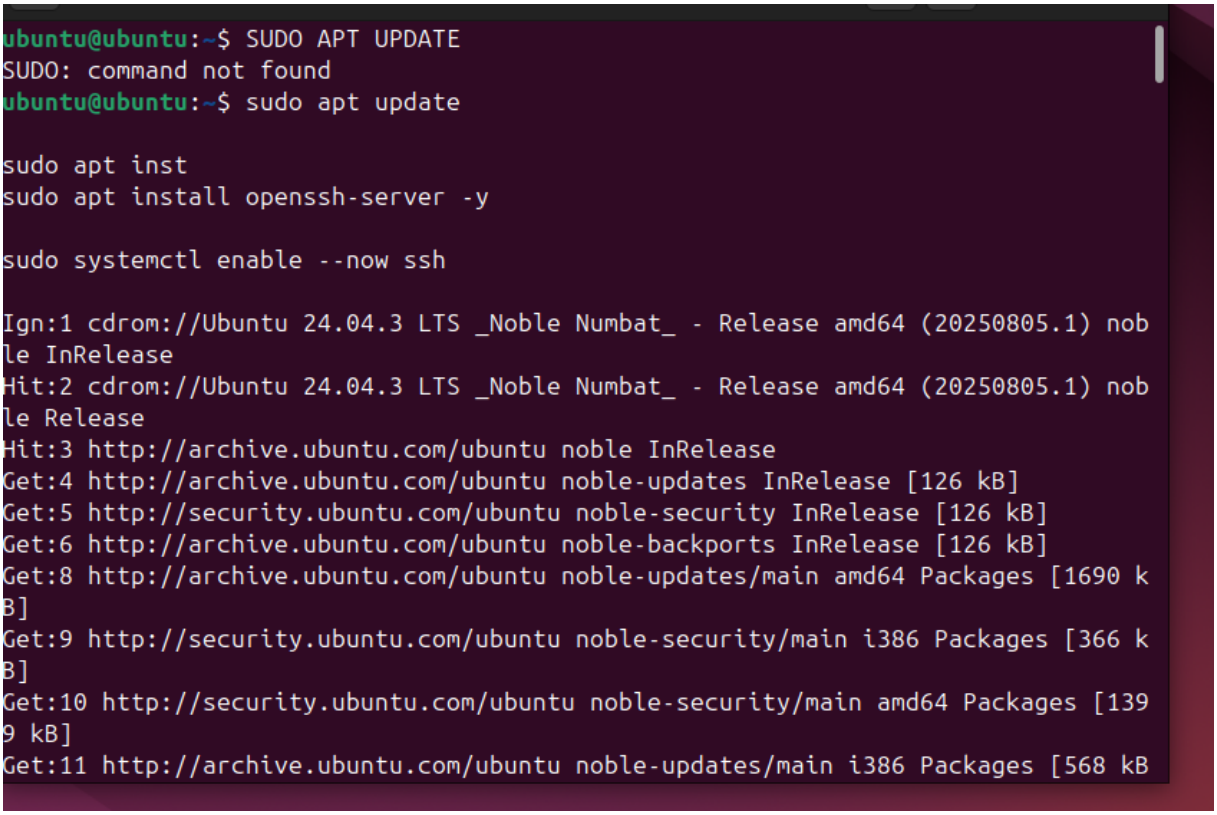
Template Week 6 – Networking

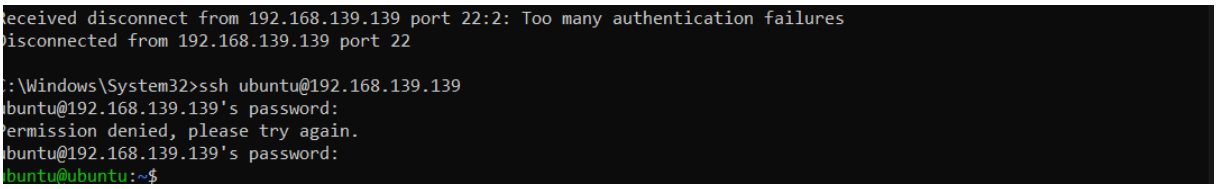
Student number:

Assignment 6.1: Working from home

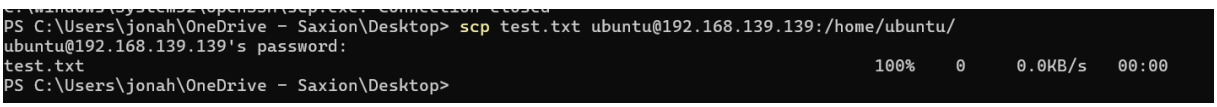
Screenshot installation openssh-server:



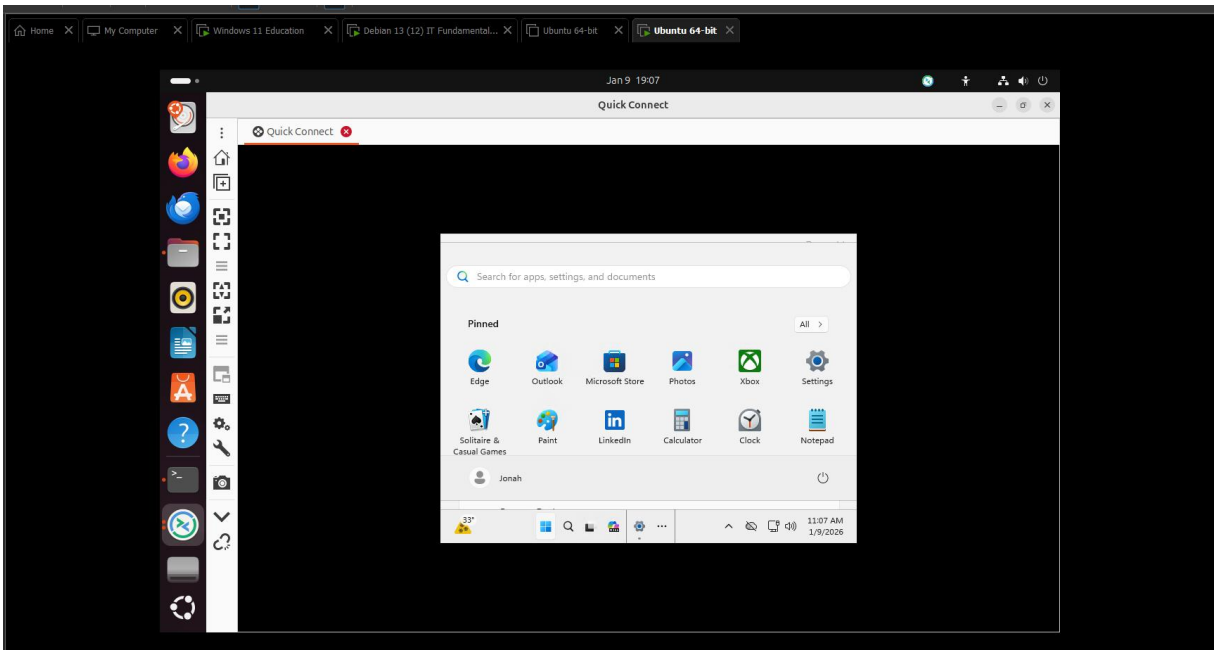
Screenshot successful SSH command execution:



Screenshot successful execution SCP command:

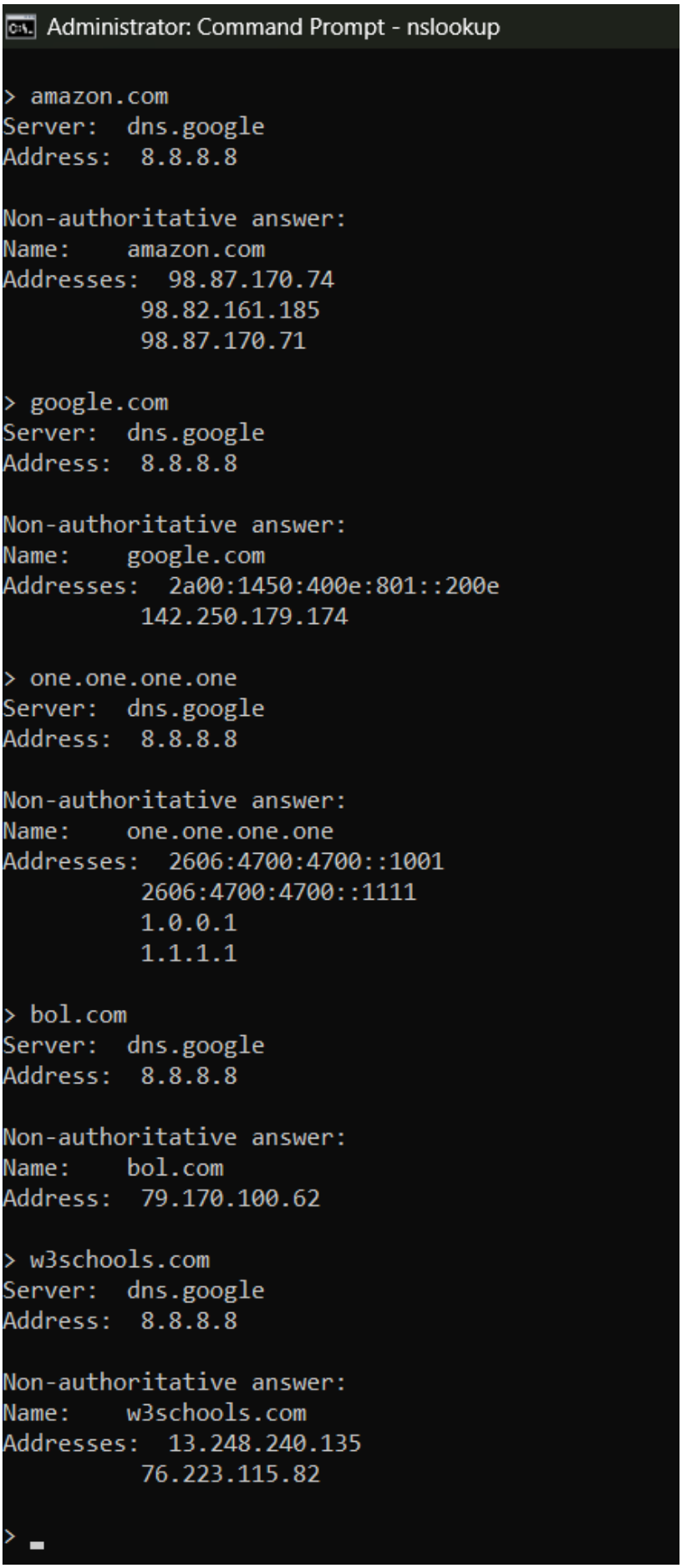


Screenshot remmina:

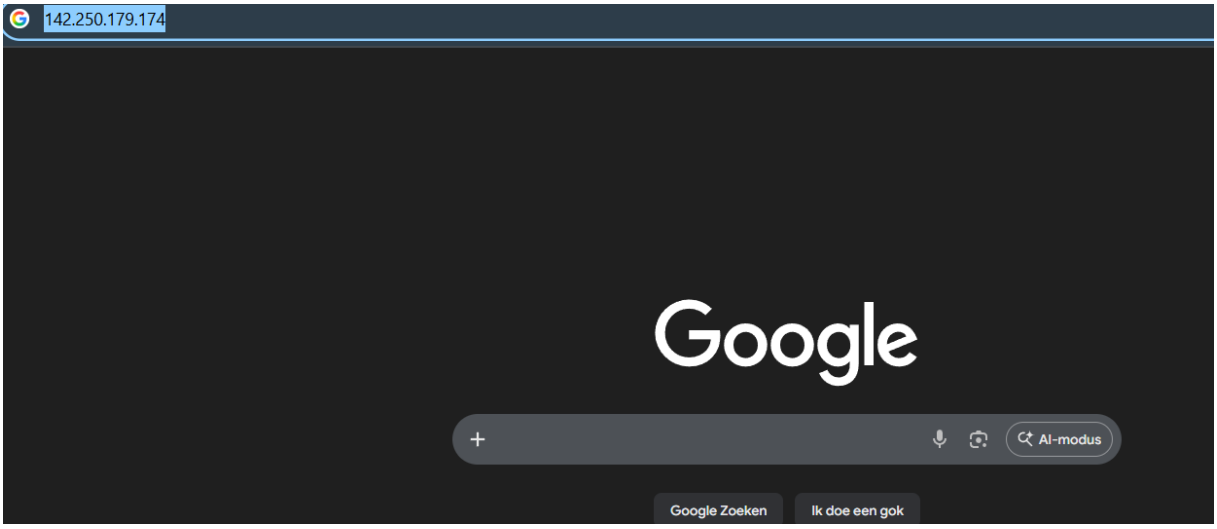


Assignment 6.2: IP addresses websites

Relevant screenshots nslookup command:


Administrator: Command Prompt - nslookup
> amazon.com
Server: dns.google
Address: 8.8.8.8
Non-authoritative answer:
Name: amazon.com
Addresses: 98.87.170.74
          98.82.161.185
          98.87.170.71
> google.com
Server: dns.google
Address: 8.8.8.8
Non-authoritative answer:
Name: google.com
Addresses: 2a00:1450:400e:801::200e
          142.250.179.174
> one.one.one.one
Server: dns.google
Address: 8.8.8.8
Non-authoritative answer:
Name: one.one.one.one
Addresses: 2606:4700:4700::1001
          2606:4700:4700::1111
          1.0.0.1
          1.1.1.1
> bol.com
Server: dns.google
Address: 8.8.8.8
Non-authoritative answer:
Name: bol.com
Address: 79.170.100.62
> w3schools.com
Server: dns.google
Address: 8.8.8.8
Non-authoritative answer:
Name: w3schools.com
Addresses: 13.248.240.135
          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?

128, ipv4 adres heeft 32 bits, met /25 prefix 32 – 25 = 7. 2 tot de macht 7 = 128

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

192.168.110.129 tot 192.168.110.254

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

```
ubuntu@ubuntu:~$ ipcalc 192.168.110.128/25
Address:   192.168.110.128      11000000.10101000.01101110.1 0000000
Netmask:   255.255.255.128 = 25 11111111.11111111.11111111.1 0000000
Wildcard:  0.0.0.127            00000000.00000000.00000000.0 1111111
=>
Network:   192.168.110.128/25   11000000.10101000.01101110.1 0000000
HostMin:   192.168.110.129      11000000.10101000.01101110.1 0000001
HostMax:   192.168.110.254      11000000.10101000.01101110.1 1111110
Broadcast: 192.168.110.255      11000000.10101000.01101110.1 1111111
Hosts/Net: 126                   Class C, Private Internet

ubuntu@ubuntu:~$
```

Explain the above calculation in your own words.

Ipv4 adres = 32 bits, met /25 subnet mask gebruik je 25 bits voor netwerk, de rest 7 voor host adressen. 2 tot de macht 7 = 128

Het netwerk adres is waar alle host bits 0 zijn, dus 192.168.110.128

Broadcast adres alle host bits 1, dus 192.168.110.255
bruikbare ip adressen is alles daar tussen in.

Ipcalc command bevestigd dit:

Network: 192.168.110.128/25

HostMin: 192.168.110.129

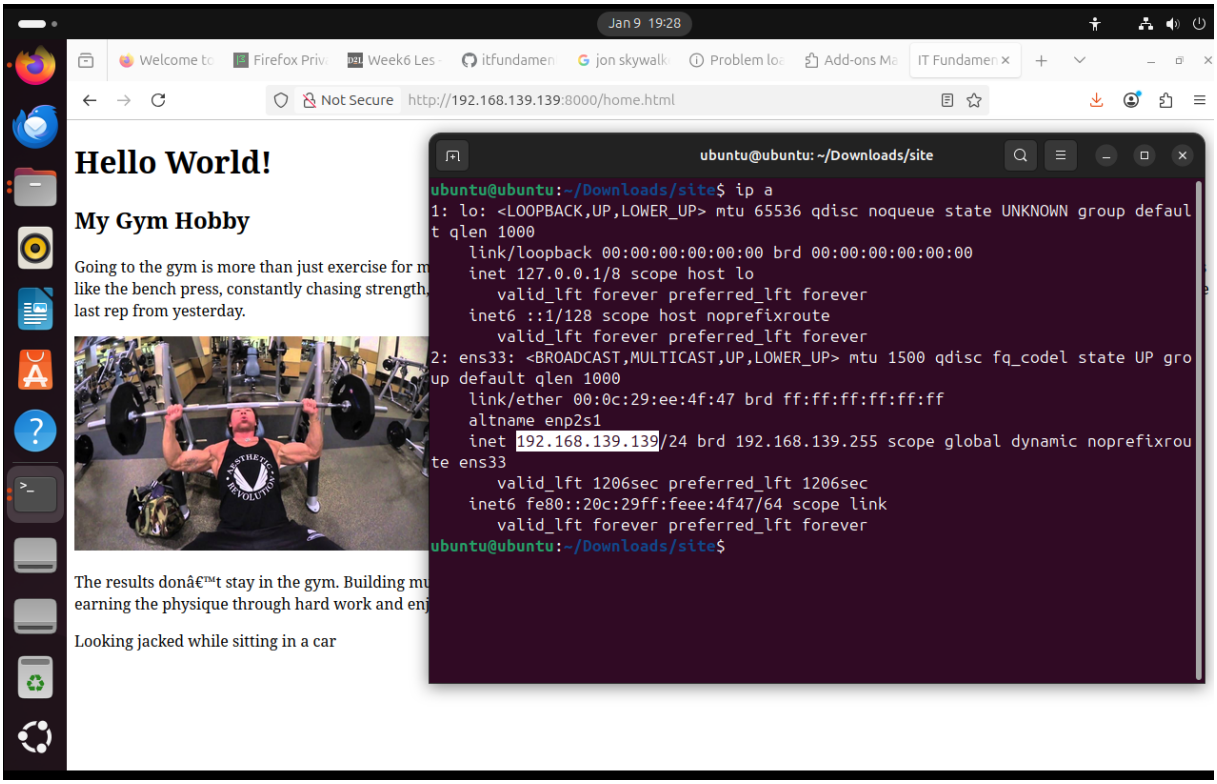
HostMax: 192.168.110.254

Broadcast: 192.168.110.255

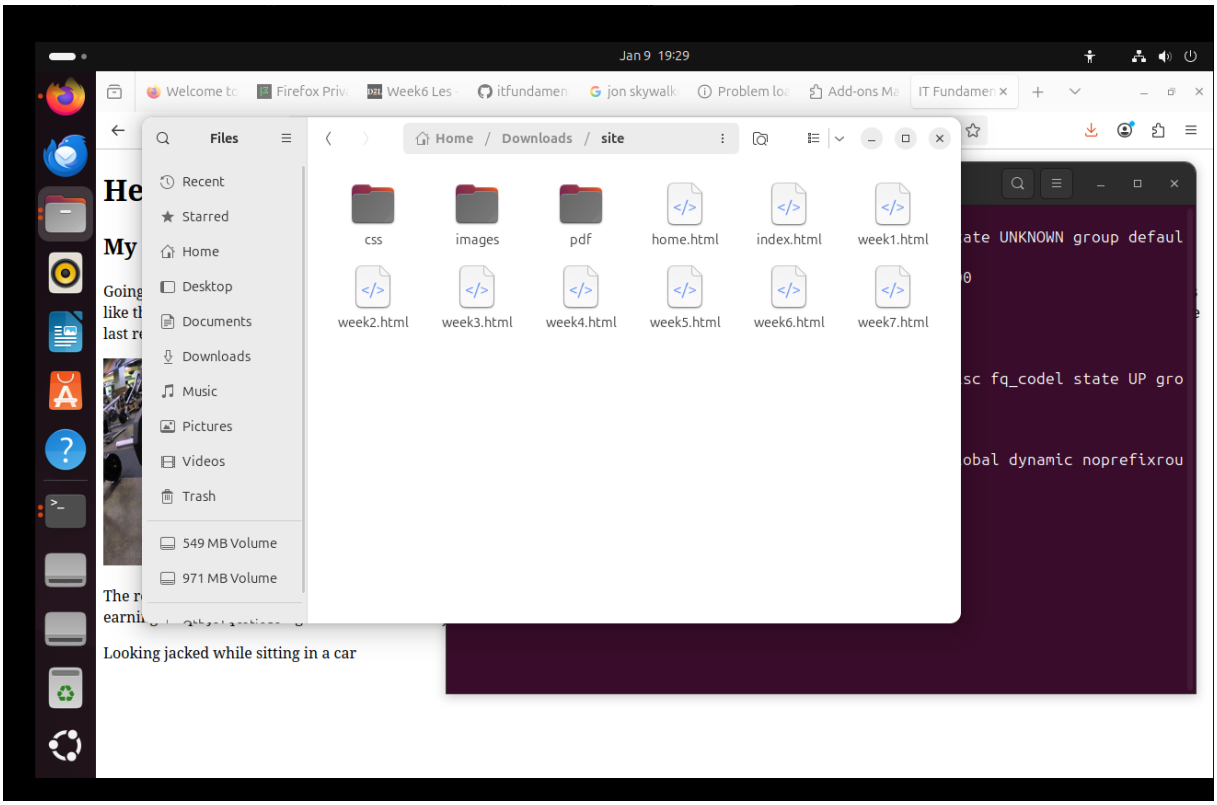
Hosts/Net: 126 bruikbare hosts

Assignment 6.4: HTML

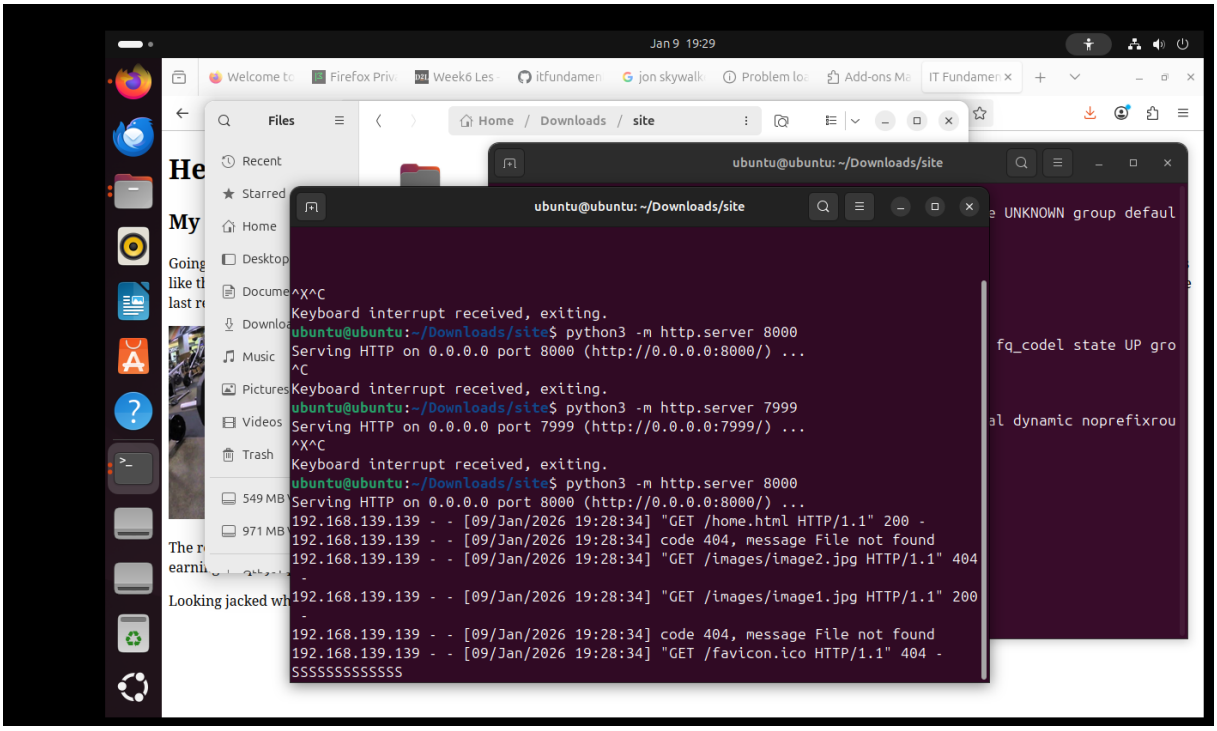
Screenshot IP address Ubuntu VM:



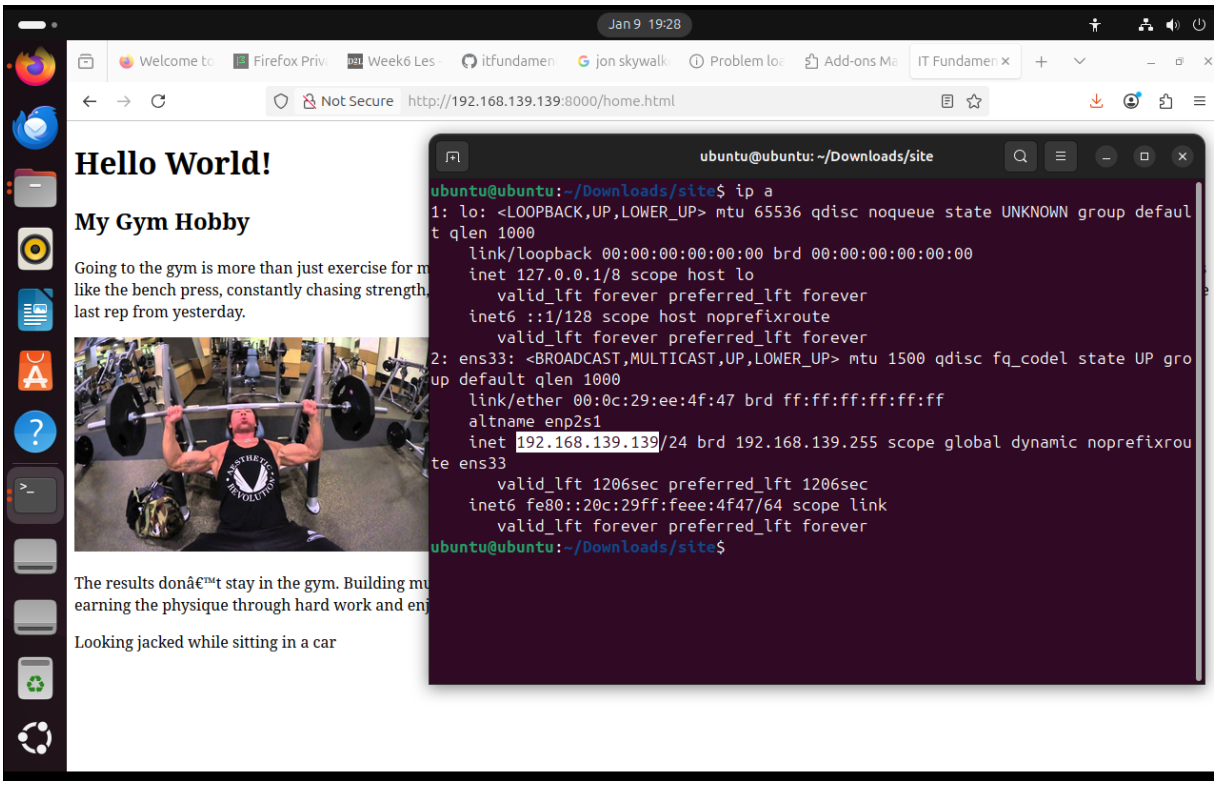
Screenshot of Site directory contents:



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 (2⁵).
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 java.util.Scanner;

public class test {
```

```
    public static boolean isOdd(int number) {
        return (number & 1) == 1;
    }

    public static boolean isPowerOfTwo(int number) {
        if (number <= 0) return false;
        return (number & (number - 1)) == 0;
    }

    public static int twosComplement(int number) {
        return ~number + 1;
    }

    private static int[] parseIPv4(String s) {
        String[] parts = s.trim().split("\\.");
        if (parts.length != 4) throw new IllegalArgumentException("Invalid IPv4 format");
        int[] out = new int[4];
        for (int i = 0; i < 4; i++) {
            out[i] = Integer.parseInt(parts[i]);
            if (out[i] < 0 || out[i] > 255) throw new IllegalArgumentException("IPv4 number out of range");
        }
        return out;
    }

    private static String toBinary8(int n) {
        String b = Integer.toBinaryString(n);
        while (b.length() < 8) b = "0" + b;
        return b;
    }

    public static void calcNetworkSegment(String ipStr, String maskStr) {
        int[] ip = parseIPv4(ipStr);
        int[] mask = parseIPv4(maskStr);

        int[] net = new int[4];
        for (int i = 0; i < 4; i++) {
            net[i] = ip[i] & mask[i];
        }

        String ipBin = toBinary8(ip[0]) + "." + toBinary8(ip[1]) + "." + toBinary8(ip[2]) + "." +
toBinary8(ip[3]);
        String maskBin = toBinary8(mask[0]) + "." + toBinary8(mask[1]) + "." + toBinary8(mask[2]) + "." +
toBinary8(mask[3]);
        String netBin = toBinary8(net[0]) + "." + toBinary8(net[1]) + "." + toBinary8(net[2]) + "." +
toBinary8(net[3]);

        String networkAddr = net[0] + "." + net[1] + "." + net[2] + "." + net[3];
```

```
        int blockSize = 256 - mask[3];
        int start = net[3];
        int end = start + (blockSize - 1);

        String rangeStart = net[0] + "." + net[1] + "." + net[2] + "." + start;
        String rangeEnd = net[0] + "." + net[1] + "." + net[2] + "." + end;

        System.out.println("IP Address:   " + ipBin);
        System.out.println("Subnet Mask:  " + maskBin);
        System.out.println("-------------------------------------------");
        System.out.println("Network Addr: " + netBin);
        System.out.println();
        System.out.println("Network address (decimal): " + networkAddr);
        System.out.println("Range: " + rangeStart + " to " + rangeEnd);
    }

    public static void main(String[] args) {
        Scanner input = new Scanner(System.in);
        boolean running = true;

        while (running) {
            System.out.println("==== Bit Calculations ====");
            System.out.println("1. Is number odd?");
            System.out.println("2. Is number a power of 2?");
            System.out.println("3. Two's complement of number");
            System.out.println("4. Calculate network segment (IP + subnet mask)");
            System.out.println("0. Exit");
            System.out.print("Choose an option: ");

            int choice = input.nextInt();
            input.nextLine(); // flush newline

            if (choice == 0) {
                running = false;
                System.out.println("Exiting program...");
            } else if (choice >= 1 && choice <= 3) {
                System.out.print("Enter an integer number: ");
                int number = input.nextInt();
                input.nextLine();

                if (choice == 1) {
                    if (isOdd(number)) System.out.println(number + " is odd");
                    else System.out.println(number + " is even");
                } else if (choice == 2) {
                    if (isPowerOfTwo(number)) System.out.println(number + " is a power of 2");
                    else System.out.println(number + " is NOT a power of 2");
                } else {
```

```
                int result = twosComplement(number);
                System.out.println("Two's complement of " + number + " = " + result);
            }
            System.out.println();
        } else if (choice == 4) {
            try {
                System.out.print("Enter IP address (e.g. 192.168.1.100): ");
                String ipStr = input.nextLine();

                System.out.print("Enter subnet mask (e.g. 255.255.255.224): ");
                String maskStr = input.nextLine();

                System.out.println();
                calcNetworkSegment(ipStr, maskStr);
                System.out.println();
            } catch (Exception e) {
                System.out.println("Error: " + e.getMessage());
                System.out.println();
            }
        } else {
            System.out.println("Unknown option, please try again.\n");
        }
    }

    input.close();
  }
}
```

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