcd         17003/udp                       # SGI Group membership daemon
sgi-cad         17004/tcp                       # Cluster Admin daemon
binkp           24554/tcp                       # binkp fidonet protocol
asp             27374/tcp                       # Address Search Protocol
asp             27374/udp
```

4- Locate .txt files in /home/<username> and remove temporary ones if needed.

```
ghannam@Ghannam:~$ find -type f -name "*.txt"
./ros2_ws/build/my_py_pkg/my_py_pkg.egg-info/requires.txt
./ros2_ws/build/my_py_pkg/my_py_pkg.egg-info/SOURCES.txt
./ros2_ws/build/my_py_pkg/my_py_pkg.egg-info/dependency_links.txt
./ros2_ws/build/my_py_pkg/my_py_pkg.egg-info/top_level.txt
./ros2_ws/build/my_py_pkg/my_py_pkg.egg-info/entry_points.txt
./ros2_ws/install/my_py_pkg/lib/python3.12/site-packages/my_py_pkg-0.0.0-py3.12.egg-info/requires.txt
./ros2_ws/install/my_py_pkg/lib/python3.12/site-packages/my_py_pkg-0.0.0-py3.12.egg-info/SOURCES.txt
./ros2_ws/install/my_py_pkg/lib/python3.12/site-packages/my_py_pkg-0.0.0-py3.12.egg-info/dependency_links.txt
./ros2_ws/install/my_py_pkg/lib/python3.12/site-packages/my_py_pkg-0.0.0-py3.12.egg-info/top_level.txt
./ros2_ws/install/my_py_pkg/lib/python3.12/site-packages/my_py_pkg-0.0.0-py3.12.egg-info/entry_points.txt
./.cache/tracker3/files/first-index.txt
./.cache/tracker3/files/last-crawl.txt
./.pki/nssdb/pkcs11.txt
./venvs/mqtt/lib/python3.12/site-packages/paho_mqtt-1.6.1.dist-info/licenses/LICENSE.txt
./venvs/mqtt/lib/python3.12/site-packages/paho_mqtt-1.6.1.dist-info/top_level.txt
./venvs/mqtt/lib/python3.12/site-packages/pip/_vendor/vendor.txt
./venvs/mqtt/lib/python3.12/site-packages/pip-24.0.dist-info/LICENSE.txt
./venvs/mqtt/lib/python3.12/site-packages/pip-24.0.dist-info/AUTHORS.txt
./venvs/mqtt/lib/python3.12/site-packages/pip-24.0.dist-info/top_level.txt
./venvs/mqtt/lib/python3.12/site-packages/pip-24.0.dist-info/entry_points.txt
./snap/firefox/common/.mozilla/firefox/6j06oucs.default/pkcs11.txt
./.vscode/extensions/twxs.cmake-0.0.17/node_modules/opener/LICENSE.txt
./.vscode/extensions/twxs.cmake-0.0.17/node_modules/peg-cmake/test/CMakeLists.txt
./.vscode/extensions/twxs.cmake-0.0.17/LICENSE.txt
./.vscode/extensions/ms-python.vscode-pylance-2025.7.1/NOTICE.txt
./.vscode/extensions/ms-python.vscode-pylance-2025.7.1/LICENSE.txt
./.vscode/extensions/ms-python.vscode-pylance-2025.7.1/dist/typeshed-fallback/stubs/vobject/@tests/stubtest_allowlist.txt
./.vscode/extensions/ms-python.vscode-pylance-2025.7.1/dist/typeshed-fallback/commit.txt
./.vscode/extensions/ms-python.vscode-pylance-2025.7.1/dist/bundled/stubs/skimage/stubtest_allowlist.txt
./.vscode/extensions/ms-python.vscode-pylance-2025.7.1/dist/bundled/stubs/vispy/stubtest_allowlist.txt
./.vscode/extensions/ms-python.vscode-pylance-2025.7.1/dist/bundled/stubs/sklearn/stubtest_allowlist.txt
./.vscode/extensions/ms-python.vscode-pylance-2025.7.1/dist/bundled/stubs/sympy-stubs/stubtest_allowlist.txt
./.vscode/extensions/ms-python.vscode-pylance-2025.7.1/dist/vendor.bundle.is.LICENSE.txt
```

5- Create hard and symbolic links for temperature.log.

```
ghannam@Ghannam:~/iot_logger$ ls
data  logs  scripts
ghannam@Ghannam:~/iot_logger$ ln -s logs/temperature.log soft_temp.log
ghannam@Ghannam:~/iot_logger$ ls
data  logs  scripts  soft_temp.log
ghannam@Ghannam:~/iot_logger$ ln logs/temperature.log hard_temp.log
ghannam@Ghannam:~/iot_logger$ ls
data  hard_temp.log  logs  scripts  soft_temp.log
ghannam@Ghannam:~/iot_logger$
```

6- Display directory structure to confirm organization.

```
ghannam@Ghannam:~/iot_logger$ tree
.
├── data
│   └── services
├── hard_temp.log
├── logs
│   └── temperature.log
├── scripts
│   └── sensor_script.py
└── soft_temp.log -> logs/temperature.log

4 directories, 5 files
ghannam@Ghannam:~/iot_logger$
```

7- Types of files in Linux

Regular file (-)

Most common type: contains data like text, images, source code, etc.

Example: notes.txt

Directory (d)

A special file that contains other files or directories.

Example: /home.

Symbolic link (l)

A shortcut (reference) pointing to another file or directory.

Example: /lib → /usr/lib.

Character device (c)

Represents devices that handle data character by character (like keyboards, serial ports).

Example: /dev/tty, /dev/random.

Block device (b)

Represents devices that handle data in blocks (like hard drives, USB storage).

Example: /dev/sda, /dev/mmcblk0.

I use "ls -l" to check the file type

Hard Link vs Symbolic Link

1. Hard Link
- A direct pointer to the file's data (inode).
- If you delete the original file, the data still exists as long as at least one hard link remains.
- Cannot link to directories (to avoid loops).
use case:
Backups — you want multiple filenames pointing to the same file to save space.

2- Symbolic Link (Soft Link)
A shortcut (reference) to another file's path.
If the original file is deleted, the symlink becomes broken (dangling).
Can link to directories.
Can cross different filesystems/partitions.
Example use case:
Shortcuts or redirection — e.g., linking config files or software versions.

rmdir VS rm -r

1- rmdir
Stands for remove directory.
Can only delete empty directories.

2- rm -r
rm stands for remove.
-r (recursive) means delete the directory and everything inside it (subdirectories, files).