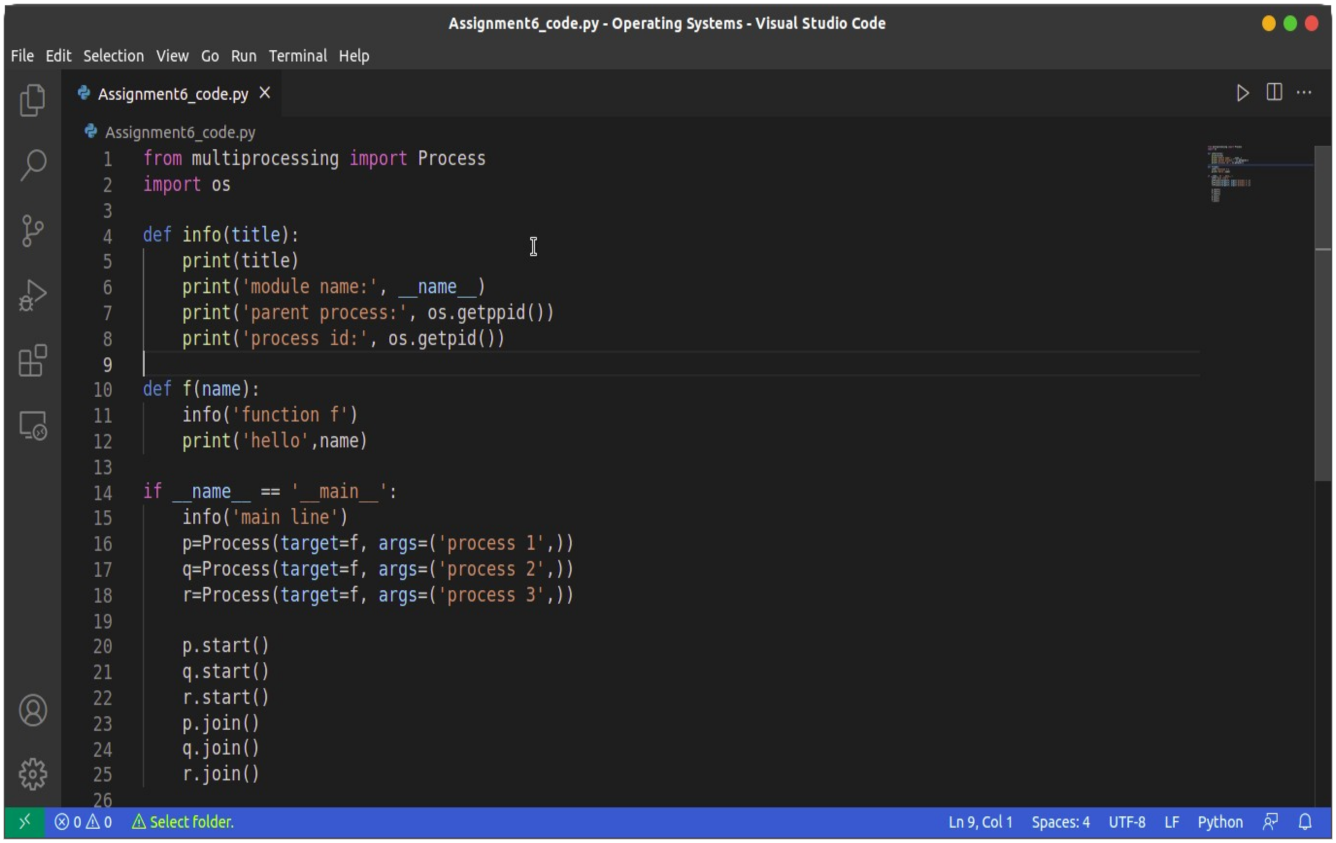
Subhan Khalid

P200086 BSCS 4A

Code:


Assignment6_code.py - Operating Systems - Visual Studio Code
File Edit Selection View Go Run Terminal Help
Assignment6_code.py
Assignment6_code.py
⊗ 0 ⚠ 0
⚠ Select folder.
Ln 9, Col 1
Spaces: 4
UTF-8
LF
Python

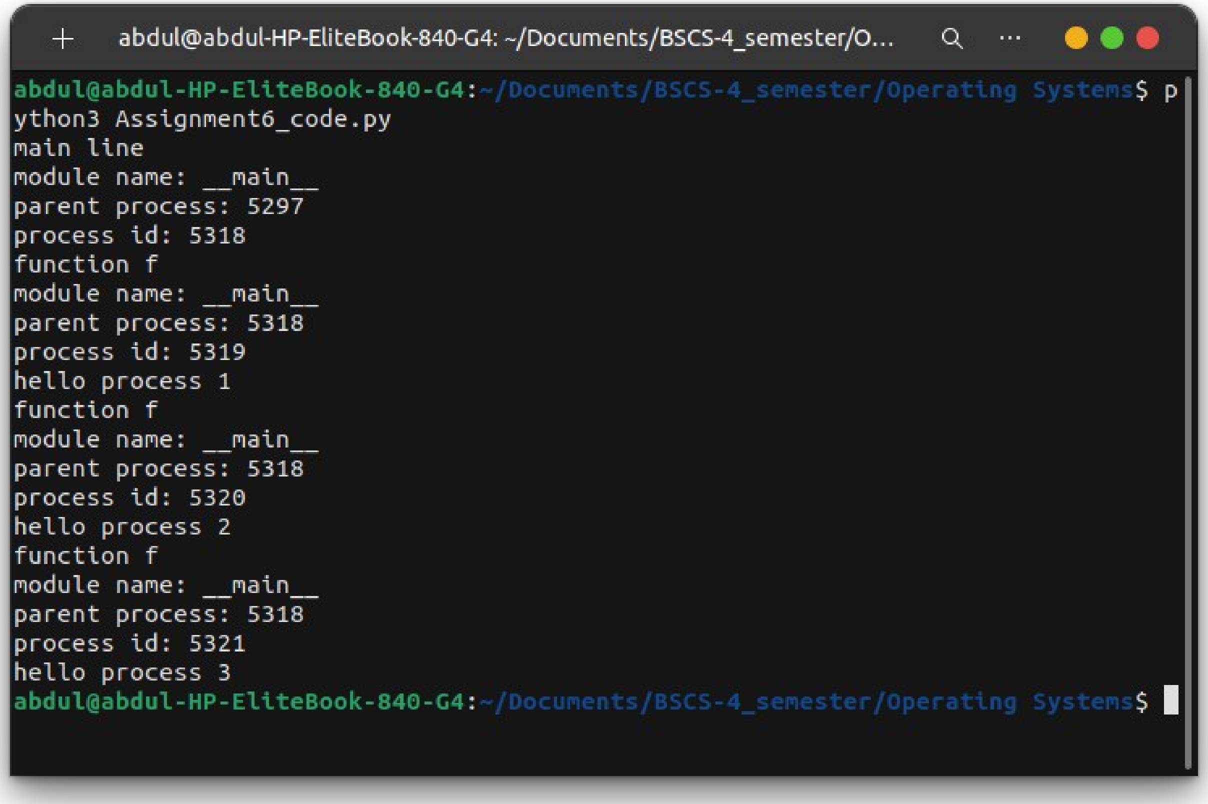
```python
from multiprocessing import Process
import os

def info(title):
    print(title)
    print('module name:', __name__)
    print('parent process:', os.getppid())
    print('process id:', os.getpid())

def f(name):
    info('function f')
    print('hello',name)

if __name__ == '__main__':
    info('main line')
    p=Process(target=f, args=('process 1',))
    q=Process(target=f, args=('process 2',))
    r=Process(target=f, args=('process 3',))

    p.start()
    q.start()
    r.start()
    p.join()
    q.join()
    r.join()
```



abdul@abdul-HP-EliteBook-840-G4: ~/Documents/BSCS-4_semester/O...

```
abdul@abdul-HP-EliteBook-840-G4:~/Documents/BSCS-4_semester/Operating Systems$ p
ython3 Assignment6_code.py
main line
module name:  __main__
parent process:  5297
process id:  5318
function f
module name:  __main__
parent process:  5318
process id:  5319
hello process 1
function f
module name:  __main__
parent process:  5318
process id:  5320
hello process 2
function f
module name:  __main__
parent process:  5318
process id:  5321
hello process 3
abdul@abdul-HP-EliteBook-840-G4:~/Documents/BSCS-4_semester/Operating Systems$ 
```

The process class is a class which can be used to create another process for the purpose of multiprocessing. Its working start by creating a process object and then calling the start method of the process class for that object.