

Hilos y procesos

Richard Exposito

1) Teneis que hacer un codigo en python que:

- Cree 1 hilo.
- Cree 1 proceso.
- Uno de estos tiene que tener un bucle infinito.

```
import threading
import multiprocessing
import time

def hilo_infinito():
    while True:
        print("Hilo infinito")
        time.sleep(1)

def proceso_infinito():
    while True:
        print("Proceso infinito")
        time.sleep(1)

if __name__ == "__main__":
    hilo = threading.Thread(target=hilo_infinito)
    proceso = multiprocessing.Process(target=proceso_infinito)

    hilo.start()
    proceso.start()
```

- ¿Como pararias un programa infinito?

Pararía el programa infinito con ctrl+C

2) Utilizando UBUNTU SERVER

En el ejercicio anterior habeis hecho un programa infinito, responde a estas cuestiones:

- Lanza el comando programa infinito en background. (Pon el comando y una captura de pantalla).

```
richard@richard:~$ nohup python3 bucle.py &
[1] 1042
```

- Que pasa si lanzas el programa en background y cierras la sesion? Sigue corriendo el programa?

Como he utilizado el nohup, al cerrar sesion el programa sigue corriendo

- Crea un servicio de linux para que ese programa siempre este en ejecucion.
Muestra el codigo .service. (Demuestra lo que has hecho y explica que es).

Creamos el archivo .service

```
[Unit]
Description= Bucle

[Service]
ExecStart=/usr/bin/python3 /home/richard/bucle.py
Restart=always

[Install]
WantedBy=multi-user.target
```

Copiamos el .service y lo pegamos a /etc/systemd/system

```
root@richard:/home/richard# cp bucle.service /etc/systemd/system_
```

Reiniciamos el systemd

```
root@richard:/home/richard# systemctl daemon-reload
```

Activamos el .service para que se ejecute cada vez que se inicie

```
root@richard:/etc/systemd/system# systemctl enable bucle.service
Created symlink /etc/systemd/system/multi-user.target.wants/bucle.service → /etc/systemd/system/bucle.service.
```