QUESTÃO 01

```
#!/usr/bin/python

'Setting the position of nodes'

from mininet.node import Controller
from mininet.log import setLogLevel, info
from mininet.wifi.cli import CLI_wifi
from mininet.wifi.node import OVSKernelAP
from mininet.wifi.net import Mininet_wifi


def topology():

    net = Mininet_wifi(controller=Controller, accessPoint=OVSKernelAP)

    info("*** Creating nodes\n")
    net.addStation('sta1', mac='00:00:00:00:00:02', ip='10.0.0.1/8',
                   position='30,50,0')
    net.addStation('sta2', mac='00:00:00:00:00:03', ip='10.0.0.2/8',
                   position='50,70,0')
    net.addStation('sta3', mac='00:00:00:00:00:04', ip='10.0.0.3/8',
                   position='100,80,0')
    net.addStation('sta4', mac='00:00:00:00:00:05', ip='10.0.0.4/8',
                   position='120,70,0')

    ap1 = net.addAccessPoint('ap1', ssid='new-ssid1', mode='g', channel='1',
                             position='50,50,0')
    ap2 = net.addAccessPoint('ap2', ssid='new-ssid2', mode='g', channel='1',
                             position='120,50,0')

    c1 = net.addController('c1', controller=Controller)
    h1 = net.addHost('h1', ip='10.0.0.3/8')
    h2 = net.addHost('h2', ip='10.0.0.4/8')

    net.propagationModel(model="logDistance", exp=4.5)

    info("*** Configuring wifi nodes\n")
    net.configureWifiNodes()

    info("*** Creating links\n")
    net.addLink(ap1, h1)

    net.plotGraph(max_x=200, max_y=200)

    info("*** Starting network\n")
    net.build()
    c1.start()
    ap1.start([c1])
    ap2.start([c1])
    info("*** Running CLI\n")
```

```
    CLI_wifi(net)

    info("*** Stopping network\n")
    net.stop()


if __name__ == '__main__':
    setLogLevel('info')
    topology()
```

QUESTÃO 02

Não, pois cada uma estação está conectada ao seu ap e com um ip com isso não tem como se conectar com mais de um ap.

QUESTÃO 03

```
#!/usr/bin/python

"""This example shows how to enable 4-address
Warning: It works only when network manager is stopped"""

from mininet.node import Controller
from mininet.log import setLogLevel, info
from mininet.wifi.node import OVSKernelAP
from mininet.wifi.link import wmediumd, _4address
from mininet.wifi.cli import CLI_wifi
from mininet.wifi.net import Mininet_wifi
from mininet.wifi.wmediumdConnector import interference


def topology():
    "Create a network."
    net = Mininet_wifi( controller=Controller, accessPoint=OVSKernelAP,
                        link=wmediumd, wmediumd_mode=interference,
                        configure4addr=True, autoAssociation=False )

    info("*** Creating nodes\n")
    ap1 = net.addAccessPoint( 'ap1', _4addr="ap", ssid="wds-ssid1",
                              mode="g", channel="1", position='80,70,0' )
    ap2 = net.addAccessPoint( 'ap2', _4addr="client", ssid="wds-ssid2",
                              mode="g", channel="1", position='150,100,0' )
    ap3 = net.addAccessPoint( 'ap3', _4addr="client", ssid="wds-ssid3",
                              mode="g", channel="1", position='180,90,0' )
    sta1 = net.addStation( 'sta1', ip="192.168.0.1/24", position='85,30,0' )
    sta2 = net.addStation( 'sta2', ip="192.168.0.2/24", position='100,40,0' )
    sta3 = net.addStation( 'sta3', ip="192.168.0.3/24", position='120,75,0' )
    sta4 = net.addStation( 'sta4', ip="192.168.0.4/24", position='130,70,0' )
    sta5 = net.addStation( 'sta5', ip="192.168.0.5/24", position='140,90,0' )
    sta6 = net.addStation( 'sta6', ip="192.168.0.6/24", position='160,85,0' )
    c0 = net.addController('c0', controller=Controller, ip='127.0.0.1',
                           port=6633)
```

```
    info("*** Configuring Propagation Model\n")
    net.propagationModel(model="logDistance", exp=4.5)

    info("*** Configuring wifi nodes\n")
    net.configureWifiNodes()

    info("*** Adding Link\n")
    net.addLink(ap1, ap2, cls=_4address)
    net.addLink(ap1, ap3, cls=_4address)
    net.addLink(sta1, ap1)
    net.addLink(sta2, ap1)
    net.addLink(sta3, ap2)
    net.addLink(sta4, ap2)
    net.addLink(sta5, ap3)
    net.addLink(sta6, ap3)

    net.plotGraph(max_x=200, max_y=200)

    info("*** Starting network\n")
    net.build()
    c0.start()
    ap1.start( [c0] )
    ap2.start( [c0] )
    ap3.start( [c0] )

    info("*** Running CLI\n")
    CLI_wifi(net)

    info("*** Stopping network\n")
    net.stop()


if __name__ == '__main__':
    setLogLevel( 'info' )
    topology()
```