

# Template Week 6 – Networking

Student number: 562594

## Assignment 6.1: Working from home

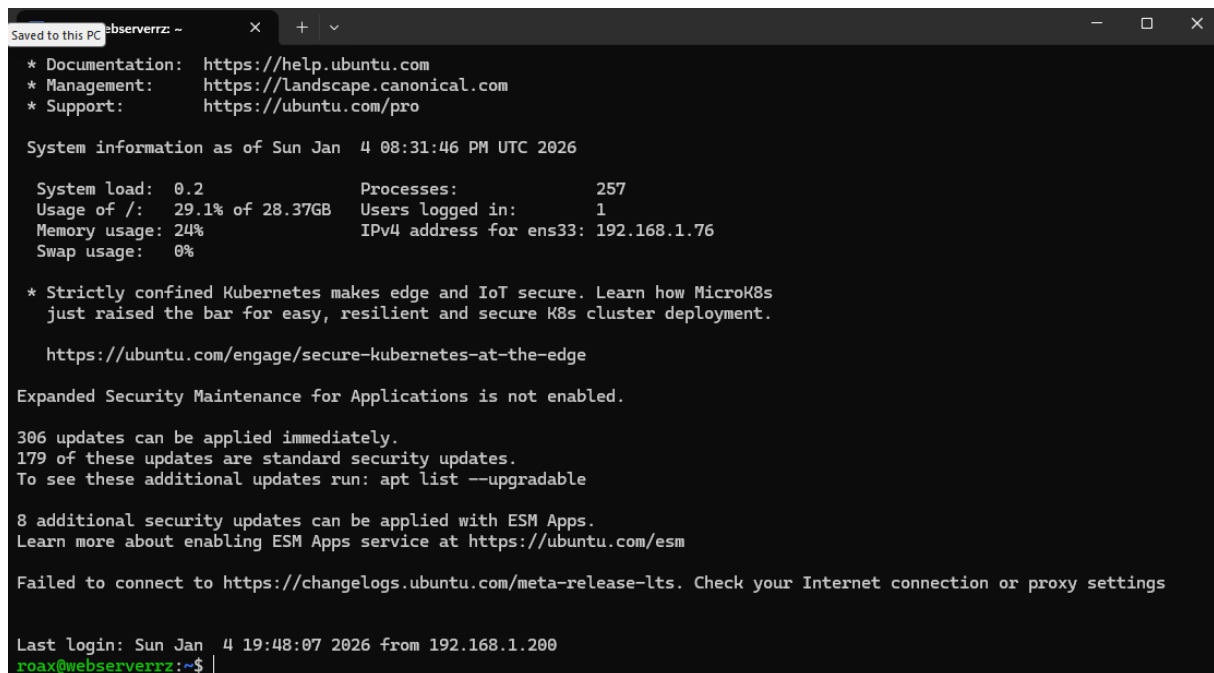
Screenshot installation openssh-server:

```
roax@webserverrz:~$ ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
        valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host noprefixroute
        valid_lft forever preferred_lft forever
2: ens33: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 00:0c:29:43:b3:44 brd ff:ff:ff:ff:ff:ff
    altname enp2s1
    inet 192.168.1.76/24 metric 100 brd 192.168.1.255 scope global dynamic ens33
        valid_lft 86219sec preferred_lft 86219sec
    inet6 fe80::20c:29ff:fe43:b344/64 scope link
        valid_lft forever preferred_lft forever
roax@webserverrz:~$ systemctl status ssh
• ssh.service - OpenBSD Secure Shell server
   Loaded: loaded (/usr/lib/systemd/system/ssh.service; disabled; preset: enabled)
   Active: active (running) since Sun 2026-01-04 19:47:52 UTC; 45min ago
     TriggeredBy: • ssh.socket
                Docs: man:sshd(8)
                   man:sshd_config(5)
   Process: 4483 ExecStartPre=/usr/sbin/sshd -t (code=exited, status=0/SUCCESS)
  Main PID: 4485 (sshd)
    Tasks: 1 (limit: 2218)
   Memory: 3.1M (peak: 4.1M)
      CPU: 671ms
   CGroup: /system.slice/ssh.service
           └─4485 "sshd: /usr/sbin/sshd -D [listener] 0 of 10-100 startups"

Jan 04 19:47:52 webserverrz systemd[1]: Starting ssh.service - OpenBSD Secure Shell server...
Jan 04 19:47:52 webserverrz sshd[4485]: Server listening on :: port 22.
Jan 04 19:47:52 webserverrz systemd[1]: Started ssh.service - OpenBSD Secure Shell server.
Jan 04 19:48:04 webserverrz sshd[4486]: Accepted password for roax from 192.168.1.200 port 44092 ssh2
Jan 04 19:48:04 webserverrz sshd[4486]: pam_unix(sshd:session): session opened for user roax(uid=1000) by roax(uid=0)
Jan 04 20:31:46 webserverrz sshd[10187]: Accepted password for roax from 192.168.1.230 port 51051 ssh2
Jan 04 20:31:46 webserverrz sshd[10187]: pam_unix(sshd:session): session opened for user roax(uid=1000) by roax(uid=0)
roax@webserverrz:~$
```

Openssh-server installatie op de ubuntu server vm.

Screenshot successful SSH command execution:



```

Saved to this PC  webserverrz ~
* Documentation: https://help.ubuntu.com
* Management:   https://landscape.canonical.com
* Support:      https://ubuntu.com/pro

System information as of Sun Jan  4 08:31:46 PM UTC 2026

System load:  0.2          Processes:           257
Usage of /:   29.1% of 28.37GB Users logged in:       1
Memory usage: 24%         IPv4 address for ens33: 192.168.1.76
Swap usage:   0%

* Strictly confined Kubernetes makes edge and IoT secure. Learn how MicroK8s
  just raised the bar for easy, resilient and secure K8s cluster deployment.

https://ubuntu.com/engage/secure-kubernetes-at-the-edge

Expanded Security Maintenance for Applications is not enabled.

306 updates can be applied immediately.
179 of these updates are standard security updates.
To see these additional updates run: apt list --upgradable

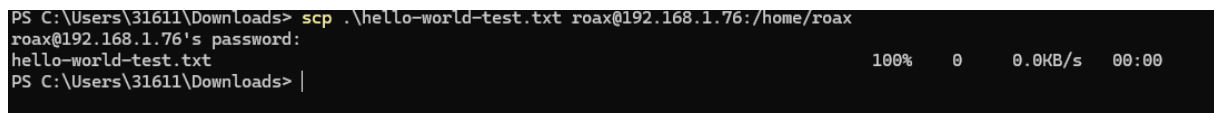
8 additional security updates can be applied with ESM Apps.
Learn more about enabling ESM Apps service at https://ubuntu.com/esm

Failed to connect to https://changelogs.ubuntu.com/meta-release-lts. Check your Internet connection or proxy settings

Last login: Sun Jan  4 19:48:07 2026 from 192.168.1.200
roax@webserverrz:~$
```

Van windows host naar ubuntu server vm.

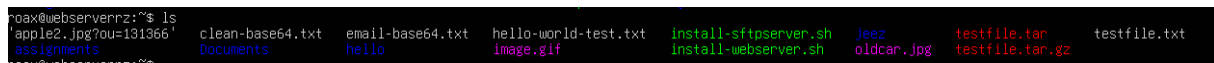
Screenshot successful execution SCP command:



```

PS C:\Users\31611\Downloads> scp .\hello-world-test.txt roax@192.168.1.76:/home/roax
roax@192.168.1.76's password:
hello-world-test.txt                                     100%   0    0.0KB/s   00:00
PS C:\Users\31611\Downloads> |
```

File gestuurd van windows naar ubuntu server

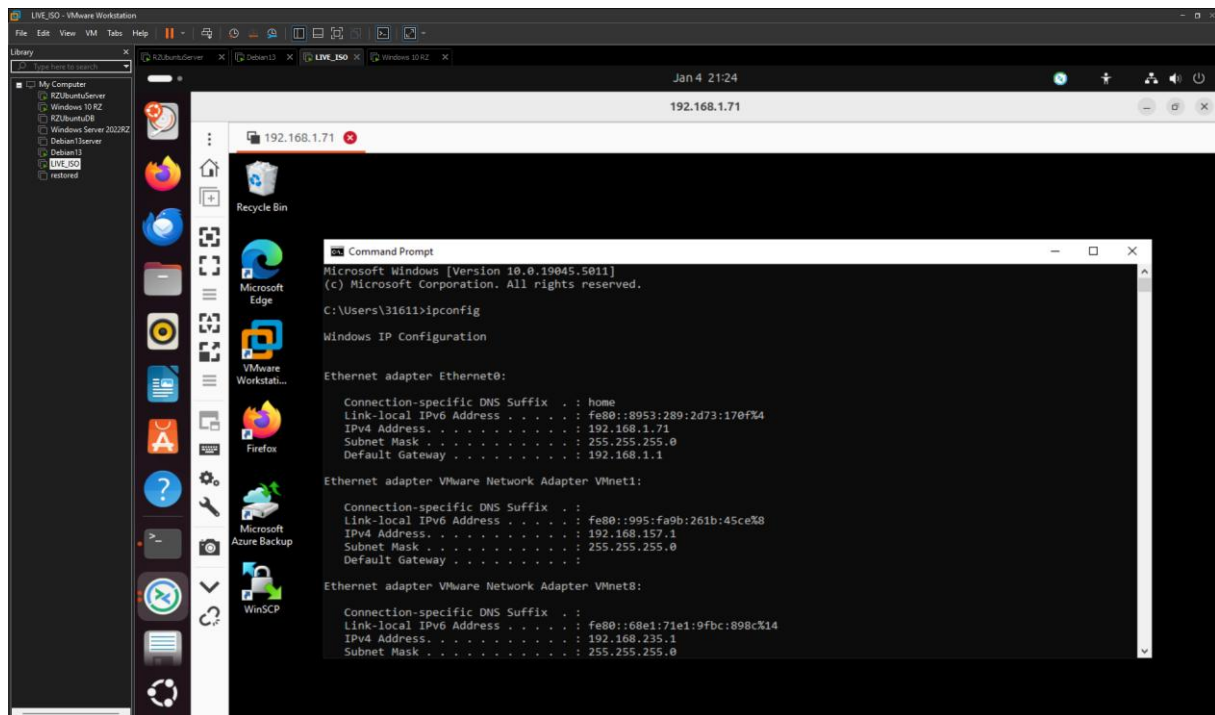


```

roax@webserverrz:~$ ls
apple2.jpg?ou=131366'  clean-base64.txt  email-base64.txt  hello-world-test.txt  install-sftpserver.sh  jeez  testfile.tar  testfile.txt
assignments            Documents         hello             image.gif            install-webserver.sh  oldcar.jpg  testfile.tar.gz
```

File in de ubuntu server vm.

Screenshot remmina:



Vnc connection van de ubuntu vm naar windows vm.

## Assignment 6.2: IP addresses websites

Relevant screenshots nslookup command:

```

roax@webserverrrz:~$ nslookup google.com
Server:      127.0.0.53
Address:     127.0.0.53#53

Non-authoritative answer:
Name:   google.com
Address: 142.250.179.206
Name:   google.com
Address: 2a00:1450:400e:803::200e

roax@webserverrrz:~$ 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
  
```

```
roax@webserverrz:~$ nslookup 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::1001
Name:   one.one.one.one
Address: 2606:4700:4700::1111
```

```
roax@webserverrz:~$ nslookup 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
```

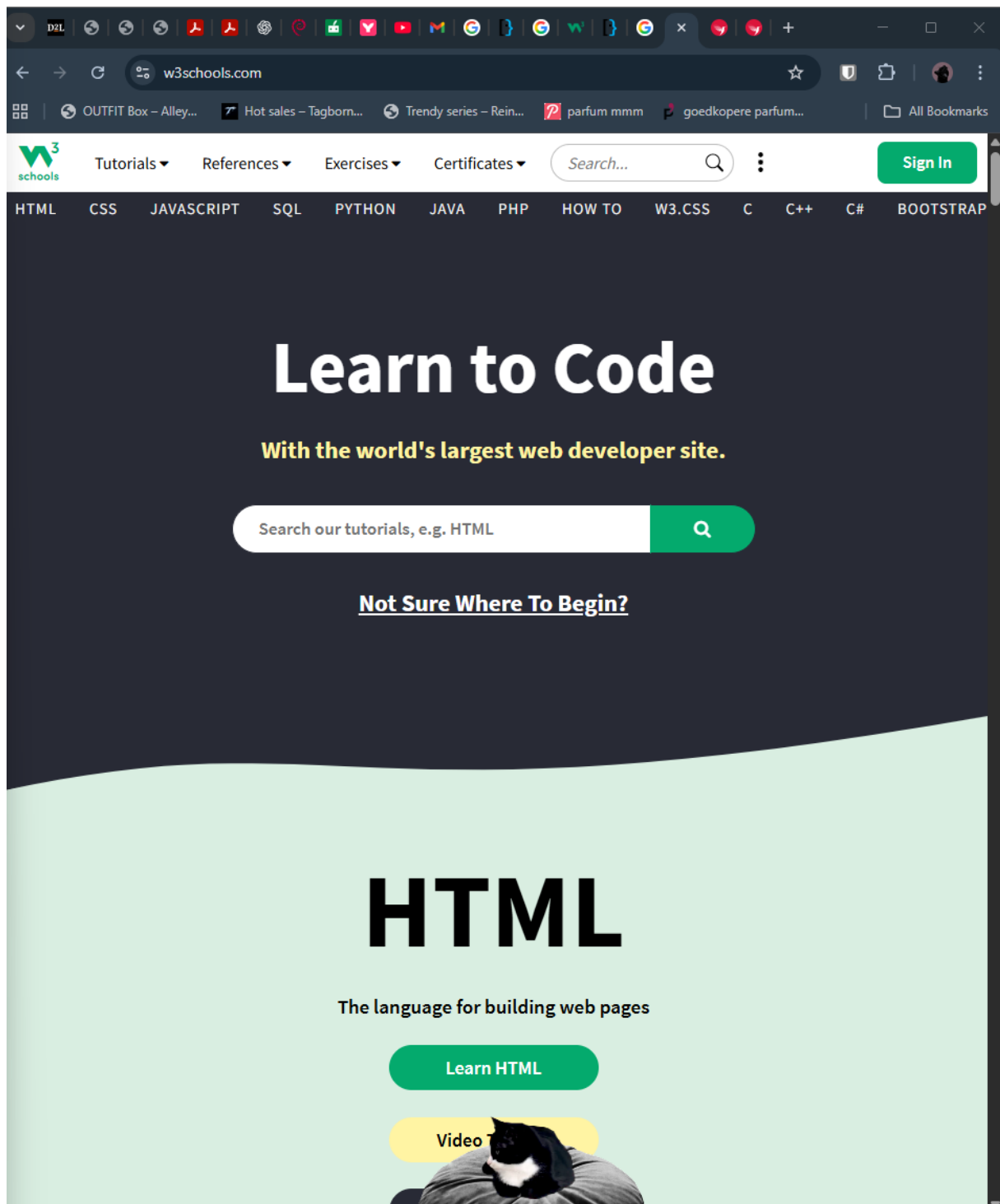
```
roax@webserverrz:~$ nslookup bol.com
Server:      127.0.0.53
Address:     127.0.0.53#53
```

```
Non-authoritative answer:
Name:   bol.com
Address: 79.170.100.42
```

```
roax@webserverrz:~$ nslookup w3schools.com
Server:      127.0.0.53
Address:     127.0.0.53#53
```

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

Screenshot website visit via IP address:



Autoredirected to the full domain.

### Assignment 6.3: subnetting

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

128 IP addresses

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

192.168.110.129 – 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.

*25 subnet means 25 bits for network and 7 bits for hosts. 7 bits for hosts gives  $2^7 = 128$  total IP addresses. The first IP is network address and the last IP is broadcast address, so they cannot be assigned to computers. The remaining IPs are usable for devices.*

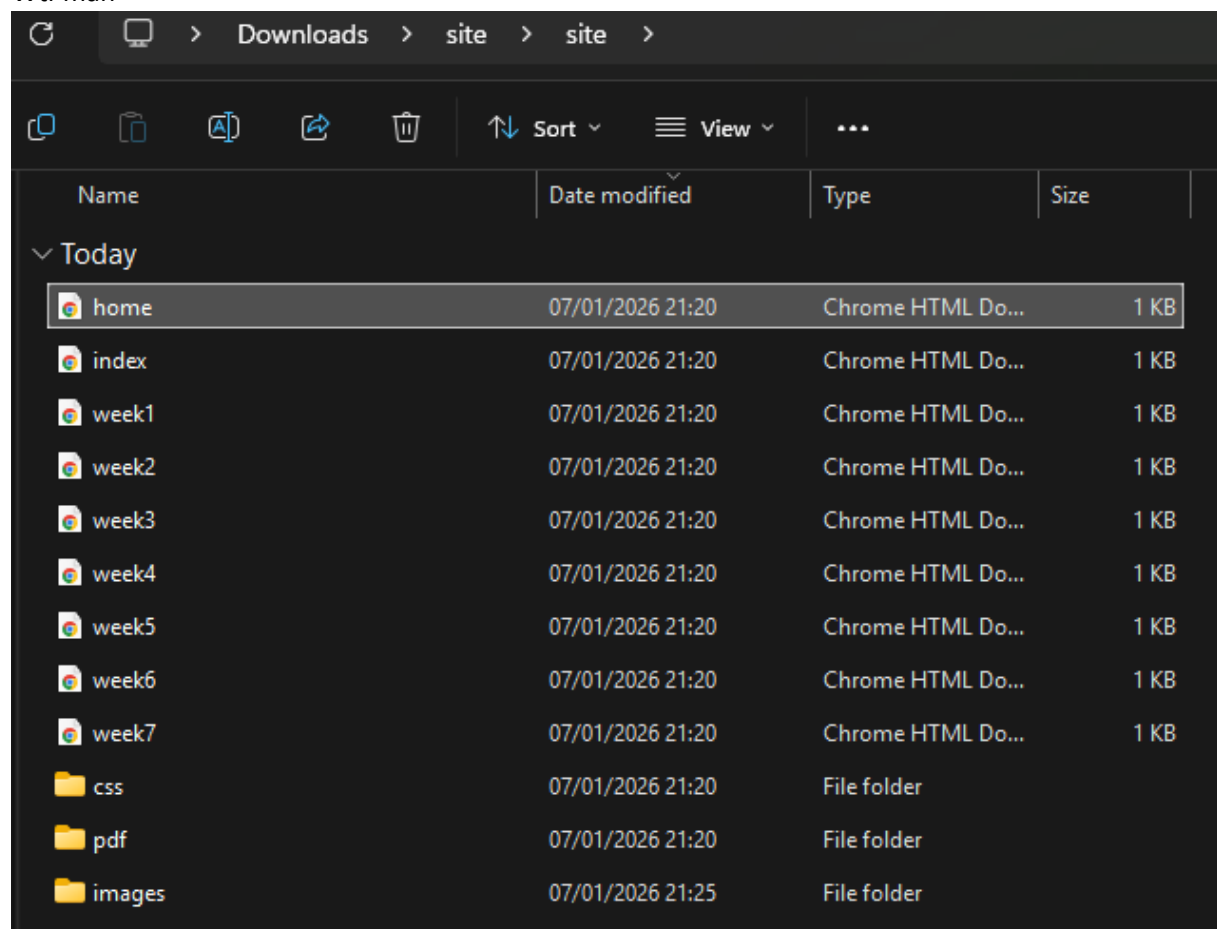
#### Assignment 6.4: HTML

Screenshot IP address Ubuntu VM:

```
inet 192.168.1.76/24 m
valid lif 85724000
```

Screenshot of Site directory contents:

Wtf man



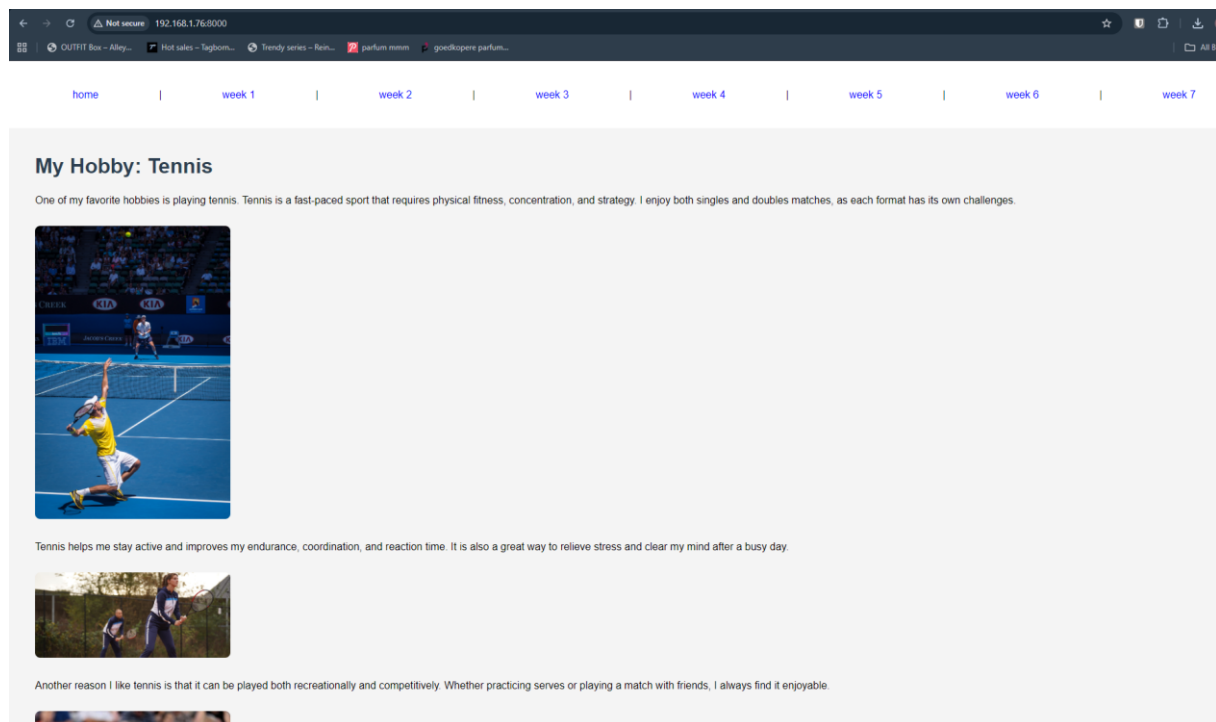
The screenshot shows a file explorer window with the address bar displaying 'Downloads > site > site'. The window contains a table of files and folders. The table has columns for Name, Date modified, Type, and Size. The files listed are 'home', 'index', 'week1', 'week2', 'week3', 'week4', 'week5', 'week6', and 'week7', all of which are Chrome HTML documents. There are also three folders: 'css', 'pdf', and 'images', all of which are file folders. The 'home' file is currently selected.

Name	Date modified	Type	Size
home	07/01/2026 21:20	Chrome HTML Do...	1 KB
index	07/01/2026 21:20	Chrome HTML Do...	1 KB
week1	07/01/2026 21:20	Chrome HTML Do...	1 KB
week2	07/01/2026 21:20	Chrome HTML Do...	1 KB
week3	07/01/2026 21:20	Chrome HTML Do...	1 KB
week4	07/01/2026 21:20	Chrome HTML Do...	1 KB
week5	07/01/2026 21:20	Chrome HTML Do...	1 KB
week6	07/01/2026 21:20	Chrome HTML Do...	1 KB
week7	07/01/2026 21:20	Chrome HTML Do...	1 KB
css	07/01/2026 21:20	File folder	
pdf	07/01/2026 21:20	File folder	
images	07/01/2026 21:25	File folder	

Screenshot python3 webserver command:

```
roax@webserverrz:~/site$ python3 -m http.server 8000
Serving HTTP on 0.0.0.0 port 8000 (http://0.0.0.0:8000/) ...
192.168.1.230 - - [07/Jan/2026 20:39:36] "GET / HTTP/1.1" 304 -
192.168.1.230 - - [07/Jan/2026 20:39:36] "GET /css/mypdfstyle.css HTTP/1.1" 304 -
192.168.1.230 - - [07/Jan/2026 20:39:36] "GET /home.html HTTP/1.1" 200 -
192.168.1.230 - - [07/Jan/2026 20:39:36] "GET /images/tennis1.jpg HTTP/1.1" 200 -
192.168.1.230 - - [07/Jan/2026 20:39:36] "GET /images/tennis2.jpg HTTP/1.1" 200 -
192.168.1.230 - - [07/Jan/2026 20:39:36] "GET /images/tennis3.jpg HTTP/1.1" 200 -
```

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

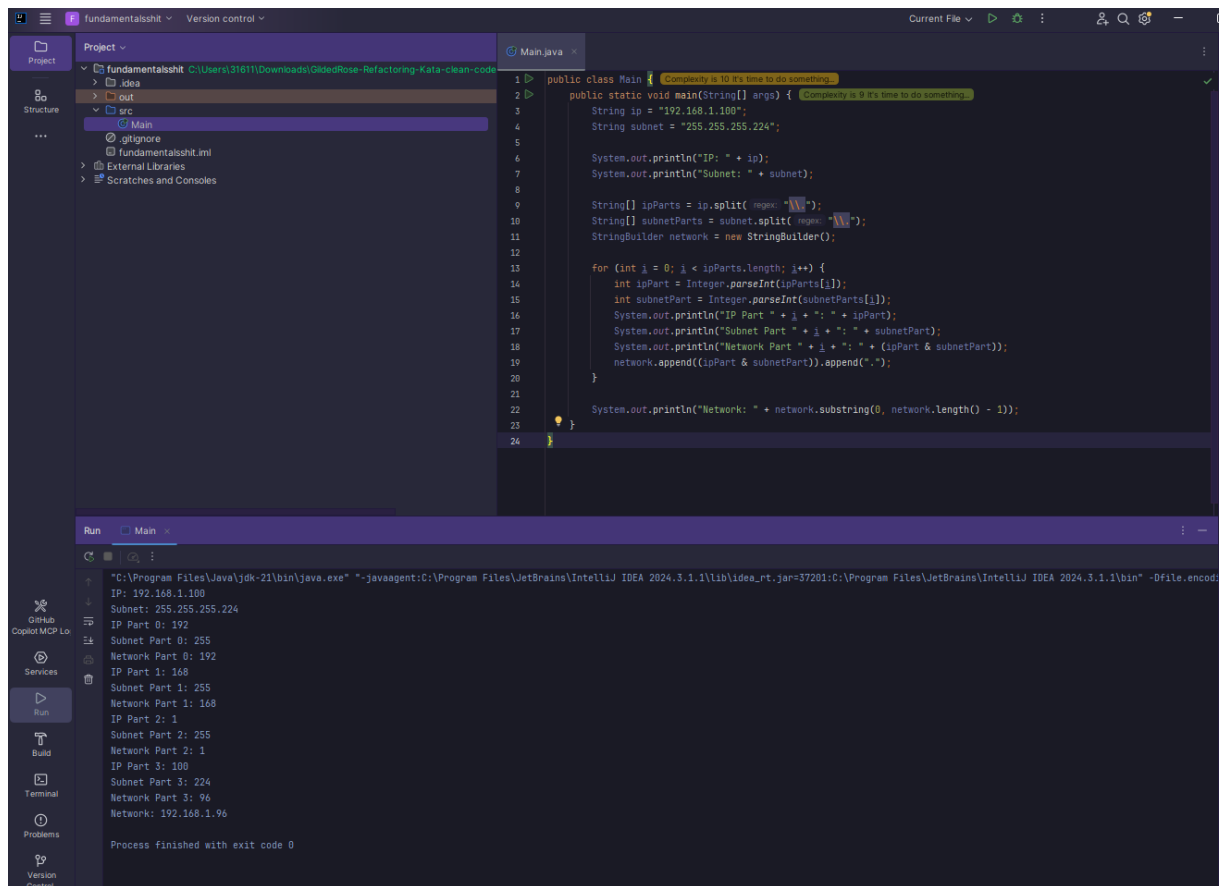
-----  
Network Addr: 11000000.10101000.00000001.01100000

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.

```
public class Main {  
    public static void main(String[] args) {  
        String ip = "192.168.1.100";  
        String subnet = "255.255.255.224";  
  
        System.out.println("IP: " + ip);  
        System.out.println("Subnet: " + subnet);  
  
        String[] ipParts = ip.split("\\.");  
        String[] subnetParts = subnet.split("\\.");  
        StringBuilder network = new StringBuilder();  
  
        for (int i = 0; i < ipParts.length; i++) {  
            int ipPart = Integer.parseInt(ipParts[i]);  
            int subnetPart = Integer.parseInt(subnetParts[i]);  
            System.out.println("IP Part " + i + ": " + ipPart);  
            System.out.println("Subnet Part " + i + ": " + subnetPart);  
            System.out.println("Network Part " + i + ": " + (ipPart & subnetPart));  
            network.append((ipPart & subnetPart)).append(".");  
        }  
  
        System.out.println("Network: " + network.substring(0, network.length() - 1));  
    }  
}
```





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