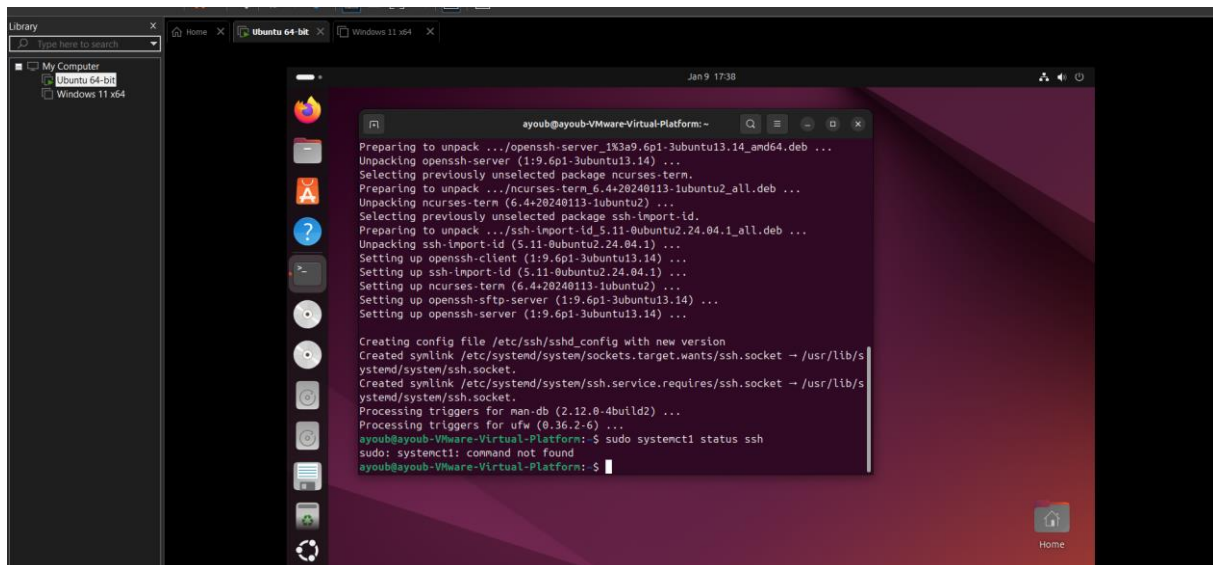


Template Week 6 – Networking

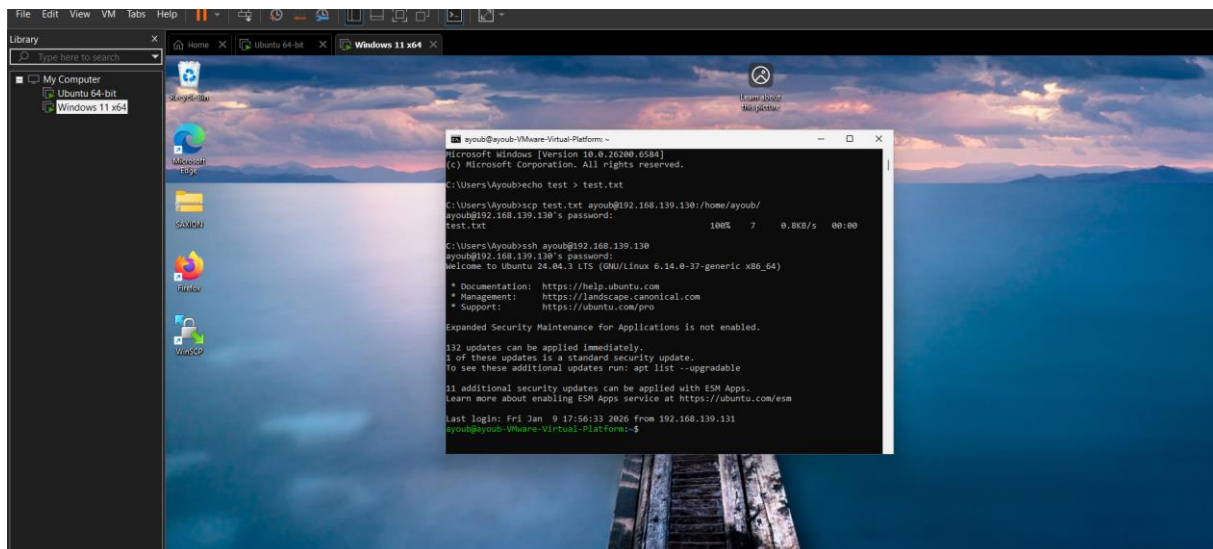
Student number:

Assignment 6.1: Working from home

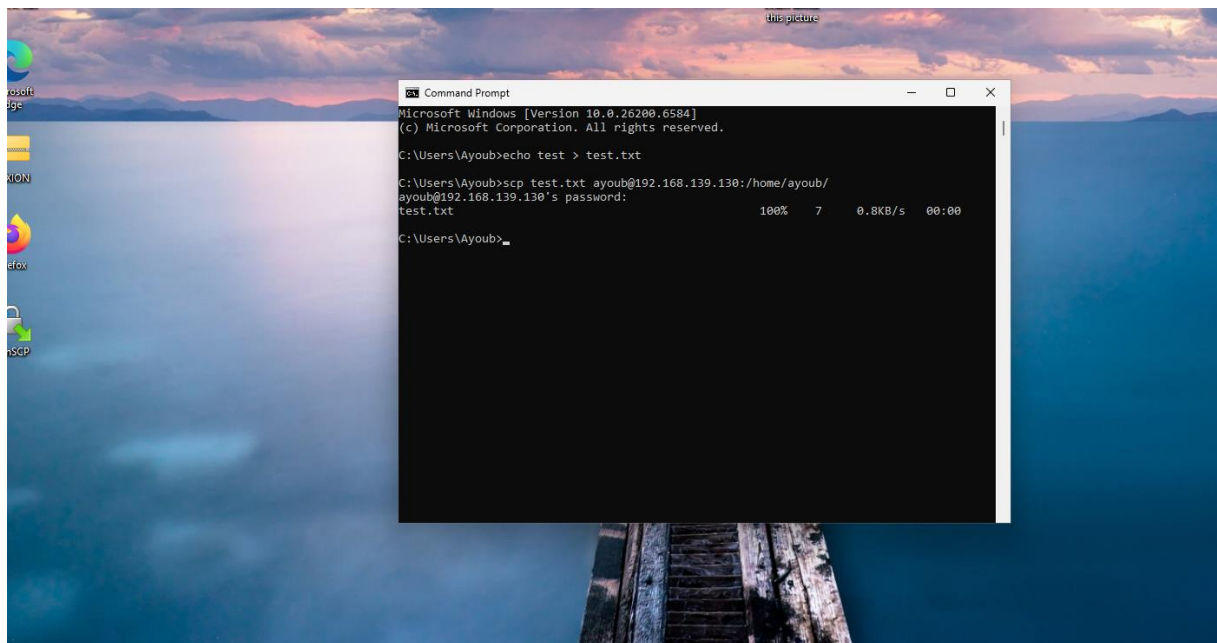
Screenshot installation openssh-server:



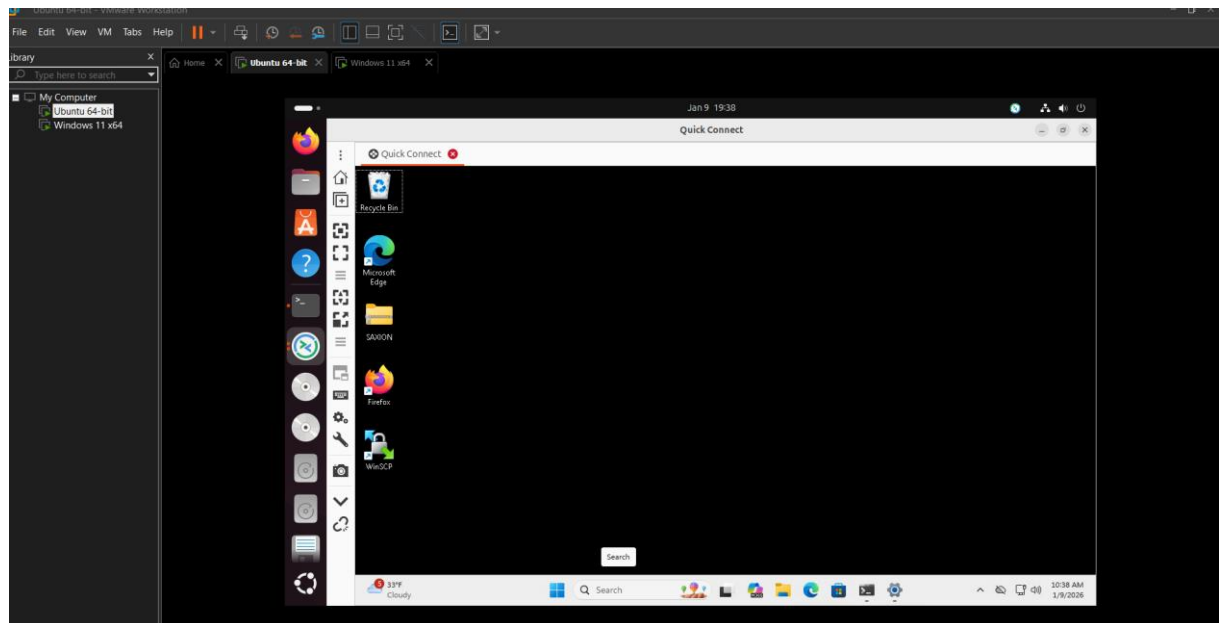
Screenshot successful SSH command execution:



Screenshot successful execution SCP command:



Screenshot remmina:



Assignment 6.2: IP addresses websites

Relevant screenshots nslookup command:

```
Jan 9 19:44
ayoub@ayoub-VMware-Virtual-Platform: ~
ayoub@ayoub-VMware-Virtual-Platform:~$ nslookup
> amazon.com
Server:      127.0.0.53
Address:     127.0.0.53#53

Non-authoritative answer:
Name:   amazon.com
Address: 98.82.161.185
Name:   amazon.com
Address: 98.87.170.71
Name:   amazon.com
Address: 98.87.170.74
> google.com
Server:      127.0.0.53
Address:     127.0.0.53#53

Non-authoritative answer:
Name:   google.com
Address: 142.251.142.206
Name:   google.com
Address: 2a00:1450:400e:805::200e
> one.one.one.one
Server:      127.0.0.53
Address:     127.0.0.53#53

Non-authoritative answer:
Name:   one.one.one.one
Address: 1.1.1.1
Name:   one.one.one.one
Address: 1.0.0.1
Name:   one.one.one.one
Address: 2606:4700:4700::1111
```

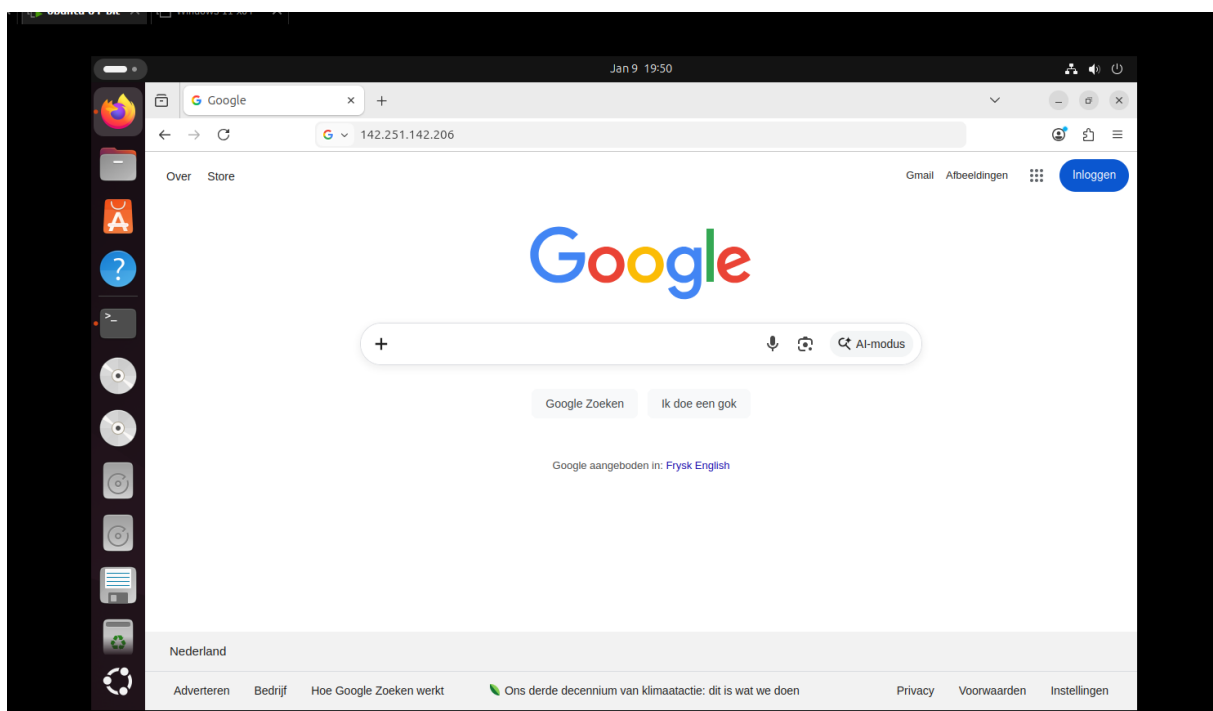
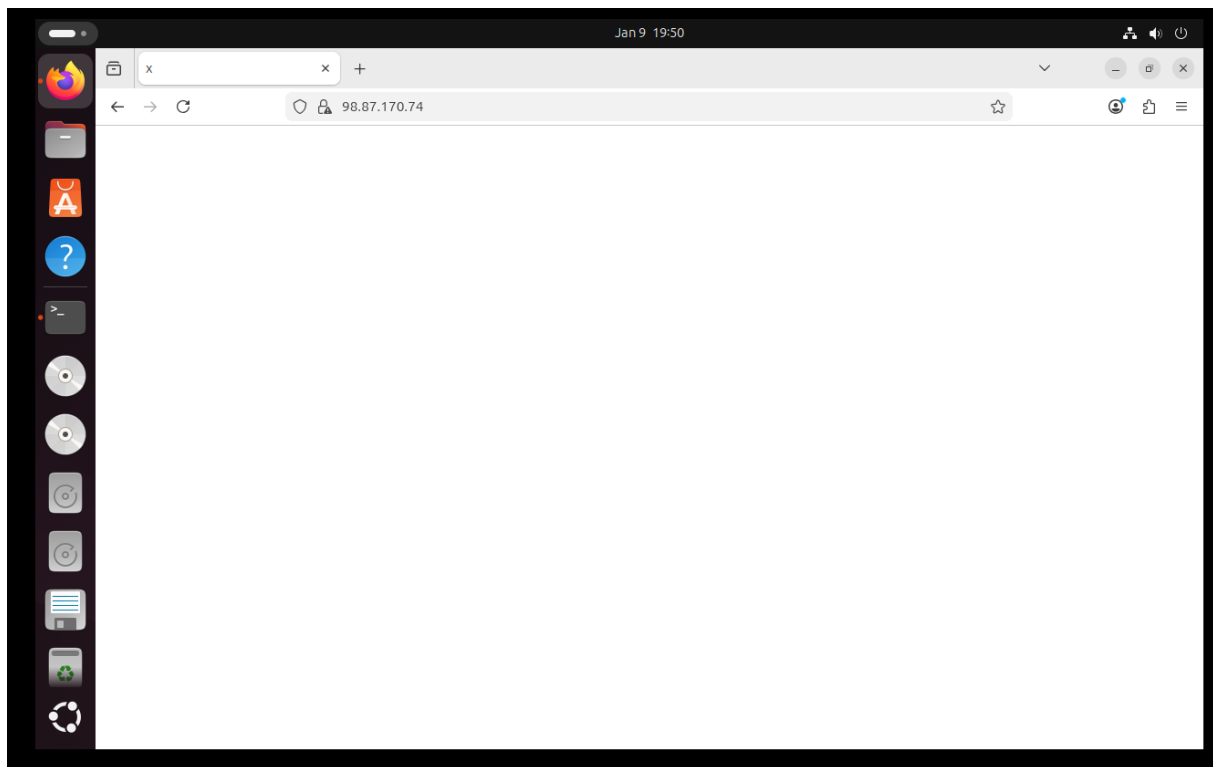
```
Name:   one.one.one.one
Address: 2606:4700:4700::1001
> dns.google.com
Server:      127.0.0.53
Address:     127.0.0.53#53

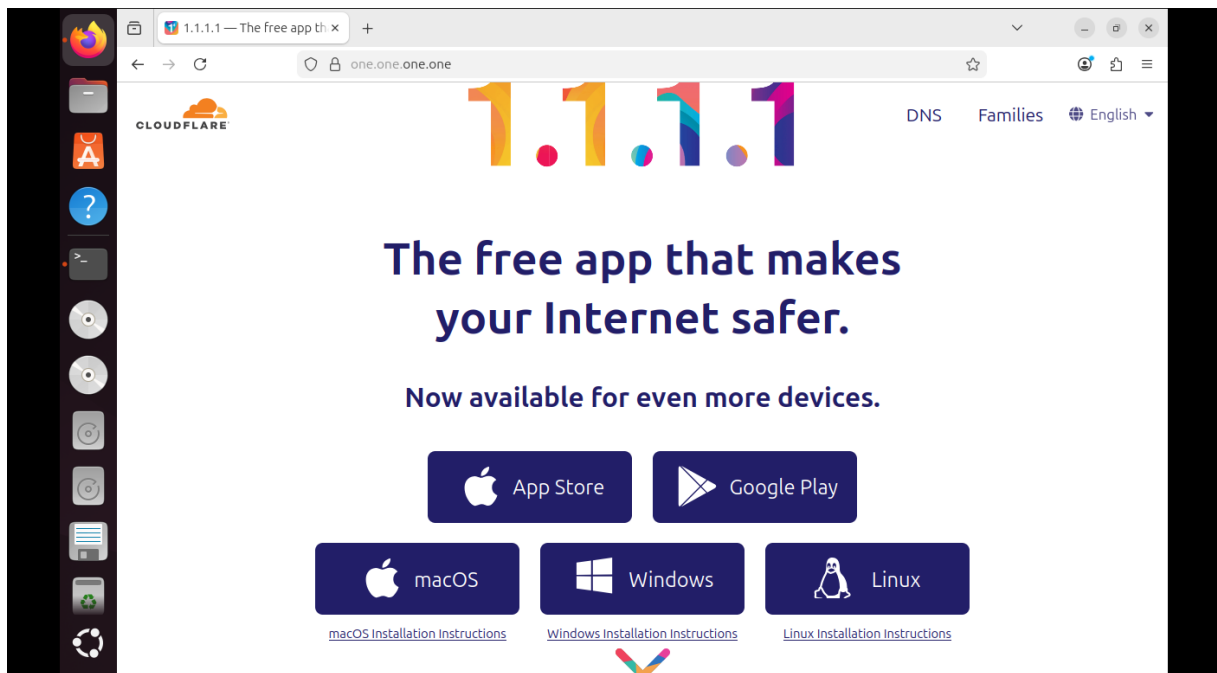
Non-authoritative answer:
Name:   dns.google.com
Address: 8.8.4.4
Name:   dns.google.com
Address: 8.8.8.8
Name:   dns.google.com
Address: 2001:4860:4860::8844
Name:   dns.google.com
Address: 2001:4860:4860::8888
> bol.com
Server:      127.0.0.53
Address:     127.0.0.53#53

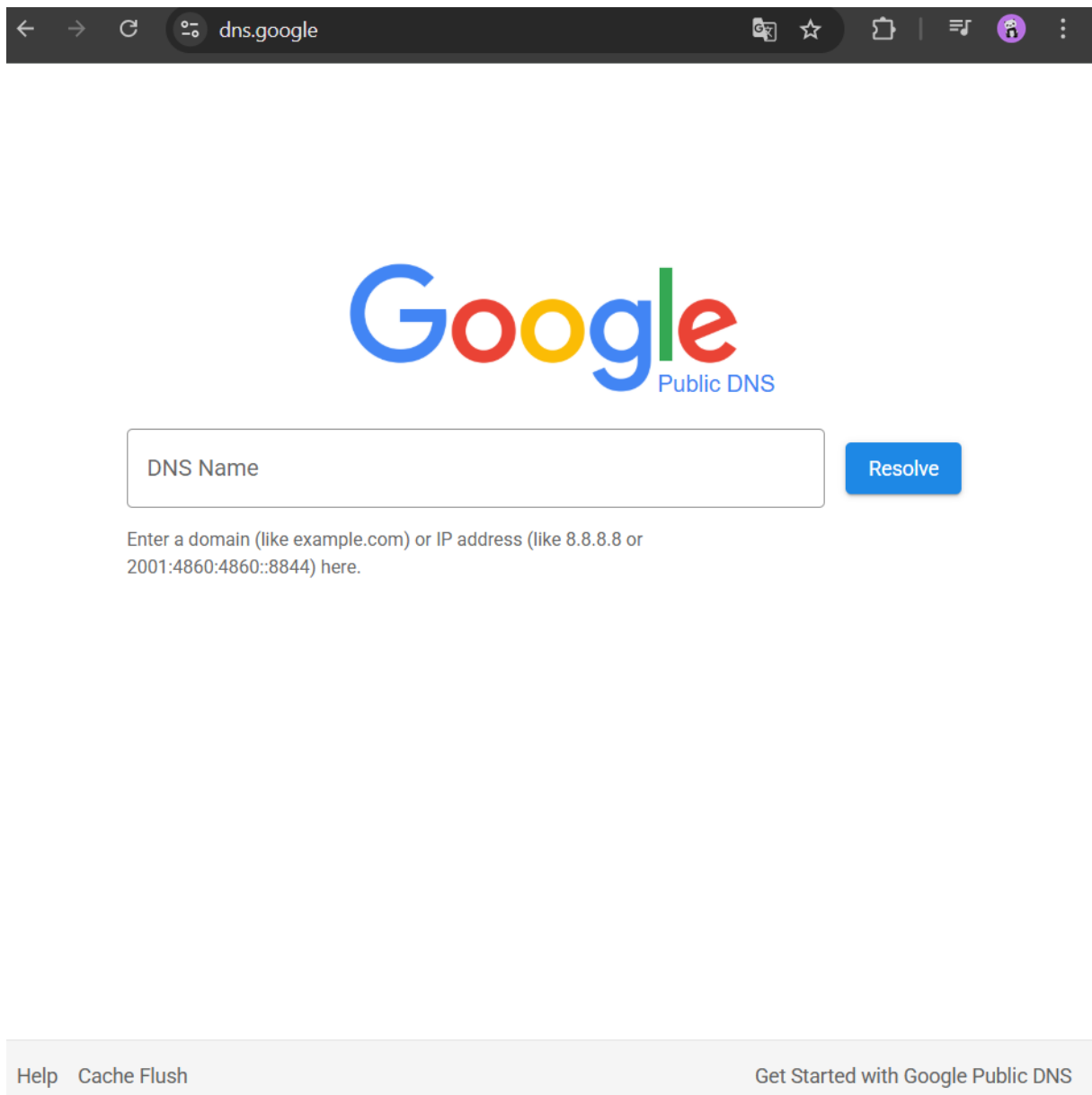
Non-authoritative answer:
Name:   bol.com
Address: 79.170.100.62
> w3schools.com
Server:      127.0.0.53
Address:     127.0.0.53#53

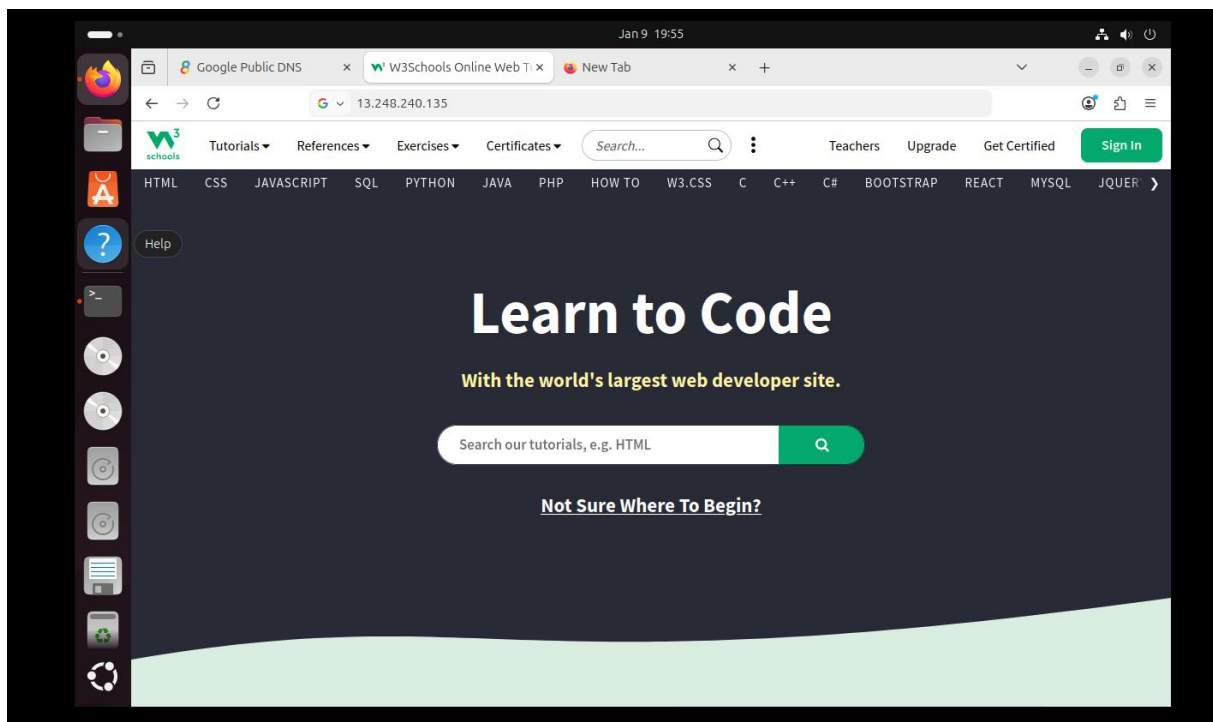
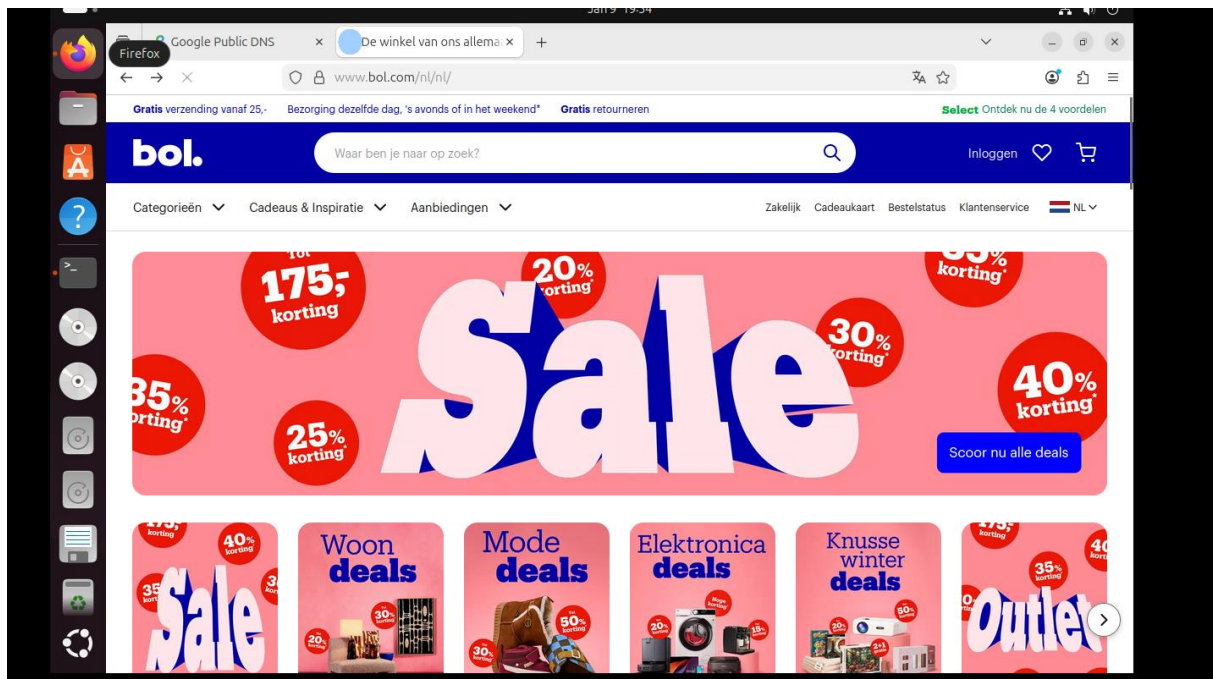
Non-authoritative answer:
Name:   w3schools.com
Address: 13.248.240.135
Name:   w3schools.com
Address: 76.223.115.82
>
```

Screenshot website visit via IP address:









Assignment 6.3: subnetting

How many IP addresses are in this network configuration 192.168.110.128/25?

126

What is the usable IP range to hand out to the connected computers?

192.168.110.128

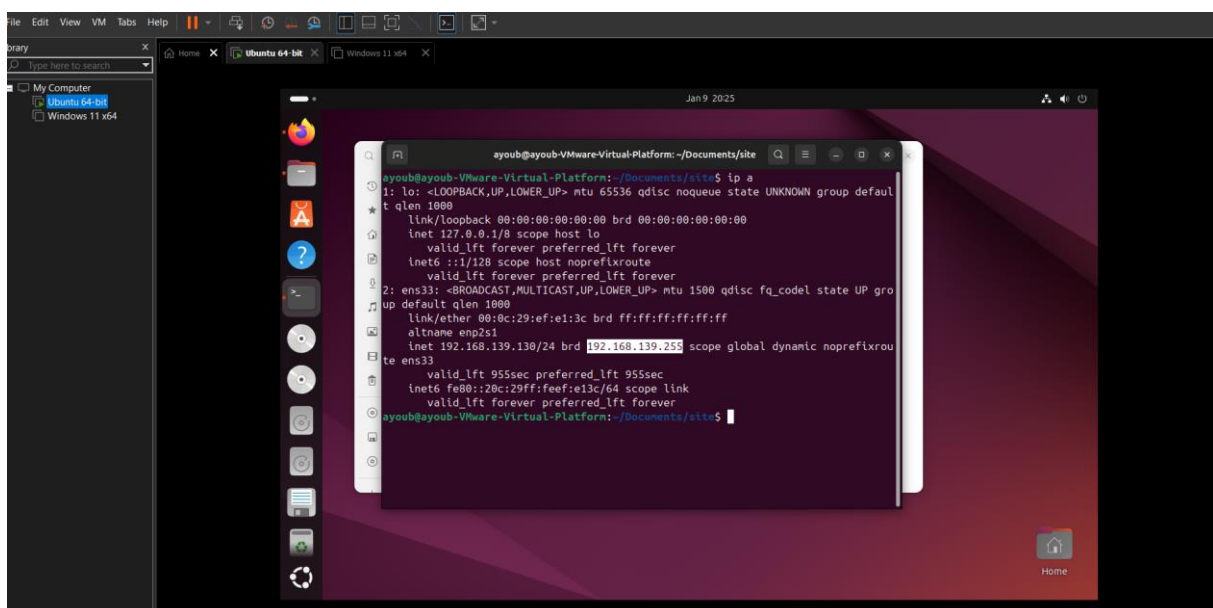
Check your two previous answers with this Linux command: `ipcalc 192.168.110.128/25`

Explain the above calculation in your own words.

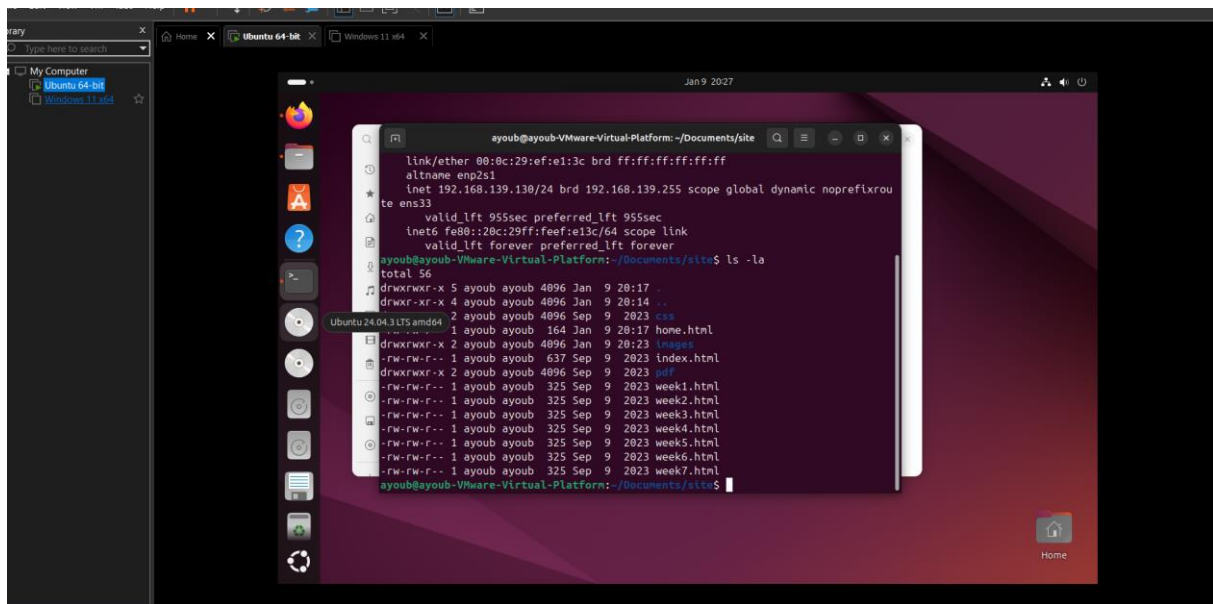
Zo kun je zien dat host min 192.168.110.129 en host max is 192.168.110.254, daar zitten precies 125 possible adressen.

Assignment 6.4: HTML

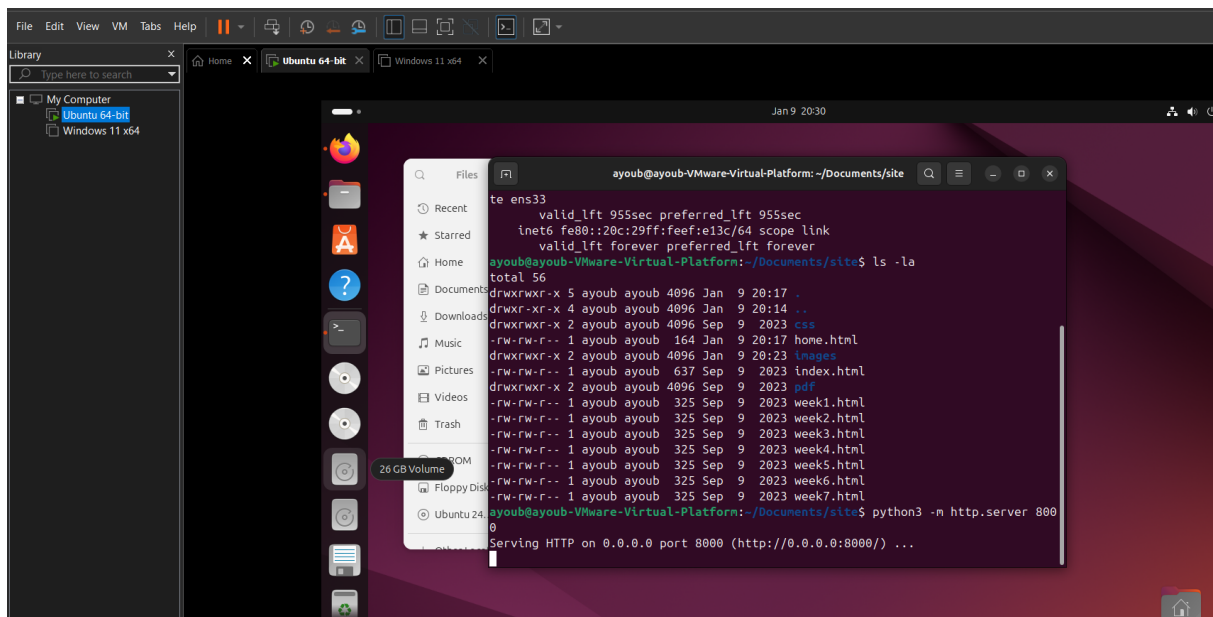
Screenshot IP address Ubuntu VM:



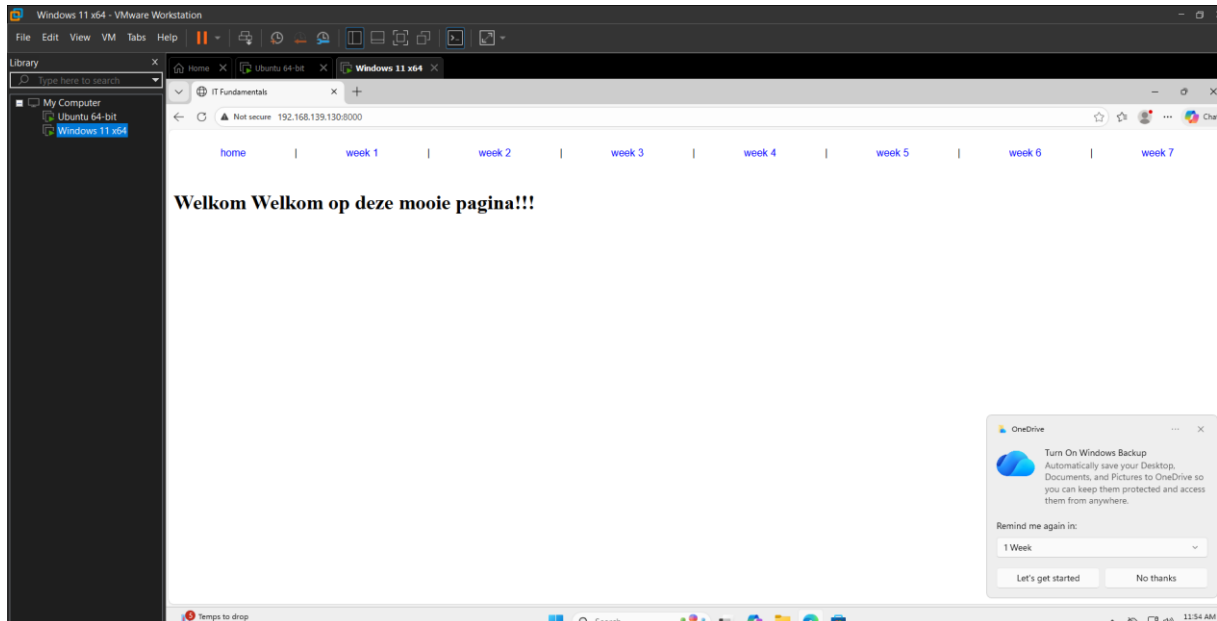
Screenshot of Site directory contents:



Screenshot python3 webserver command:



Screenshot web browser visits your site



Assignment 6.5: Network segment

Remember that bitwise java application you've made in week 2? Expand that application so that you can also calculate a network segment as explained in the PowerPoint slides of week 6. Use the bitwise & AND operator. You need to be able to input two Strings. An IP address and a subnet.

IP: 192.168.1.100 and subnet: 255.255.255.224 for /27

Example: 192.168.1.100/27

Calculate the network segment

IP Address: 11000000.10101000.00000001.01100100

Subnet Mask: 11111111.11111111.11111111.11100000

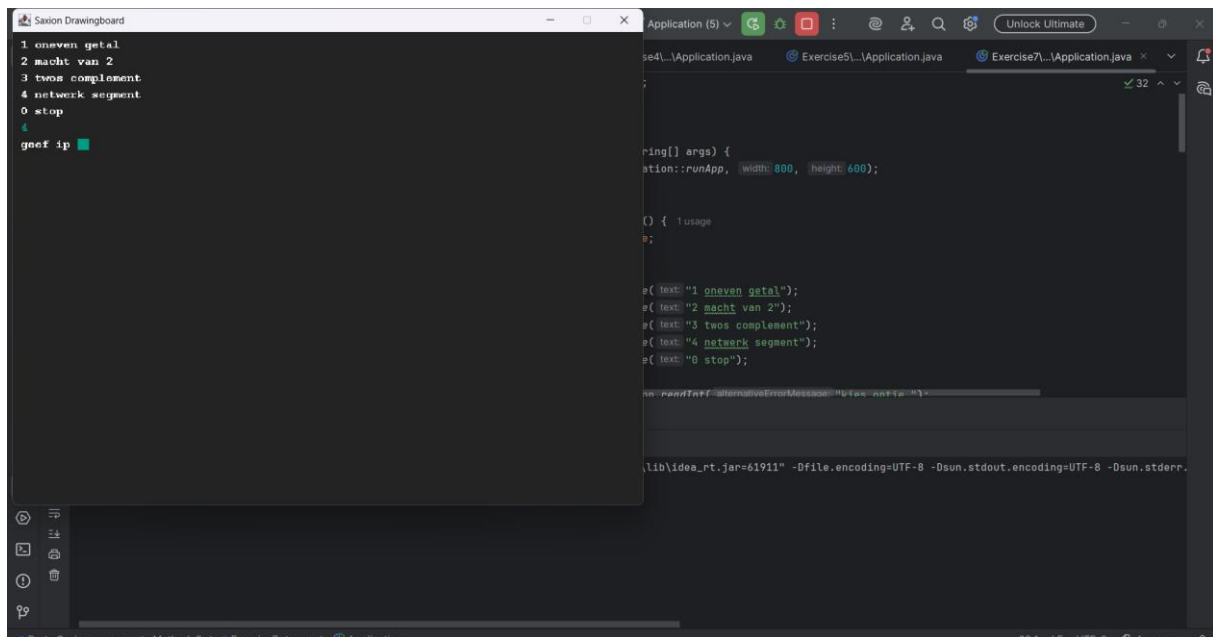
Network Addr: 11000000.10101000.00000001.01100000

This gives 192.168.1.96 in decimal as the network address.

For a /27 subnet, each segment (or subnet) has 32 IP addresses (2^5).

The range of this network segment is from 192.168.1.96 to 192.168.1.127.

Paste source code here, with a screenshot of a working application.



(ik heb veel veel moeite Gehad met deze opdracht en heb uiteindelijk de hulp van ai gebruikt die heeft mij Hiermee geholpen die maakte paar Kleine foutjes die ik wel had verbeterd maar groten deels van de code moet ik credits geven aan deepseek ai)

```
import nl.saxion.app.SaxionApp;
```

```
public class Application {
```

```
    public static void main(String[] args) {
        SaxionApp.start(Application::runApp, 800, 600);
    }
```

```
    private static void runApp() {
        boolean doorgann = true;
```

```
        while (doorgann) {
            SaxionApp.println("1 oneven getal");
            SaxionApp.println("2 macht van 2");
            SaxionApp.println("3 twos complement");
            SaxionApp.println("4 netwerk segment");
            SaxionApp.println("0 stop");
```

```
            int keus = SaxionApp.readInt("kies optie ");
```

```
            if (keus == 1) {
                int getal = SaxionApp.readInt("geef getal ");
                if ((getal & 1) == 1)
                    SaxionApp.println("oneven");
                else
                    SaxionApp.println("even");
            }
```

```

} else if (keus == 2) {
    int getal = SaxionApp.readInt("geef getal ");
    if (getal > 0 && (getal & (getal - 1)) == 0)
        SaxionApp.println("is macht van 2");
    else
        SaxionApp.println("is geen macht van 2");

} else if (keus == 3) {
    int getal = SaxionApp.readInt("geef getal ");
    SaxionApp.println("'" + (~getal) + 1);

} else if (keus == 4) {

    SaxionApp.print("geef ip ");
    String ipTex = SaxionApp.readString();

    SaxionApp.print("geef subnet ");
    String subnnetTex = SaxionApp.readString();

    int mask = 0xFFFFFE0;

    if (!(subnnetTex.equals("/27") || subnnetTex.equals("255.255.255.224"))) {
        SaxionApp.println("subnet niet herkend ik gebruik /27");
    }

    String[] parts = ipTex.split("\\.");
    int a = Integer.parseInt(parts[0]);
    int b = Integer.parseInt(parts[1]);
    int c = Integer.parseInt(parts[2]);
    int d = Integer.parseInt(parts[3]);

    int ip = (a << 24) | (b << 16) | (c << 8) | d;

    int net = ip & mask;
    int broad = net + 31;

    int n1 = (net >> 24) & 255;
    int n2 = (net >> 16) & 255;
    int n3 = (net >> 8) & 255;
    int n4 = net & 255;

    int b1 = (broad >> 24) & 255;
    int b2 = (broad >> 16) & 255;
    int b3 = (broad >> 8) & 255;
    int b4 = broad & 255;

    SaxionApp.println("netwerk adres " + n1 + "." + n2 + "." + n3 + "." + n4);

```

```

        SaxionApp.println("range " + n1 + "." + n2 + "." + n3 + "." + n4 + " tot " + b1 + "." + b2 + "."
+ b3 + "." + b4);
        SaxionApp.println("aantal ip adressen 32");

    } else if (keus == 0) {
        doorgann = false;
    } else {
        SaxionApp.println("ongeldige keuze");
    }

    SaxionApp.println("");
}
}
}

```

Ready? Save this file and export it as a pdf file with the name: [week6.pdf](#)