

Nama	Imam Aprido Simarmata
NIM	11S18034

Threading in Python with Synchronization

Without Lock

```
import threading
import logging
import time

lock = threading.Lock()

daftar_mhs = []

usm_1 = ["A", "B", "C", "D", "E", "F", "G"]
tidak_daftar_ulang = ["A", "B"]
# [A, C, D, E, F]

usm_2 = ["B", "H", "J"]
# [A, C, D, E, F, B, H, J]

def insert_usm_1():
    print("t1 starting")
    global daftar_mhs
    for cama in usm_1:
        daftar_mhs.append(cama)
        time.sleep(1)
    print("t1 end")

def remove_tidak_daftar_ulang():
    print("t2 starting")
    global daftar_mhs
    global tidak_daftar_ulang
    for calon_mhs in tidak_daftar_ulang:
        print(f"{calon_mhs} akan dihapus dari {daftar_mhs}")
        daftar_mhs.remove(calon_mhs)
    print("t2 end")

if __name__ == "__main__":

    thread1 = threading.Thread(target = insert_usm_1, args = ())
```

```

thread2 = threading.Thread(target = remove_tidak_daftar_ulang, args = ())
thread1.start()
thread2.start()
thread1.join()
thread2.join()
print(daftar_mhs)

```

Output

```

PS C:\Users\aprid\Documents\Lit> python3 .\PrakSPT.py
t1 starting
Exception in thread Thread-1:
Traceback (most recent call last):
  File "C:\Program Files\WindowsApps\PythonSoftwareFoundation.Python.3.9_3.9.3312.0_x64__qbz5n2kfra8p0\lib\threading.py", line 973, in _bootstrap_inner
t2 starting
A akan dihapus dari ['A']
B akan dihapus dari []
  self.run()
Exception in thread Thread-2:
Traceback (most recent call last):
  File "C:\Program Files\WindowsApps\PythonSoftwareFoundation.Python.3.9_3.9.3312.0_x64__qbz5n2kfra8p0\lib\threading.py", line 973, in _bootstrap_inner
  File "C:\Program Files\WindowsApps\PythonSoftwareFoundation.Python.3.9_3.9.3312.0_x64__qbz5n2kfra8p0\lib\threading.py", line 910, in run
    self.run()
  File "C:\Program Files\WindowsApps\PythonSoftwareFoundation.Python.3.9_3.9.3312.0_x64__qbz5n2kfra8p0\lib\threading.py", line 910, in run
    self._target(*self._args, **self._kwargs)
  File "C:\Users\aprid\Documents\Lit\PrakSPT.py", line 22, in insert_usm_1
    self._target(*self._args, **self._kwargs)
    time.sleep(1)
AttributeError: 'builtin_function_or_method' object has no attribute 'sleep'
  File "C:\Users\aprid\Documents\Lit\PrakSPT.py", line 34, in remove_tidak_daftar_ulang
    daftar_mhs.remove(calon_mhs)
ValueError: list.remove(x): x not in list
[]
PS C:\Users\aprid\Documents\Lit> █

```

Karena `insert_usm_1` akan *sleep* selama satu detik setiap kali menambahkan satu **cama** ke **daftar_mhs**, di saat yang sama **remove_tidak_daftar_ulang** juga berusaha memodifikasi **daftar_mhs** untuk menghapus beberapa **cama** yang tidak daftar ulang. Namun, karena **cama** yang tidak daftar ulang yaitu **B** belum ada di **daftar_mhs**, terjadi Exception. Ini yang disebut dengan *race condition*.

Untuk itu, sebelum `insert_usm_1` mengerjakan tugasnya, list **daftar_mhs** sebaiknya tidak dapat digunakan oleh thread lain, yang maksudnya adalah **remove_tidak_daftar_ulang**. Digunakan **lock** untuk memberitahukan, bahwa **insert_usm_1** sedang berjalan, sehingga dengan demikian **remove_tidak_daftar_ulang** dapat menunggu sampai **insert_usm_1** sudah selesai ditandai dengan **lock** yang di-release.

With lock

```

import threading
import logging
import time

```

```

lock = threading.Lock()

daftar_mhs = []

usm_1 = ["A", "B", "C", "D", "E", "F", "G" ]
tidak_daftar_ulang = ["A", "B"]
# [A, C, D, E, F]

usm_2 = ["B", "H", "J"]
# [A, C, D, E, F, B, H, J]

def insert_usm_1():
    lock.acquire()
    print("t1 starting")
    global daftar_mhs
    for cama in usm_1:
        daftar_mhs.append(cama)
        time.sleep(1)
    lock.release()
    print("t1 end")

def remove_tidak_daftar_ulang():
    lock.acquire()
    print("t2 starting")
    global daftar_mhs
    global tidak_daftar_ulang
    for calon_mhs in tidak_daftar_ulang:
        print(f"{calon_mhs} akan dihapus dari {daftar_mhs}")
        daftar_mhs.remove(calon_mhs)
    lock.release()
    print("t2 end")

if __name__ == "__main__":

    thread1 = threading.Thread(target = insert_usm_1, args = ())
    thread2 = threading.Thread(target = remove_tidak_daftar_ulang, args = ())
    thread1.start()
    thread2.start()
    thread1.join()
    thread2.join()
    print(daftar_mhs)

```

Output

```
PS C:\Users\aprid\Documents\Lit> python3 .\PrakSPT.py
t1 starting
t1 end
t2 starting
A akan dihapus dari ['A', 'B', 'C', 'D', 'E', 'F', 'G']
B akan dihapus dari ['B', 'C', 'D', 'E', 'F', 'G']
t2 end
['C', 'D', 'E', 'F', 'G']
PS C:\Users\aprid\Documents\Lit>
```