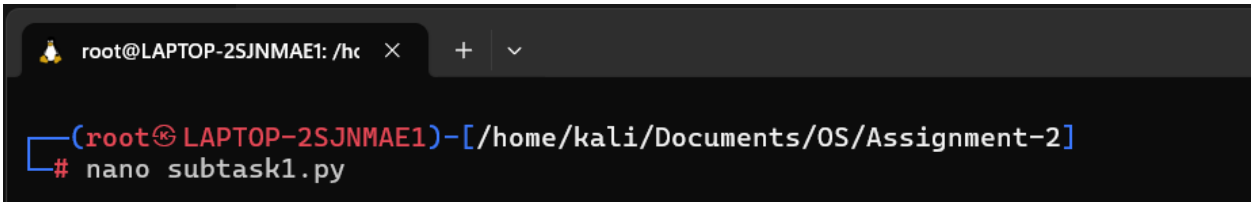
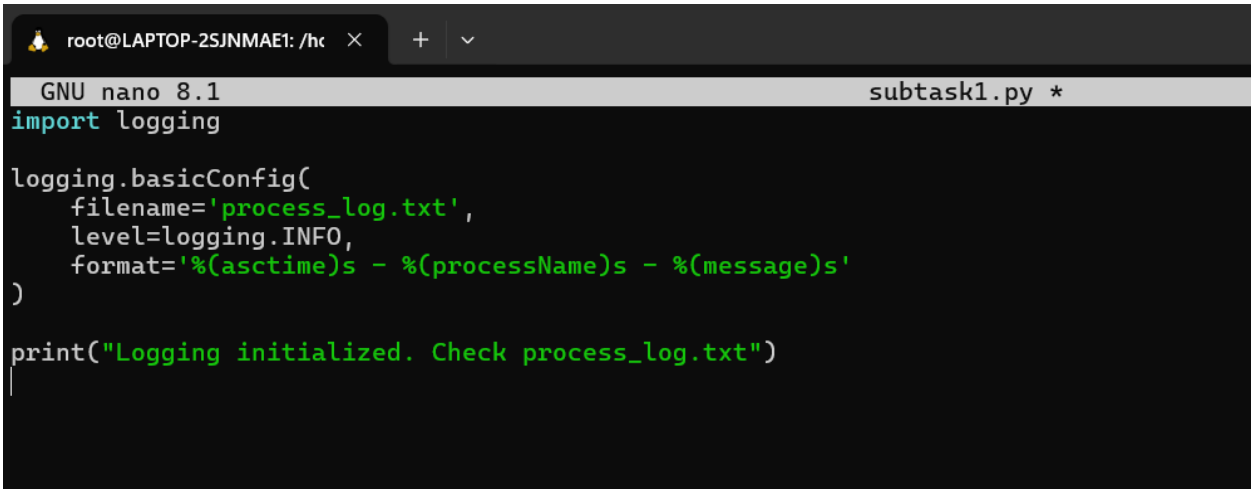
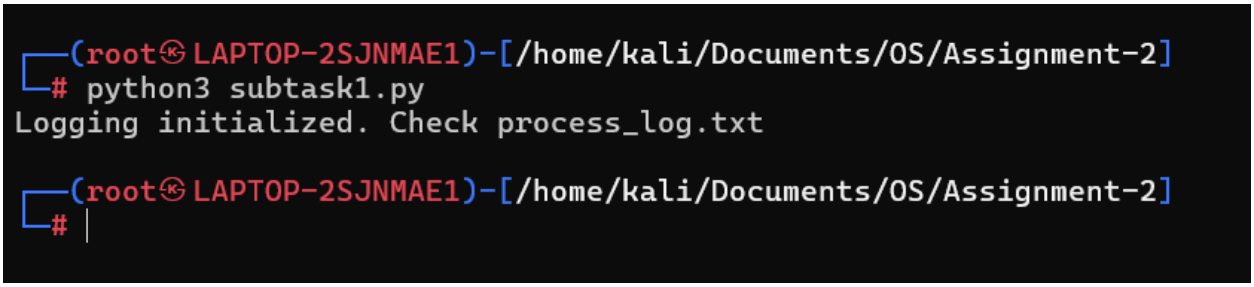
Operating Systems Lab Assignment – 2

Sub-Task 1: Initialize the logging configuration to capture timestamped messages.

Sol.-







Sub-Task 2: Define a function that simulates a process task (e.g., sleep for 2 seconds).

Sol.-

```
┌──(root㉿LAPTOP-2SJNMAE1)-[/home/kali/Documents/OS/Assignment-2]
└─# nano subtask2.py
```

root@LAPTOP-2SJNMAE1: /hc

```
 GNU nano 8.1                                          subtask2.py *
import time
import logging

logging.basicConfig(
    filename='process_log.txt',
    level=logging.INFO,
    format='%(asctime)s - %(processName)s - %(message)s'
)

def system_process(task_name):
    logging.info(f"{task_name} started")
    time.sleep(2)
    logging.info(f"{task_name} ended")

print("Function system_process() defined successfully.")
```

```
┌──(root㉿LAPTOP-2SJNMAE1)-[/home/kali/Documents/OS/Assignment-2]
└─# python3 subtask2.py
Function system_process() defined successfully.

┌──(root㉿LAPTOP-2SJNMAE1)-[/home/kali/Documents/OS/Assignment-2]
└─#
```

Sub-task3: Create at least two processes and start them concurrently.

Sol.-

```
┌──(root㉿LAPTOP-2SJNMAE1)-[/home/kali/Documents/OS/Assignment-2]
└─# nano subtask3.py
```

root@LAPTOP-2SJNMAE1: /h

```
 GNU nano 8.1                                          subtask3.py *
import multiprocessing
import logging
import time

logging.basicConfig(
    filename='process_log.txt',
    level=logging.INFO,
    format='%(asctime)s - %(processName)s - %(message)s'
)

def system_process(task_name):
    logging.info(f"{task_name} started")
    time.sleep(2)
    logging.info(f"{task_name} ended")

if __name__ == '__main__':
    print("System Starting...")

    p1 = multiprocessing.Process(target=system_process, args=('Process-1',))
    p2 = multiprocessing.Process(target=system_process, args=('Process-2',))

    p1.start()
    p2.start()
```

```
┌──(root㉿LAPTOP-2SJNMAE1)-[/home/kali/Documents/OS/Assignment-2]
└─# python3 subtask3.py
System Starting...

┌──(root㉿LAPTOP-2SJNMAE1)-[/home/kali/Documents/OS/Assignment-2]
└─#
```

Sub-Task 4: Ensure proper termination and joining of processes, and verify the output in the log file.

Sol.-

```
┌──(root㉿LAPTOP-2SJNMAE1)-[/home/kali/Documents/OS/Assignment-2]
└─# nano subtask4.py
```

root@LAPTOP-2SJNMAE1: /hc

GNU nano 8.1 subtask4.py *

```python
import multiprocessing
import logging
import time

logging.basicConfig(
    filename='process_log.txt',
    level=logging.INFO,
    format='%(asctime)s - %(processName)s - %(message)s'
)

def system_process(task_name):
    logging.info(f"{task_name} started")
    time.sleep(2)
    logging.info(f"{task_name} ended")

if __name__ == '__main__':
    print("System Starting...")

    p1 = multiprocessing.Process(target=system_process, args=('Process-1',))
    p2 = multiprocessing.Process(target=system_process, args=('Process-2',))

    p1.start()
    p2.start()

    p1.join()
    p2.join()

    print("System Shutdown.")
```

```
┌──(root㉿LAPTOP-2SJNMAE1)-[/home/kali/Documents/OS/Assignment-2]
└─# python3 subtask4.py
System Starting...
System Shutdown.

┌──(root㉿LAPTOP-2SJNMAE1)-[/home/kali/Documents/OS/Assignment-2]
└─#
```