

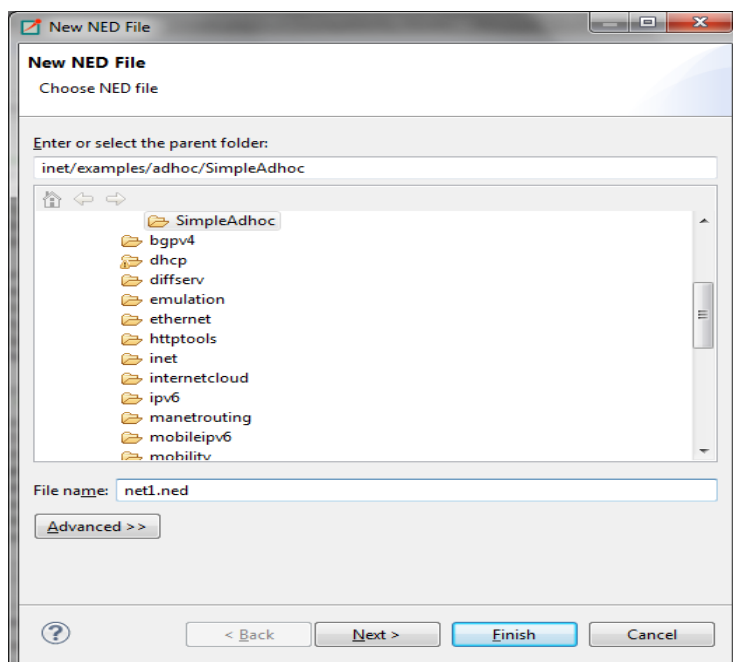
Practical No:4

Description:

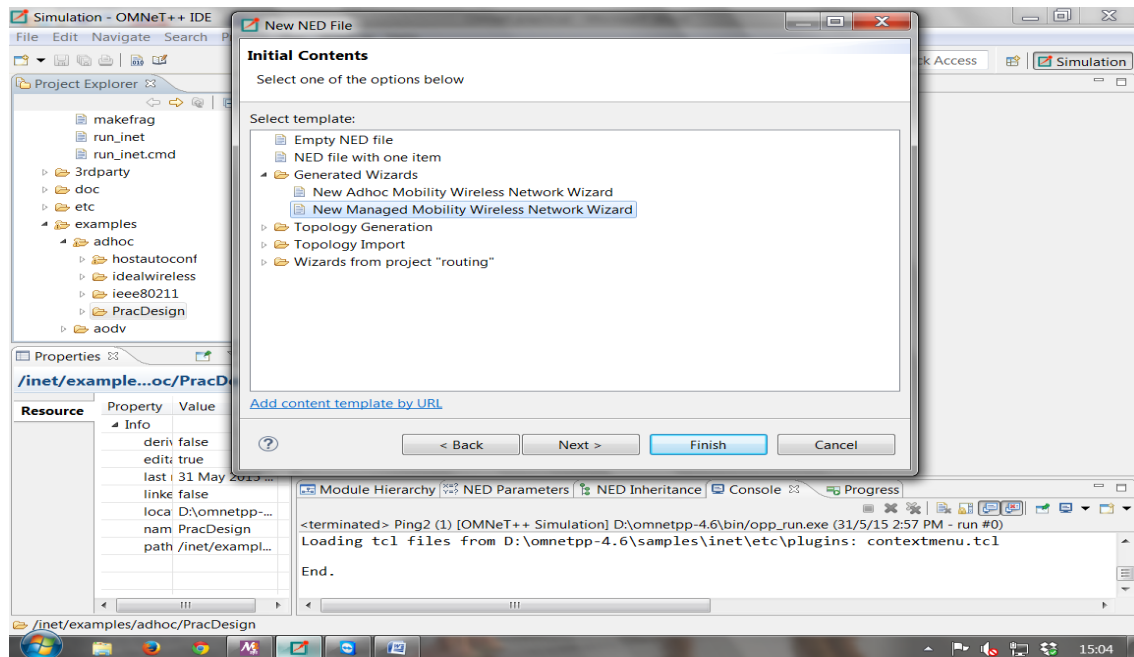
- Simulator used : Omnet++
- Simulator can be downloaded from below link:
<https://omnetpp.org/omnetpp> (recommended version is omnet++ 4.2.2).
- After installing Omnet++, we need to install inet framework which is specially designed for wireless simulation. You can download inet framework from below link.
<https://inet.omnetpp.org/Download.html>
- After downloading there are certain steps to be followed to include this framework in omnet++ as follows:
- Download the INET sources.
- Unpack it into the directory of your choice: (tar xvfz inet-<version>.tgz) (recommended is tar or .tgz)
- Recommended version is inet 2.1
- Start the Omnet++ IDE, and import the project via *File -> Import -> Existing Projects to the Workspace*. A project named inet should appear.
- Build with *Project -> Build*, or hit ctrl+b
- Now you should be able to launch example simulations.

Steps for practical:

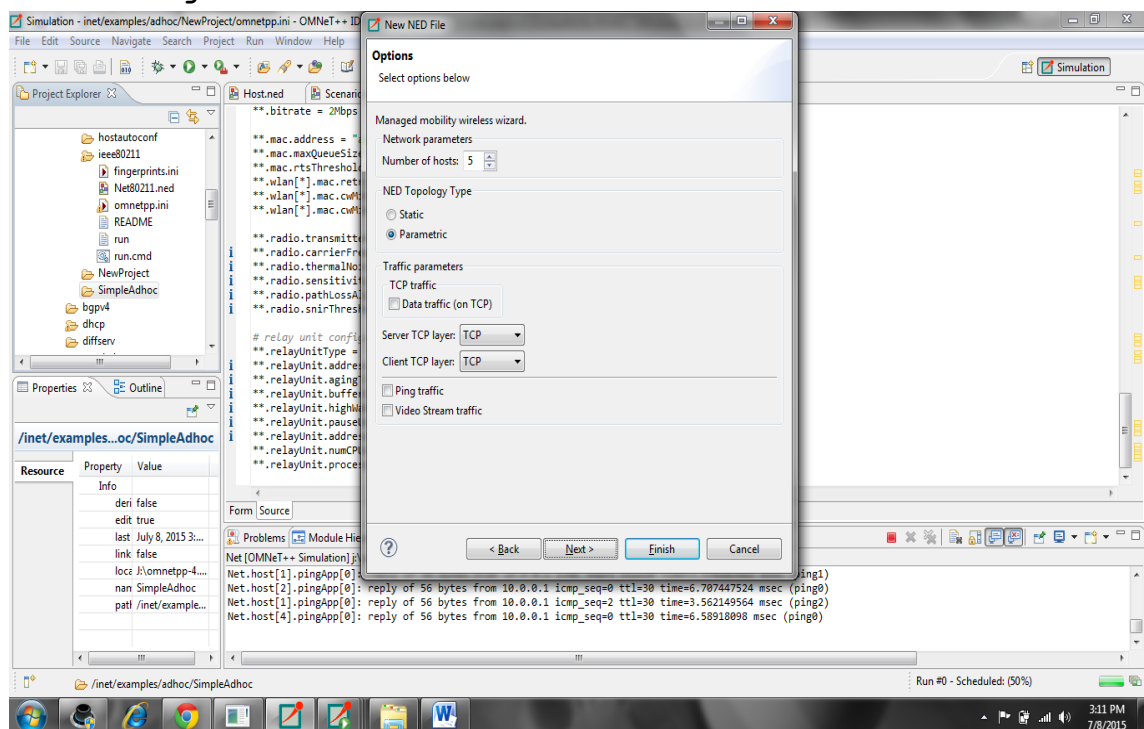
- Then open inet/examples/
- Right click on adhoc -create new folder as SimpleAdhoc.
- Right click on your newly created folder and select NED file. Give name as Net1.



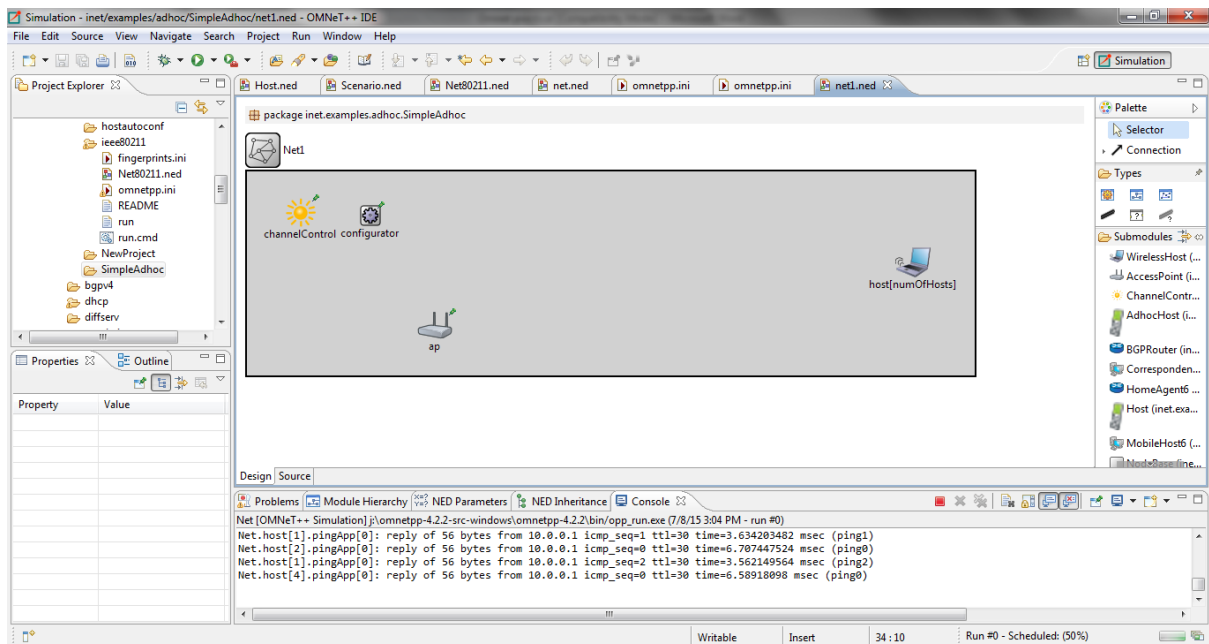
click on new manages mobility wireless network wizard.



then configure as follows



then click on finish.



below is the code that will be available in source part of net1.ned once configured.

```
package inet.examples.adhoc.SimpleAdhoc;

// numOfHosts: 5

import inet.networklayer.autorouting.ipv4.Ipv4NetworkConfigurator;
import inet.nodes.inet.WirelessHost;
import inet.nodes.wireless.AccessPoint;
import inet.world.radio.ChannelControl;

network Net
{
    parameters:
        int numOfHosts;

    submodules:
        host[numOfHosts]: WirelessHost
        {
            @display("r=, ,#707070");
        }

        ap: AccessPoint
        {
            @display("p=213,174;r=, ,#707070");
        }

        channelControl: ChannelControl
        {
            numChannels = 2;
            @display("p=61,46");
        }

        configurator: Ipv4NetworkConfigurator
        {
            @display("p=140,50");
        }
}
```

On design part you will find components appearing according to the code as the above snapshot.

Same as do this in omnetpp.ini file :

Source code for omnetpp.ini:

```
[General]
network = Net1

*.numOfHosts = 5

#debug-on-errors = true
tkenv-plugin-path = ../../../../etc/plugins

**.constraintAreaMinX = 0m
**.constraintAreaMinY = 0m
**.constraintAreaMinZ = 0m
**.constraintAreaMaxX = 600m
**.constraintAreaMaxY = 400m
**.constraintAreaMaxZ = 0m
**.debug = true
**.coreDebug = false
**.host*.**.channelNumber = 0

# channel physical parameters
*.channelControl.carrierFrequency = 2.4GHz
*.channelControl.pMax = 2.0mW
*.channelControl.sat = -110dBm
*.channelControl.alpha = 2

# mobility
**.host*.mobilityType = "MassMobility"
**.host*.mobility.initFromDisplayString = false
**.host*.mobility.changeInterval = truncnormal(2s, 0.5s)
**.host*.mobility.changeAngleBy = normal(0deg, 30deg)
**.host*.mobility.speed = truncnormal(20mps, 8mps)
**.host*.mobility.updateInterval = 100ms

# ping app (host[0] pinged by others)
*.host[0].numPingApps = 0
*.host[*].numPingApps = 2
*.host[*].pingApp[*].destAddr = "host[0]"
**.pingApp[0].startTime = uniform(1s, 5s)
**.pingApp[1].startTime = 5s+uniform(1s, 5s)
**.pingApp[*].printPing = true

# nic settings
**.wlan[*].bitrate = 2Mbps

**.wlan[*].mgmt.frameCapacity = 10

**.wlan[*].mac.address = "auto"
**.wlan[*].mac.maxQueueSize = 14
**.wlan[*].mac.rtsThresholdBytes = 3000B
**.wlan[*].mac.retryLimit = 7
**.wlan[*].mac.cwMinData = 7

**.wlan[*].radio.transmitterPower = 2mW
**.wlan[*].radio.thermalNoise = -110dBm
**.wlan[*].radio.sensitivity = -85dBm
**.wlan[*].radio.pathLossAlpha = 2
**.wlan[*].radio.snirThreshold = 4dB
```

```

[Config Ping1]
description = "host1 pinging host0"

[Config Ping2] # __interactive__
description = "n hosts"
# leave numHosts undefined here

**.mobility.constraintAreaMinZ = 0m
**.mobility.constraintAreaMaxZ = 0m
**.mobility.constraintAreaMinX = 0m
**.mobility.constraintAreaMinY = 0m
**.mobility.constraintAreaMaxX = 600m
**.mobility.constraintAreaMaxY = 400m
**.debug = false
**.coreDebug = false

**.channelNumber = 0

# channel physical parameters
*.channelControl.carrierFrequency = 2.4GHz
*.channelControl.pMax = 20.0mW
*.channelControl.sat = -110dBm
*.channelControl.alpha = 2

# mobility

**.host[*].mobilityType = "MassMobility"
**.host[*].mobility.changeInterval = truncnormal(2s, 0.5s)
**.host[*].mobility.changeAngleBy = normal(0deg, 30deg)
**.host[*].mobility.speed = truncnormal(20mps, 8mps)
**.host[*].mobility.updateInterval = 100ms

# nic settings
**.bitrate = 2Mbps

**.mac.address = "auto"
**.mac.maxQueueSize = 14
**.mac.rtsThresholdBytes = 3000B
**.wlan[*].mac.retryLimit = 7
**.wlan[*].mac.cwMinData = 7
**.wlan[*].mac.cwMinMulticast = 31

**.radio.transmitterPower = 20.0mW
**.radio.carrierFrequency = 2.4GHz
**.radio.thermalNoise = -110dBm
**.radio.sensitivity = -85dBm
**.radio.pathLossAlpha = 2
**.radio.snirThreshold = 4dB

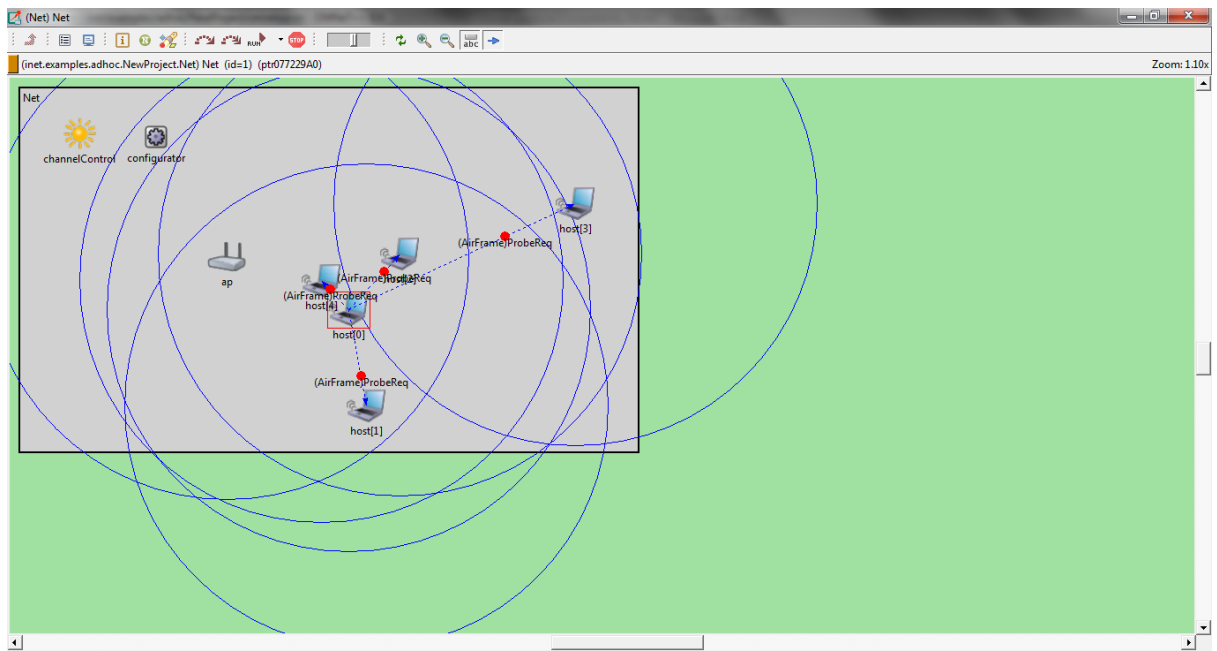
# relay unit configuration
**.relayUnitType = "MACRelayUnitNP"
**.relayUnit.addressTableSize = 100
**.relayUnit.agingTime = 120s
**.relayUnit.bufferSize = 1MiB
**.relayUnit.highWatermark = 512KiB
**.relayUnit.pauseUnits = 300 # pause for 300*512 bit (19200 byte) time
**.relayUnit.addressTableFile = ""
**.relayUnit.numCPUs = 2
**.relayUnit.processingTime = 2us

```

EXECUTION:

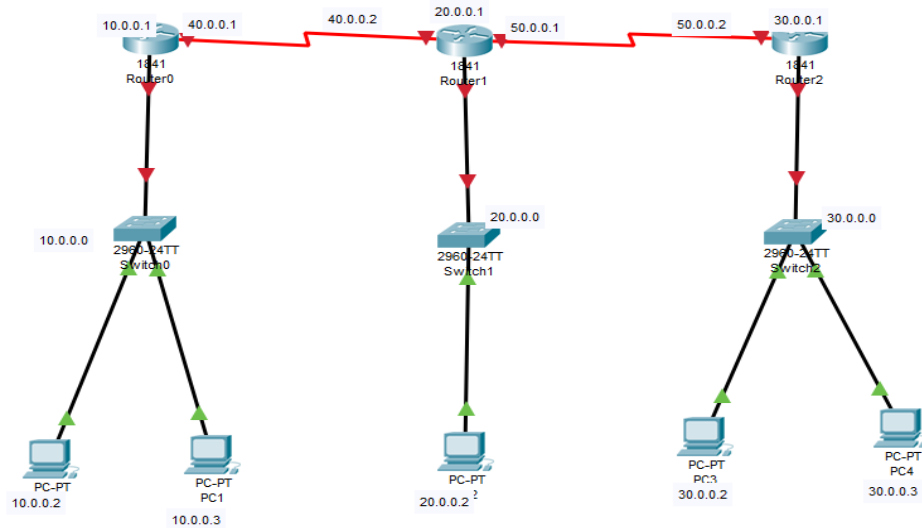
Now try to execute by right click on ned file Run as-1-Omnet++ simulation.

OUTPUT :



Practical No:5

Connect the Routers Using Serializable Wire and give ip address



Router0→Config→RIP→Add Networks

Router0

Physical Config CLI Attributes

GLOBAL

- Settings
- Algorithm Settings

ROUTING

- Static
- RIP**

SWITCHING

- VLAN Database

INTERFACE

- FastEthernet0/0
- FastEthernet0/1
- Serial0/0/0

RIP Routing

Network

Add

Network Address

10.0.0.0

40.0.0.0

Remove

Equivalent IOS Commands

```
Router(config-if)#
%LINK-5-CHANGED: Interface Serial0/0/0, changed state to up
%LINEPROTO-5-UPDOWN: Line protocol on Interface Serial0/0/0, changed state to up
Router(config-if)#exit
Router(config)#router rip
Router(config-router)#network 40.0.0.0
Router(config-router)#network 10.0.0.0
Router(config-router)#
```

☐ Top

Router1→Config→RIP→Add Networks

The screenshot shows the configuration window for Router1. The 'Config' tab is active, and the 'RIP' option is selected under the 'ROUTING' section. The 'RIP Routing' section displays a table of network addresses: 20.0.0.0, 40.0.0.0, and 50.0.0.0. An 'Add' button is visible next to the table, and a 'Remove' button is at the bottom right. The 'Equivalent IOS Commands' section shows the following commands:

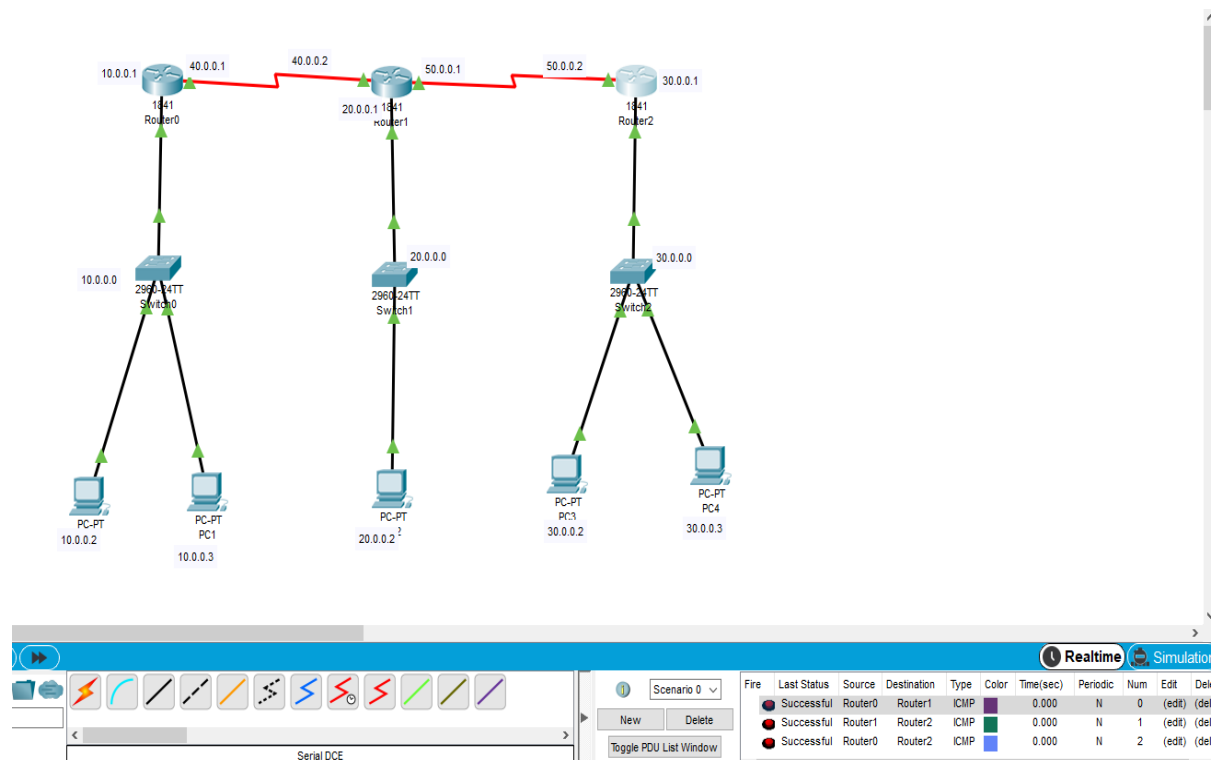
```
Router(config-router)#network 40.0.0.0
Router(config-router)#network 50.0.0.0
Router(config-router)#network 20.0.0.0
Router(config-router)#
Router(config-router)#end
Router#configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)#router rip
Router(config-router)#
%SYS-5-CONFIG_I: Configured from console by console
```

Router2→Config→RIP→Add Networks

The screenshot shows the configuration window for Router2. The 'Config' tab is active, and the 'RIP' option is selected under the 'ROUTING' section. The 'RIP Routing' section displays a table of network addresses: 30.0.0.0 and 50.0.0.0. An 'Add' button is visible next to the table, and a 'Remove' button is at the bottom right. The 'Equivalent IOS Commands' section shows the following commands:

```
Router>enable
Router#
Router#configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)#router rip
Router(config-router)#
```


Final Output:



Practical No: 6

Steps for practical:

1. open inet/examples/adhoc/manetrouting
2. Right click on manetrouting -create new folder as MobileNet.
3. Right click on your newly created folder and select NED file. Give name as Net1.
select new adhoc mobility wireless network wizard

Net1.ned

```
package inet.examples.manetrouting.mobilenet;

// numOfHosts: 10
// parametric: true
// static:      false

import inet.networklayer.autorouting.ipv4.Ipv4NetworkConfigurator;
import inet.nodes.inet.AdhocHost;
import inet.world.radio.ChannelControl;

network Manet
{
    parameters:
        int numHosts;
    submodules:
        host[numHosts]: AdhocHost
        {
            parameters:
                @display("r=,,#707070");
        }

        channelControl: ChannelControl
        {
            parameters:
                @display("p=60,50");
        }

        configurator: Ipv4NetworkConfigurator
        {
            @display("p=140,50");
        }
}
```

a file omnetpp.ini will be created with the following code :

omnetpp.ini :-

```
[General]
network = Manet
#record-eventlog = true
#eventlog-message-detail-pattern = *:(not declaredOn(cMessage) and not
declaredOn(cNamedObject) and not declaredOn(cObject))

*.numHosts = 10

num-rngs = 3
**.mobility.rng-0 = 1
**.wlan[*].mac.rng-0 = 2
#debug-on-errors = true

tkenv-plugin-path = ../../../../etc/plugins

**.channelNumber = 0
```

```

# channel physical parameters
*.channelControl.carrierFrequency = 2.4GHz
*.channelControl.pMax = 2.0mW
*.channelControl.sat = -110dBm
*.channelControl.alpha = 2
*.channelControl.numChannels = 1

# mobility
**.host[*].mobilityType = "MassMobility"
**.mobility.constraintAreaMinZ = 0m
**.mobility.constraintAreaMaxZ = 0m
**.mobility.constraintAreaMinX = 0m
**.mobility.constraintAreaMinY = 0m
**.mobility.constraintAreaMaxX = 600m
**.mobility.constraintAreaMaxY = 400m
**.mobility.changeInterval = truncnormal(2s, 0.5s)
**.mobility.changeAngleBy = normal(0deg, 30deg)
**.mobility.speed = truncnormal(20mps, 8mps)
**.mobility.updateInterval = 100ms

# ping app (host[0] pinged by others)
*.host[0].pingApp[0].destAddr = ""
*.host[*].numPingApps = 1
*.host[*].pingApp[0].destAddr = "host[0]"
*.host[*].pingApp[0].startTime = uniform(1s, 5s)
*.host[*].pingApp[0].printPing = true

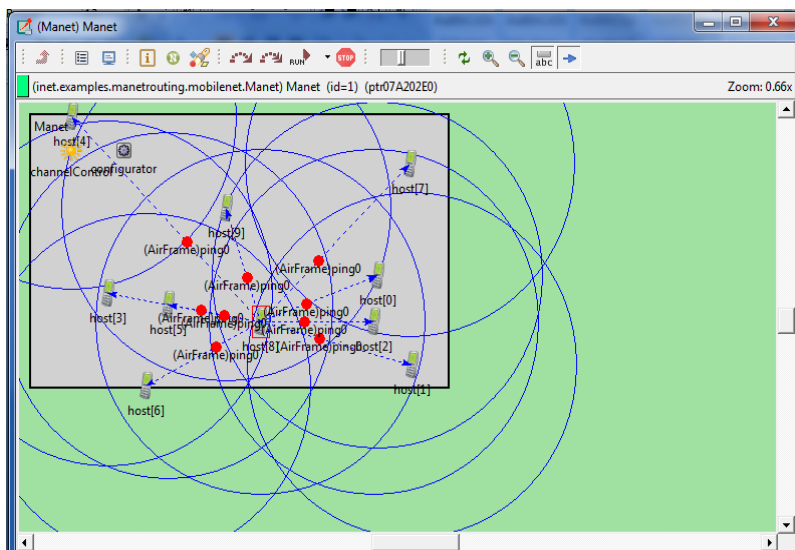
# nic settings
**.wlan[*].bitrate = 2Mbps

**.wlan[*].mgmt.frameCapacity = 10
**.wlan[*].mac.address = "auto"
**.wlan[*].mac.maxQueueSize = 14
**.wlan[*].mac.rtsThresholdBytes = 3000B
**.wlan[*].mac.retryLimit = 7
**.wlan[*].mac.cwMinData = 7
**.wlan[*].mac.cwMinMulticast = 31

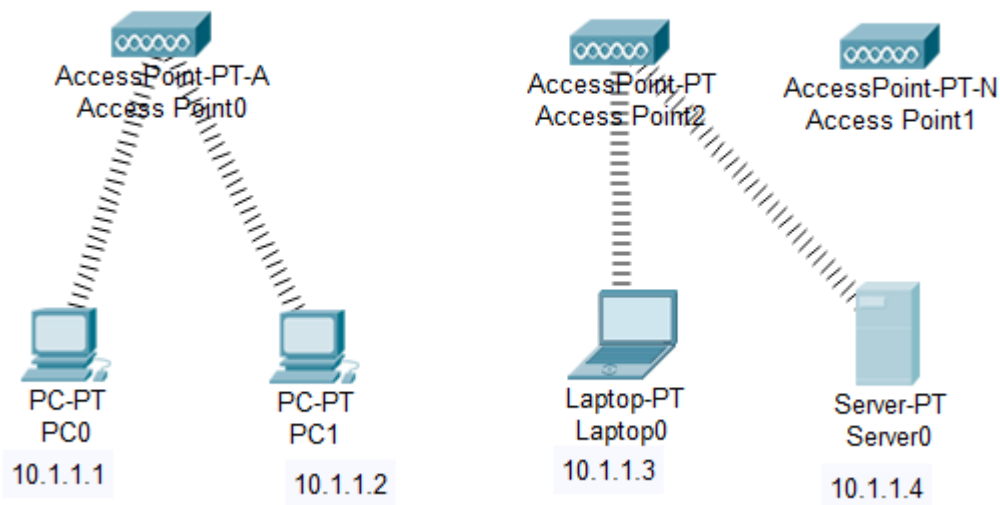
**.wlan[*].radio.transmitterPower = 2mW
**.wlan[*].radio.thermalNoise = -110dBm
**.wlan[*].radio.sensitivity = -85dBm
**.wlan[*].radio.pathLossAlpha = 2
**.wlan[*].radio.snirThreshold = 4dB

```

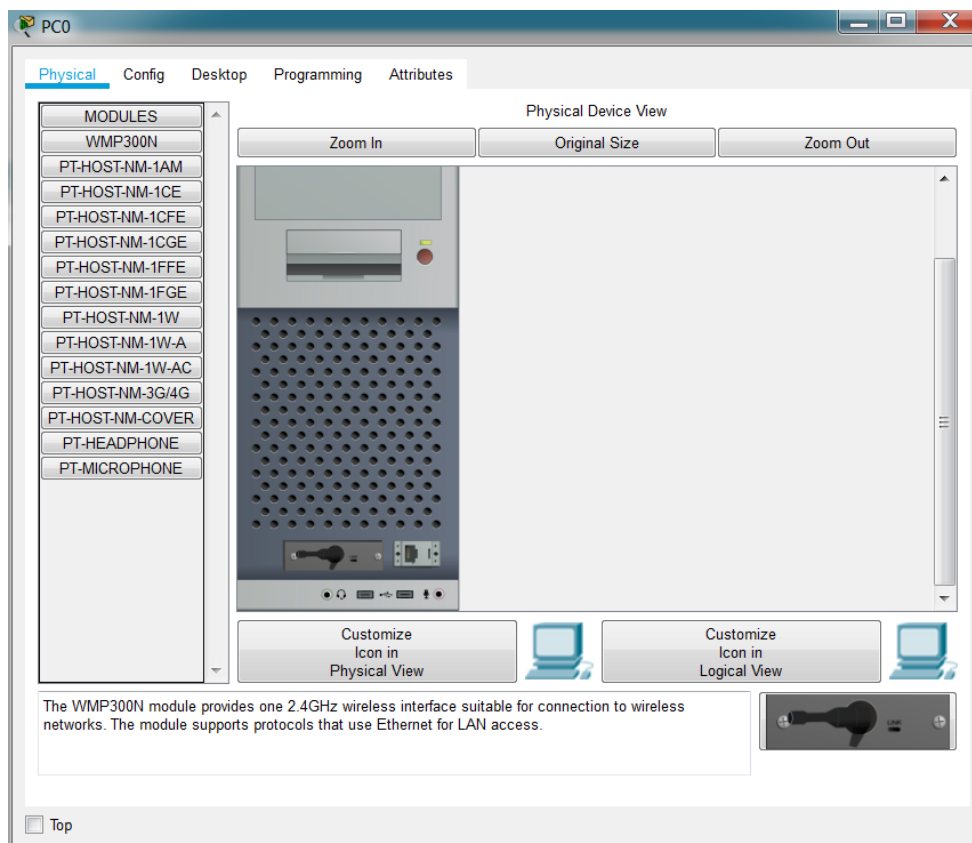
Right click on ned file and run it as omnetpp simulation



Practical No:7

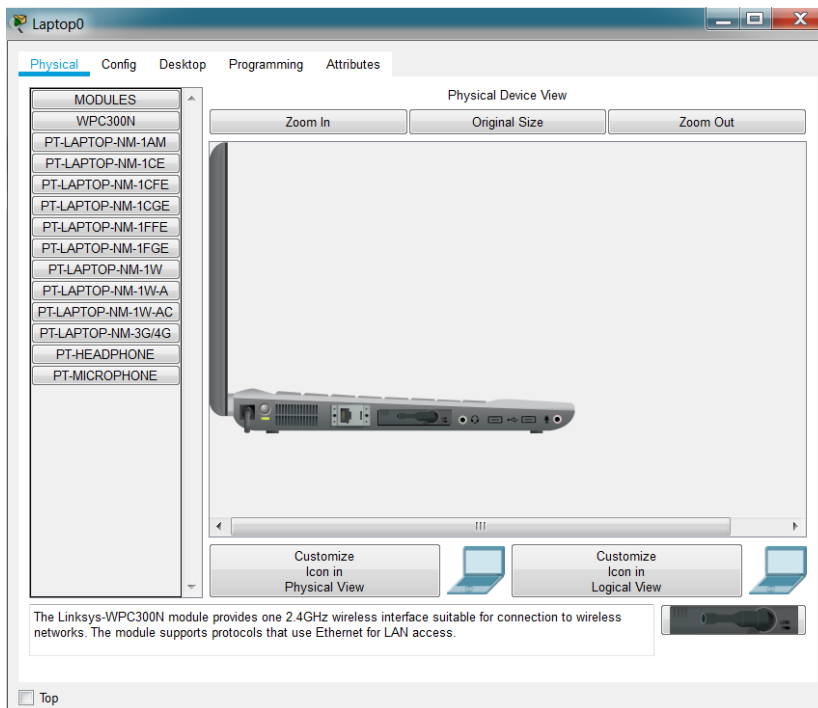


1. PC0 → Physical → Drag and Drop PT-HOST-NM-1W-A module



2. Do same for PC1

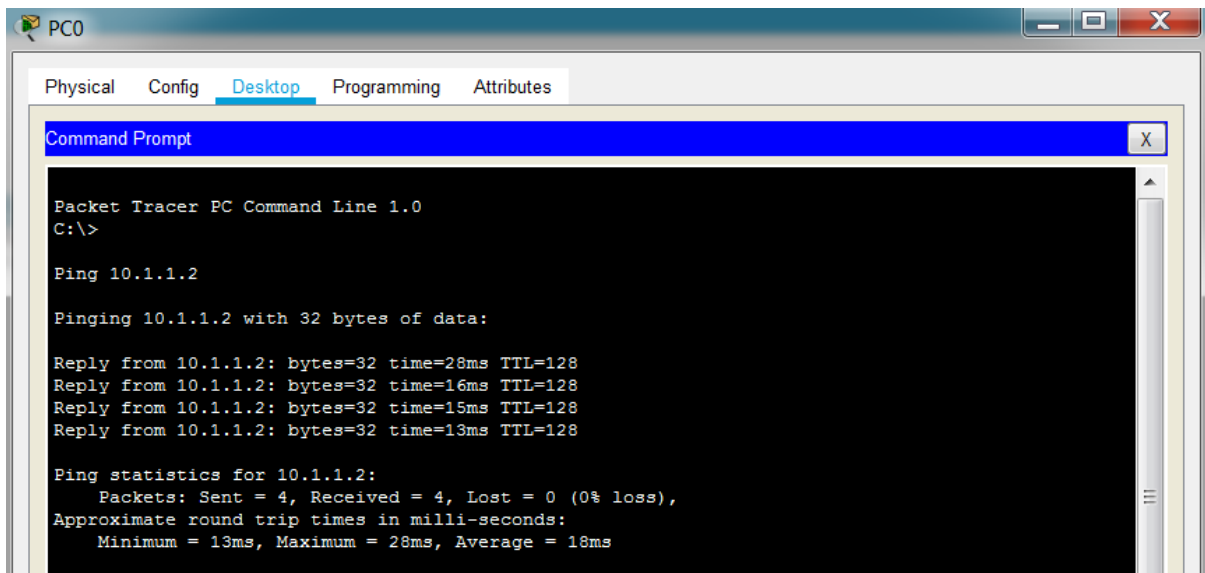
3.Laptop0→Physical→Drag and Drop PT-HOST-NM-1W module



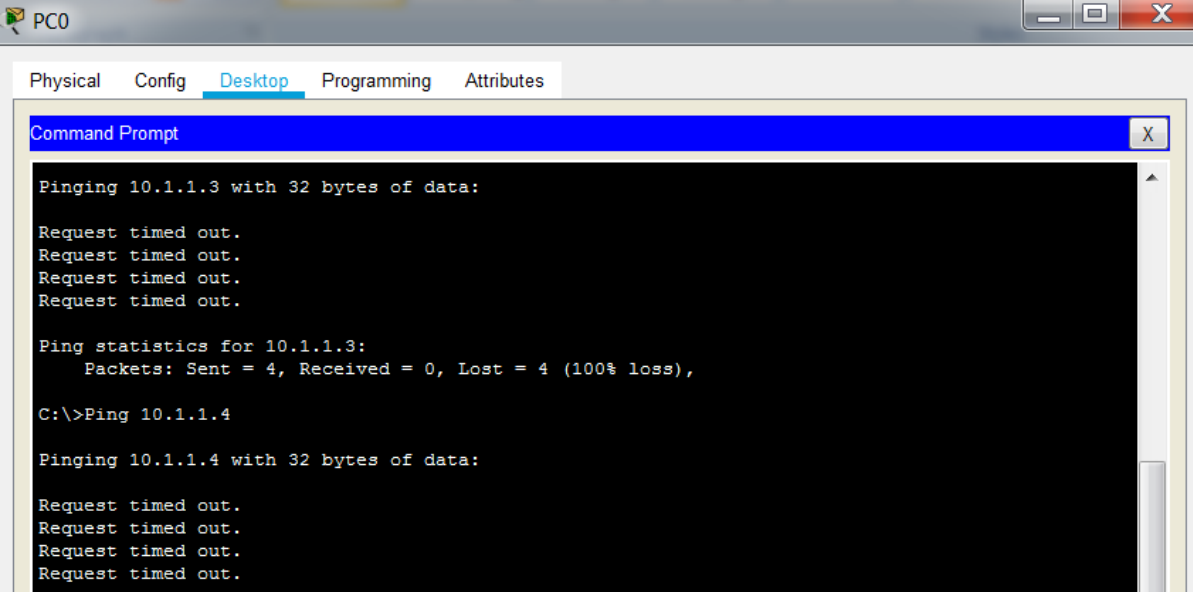
4.Server0-same as laptop0

5.Test Access PointA

a)Ping PC1(10.1.1.2) from PC0.The ping should succeed.



b) Ping Laptop0 (10.1.1.3) and Server0 (10.1.1.4) from PC0. The pings should fail.



The screenshot shows a Packet Tracer window for PC0. The 'Desktop' tab is active, displaying a Command Prompt. The user has entered two ping commands. The first command, 'ping 10.1.1.3', resulted in four 'Request timed out.' messages and statistics showing 100% loss. The second command, 'ping 10.1.1.4', also resulted in four 'Request timed out.' messages.

```
PC0
Physical Config Desktop Programming Attributes
Command Prompt
Pinging 10.1.1.3 with 32 bytes of data:
Request timed out.
Request timed out.
Request timed out.
Request timed out.

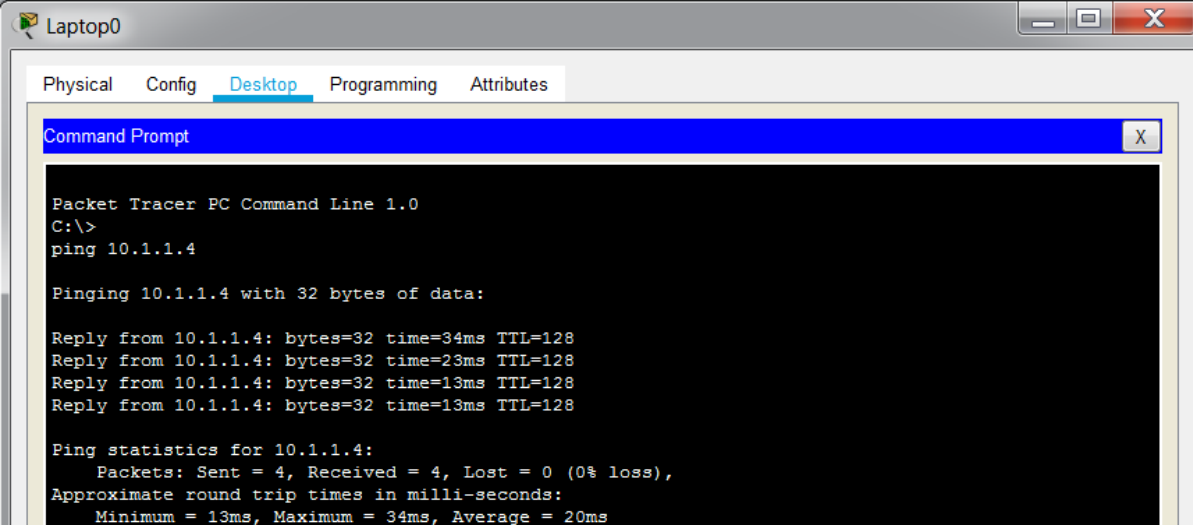
Ping statistics for 10.1.1.3:
    Packets: Sent = 4, Received = 0, Lost = 4 (100% loss),

C:\>ping 10.1.1.4

Pinging 10.1.1.4 with 32 bytes of data:
Request timed out.
Request timed out.
Request timed out.
Request timed out.
```

6. Test Access PointN

a) Ping Server0 (10.1.1.4) from Laptop0. The pings should succeed.



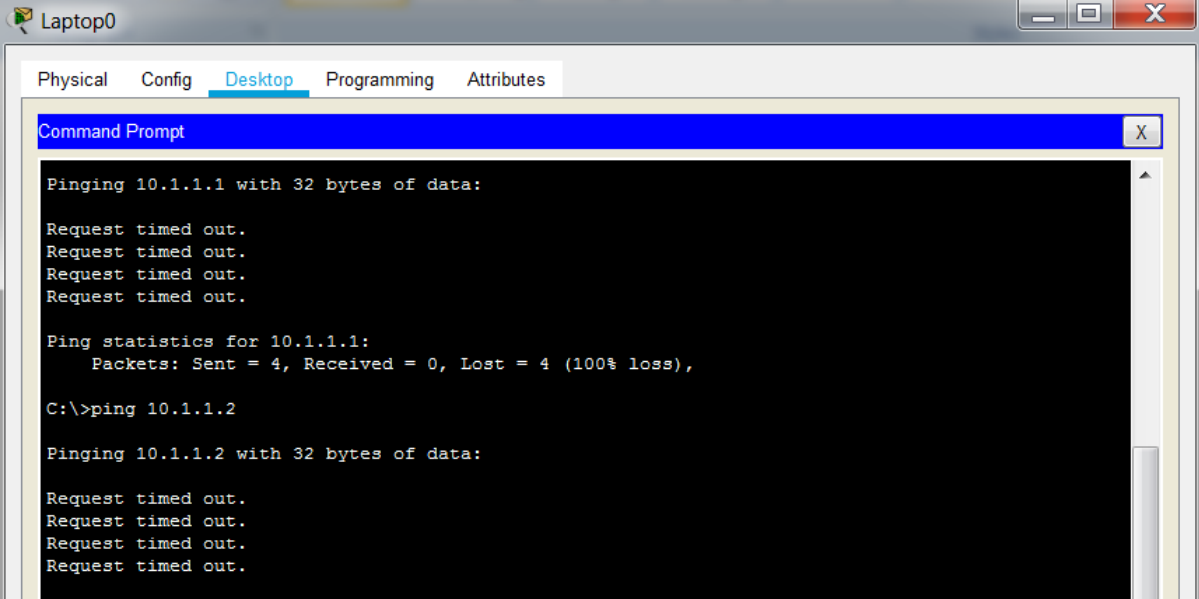
The screenshot shows a Packet Tracer window for Laptop0. The 'Desktop' tab is active, displaying a Command Prompt. The user has entered the command 'ping 10.1.1.4'. The output shows four successful replies from 10.1.1.4 with varying times and TTL values. The statistics show 0% loss.

```
Laptop0
Physical Config Desktop Programming Attributes
Command Prompt
Packet Tracer PC Command Line 1.0
C:\>
ping 10.1.1.4

Pinging 10.1.1.4 with 32 bytes of data:
Reply from 10.1.1.4: bytes=32 time=34ms TTL=128
Reply from 10.1.1.4: bytes=32 time=23ms TTL=128
Reply from 10.1.1.4: bytes=32 time=13ms TTL=128
Reply from 10.1.1.4: bytes=32 time=13ms TTL=128

Ping statistics for 10.1.1.4:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 13ms, Maximum = 34ms, Average = 20ms
```

b) Ping PC0(10.1.1.1) and PC1(10.1.1.2) from Laptop0. Pings should fail.



The screenshot shows the 'Laptop0' configuration window in Packet Tracer. The 'Desktop' tab is selected, displaying a 'Command Prompt' window. The command prompt shows the results of two ping commands. The first command, 'ping 10.1.1.1', shows four 'Request timed out.' messages and statistics indicating 100% loss. The second command, 'ping 10.1.1.2', also shows four 'Request timed out.' messages.

```
Command Prompt

Pinging 10.1.1.1 with 32 bytes of data:

Request timed out.
Request timed out.
Request timed out.
Request timed out.

Ping statistics for 10.1.1.1:
    Packets: Sent = 4, Received = 0, Lost = 4 (100% loss),

C:\>ping 10.1.1.2

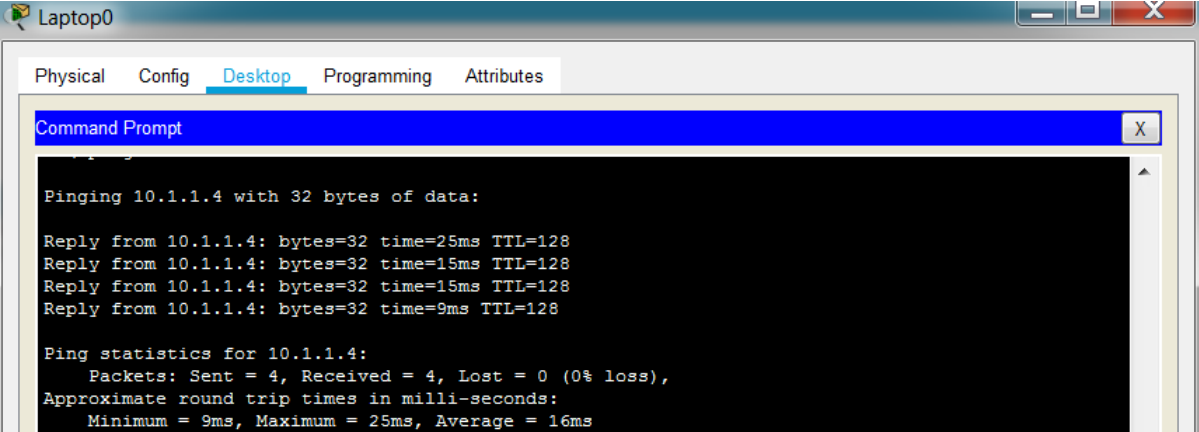
Pinging 10.1.1.2 with 32 bytes of data:

Request timed out.
Request timed out.
Request timed out.
Request timed out.
```

7. Test Access Point_B_G

a) Turn on Port1 on Access Point_B_G and turn off Port1 on Access PointN. Laptop0 and Server0 should associate with Access Point_B_G.

b) Ping Server0 from Laptop0. The Ping should succeed.



The screenshot shows the 'Laptop0' configuration window in Packet Tracer. The 'Desktop' tab is selected, displaying a 'Command Prompt' window. The command prompt shows the results of a ping command to 10.1.1.4. It shows four successful replies with varying round trip times (25ms, 15ms, 15ms, 9ms) and statistics indicating 0% loss.

```
Command Prompt

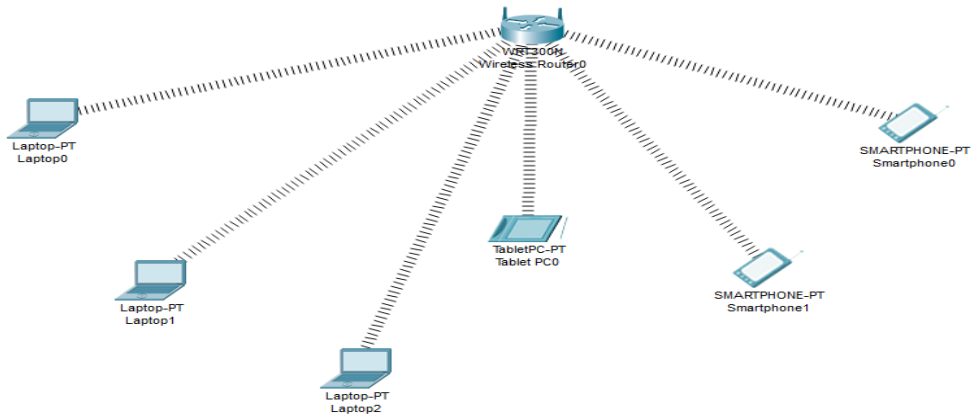
Pinging 10.1.1.4 with 32 bytes of data:

Reply from 10.1.1.4: bytes=32 time=25ms TTL=128
Reply from 10.1.1.4: bytes=32 time=15ms TTL=128
Reply from 10.1.1.4: bytes=32 time=15ms TTL=128
Reply from 10.1.1.4: bytes=32 time=9ms TTL=128

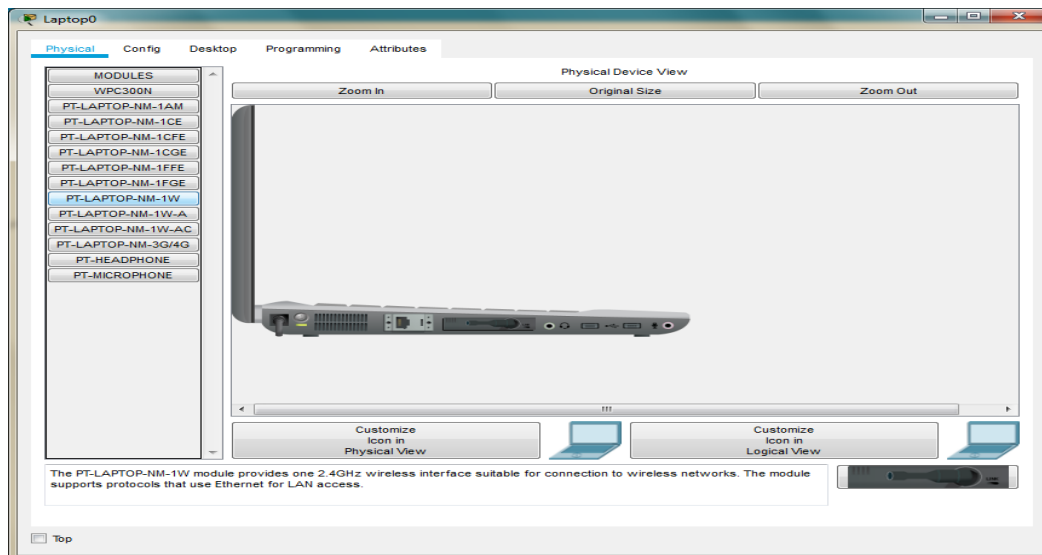
Ping statistics for 10.1.1.4:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 9ms, Maximum = 25ms, Average = 16ms
```

Practical No:8

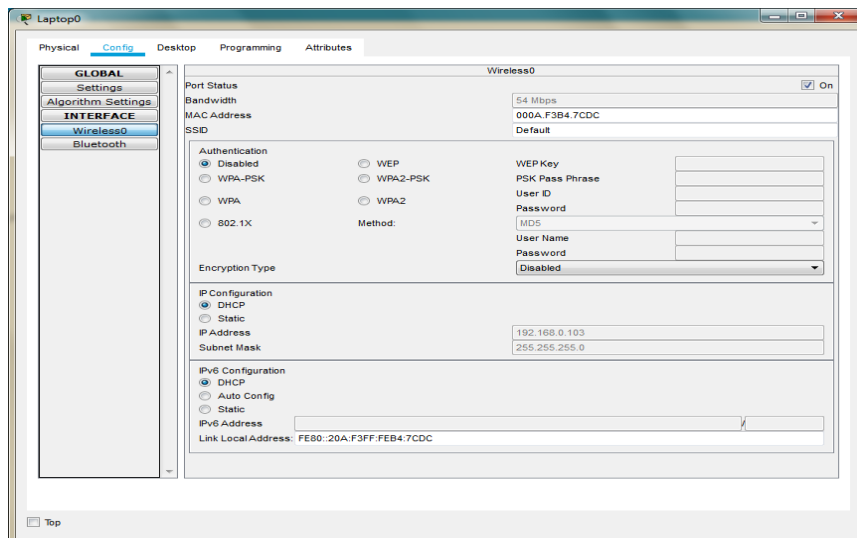
Consider the following topology

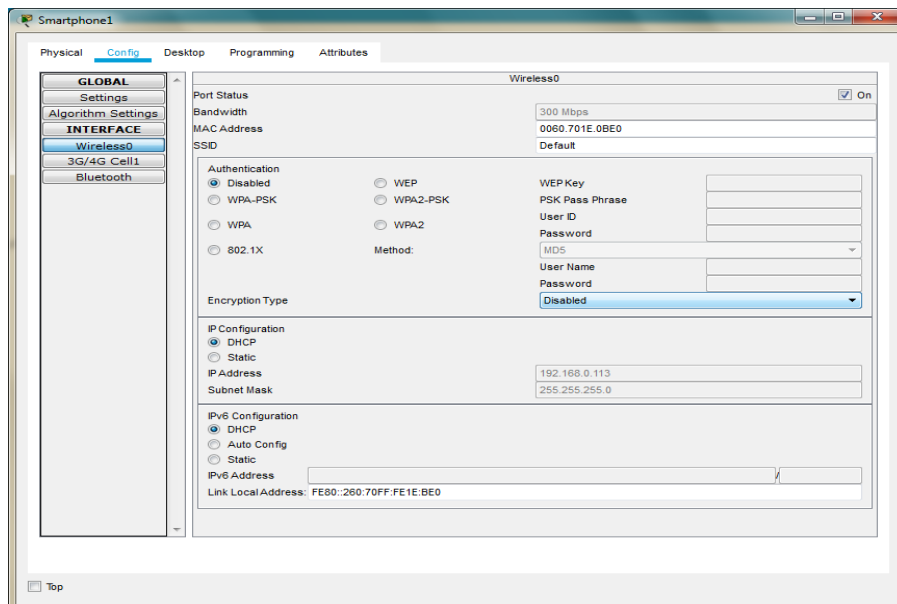
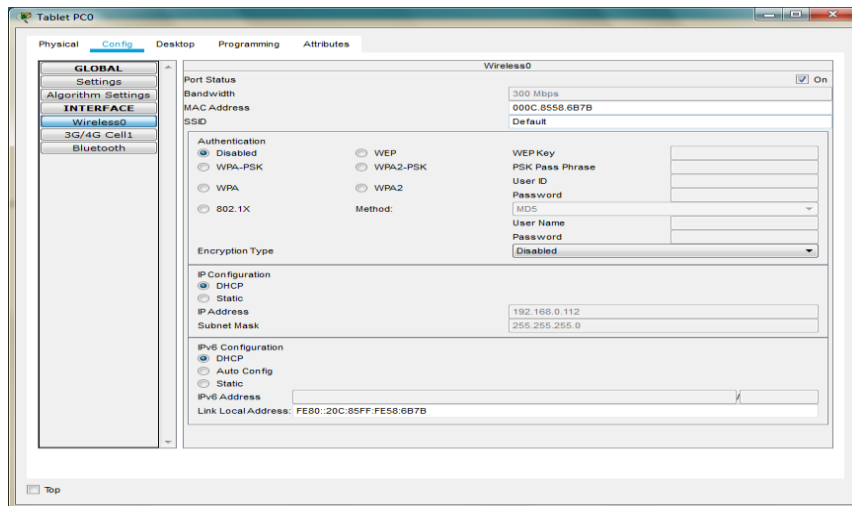


Adding the wireless interface to each Laptops



Copy the MAC address of each component as follows





We note the following MAC addresses and convert them to the following form

| Component | MAC Address | Converted MAC address |
|-------------|----------------|-----------------------|
| Laptop0 | 000A.F3B4.7CDC | 00:0A:F3:B4:7C:DC |
| Laptop1 | 0001.4269.6539 | 00:01:42:69:65:39 |
| Laptop2 | 0060.5CB8.B919 | 00:60:5C:B8:B9:19 |
| TabletPC | 000C.8558.6B7B | 00:0C:85:58:6B:7B |
| SmartPhone0 | 00D0.9774.32BD | 00:D0:97:74:32:BD |
| SmartPhone1 | 0060.701E.0BE0 | 00:60:70:1E:0B:E0 |

Now we add few addresses in the wireless MAC filter of the Wireless Router and then use the given options for either allow or deny the Wireless access

Wireless Router0

Physical Config **GUI** Attributes

Wireless-N Broadband Router

Firmware Ver

Wireless Setup Wireless Security Access Restrictions Applications & Gaming Administration

Basic Wireless Settings Wireless Security Guest Network Wireless MAC Filter Advanced Wireless

Wireless MAC Filter

Wireless Port: 2.4G

Enabled Disabled

Prevent PCs listed below from accessing the wireless network

Permit PCs listed below to access wireless network

Wireless Client List

| | | | |
|---------|-------------------|---------|-------------------|
| MAC 01: | 00:0A:F3:B4:7C:DC | MAC 26: | 00:00:00:00:00:00 |
| MAC 02: | 00:D0:97:74:32:BD | MAC 27: | 00:00:00:00:00:00 |
| MAC 03: | 00:0C:85:58:6B:7B | MAC 28: | 00:00:00:00:00:00 |
| MAC 04: | 00:00:00:00:00:00 | MAC 29: | 00:00:00:00:00:00 |
| MAC 05: | 00:00:00:00:00:00 | MAC 30: | 00:00:00:00:00:00 |

Help...

As seen in above screen shot we add the MAC address of Laptop0, TabletPC SmartPhone0 in the list so as to deny them accessing the Wireless network and then save the settings

Wireless Router0

Physical Config **GUI** Attributes

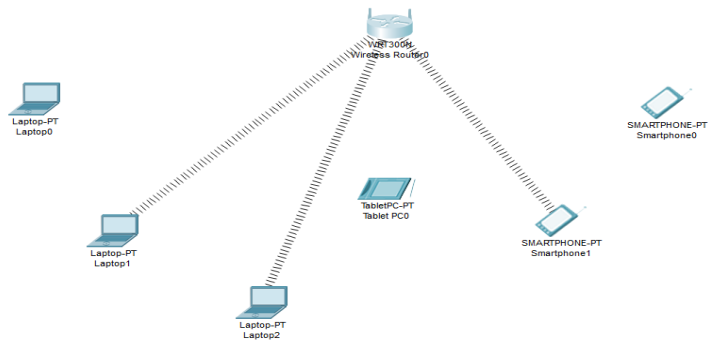
Permit PCs listed below to access wireless network

Wireless Client List

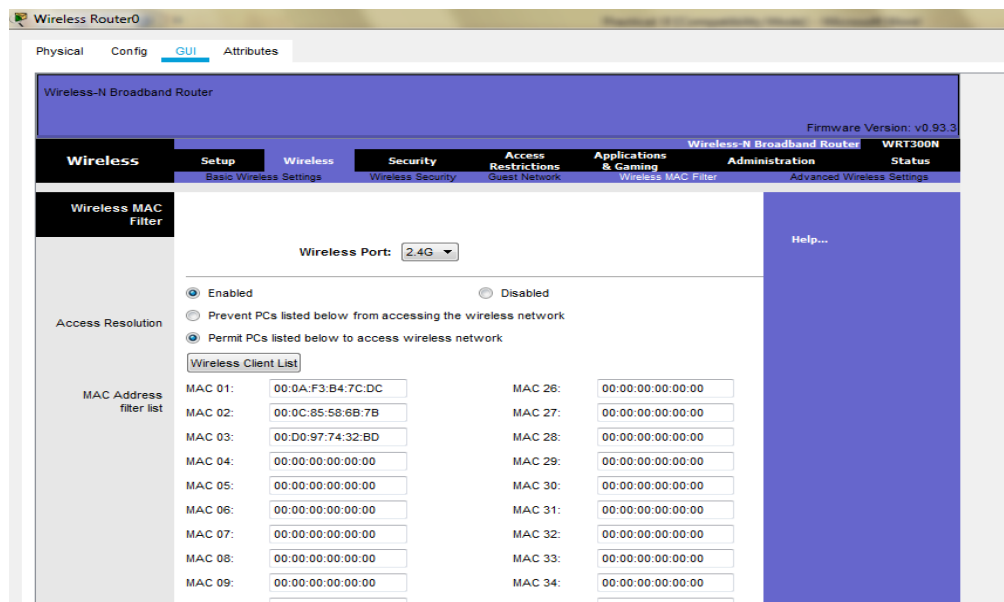
| | | | |
|---------|-------------------|---------|-------------------|
| MAC 01: | 00:0A:F3:B4:7C:DC | MAC 26: | 00:00:00:00:00:00 |
| MAC 02: | 00:0C:85:58:6B:7B | MAC 27: | 00:00:00:00:00:00 |
| MAC 03: | 00:D0:97:74:32:BD | MAC 28: | 00:00:00:00:00:00 |
| MAC 04: | 00:00:00:00:00:00 | MAC 29: | 00:00:00:00:00:00 |
| MAC 05: | 00:00:00:00:00:00 | MAC 30: | 00:00:00:00:00:00 |
| MAC 06: | 00:00:00:00:00:00 | MAC 31: | 00:00:00:00:00:00 |
| MAC 07: | 00:00:00:00:00:00 | MAC 32: | 00:00:00:00:00:00 |
| MAC 08: | 00:00:00:00:00:00 | MAC 33: | 00:00:00:00:00:00 |
| MAC 09: | 00:00:00:00:00:00 | MAC 34: | 00:00:00:00:00:00 |
| MAC 10: | 00:00:00:00:00:00 | MAC 35: | 00:00:00:00:00:00 |
| MAC 11: | 00:00:00:00:00:00 | MAC 36: | 00:00:00:00:00:00 |
| MAC 12: | 00:00:00:00:00:00 | MAC 37: | 00:00:00:00:00:00 |
| MAC 13: | 00:00:00:00:00:00 | MAC 38: | 00:00:00:00:00:00 |
| MAC 14: | 00:00:00:00:00:00 | MAC 39: | 00:00:00:00:00:00 |
| MAC 15: | 00:00:00:00:00:00 | MAC 40: | 00:00:00:00:00:00 |
| MAC 16: | 00:00:00:00:00:00 | MAC 41: | 00:00:00:00:00:00 |
| MAC 17: | 00:00:00:00:00:00 | MAC 42: | 00:00:00:00:00:00 |
| MAC 18: | 00:00:00:00:00:00 | MAC 43: | 00:00:00:00:00:00 |
| MAC 19: | 00:00:00:00:00:00 | MAC 44: | 00:00:00:00:00:00 |
| MAC 20: | 00:00:00:00:00:00 | MAC 45: | 00:00:00:00:00:00 |
| MAC 21: | 00:00:00:00:00:00 | MAC 46: | 00:00:00:00:00:00 |
| MAC 22: | 00:00:00:00:00:00 | MAC 47: | 00:00:00:00:00:00 |
| MAC 23: | 00:00:00:00:00:00 | MAC 48: | 00:00:00:00:00:00 |
| MAC 24: | 00:00:00:00:00:00 | MAC 49: | 00:00:00:00:00:00 |
| MAC 25: | 00:00:00:00:00:00 | MAC 50: | 00:00:00:00:00:00 |

Save Settings Cancel Changes

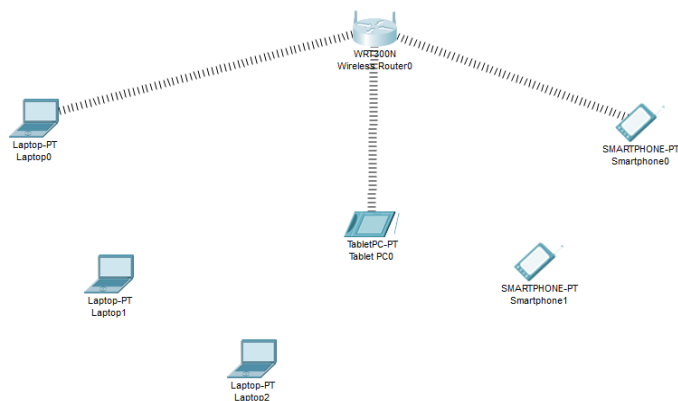
The result so obtained is as shown, the three devices denied any wireless connectivity



Similarly we can change the setting so that the above devices get wireless connectivity and the remaining devices do not get the wireless connectivity

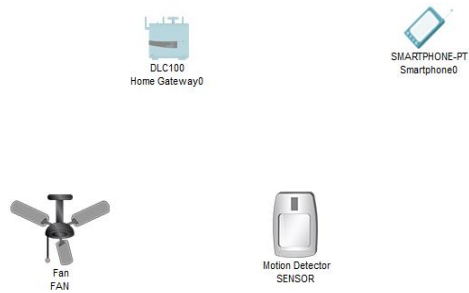


And save the setting and get the following

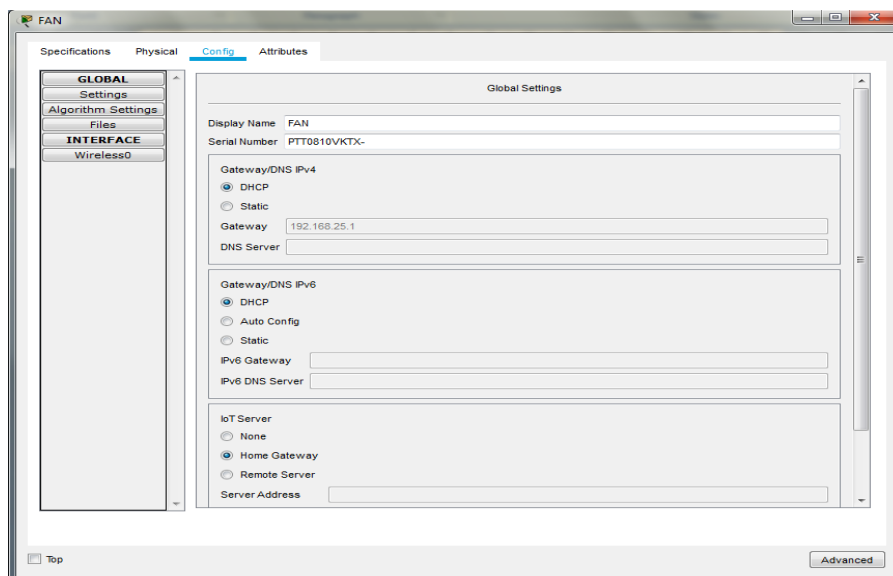


Practical No:9

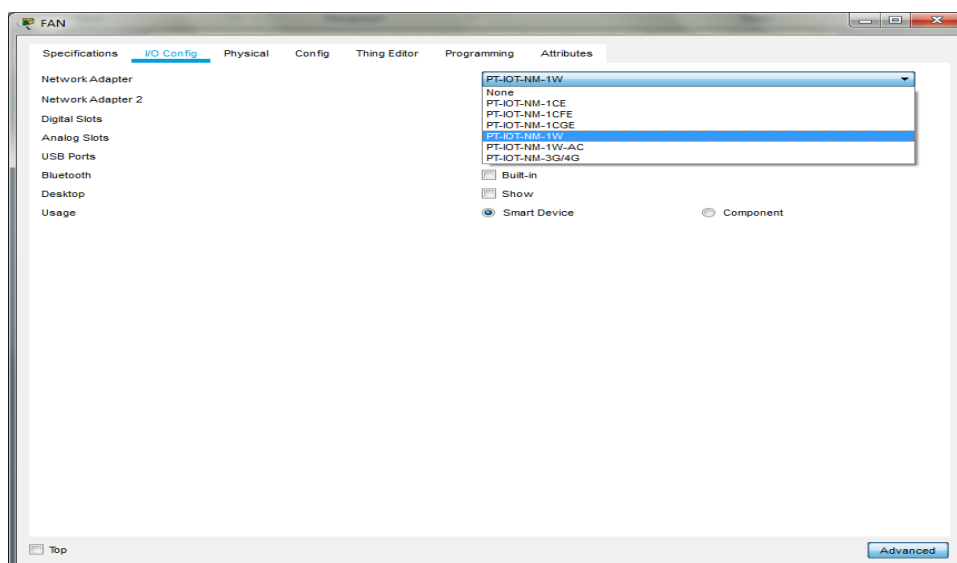
Consider the following topology



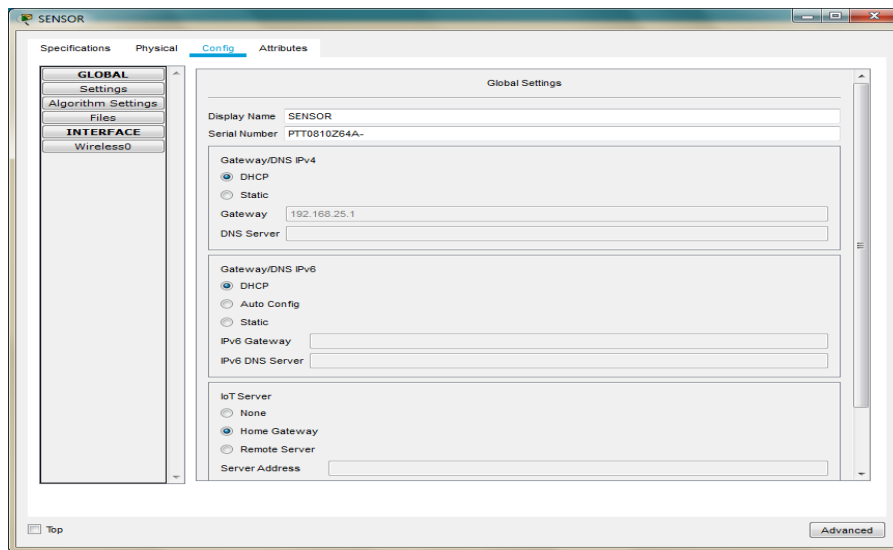
Click on the Fan and do the following



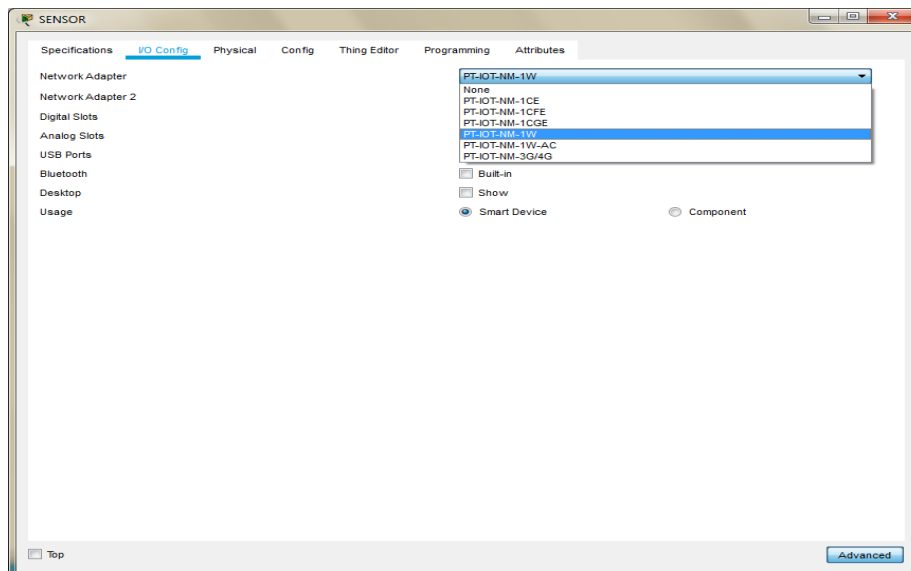
In the Advanced setting do the following for the Network adapter



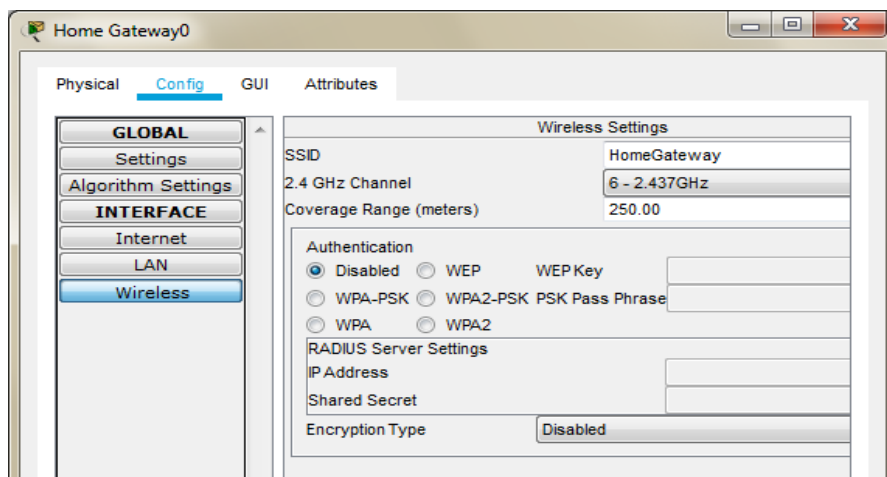
For the motion Detector sensor do the following



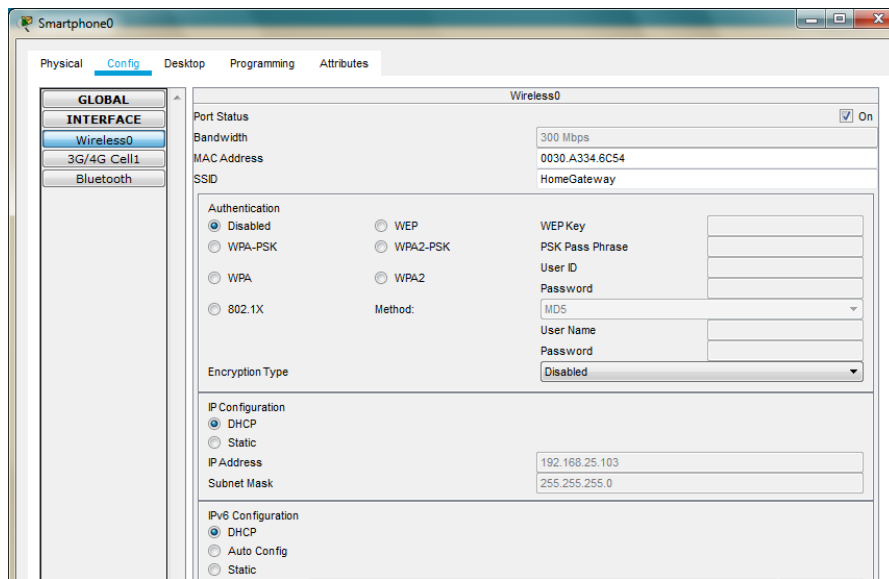
In the Advanced setting do the following for the Network adapter



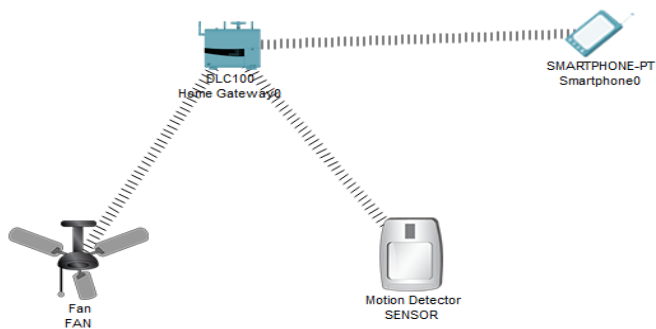
For the smartphone change the SSID to the SSID in the Home Gateway0



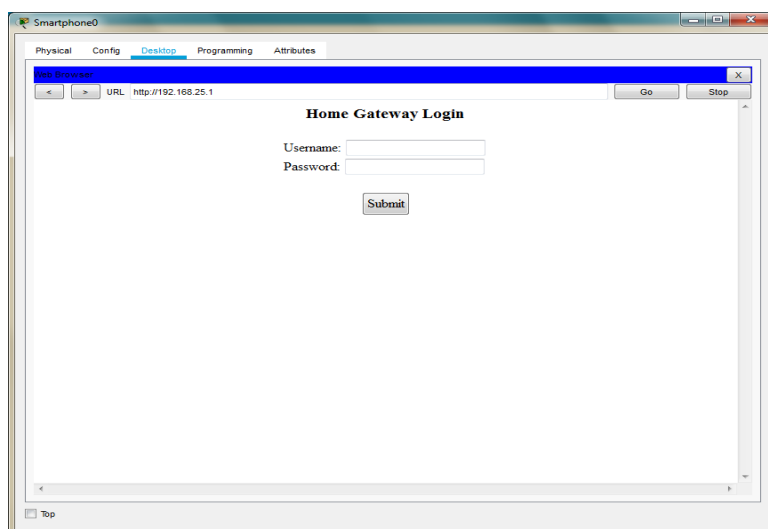
As seen above the SSID is HomeGateway, we use the same and set the SSID in the Smartphone



All the devices are now connected to the Home Gateway



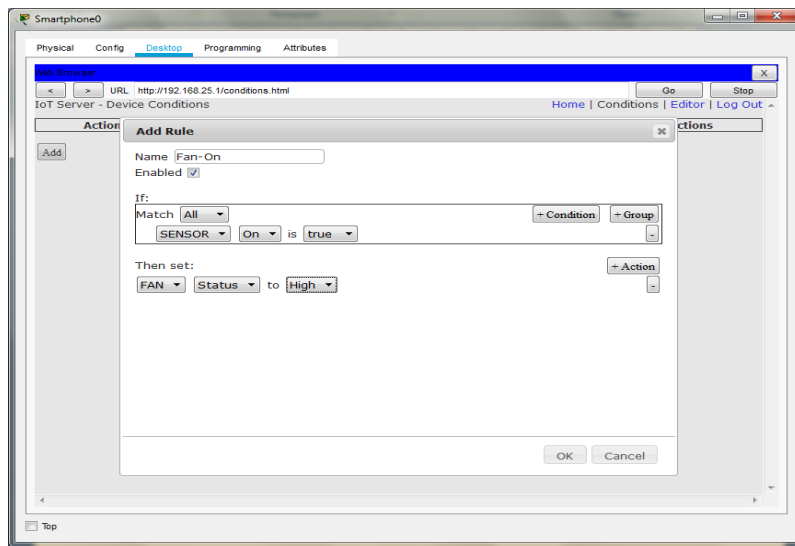
Now open the Web browser of the SmartPhone and type the IP address of the HomeGateway



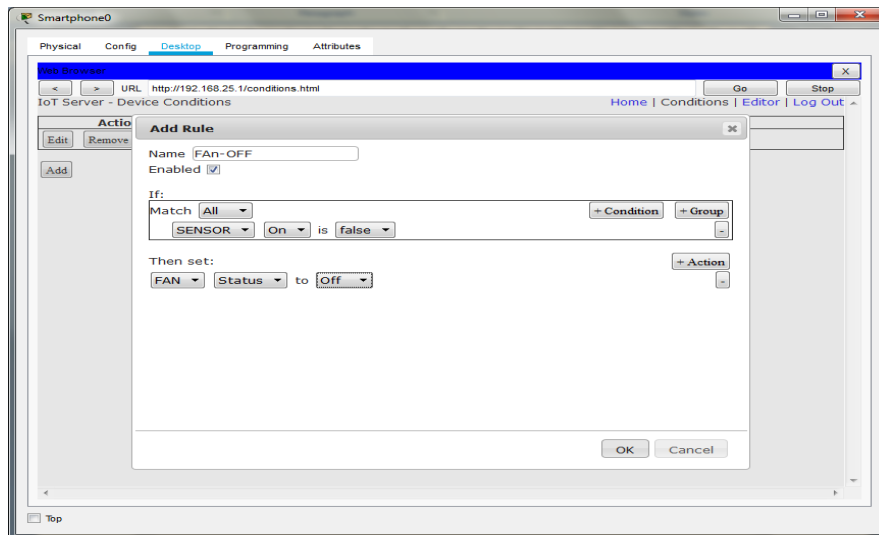
Username : admin

Password : admin

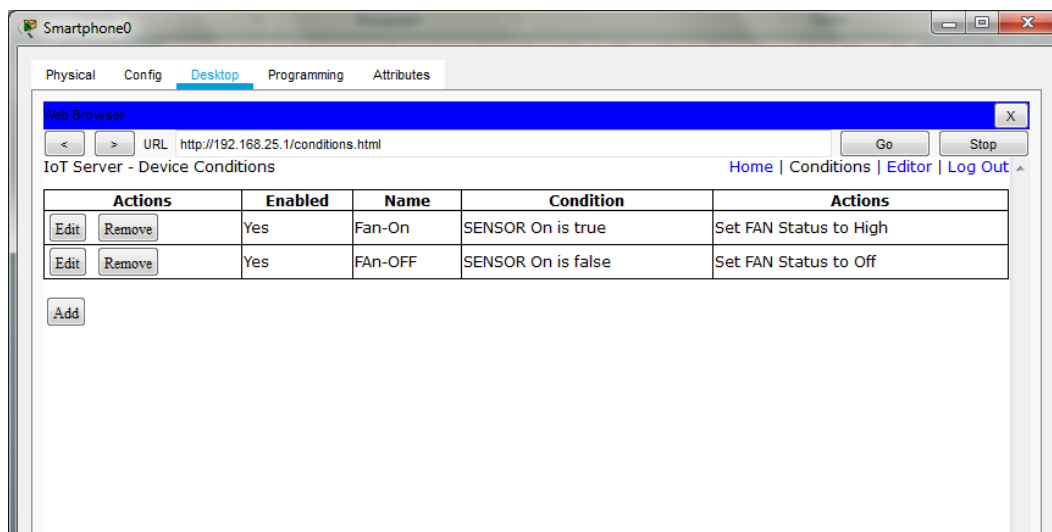
After logging click on conditions and do the following



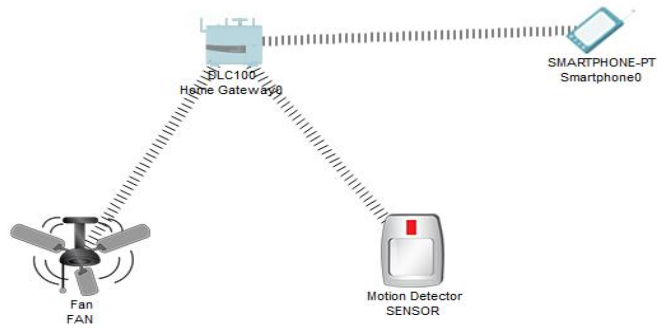
Add another condition as follows



Press the go button after adding the two conditions



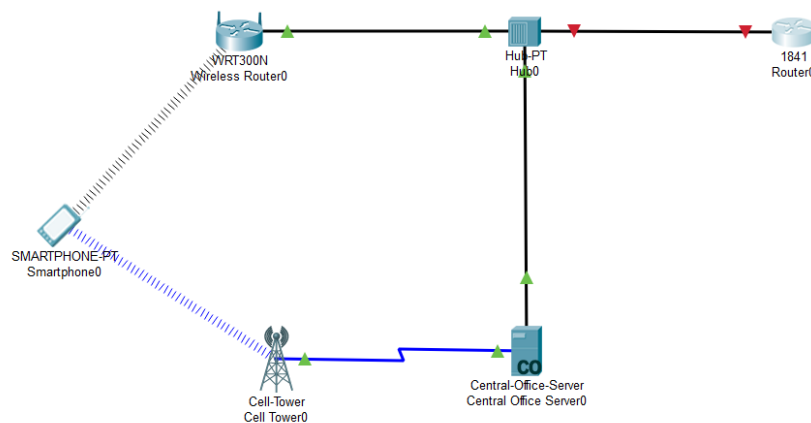
In order to turn ON the fan Press the ALT key and left-click the mouse over the Sensor



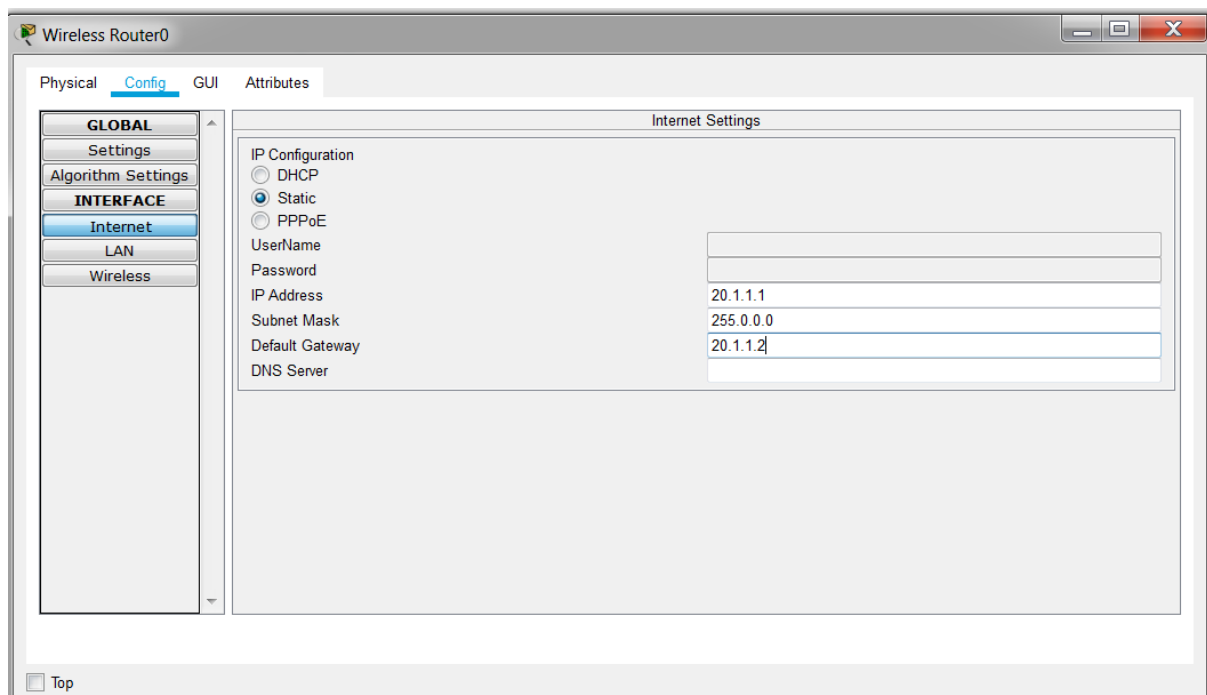
Practical No:10

Steps :-

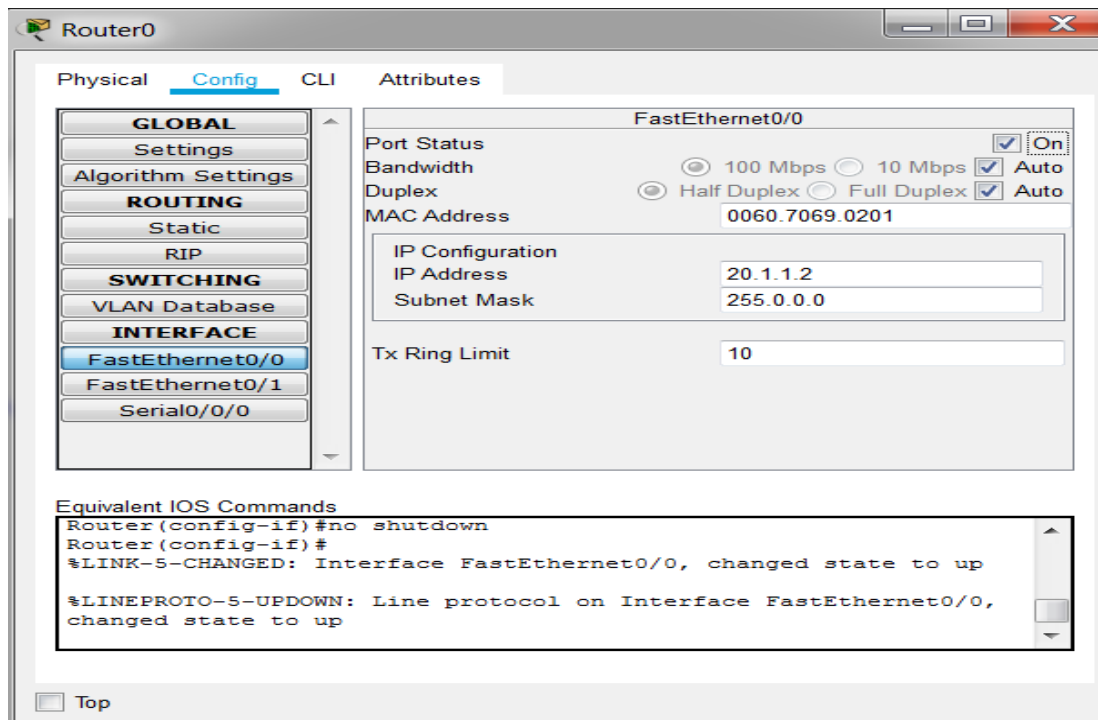
- 1) Create a network using smartphone, wireless router WRT300N, Hub-pt, 1841 Router, central-office-server, Cell-Tower.
- 2) Connect cell tower and central office server using coaxial cable.
- 3) Connect wireless router WRT300N, Hub-pt, 1841 Router, central-office-server using copper straight through wire.



- 4) Click on wireless router.in config tab select internet.in internet choose ip configuration as static and set ip address and default gateway.



5) Click on router 1841. In config tab select interface and give ip address.



6) Click on smartphone and ping router1841

