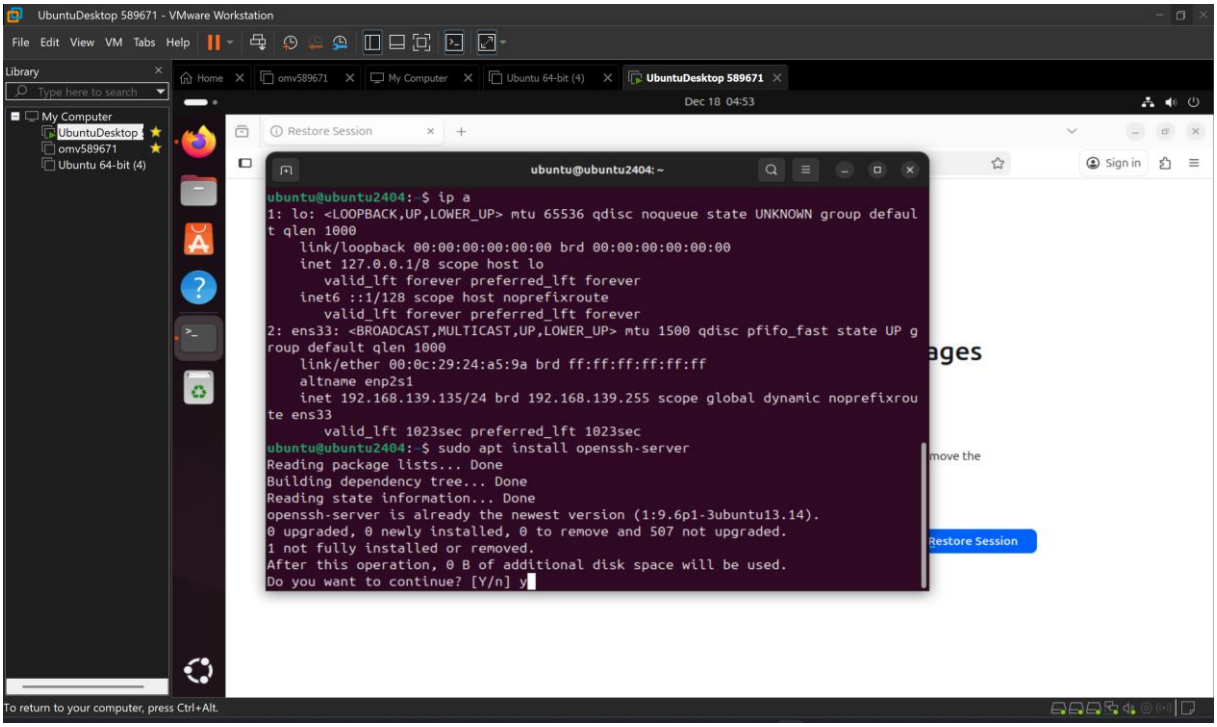
Template Week 6 – Networking

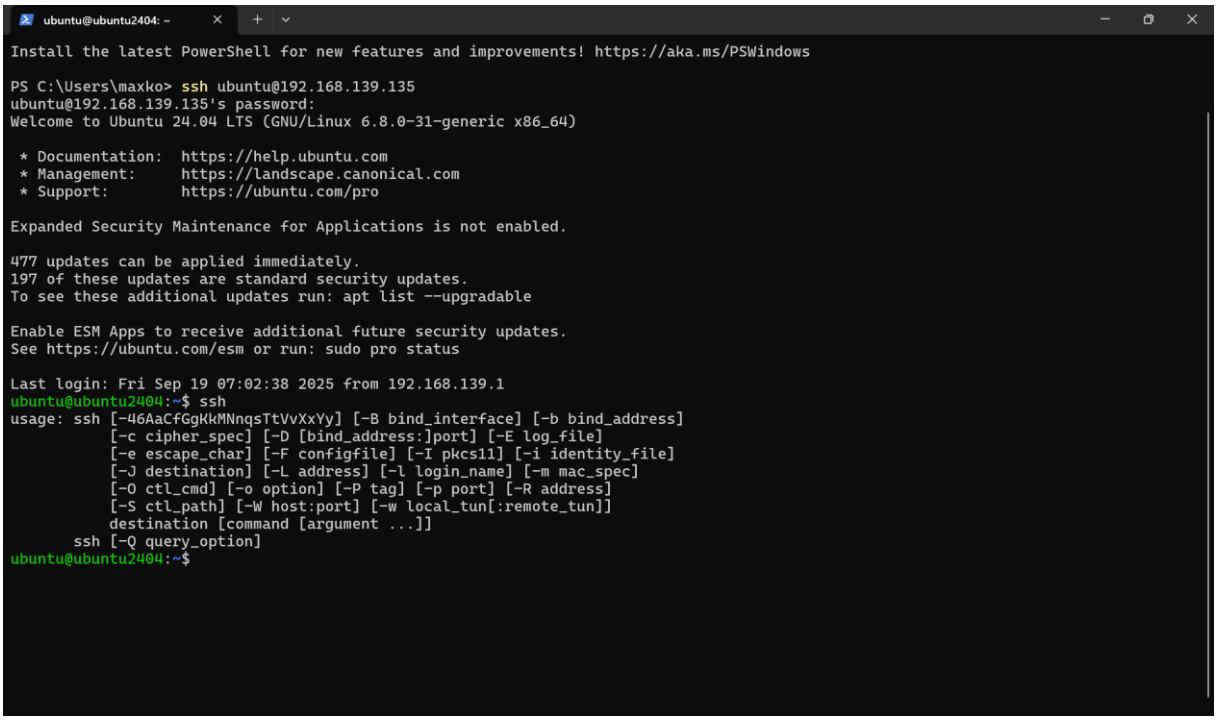
Student number: 589671

Assignment 6.1: Working from home

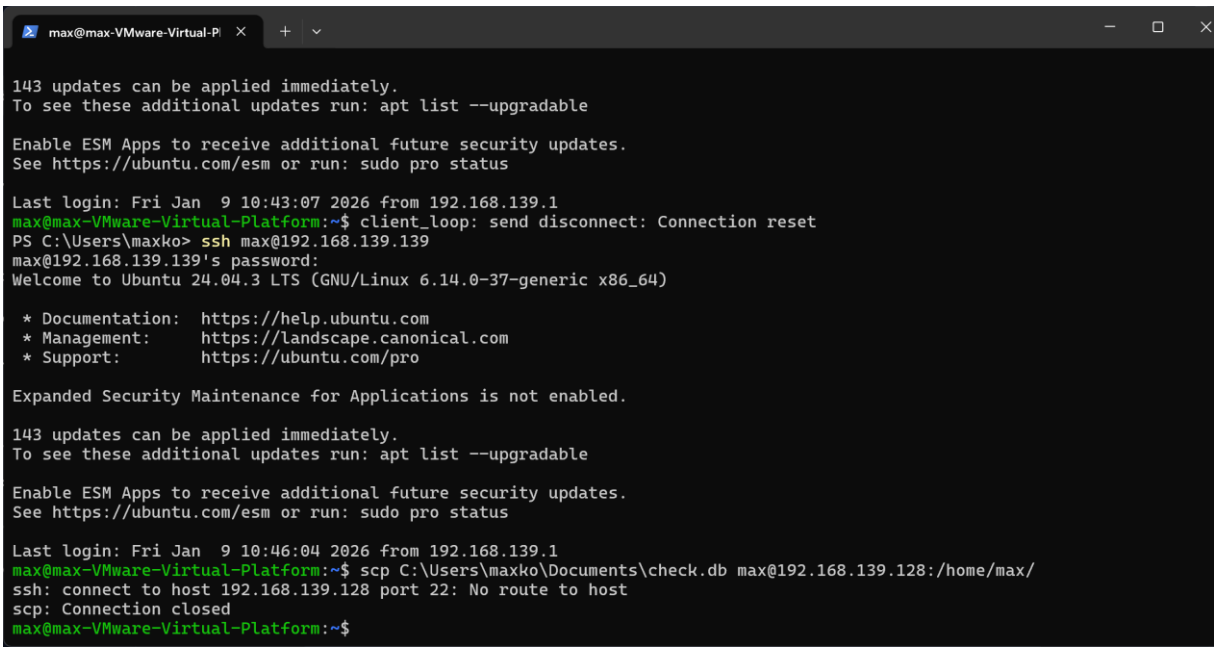
Screenshot installation openssh-server:



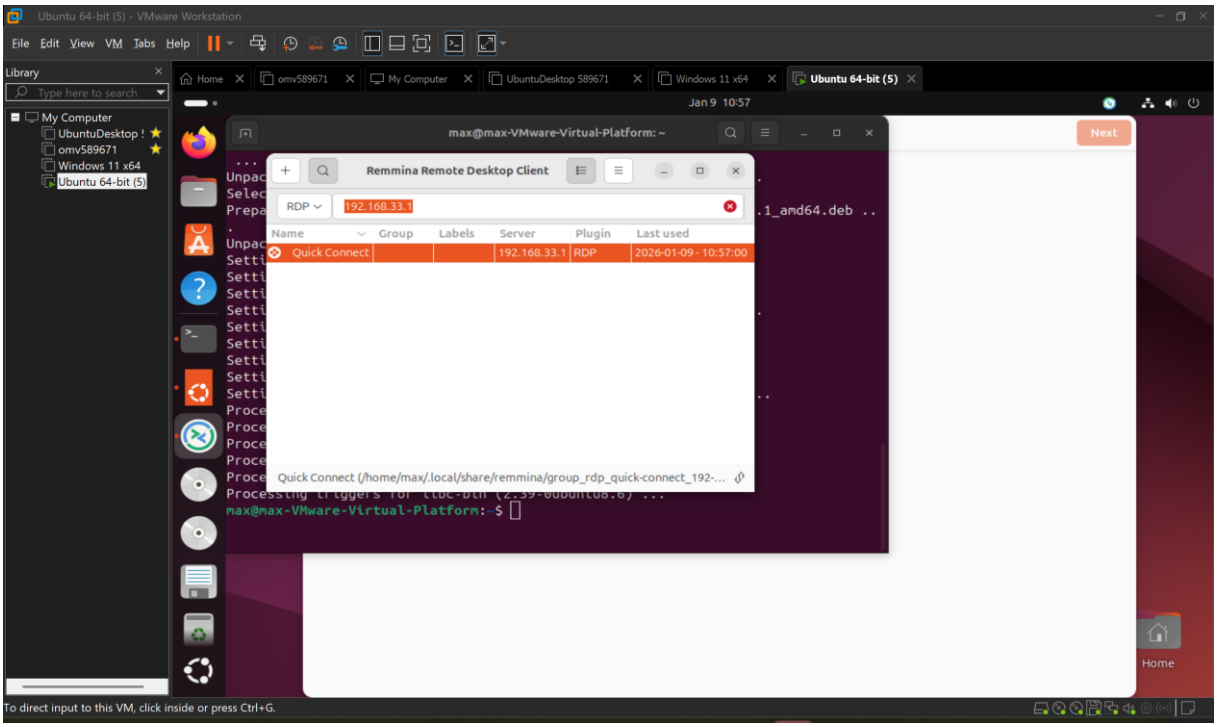
Screenshot successful SSH command execution:



Screenshot successful execution SCP command:

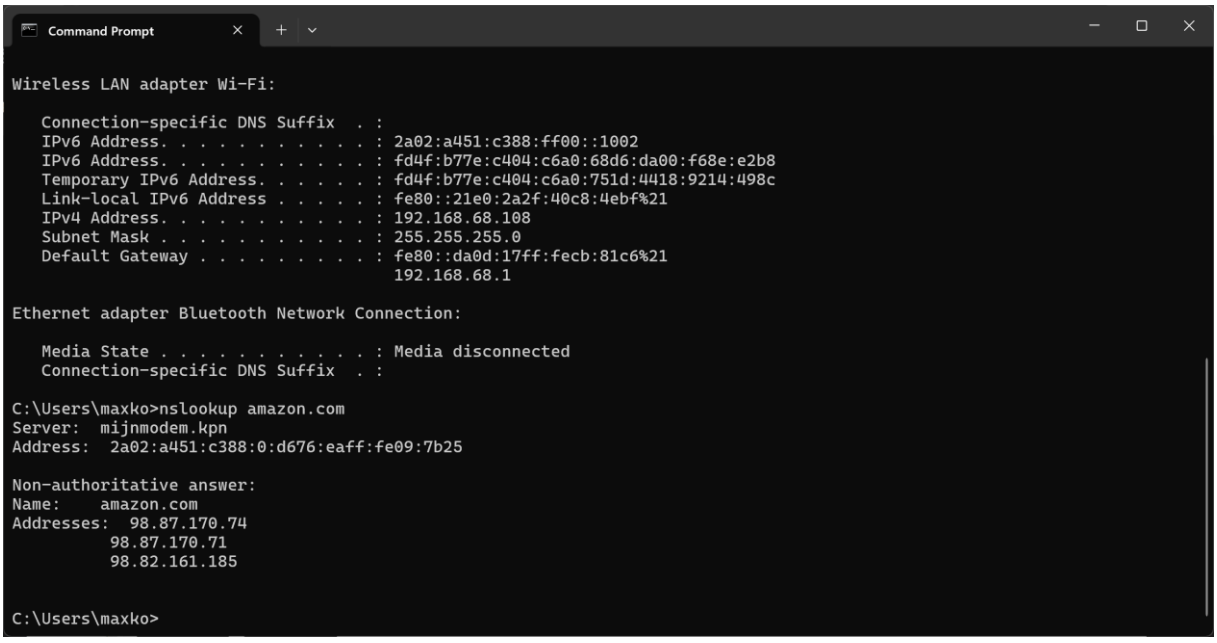


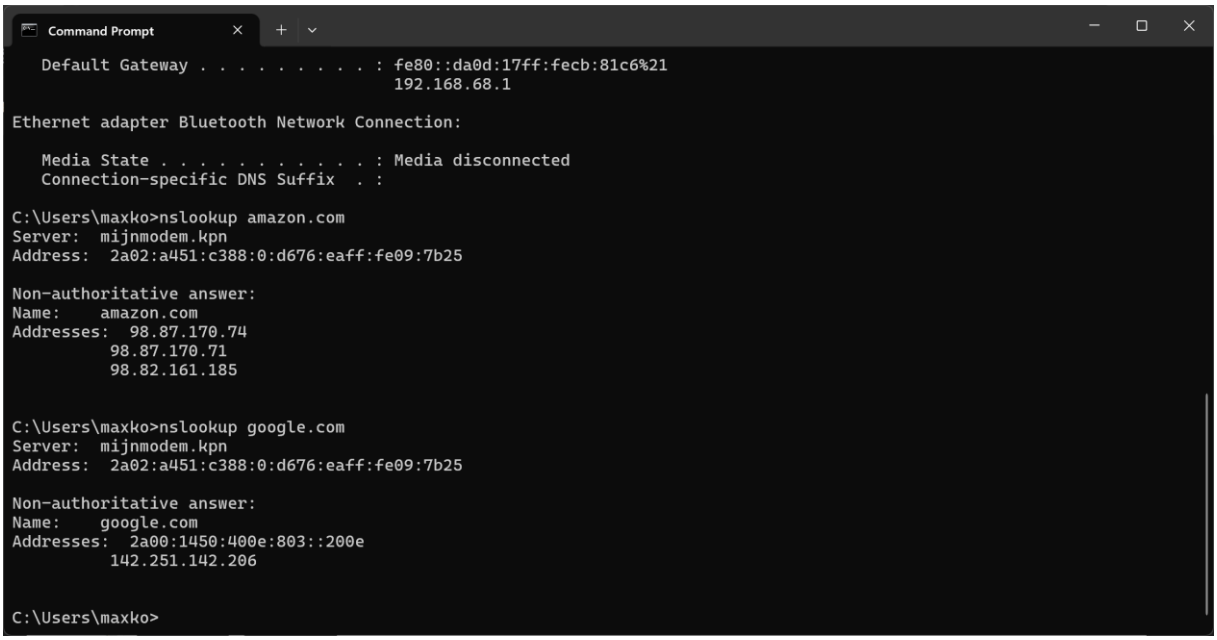
Screenshot remmina:

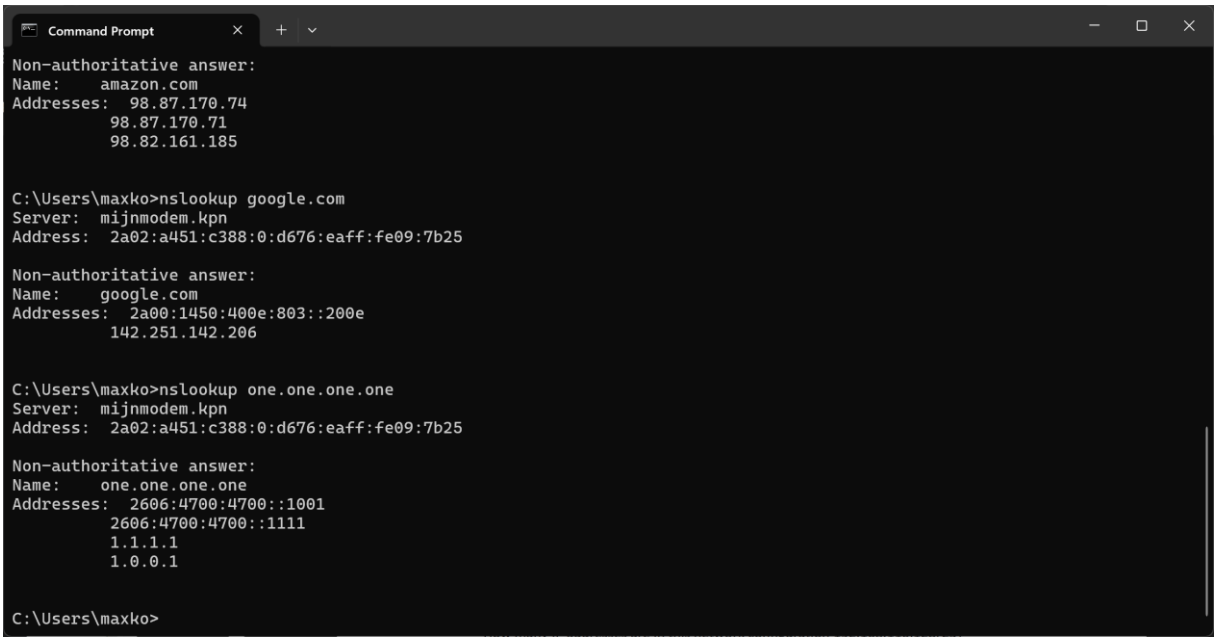


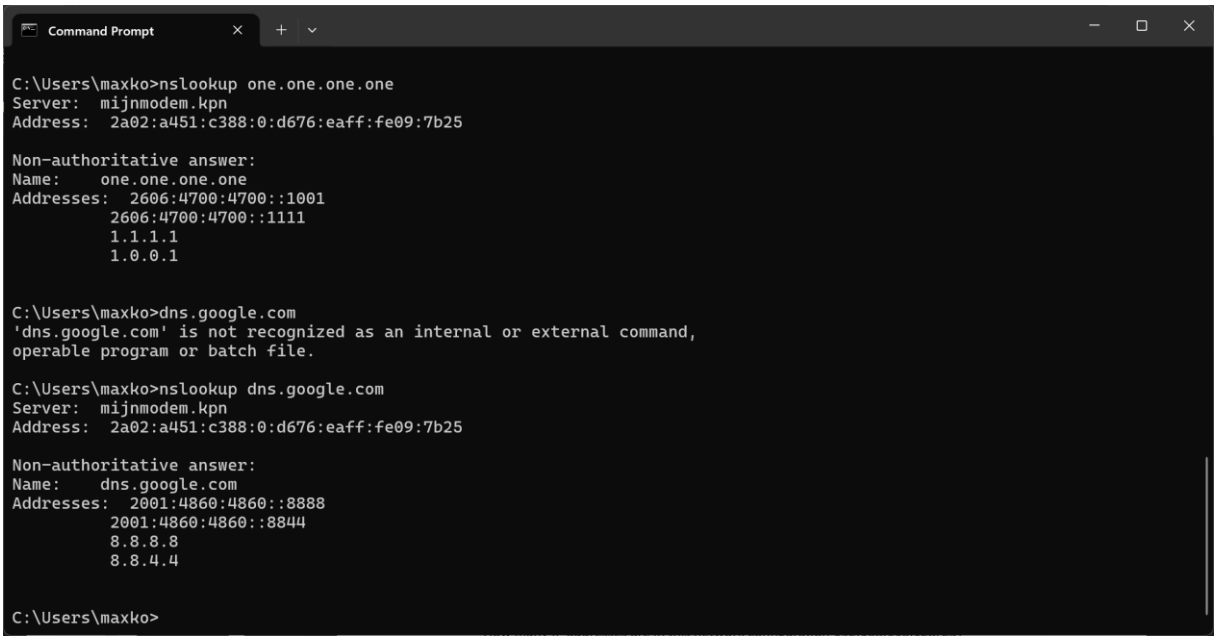
Assignment 6.2: IP addresses websites

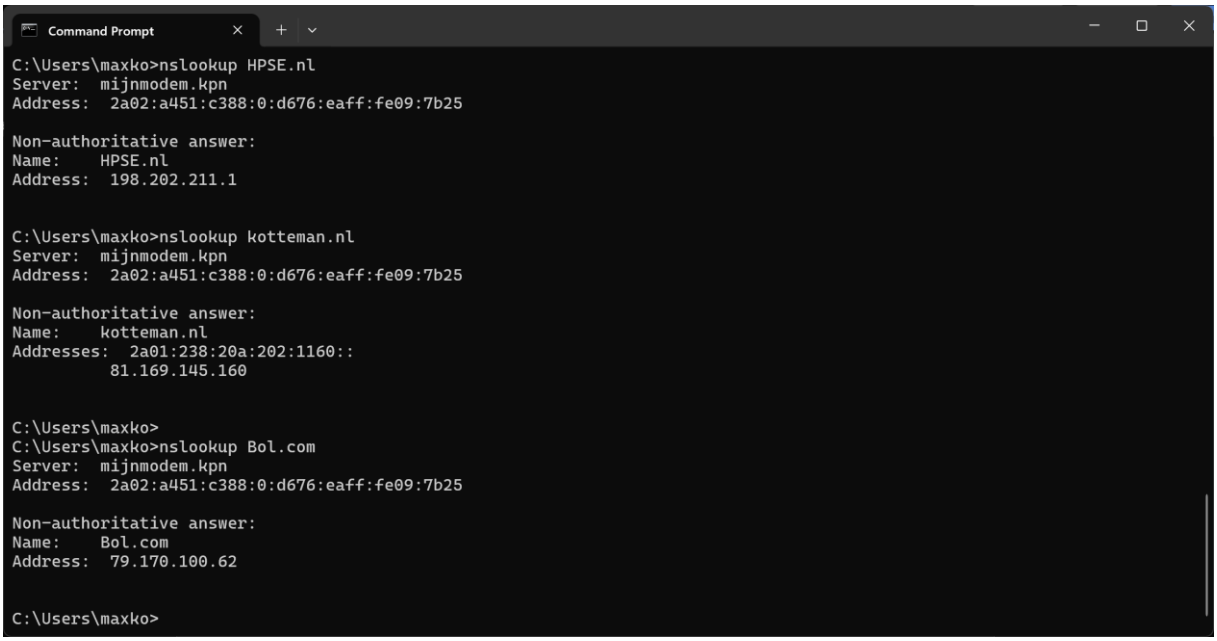
Relevant screenshots nslookup command:

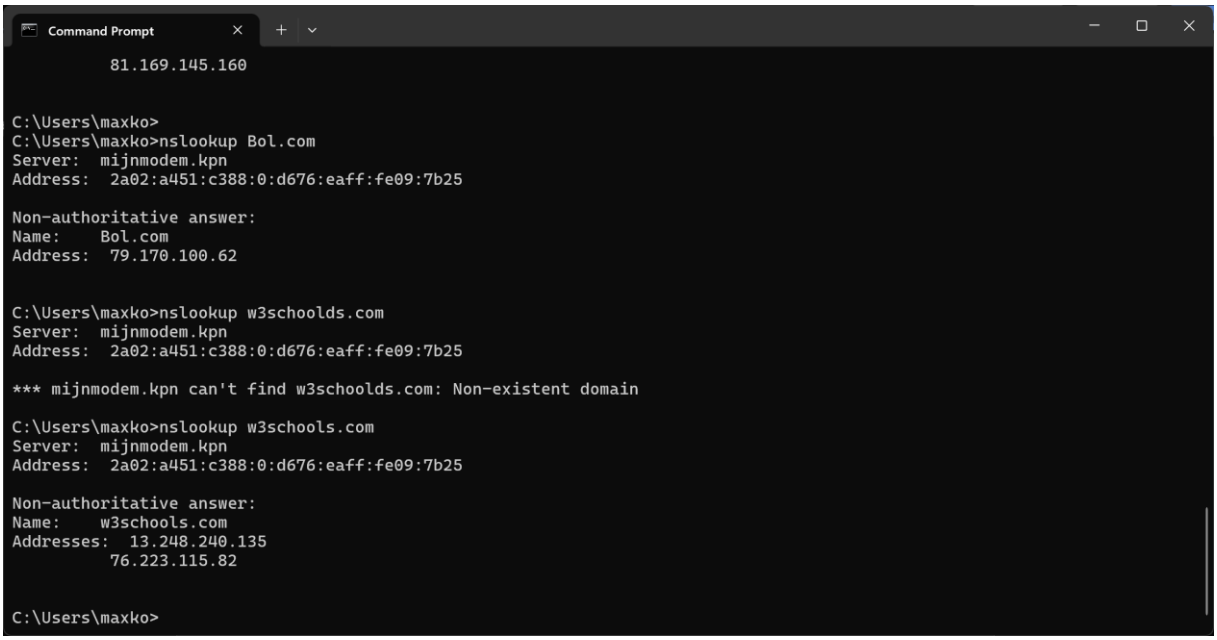



Command Prompt
Default Gateway . . . . . : fe80::da0d:17ff:fecb:81c6%21
192.168.68.1
Ethernet adapter Bluetooth Network Connection:
Media State . . . . . : Media disconnected
Connection-specific DNS Suffix . :
C:\Users\maxko>nslookup amazon.com
Server: mijnmodem.kpn
Address: 2a02:a451:c388:0:d676:eaff:fe09:7b25
Non-authoritative answer:
Name: amazon.com
Addresses: 98.87.170.74
98.87.170.71
98.82.161.185
C:\Users\maxko>nslookup google.com
Server: mijnmodem.kpn
Address: 2a02:a451:c388:0:d676:eaff:fe09:7b25
Non-authoritative answer:
Name: google.com
Addresses: 2a00:1450:400e:803::200e
142.251.142.206
C:\Users\maxko>

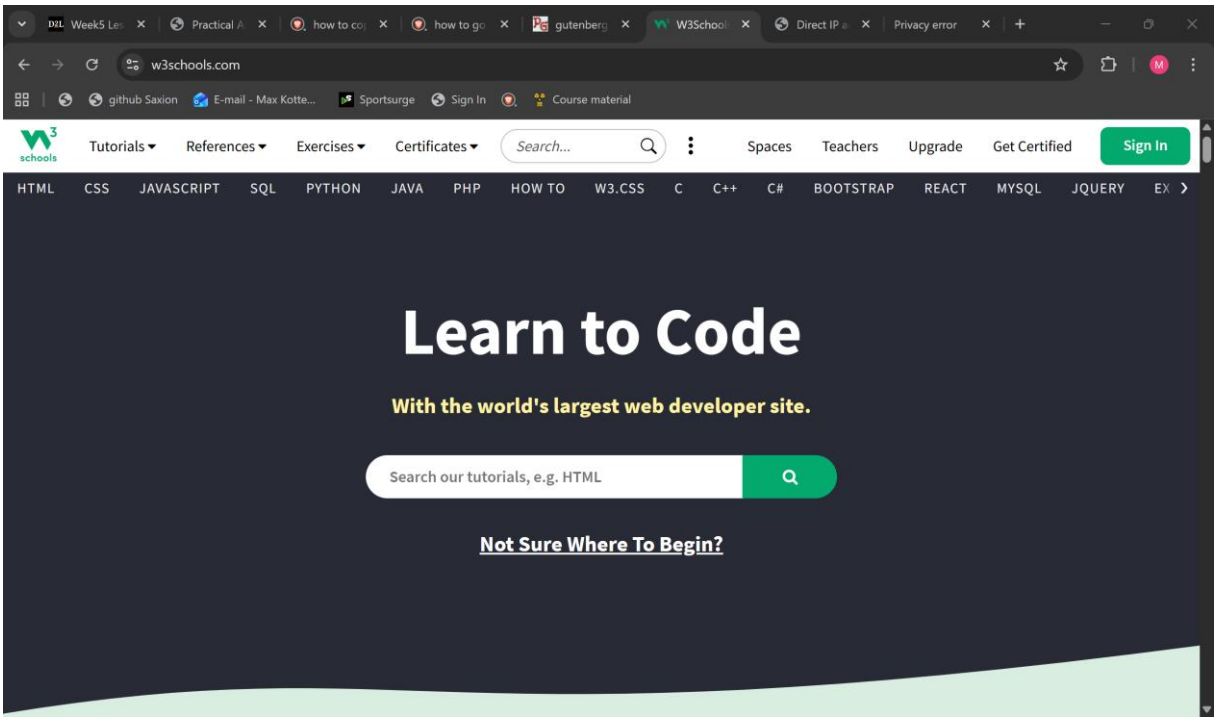

Command Prompt
Non-authoritative answer:
Name: amazon.com
Addresses: 98.87.170.74
98.87.170.71
98.82.161.185
C:\Users\maxko>nslookup google.com
Server: mijnmodem.kpn
Address: 2a02:a451:c388:0:d676:eaff:fe09:7b25
Non-authoritative answer:
Name: google.com
Addresses: 2a00:1450:400e:803::200e
142.251.142.206
C:\Users\maxko>nslookup one.one.one.one
Server: mijnmodem.kpn
Address: 2a02:a451:c388:0:d676:eaff:fe09:7b25
Non-authoritative answer:
Name: one.one.one.one
Addresses: 2606:4700:4700::1001
2606:4700:4700::1111
1.1.1.1
1.0.0.1
C:\Users\maxko>


Command Prompt
C:\Users\maxko>nslookup one.one.one.one
Server: mijnmodem.kpn
Address: 2a02:a451:c388:0:d676:eaff:fe09:7b25
Non-authoritative answer:
Name: one.one.one.one
Addresses: 2606:4700:4700::1001
          2606:4700:4700::1111
          1.1.1.1
          1.0.0.1
C:\Users\maxko>dns.google.com
'dns.google.com' is not recognized as an internal or external command,
operable program or batch file.
C:\Users\maxko>nslookup dns.google.com
Server: mijnmodem.kpn
Address: 2a02:a451:c388:0:d676:eaff:fe09:7b25
Non-authoritative answer:
Name: dns.google.com
Addresses: 2001:4860:4860::8888
          2001:4860:4860::8844
          8.8.8.8
          8.8.4.4
C:\Users\maxko>


Command Prompt
C:\Users\maxko>nslookup HPSE.nl
Server: mijnmodem.kpn
Address: 2a02:a451:c388:0:d676:eaff:fe09:7b25
Non-authoritative answer:
Name: HPSE.nl
Address: 198.202.211.1
C:\Users\maxko>nslookup kotteman.nl
Server: mijnmodem.kpn
Address: 2a02:a451:c388:0:d676:eaff:fe09:7b25
Non-authoritative answer:
Name: kotteman.nl
Addresses: 2a01:238:20a:202:1160::
          81.169.145.160
C:\Users\maxko>
C:\Users\maxko>nslookup Bol.com
Server: mijnmodem.kpn
Address: 2a02:a451:c388:0:d676:eaff:fe09:7b25
Non-authoritative answer:
Name: Bol.com
Address: 79.170.100.62
C:\Users\maxko>



Screenshot website visit via IP address:



Assignment 6.3: subnetting

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

32 – 25 = 7bits

25 bits voor network

2 tot de macht 7 = 128

Dus er zitten 128 ip adressen in dit subnet

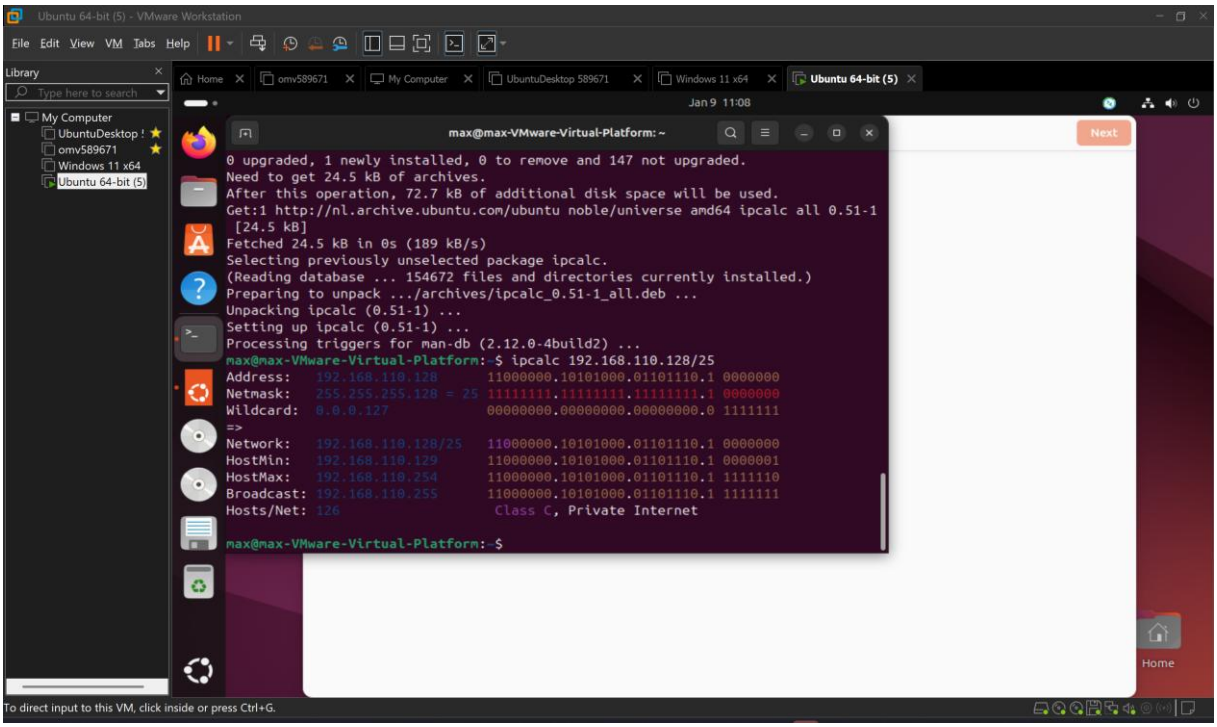
What is the usable IP range to hand out to the connected computers?

Eerste address is voor netwerk

Laatste adres broadcast adres

Dus van 192.168.110.129 tot 192.168.110.254

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

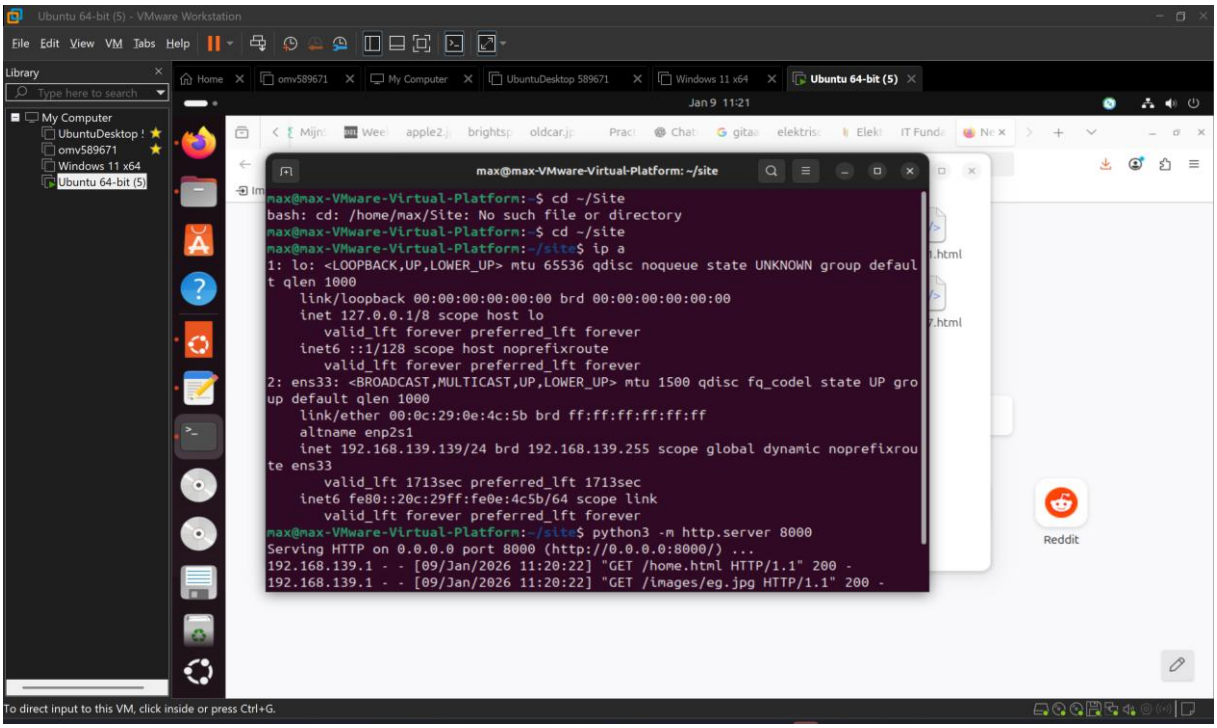


Explain the above calculation in your own words.

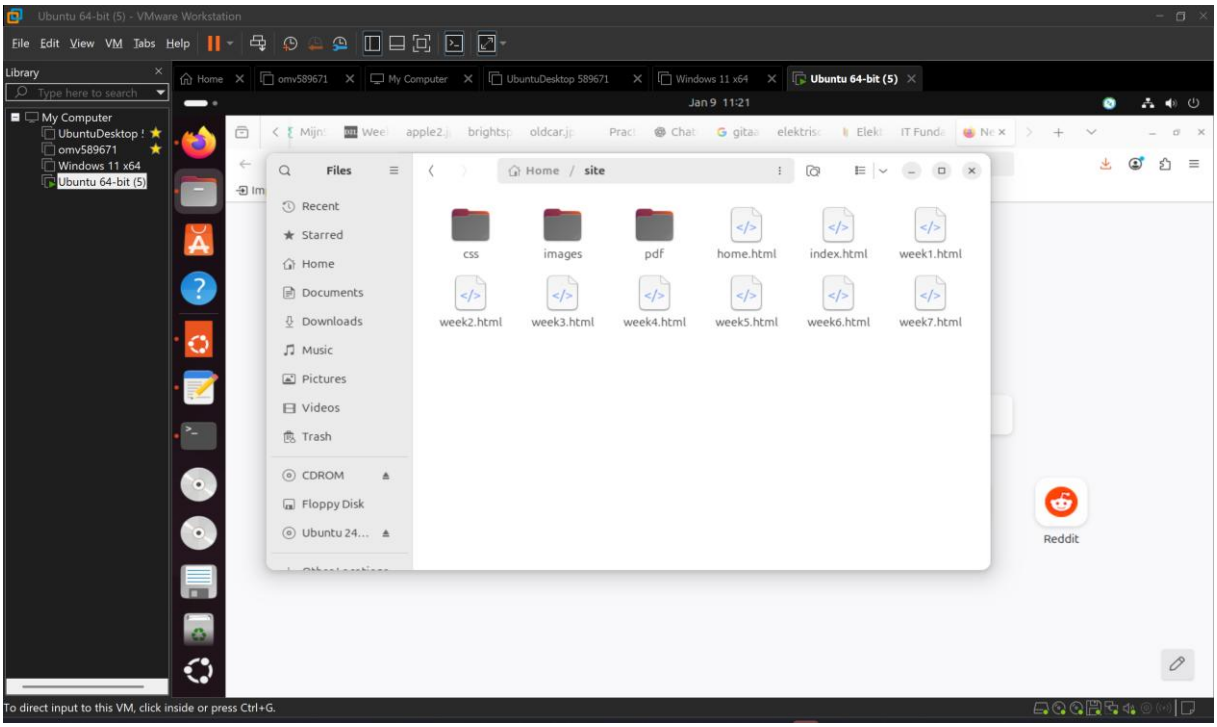
De **/25-notatie** betekent dat 25 bits voor het netwerk zijn gereserveerd en dat er 7 bits overblijven voor hosts. Met 7 hostbits zijn er 2^7=128 mogelijke adressen, waarvan het eerste het netwerkadres is en het laatste het broadcastadres. De adressen tussen deze twee vormen het bruikbare bereik voor computers in het netwerk.

Assignment 6.4: HTML

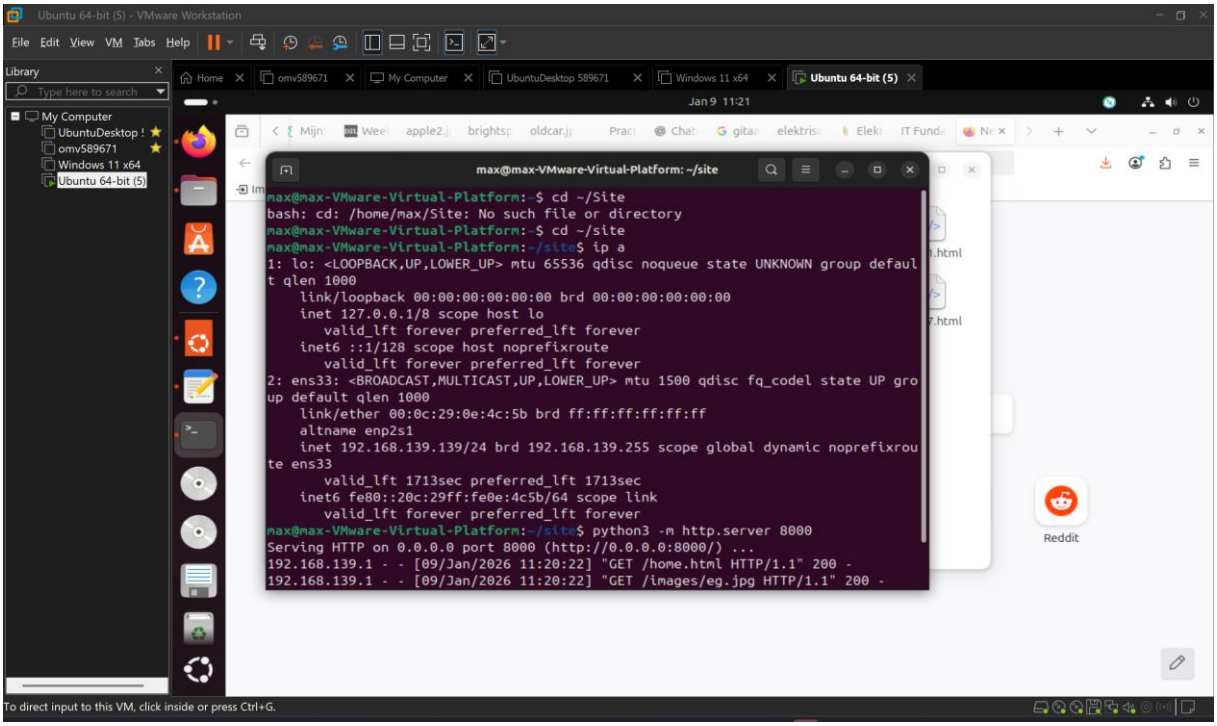
Screenshot IP address Ubuntu VM:



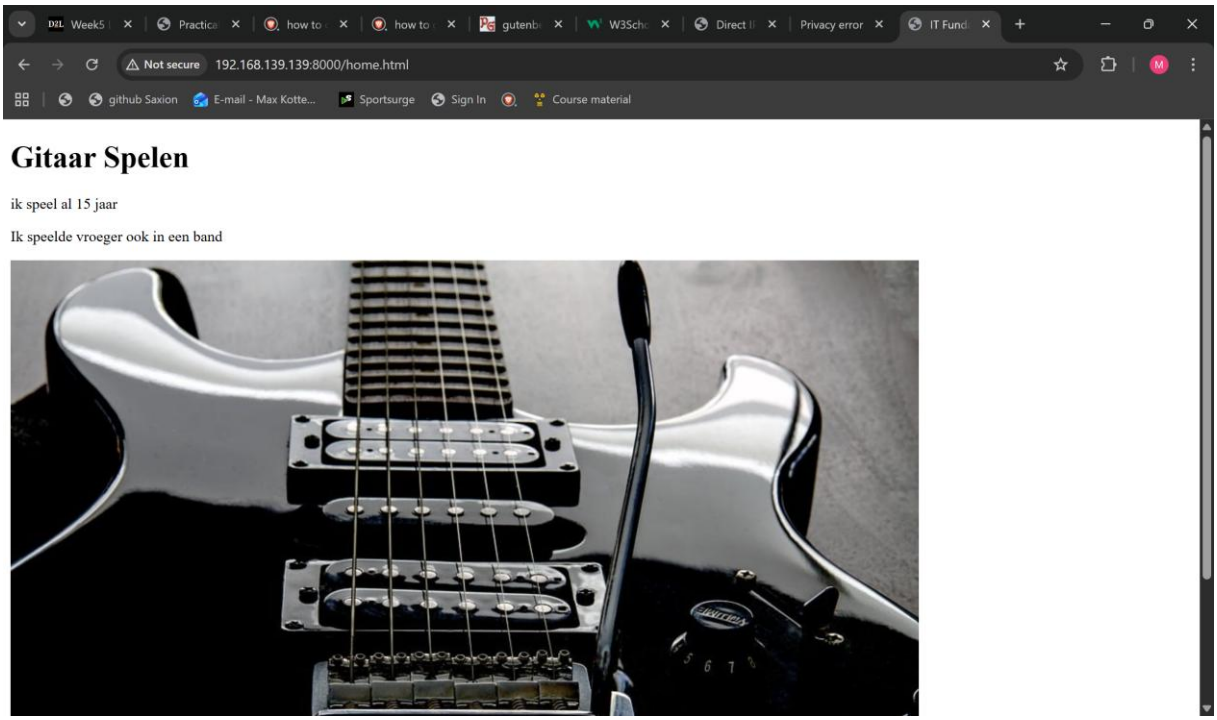
Screenshot of Site directory contents:



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.


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

import java.awt.*;
import java.util.ArrayList;
import java.util.Random;



public class Application implements Runnable {

    public static void main(String[] args) {
```

```
    SaxionApp.start(new Application());
  }

  public void run() {
    boolean blijfgaan = true;
    while (blijfgaan) {
      int gekozenSelectie = printmenu();
      String gekozenSelectieString = String.valueOf(gekozenSelectie);
      switch (gekozenSelectieString) {
        case "1":
          option1();
          break;
        case "2":
          option2();
          break;
        case "3":
          option3();
          break;
        case "4":
          option4();
          break;
      }
      SaxionApp.printLine("SELECTEER 0 OM TERUG TE GAAN");
      int teruggaan = SaxionApp.readInt();
      if (teruggaan != 0) {
        blijfgaan = false;
      }
    }
    SaxionApp.print("SYSTEEM BEEINDIGD");
  }
  public int printmenu(){
    SaxionApp.printLine("1. EVEN OF ONEVEN BEREKENEN");
    SaxionApp.printLine("2. MACHT VAN 2 UITREKENEN");
    SaxionApp.printLine("3. TWO COMPLEMENT UITREKENEN");
    SaxionApp.printLine("4. NETWERK ADRES BEREKENEN");
    int selection = SaxionApp.readInt();
    return selection;
  }
  public void option1(){
    SaxionApp.printLine("KIES EEN GETAL: ");
    int gekozennummer = SaxionApp.readInt();

    if ((gekozennummer & 1) == 0) {
      SaxionApp.printLine("HET NUMMER IS EVEN");
    } else {
      SaxionApp.printLine("HET NUMMER IS ONEVEN");
    }
```

```
}
public void option2(){
   SaxionApp.printLine("KIES EEN GETAL: ");
   int gekozennummer = SaxionApp.readInt();

   if ((gekozennummer & (gekozennummer - 1)) == 0 && gekozennummer > 0) {
      SaxionApp.printLine("DIT IS EEN MACHT VAN 2");
   } else {
      SaxionApp.printLine ("DIT IS GEEN MACHT VAN 2");
   }
}
public void option3(){
   SaxionApp.print("KIES EEN GETAL: ");
   int gekozennummer = SaxionApp.readInt();

   int negatief = ~gekozennummer + 1;

   int positief = ~negatief + 1;

   SaxionApp.printLine("Origineel: " + gekozennummer);
   SaxionApp.printLine("Negatief): " + negatief);
   SaxionApp.printLine("Positief: " + positief);
}

public void option4() {
   SaxionApp.printLine("Voer een IP adres in (bijv. 192.168.1.100): ");
   String ip = SaxionApp.readString();

   SaxionApp.printLine("Voer een subnetmasker in (bijv. 255.255.255.224): ");
   String mask = SaxionApp.readString();

   int[] ipParts = parse(ip);
   int[] maskParts = parse(mask);

   int[] network = new int[4];

   for (int i = 0; i < 4; i++) {
      network[i] = ipParts[i] & maskParts[i];
   }

   SaxionApp.printLine("Netwerk adres: " +
         network[0] + "." + network[1] + "." + network[2] + "." + network[3]);
}
private int[] parse(String s) {
   String[] parts = s.split("\\.");
   int[] nums = new int[4];
   for (int i = 0; i < 4; i++) {
      nums[i] = Integer.parseInt(parts[i]);
```

```
        }
        return nums;
    }


}
```

Ready? Save this file and export it as a pdf file with the name: **week6.pdf**