Name: Basant Tarik Salah

Instructor: Eng. Mohamed Abo-Khalil

SIC7_Task.Phase2

Part1: Tasks

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

```
basant@basant-VirtualBox:~$ cd ~/iot_logger
basant@basant-VirtualBox:~/iot_logger$ touch logs/temperature.log
basant@basant-VirtualBox:~/iot_logger$ touch scripts/sensor_script.py
basant@basant-VirtualBox:~/iot_logger$
```

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

```
basant@basant-VirtualBox:~/iot_logger$
basant@basant-VirtualBox:~/iot_logger$ cp /etc/services data/
```

```
basant@basant-VirtualBox:~/iot_logger$ grep "ssh" data/services
ssh             22/tcp                          # SSH Remote Login Protocol
basant@basant-VirtualBox:~/iot_logger$ grep "http" data/services
# Updated from https://www.iana.org/assignments/service-names-port-numbers/service-names-port-numbers.xhtml .
http            80/tcp          www             # WorldWideWeb HTTP
https           443/tcp                         # http protocol over TLS/SSL
https           443/udp                         # HTTP/3
http-alt        8080/tcp        webcache        # WWW caching service
basant@basant-VirtualBox:~/iot_logger$
```

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

```
basant@basant-VirtualBox:~/iot_logger$ grep "^t" data/services
tcpmux          1/tcp                           # TCP port service multiplexer
telnet          23/tcp
time            37/tcp          timserver
time            37/udp          timserver
tacacs          49/tcp                          # Login Host Protocol (TACACS)
tacacs          49/udp
tftp            69/udp
talk            517/udp
tinc            655/tcp                         # tinc control port
tinc            655/udp
telnets         992/tcp                         # Telnet over SSL
tproxy          8081/tcp                        # Transparent Proxy
tfido           60177/tcp                       # fidonet EMSI over telnet
```

```
basant@basant-VirtualBox:~/iot_logger$ grep "[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
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
```

4. Locate .txt files in /home/ and remove temporary ones if needed:

```
basant@basant-VirtualBox:~/iot_logger$ find ~ -name "*.txt"
/home/basant/snap/firefox/common/.mozilla/firefox/49jfxvfc.default/pkcs11.txt
/home/basant/.cache/tracker3/files/last-crawl.txt
/home/basant/.cache/tracker3/files/first-index.txt
/home/basant/gitdemo/test.txt
/home/basant/gitdemo/file.txt
basant@basant-VirtualBox:~/iot_logger$
basant@basant-VirtualBox:~/iot_logger$
```

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

```
basant@basant-VirtualBox:~/iot_logger$ ln logs/temperature.log temp_hardlink.log
basant@basant-VirtualBox:~/iot_logger$ ln -s logs/temperature.log temp_symlink.log
basant@basant-VirtualBox:~/iot_logger$
```

6. Display directory structure to confirm organization:

```
basant@basant-VirtualBox:~/iot_logger$ tree ~/iot_logger
/home/basant/iot_logger
├── data
│   └── services
├── logs
│   └── temperature.log
├── scripts
│   └── sensor_script.py
├── temp_hardlink.log
└── temp_symlink.log -> logs/temperature.log

3 directories, 5 files
```

Part 2: Open Ended Questions

1. **Explain the different types of files in Linux (regular, directory, symbolic link, device, etc.) and how to check them with commands.**

Ans.

Regular files are the normal files used usually like text files, programs, scripts.

Directory is a like a folder that contains a list of other files.

Symbolic link is a shortcut points to another file.

Device files are special files enables Linux to talk to hardware.

Pipes lets processes talk to each other by enabling direct data transfer

Sockets are used for network communication.

To check their type with commands: **ls –l**

The first letter of the line refers to different file type as follows:

- = regular file

d = directory

l = symbolic link

c = character device

b = block device

p = pipe

s = socket

2. What's the difference between a hard link and a symbolic link? Give real examples of when to use each.

Hard link is giving the same file **two names**. Both names point to the exact same data on the disk. If one is deleted, the other still works.

- It is used to create a second permanent copy for the file.

Symbolic link is creating a **shortcut** points to another file. If the original file is deleted, the created file becomes broken.

- It is used to create a shortcut that always points to the latest version of the file.

3. Is rmdir the same as rm -r when deleting directories? Explain

No, they are different.

- rmdir deletes a directory only if it is empty**.** It works only if there are no files inside.
- rm –r deletes a directory and everything inside it like files or subfolders.