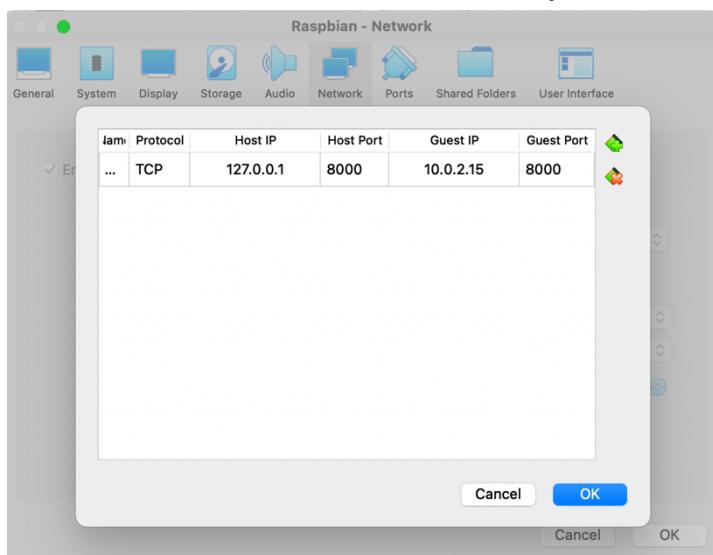


### สิ่งที่ต้องเตรียม

- ทำการติดตั้ง websockets เพื่อใช้สำหรับใช้งาน Python กับ WebSocket โดยพิมพ์คำสั่งดังต่อไปนี้

```
- pip3 install websockets  
- pip3 install asyncio
```

- ตั้งค่า Port Forwarding บนระบบปฏิบัติการที่ติดตั้งเซิร์จแล้วเรียบร้อยเพื่อให้เครื่อง host และ guest สามารถติดต่อสื่อสารกันได้ โดยไปที่โปรแกรม VirtualBox ทำการตั้งค่า Port Forwarding โดยเข้าไปที่ Settings -> Network -> Port Forwarding ดังรูปต่อไปนี้



### แบบฝึกปฏิบัติการ

WebSocket คือ โปรโตคอลการสื่อสารสำหรับการเชื่อมต่อ TCP แบบสองทิศทางแบบสองทิศทางแบบต่อเนื่องจากผู้ใช้ หรือแอปพลิเคชันฝั่งไคลเอนต์อื่น ๆ ไปยังเซิร์ฟเวอร์

- ทำการสร้าง WebSocket ฝั่ง Server (Raspberry Pi) โดยให้ นศ. สร้างไฟล์ใหม่ชื่อ server.py และเขียนโปรแกรมดังรูปต่อไปนี้

```

pi@raspberry: ~
File Edit Tabs Help
GNU nano 3.2           server.py
import asyncio
import websockets

async def handler(websocket):
    while True:
        message = await websocket.recv()
        print("Data received : "+message)
        await websocket.send("Data received : "+message)

async def main():
    async with websockets.serve(handler, "10.0.2.15", 8000):
        await asyncio.Future() # run forever

if __name__ == "__main__":
    asyncio.run(main())

```

[ Read 15 lines ]

AG Get Help ^O Write Out ^W Where Is ^K Cut Text ^J Justify ^C Cur Pos  
AX Exit ^R Read File ^\ Replace ^U Uncut Text ^T To Spell ^ Go To Line

2. รันไฟล์ Server.py เพื่อเปิดการทำงาน WebSocket Server โดยเปิด Terminal และพิมพ์คำสั่งดังต่อไปนี้

- python3 server.py

3. ทดสอบการทำงาน WebSocket โดยที่เครื่อง Host เปิด Terminal เพื่อทำการเชื่อมต่อและแลกเปลี่ยนข้อมูลไประหว่าง client และ Server ด้วย WebSocket โดยพิมพ์คำสั่งดังต่อไปนี้

- python3 -m websockets ws://127.0.0.1:8000/

4. สังเกตผลลัพธ์ของการทำงานระหว่าง client และ Server และอภิปรายผล (3 คะแนน)

5. แก้ไขไฟล์ server.py เพื่อนำค่าที่ได้จากผู้ใช้ client ไปสั่งงานบอร์ด Raspberry Pi ไปกำหนดค่าในโปรแกรม Sense Hat Emulator ตามรูปที่กำหนดให้ต่อไปนี้

```

pi@raspberry: ~
File Edit Tabs Help
GNU nano 3.2           server.py
import asyncio
import websockets
from sense_emu import SenseHat

sense = SenseHat()

red = (255, 0, 0)
blue = (0, 0, 255)

async def handler(websocket):
    while True:
        message = await websocket.recv()
        print("Data received : "+message)

        pixels = [red if i < int(message) else blue for i in range(64)]
        sense.set_pixels(pixels)

        await websocket.send("Data received : "+message)

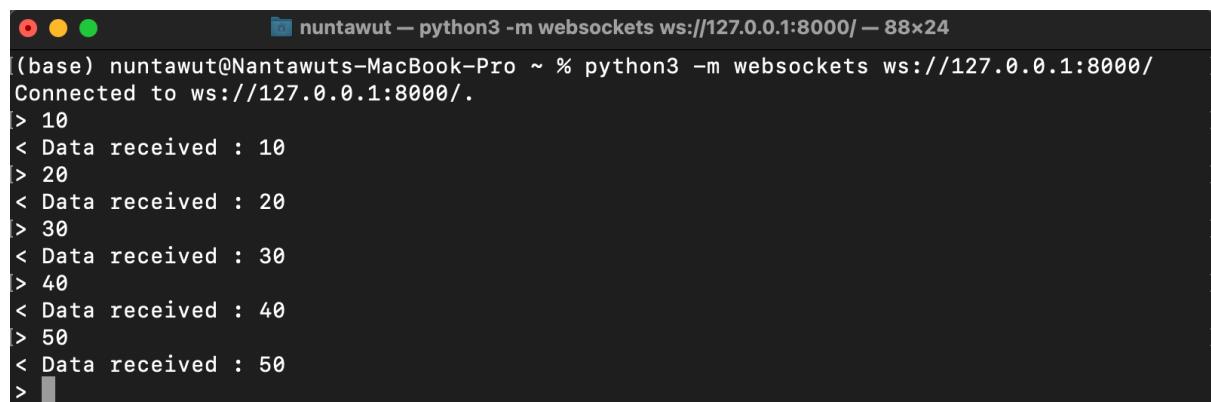
async def main():
    async with websockets.serve(handler, "10.0.2.15", 8000):
        await asyncio.Future() # run forever

if __name__ == "__main__":
    asyncio.run(main())

```

AG Get Help ^O Write Out ^W Where Is ^K Cut Text ^J Justify ^C Cur Pos  
AX Exit ^R Read File ^\ Replace ^U Uncut Text ^T To Spell ^ Go To Line

6. ทดสอบการทำงาน WebSocket โดยที่เครื่อง Host เปิด Terminal และพิมพ์คำสั่งดังและป้อนค่าต่าง ๆ ต่อไปนี้



```
nuntawut — python3 -m websockets ws://127.0.0.1:8000/ — 88x24
(base) nuntawut@Nantawuts-MacBook-Pro ~ % python3 -m websockets ws://127.0.0.1:8000/
Connected to ws://127.0.0.1:8000/.
> 10
< Data received : 10
> 20
< Data received : 20
> 30
< Data received : 30
> 40
< Data received : 40
> 50
< Data received : 50
>
```

7. สังเกตผลลัพธ์ของการทำงานในโปรแกรม Sense Hat Emulator และอภิปรายผล (3 คะแนน)

8. ดาวน์โหลดไฟล์ index.html ตามลิงค์นี้

[https://drive.google.com/file/d/11X8foYcdZcp4umkGOyK8nPcVJC7\\_5V4M/view?usp=sharing](https://drive.google.com/file/d/11X8foYcdZcp4umkGOyK8nPcVJC7_5V4M/view?usp=sharing)

9. ทำความเข้าใจการทำงานของไฟล์ index.html และสังเกตผลลัพธ์ของการทำงานในโปรแกรม Sense Hat Emulator พร้อมอภิปรายผล (4 คะแนน)