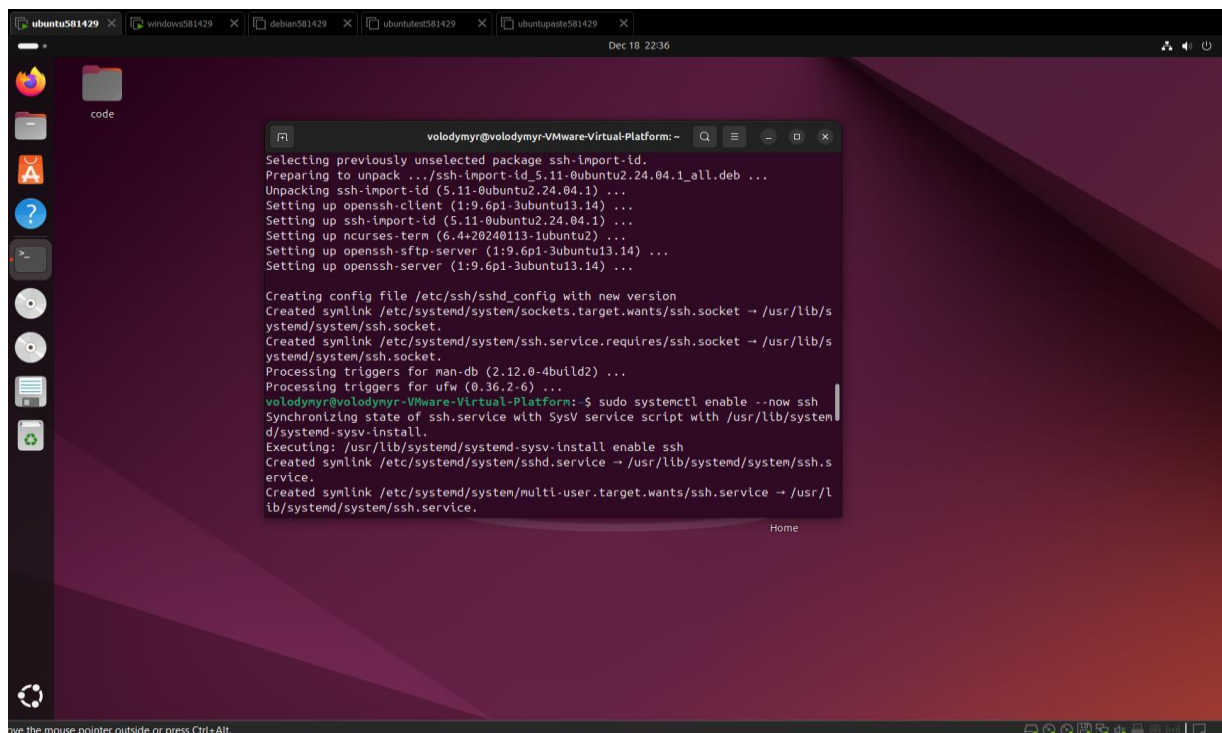


Template Week 6 – Networking

Student number: 581429

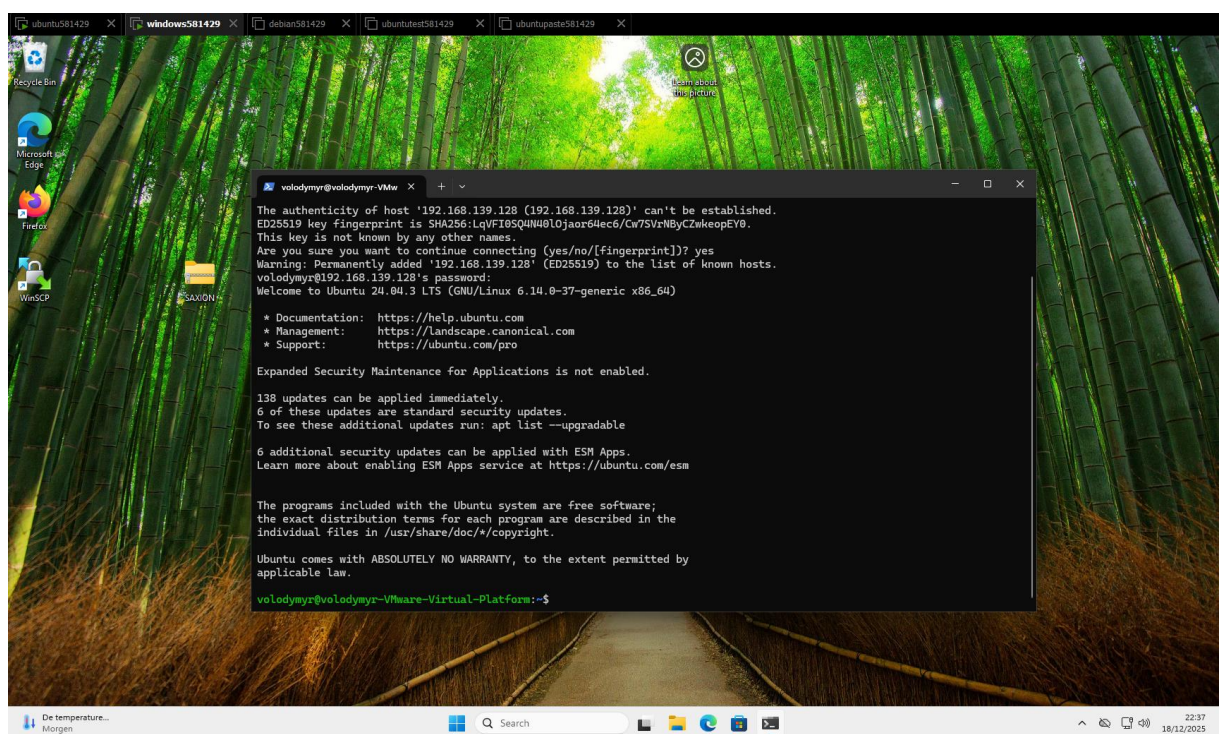
Assignment 6.1: Working from home

Screenshot installation openssh-server:



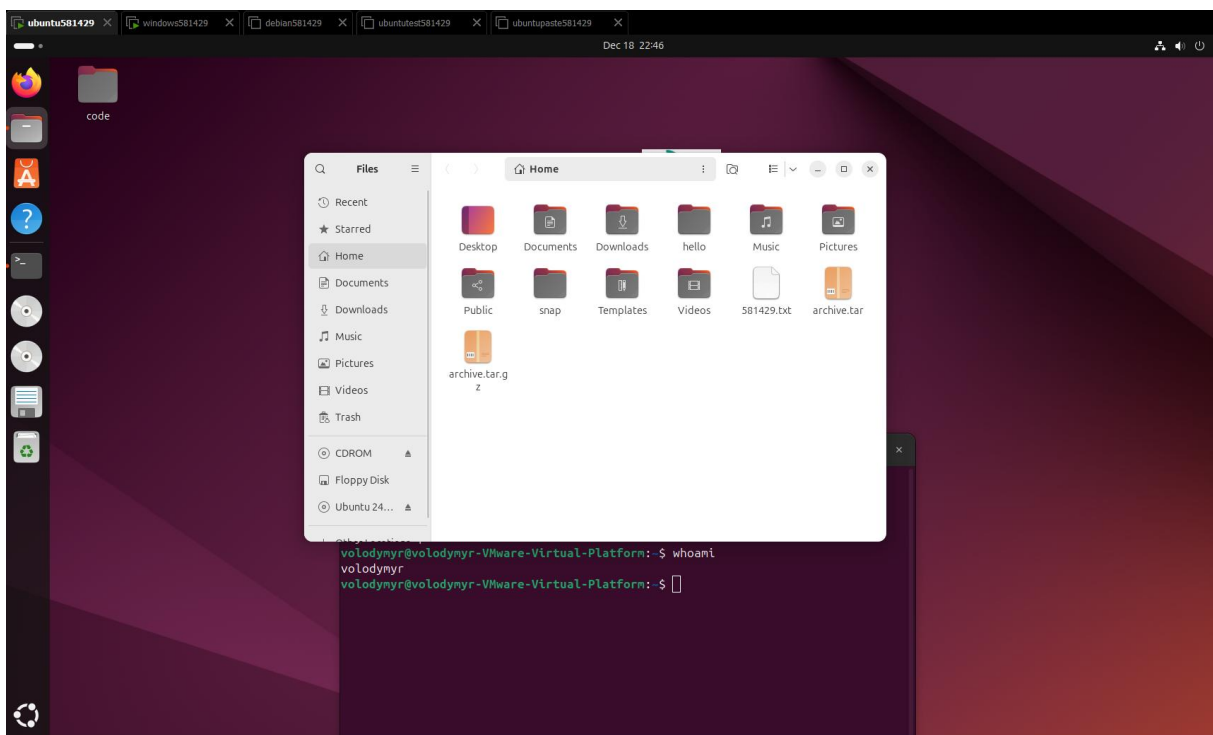
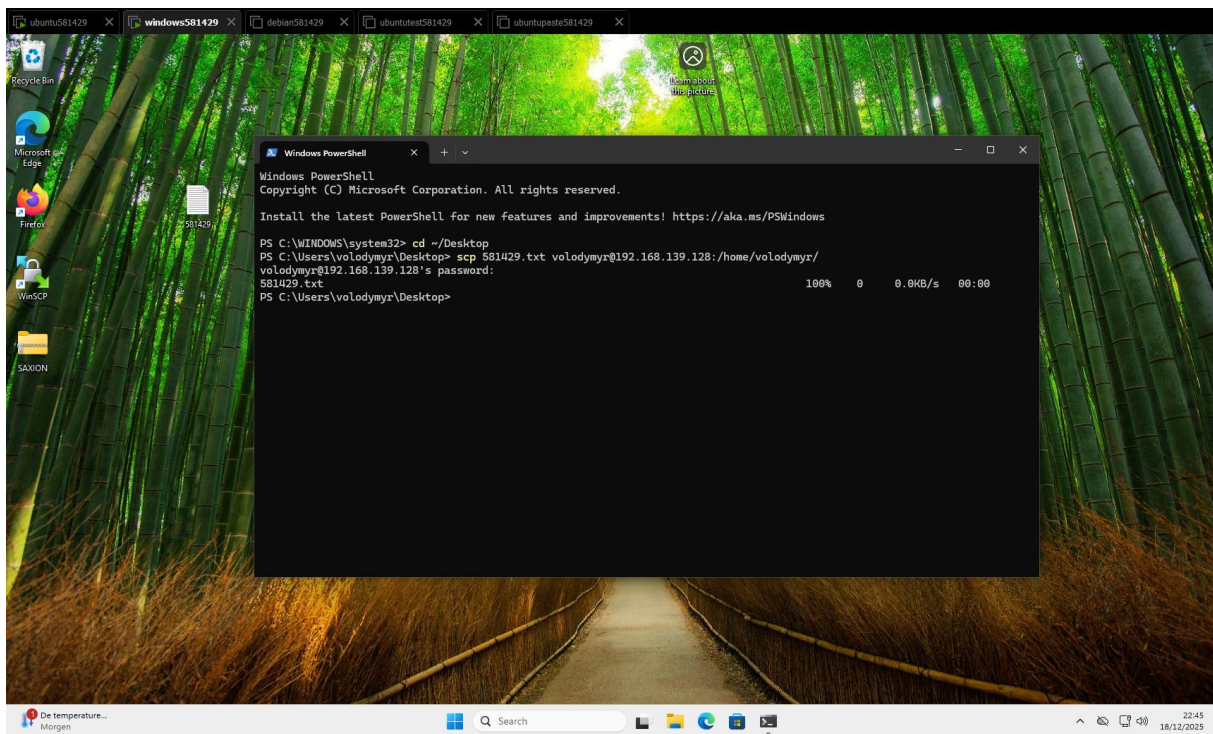
```
volodymyr@volodymyr-Virtual-Platform: ~  
Selecting previously unselected package ssh-import-id.  
Preparing to unpack .../ssh-import-id_5.11-0ubuntu2.24.04.1_all.deb ...  
Unpacking ssh-import-id (5.11-0ubuntu2.24.04.1) ...  
Setting up openssh-client (1:9.6p1-3ubuntu13.14) ...  
Setting up ssh-import-id (5.11-0ubuntu2.24.04.1) ...  
Setting up ncurses-term (6.4+20240113-1ubuntu2) ...  
Setting up openssh-sftp-server (1:9.6p1-3ubuntu13.14) ...  
Setting up openssh-server (1:9.6p1-3ubuntu13.14) ...  
  
Creating config file /etc/ssh/sshd_config with new version  
Created symlink /etc/systemd/system/sockets.target.wants/ssh.socket → /usr/lib/s  
ystemd/system/ssh.socket.  
Created symlink /etc/systemd/system/ssh.service.requires/ssh.socket → /usr/lib/s  
ystemd/system/ssh.socket.  
Processing triggers for man-db (2.12.0-4build2) ...  
Processing triggers for ufw (0.36.2-6) ...  
volodymyr@volodymyr-Virtual-Platform: ~$ sudo systemctl enable --now ssh  
Synchronizing state of ssh.service with SysV service script with /usr/lib/system  
d/systemd-sysv-install.  
Executing: /usr/lib/systemd/systemd-sysv-install enable ssh  
Created symlink /etc/systemd/system/ssh.service → /usr/lib/systemd/system/ssh.s  
ervice.  
Created symlink /etc/systemd/system/multi-user.target.wants/ssh.service → /usr/l  
ib/systemd/system/ssh.service.  
  
Home
```

Screenshot successful SSH command execution:

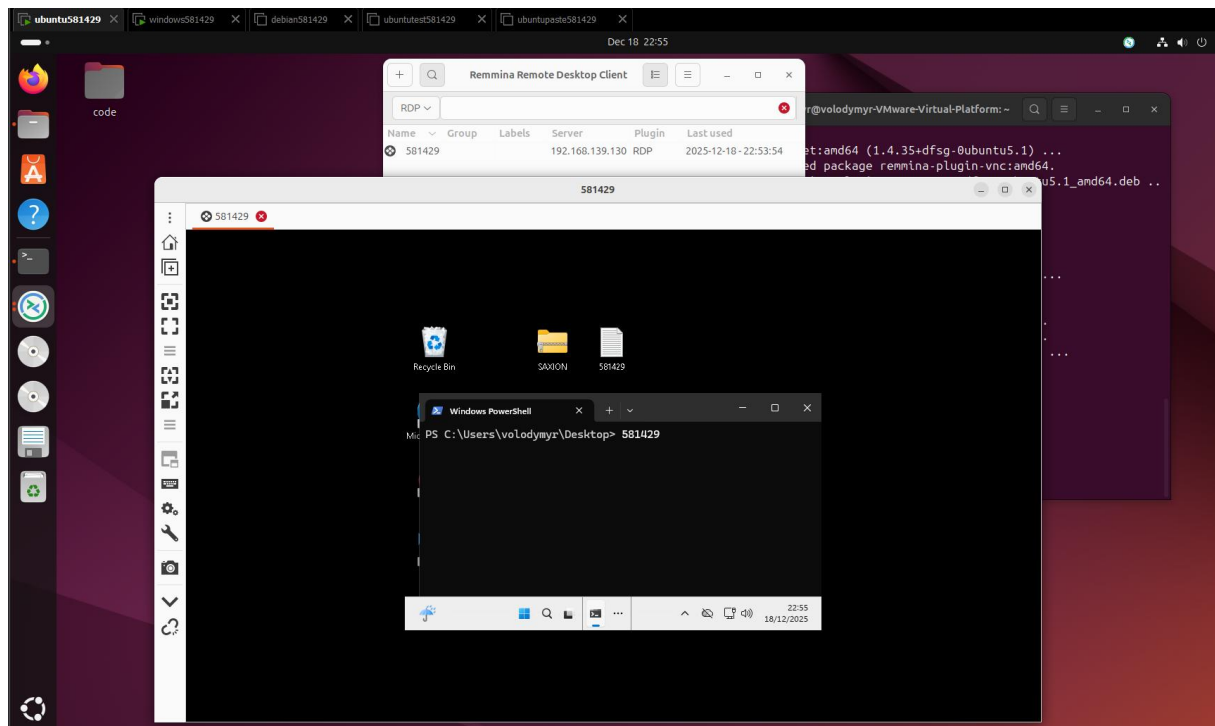


```
volodymyr@volodymyr-Virtual-Platform: ~$ ssh 192.168.139.128  
The authenticity of host '192.168.139.128 (192.168.139.128)' can't be established.  
ED25519 key fingerprint is SHA256:1qV7J85QW4H0jaor64ec6/Cw7SVzHByCZwkeopEY0.  
This key is not known by any other names.  
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes  
Warning: Permanently added '192.168.139.128' (ED25519) to the list of known hosts.  
volodymyr@192.168.139.128's password:  
Welcome to Ubuntu 24.04.3 LTS (GNU/Linux 6.14.0-37-generic x86_64)  
  
 * Documentation:  https://help.ubuntu.com  
 * Management:    https://landscape.canonical.com  
 * Support:        https://ubuntu.com/pro  
  
Expanded Security Maintenance for Applications is not enabled.  
  
138 updates can be applied immediately.  
6 of these updates are standard security updates.  
To see these additional updates run: apt list --upgradable  
  
6 additional security updates can be applied with ESM Apps.  
Learn more about enabling ESM Apps service at https://ubuntu.com/esm  
  
The programs included with the Ubuntu system are free software;  
the exact distribution terms for each program are described in the  
individual files in /usr/share/doc/*/*copyright.  
  
Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by  
applicable law.  
  
volodymyr@volodymyr-Virtual-Platform: ~$
```

Screenshot successful execution SCP command:

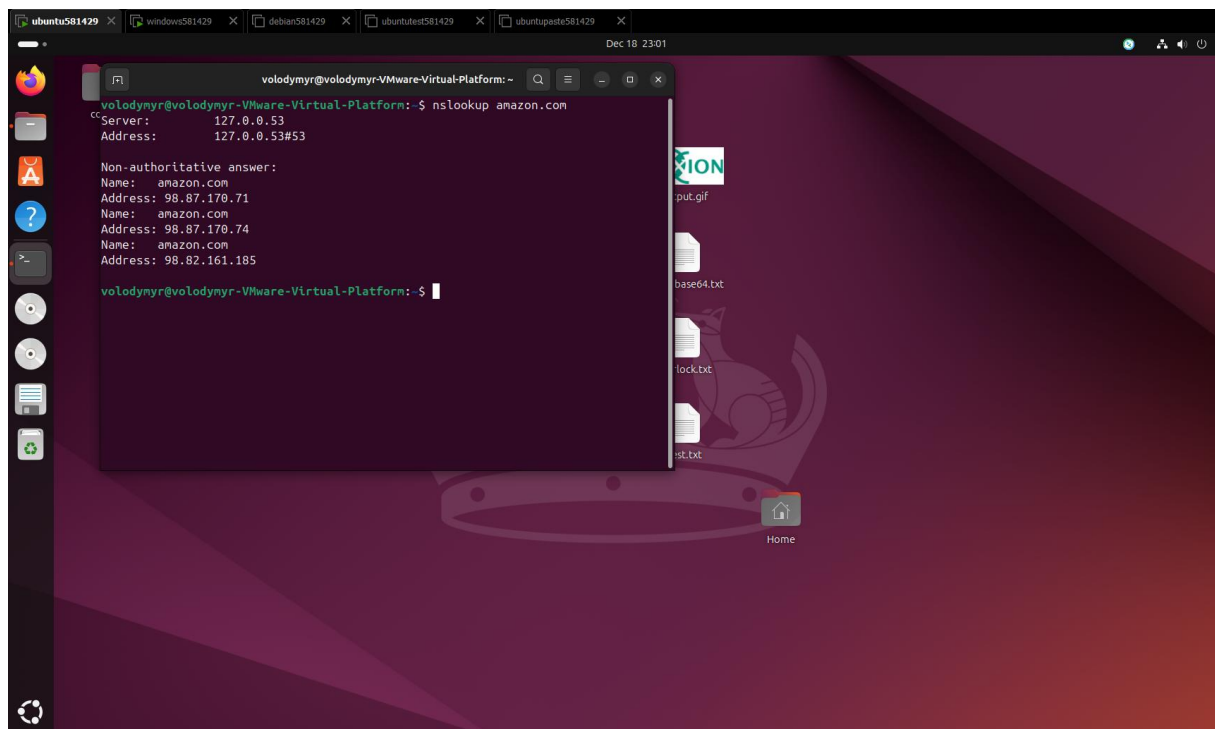


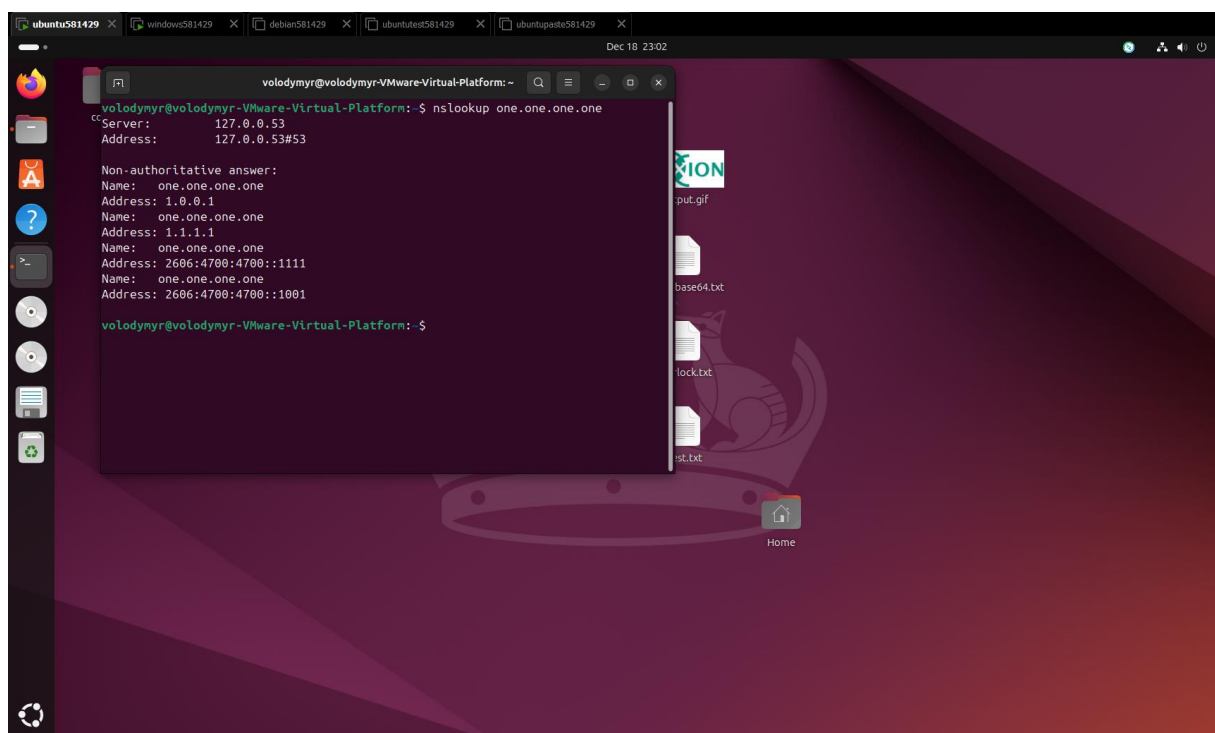
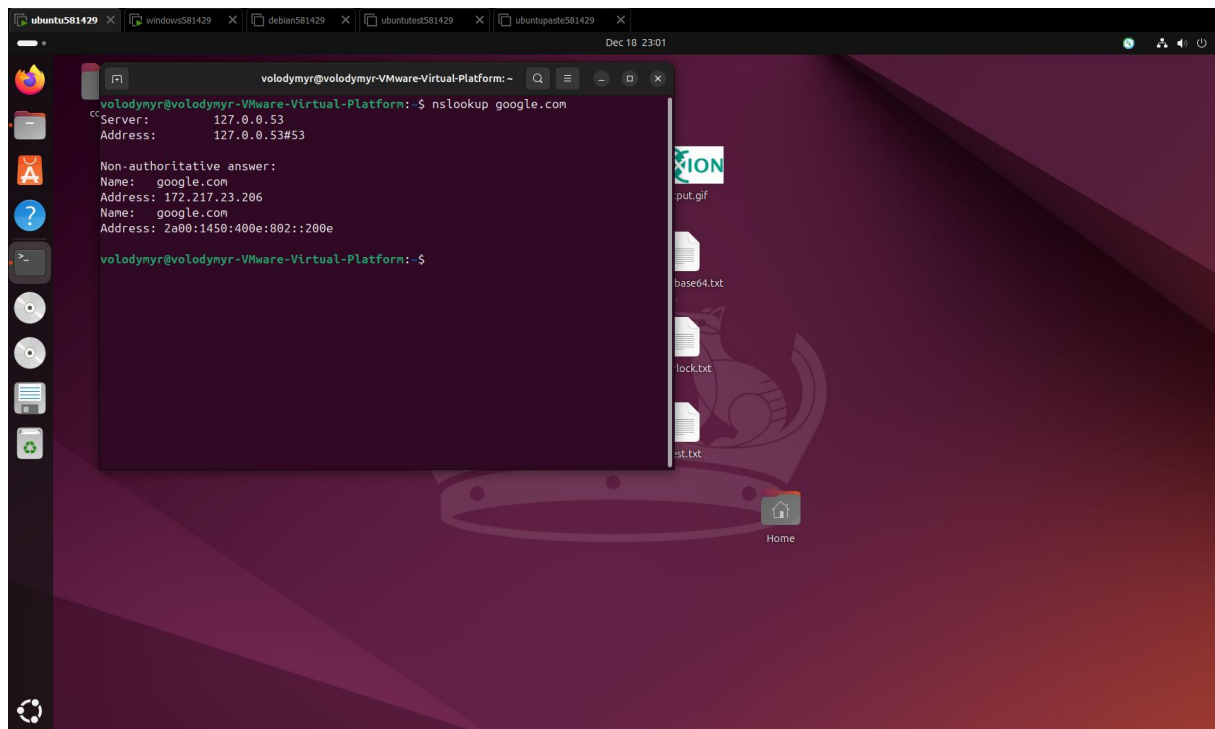
Screenshot remmina:

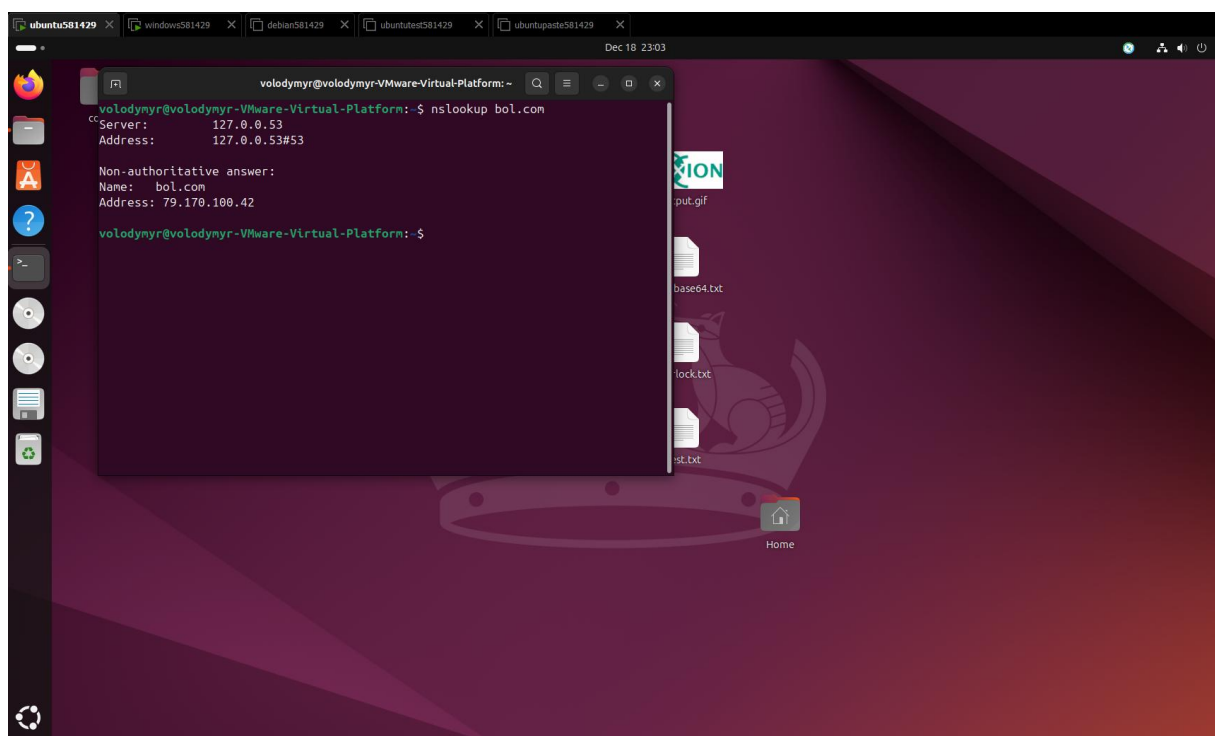
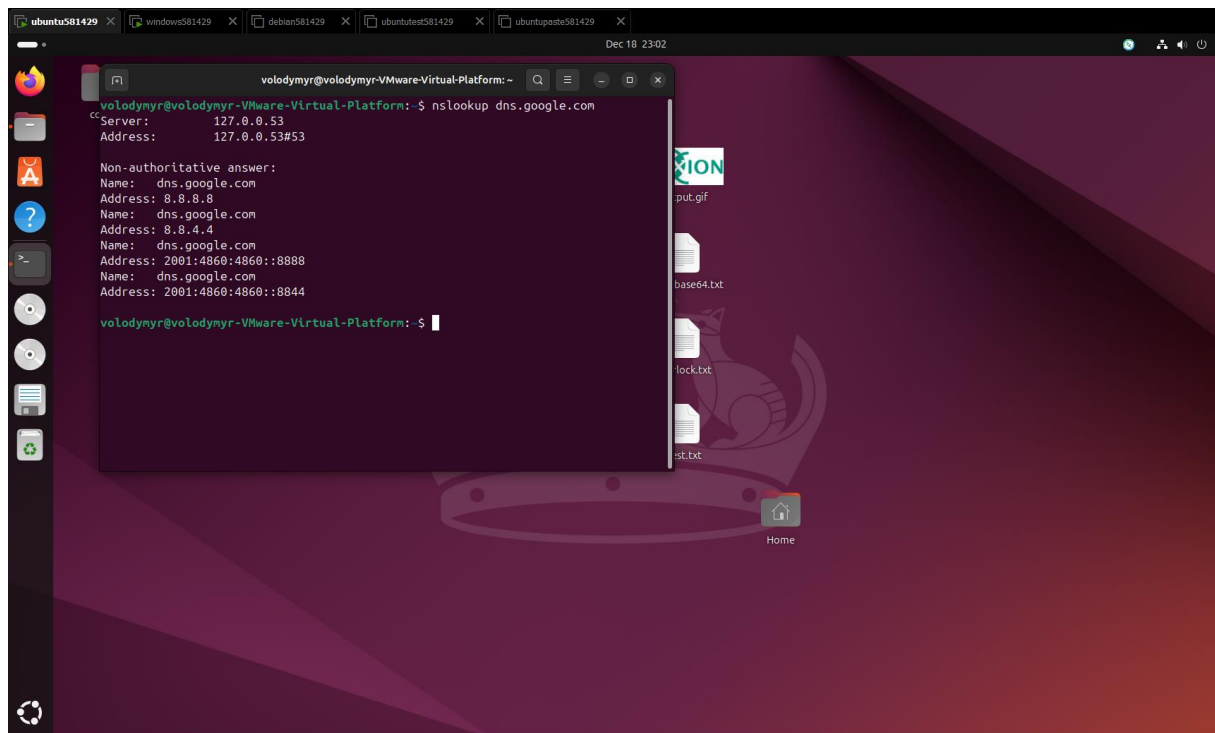


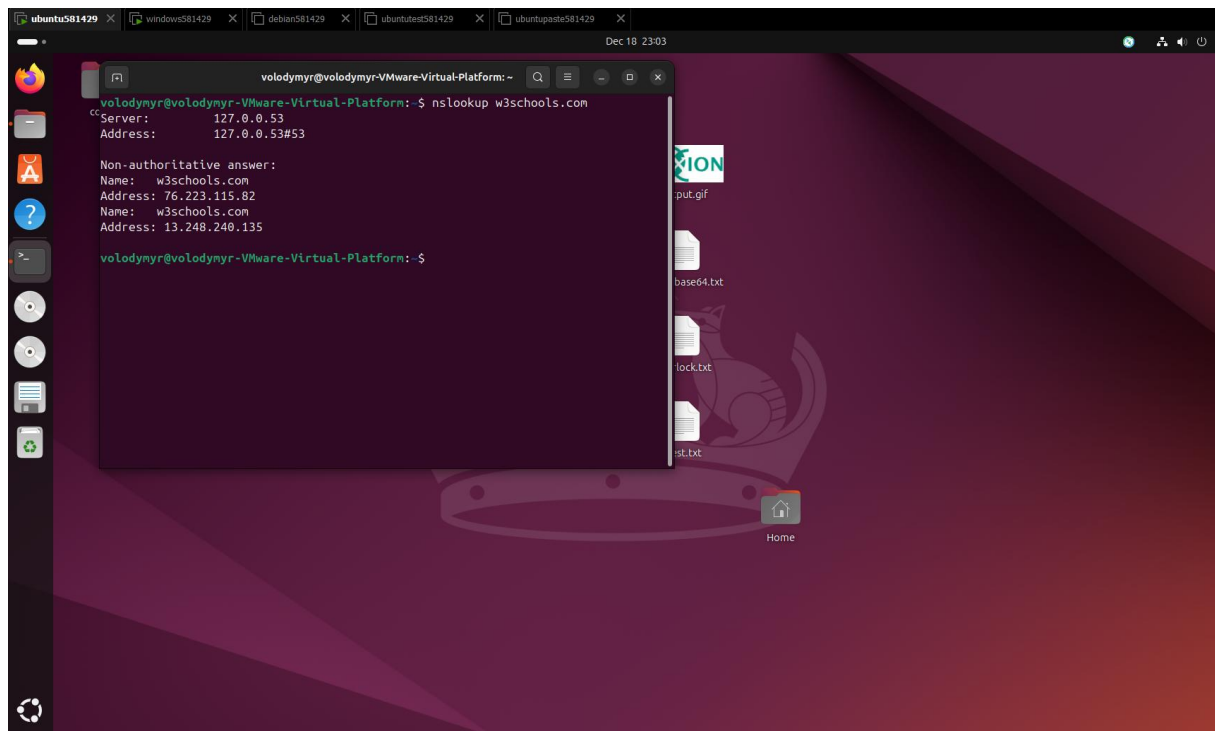
Assignment 6.2: IP addresses websites

Relevant screenshots nslookup command:

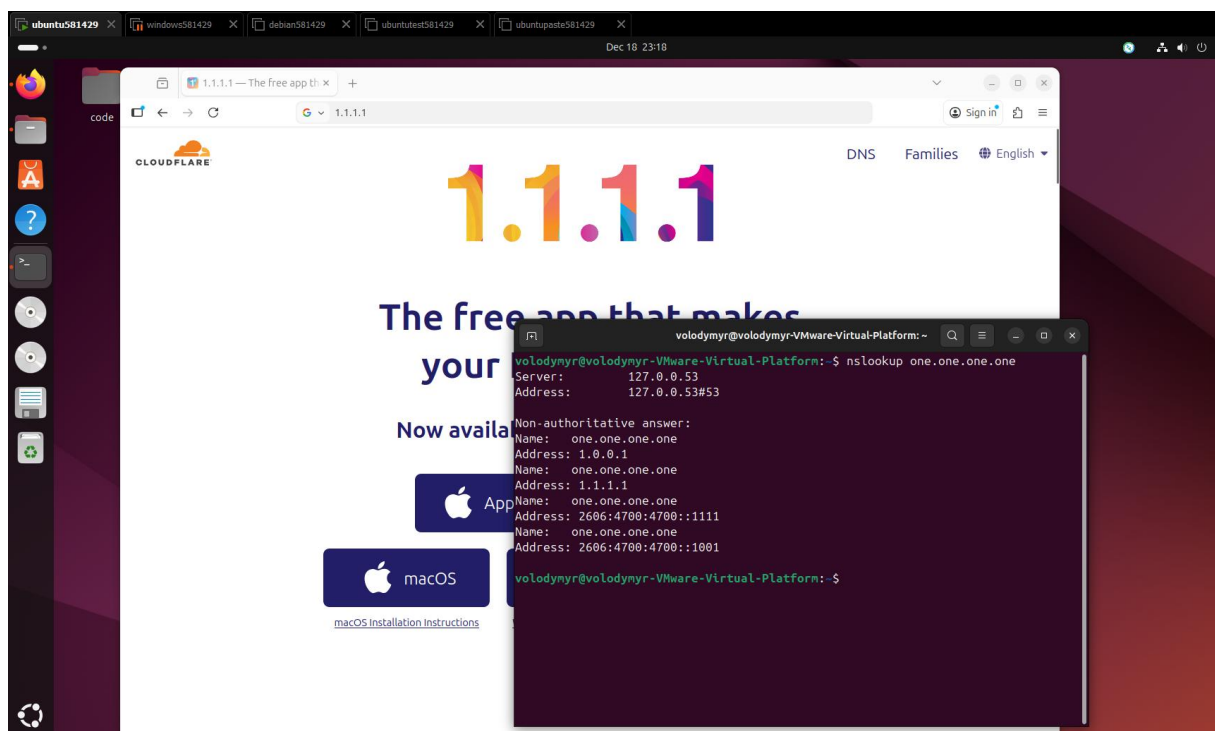








Screenshot website visit via IP address:



Assignment 6.3: subnetting

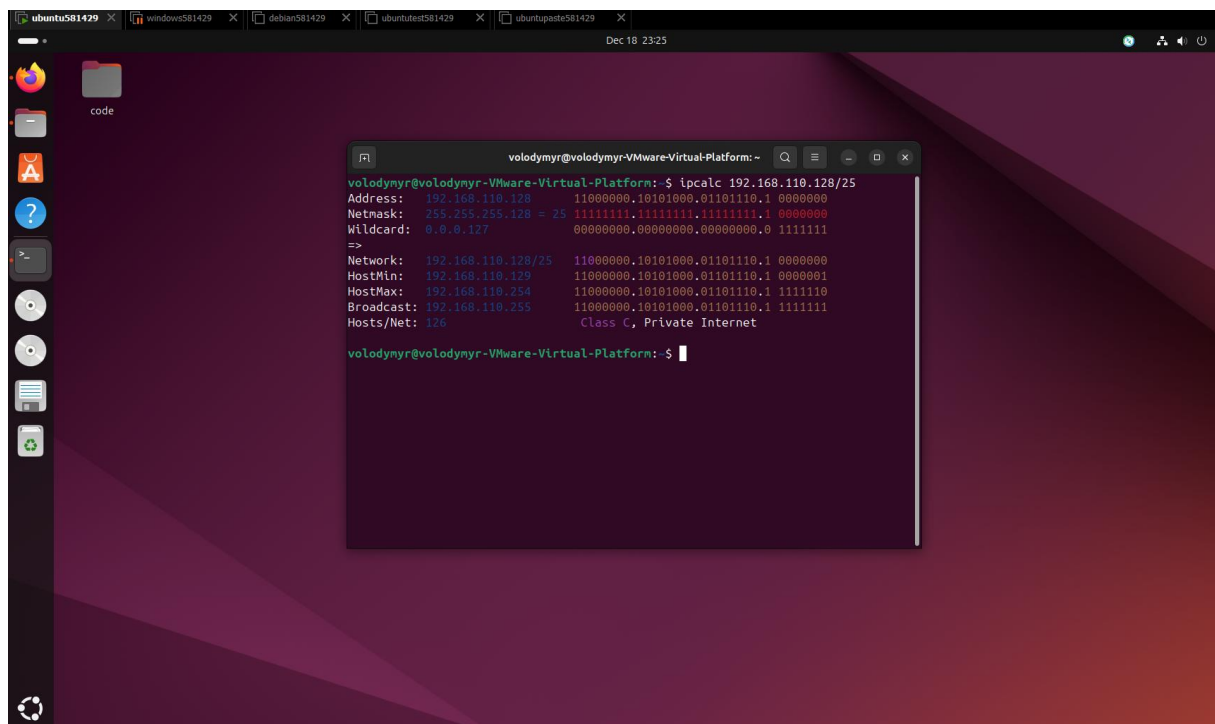
How many IP addresses are in this network configuration 192.168.110.128/25?

128

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`



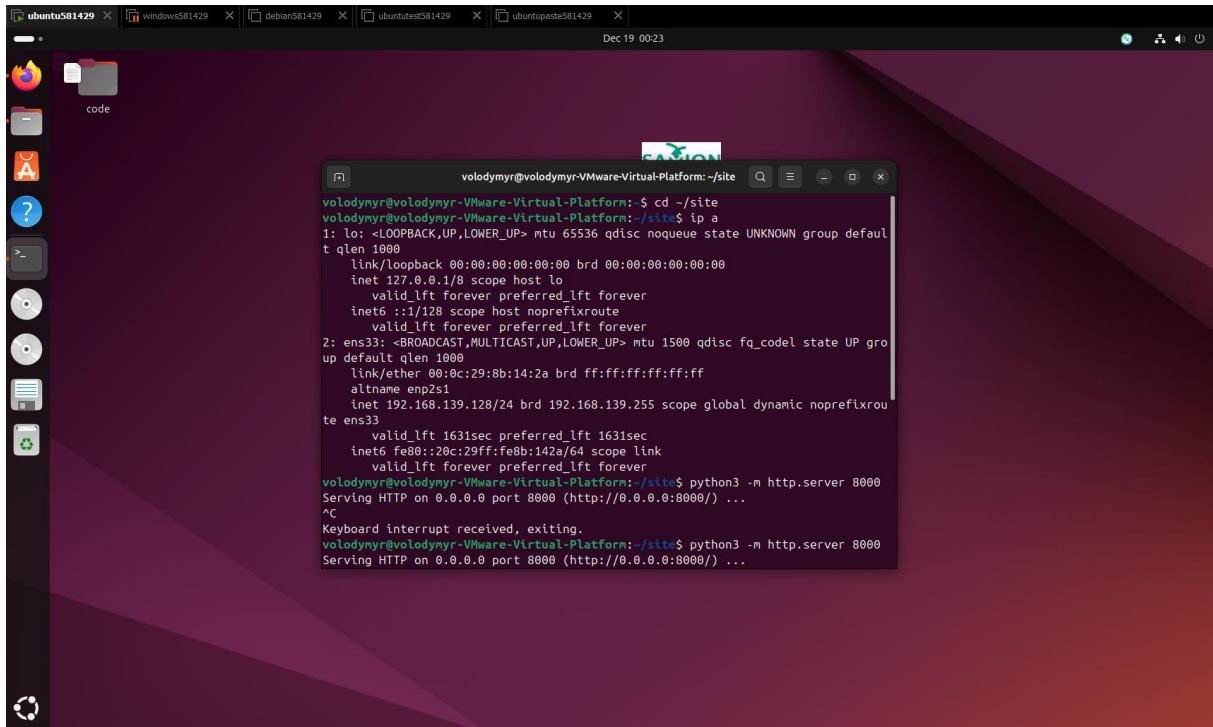
```
volodymyr@volodymyr-VMware-Virtual-Platform: ~  
volodymyr@volodymyr-VMware-Virtual-Platform: ~$ ipcalc 192.168.110.128/25  
Address: 192.168.110.128      11000000.10101000.01101110.1 0000000  
Netmask: 255.255.255.128 = 25 11111111.11111111.11111111.1 0000000  
Wildcard: 0.0.0.127          00000000.00000000.00000000.0 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  
volodymyr@volodymyr-VMware-Virtual-Platform: ~$
```

Explain the above calculation in your own words.

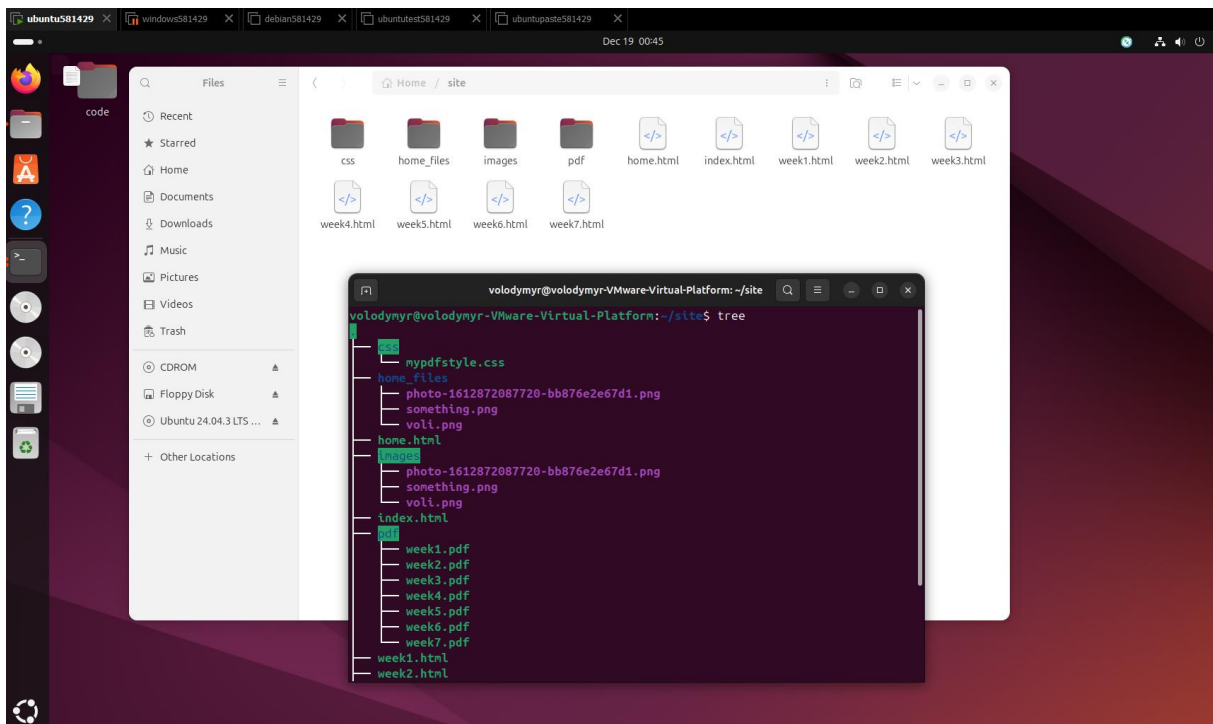
The subnet 192.168.110.128/25 means that 25 bits are used for the network and 7 bits are left for hosts. Since 7 bits are available, the total number of addresses is $2^7 = 128$. Out of these 128 addresses, one is reserved as the network ID (192.168.110.128) and one is reserved as the broadcast address (192.168.110.255). That leaves 126 usable addresses for devices. The usable range starts at 192.168.110.129 and ends at 192.168.110.254

Assignment 6.4: HTML

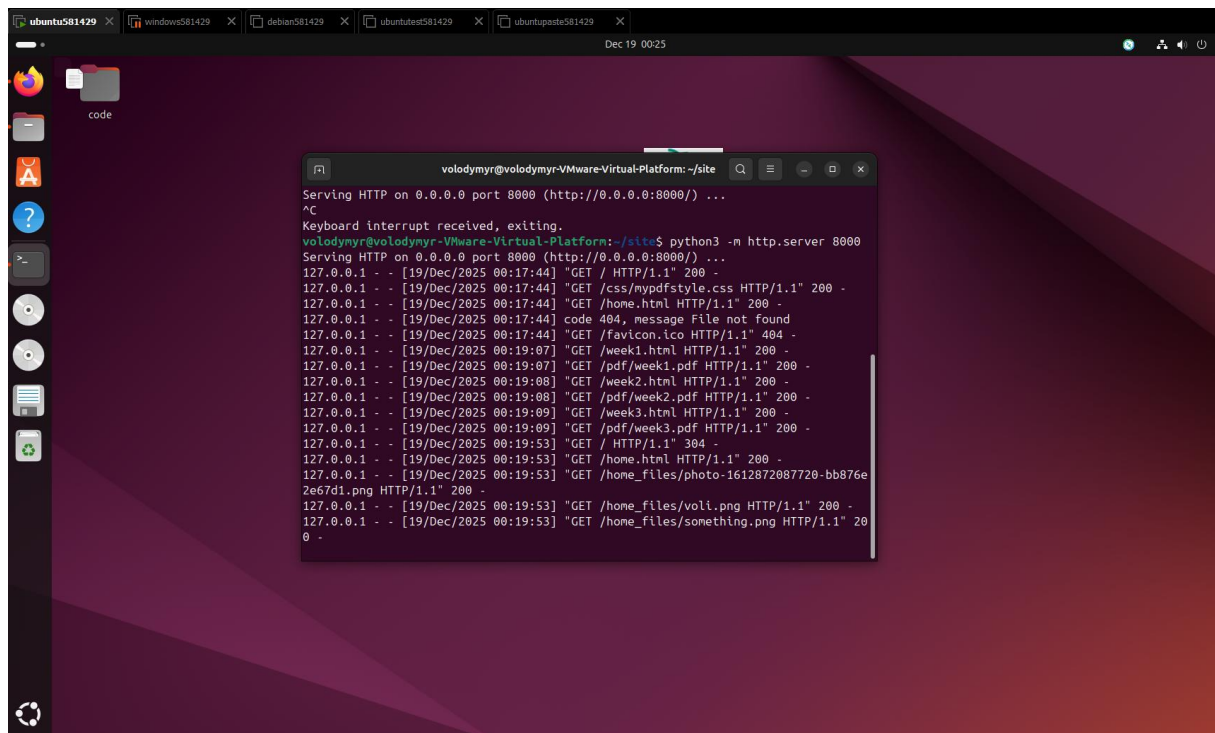
Screenshot IP address Ubuntu VM:



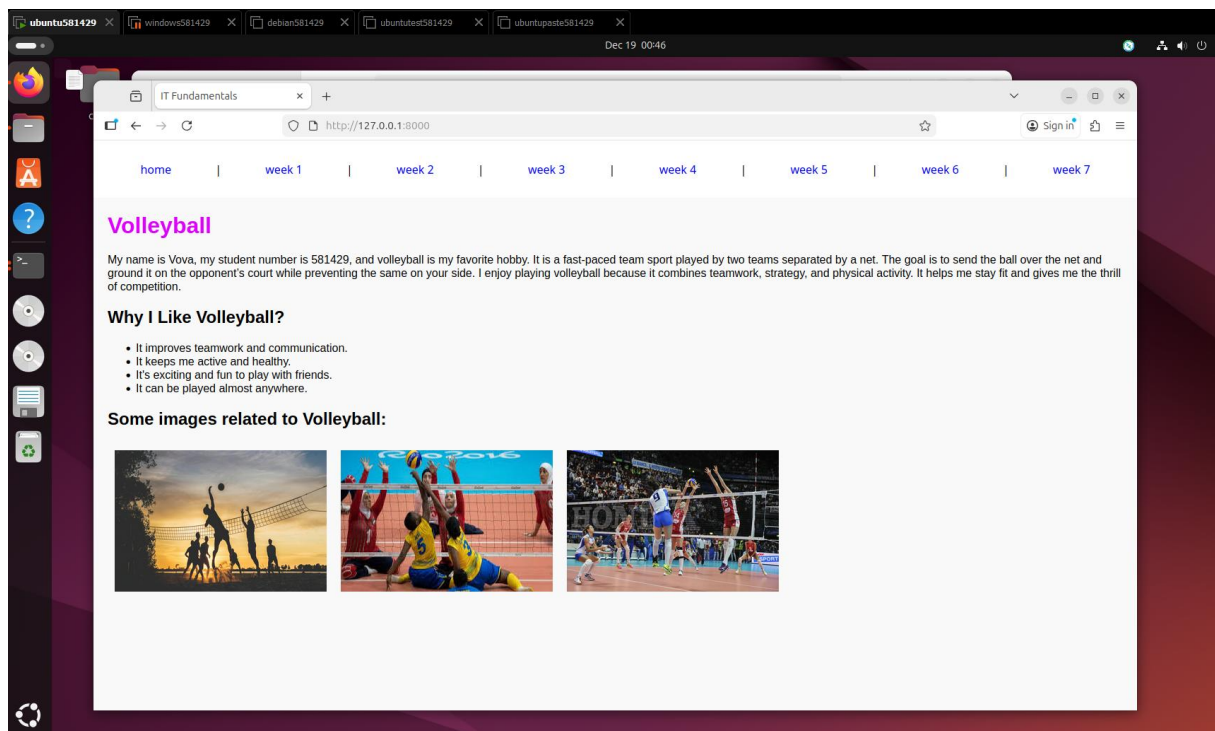
Screenshot of Site directory contents:



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.SaxionApp;

public class Application implements Runnable {

    public static void main(String[] args) {
        SaxionApp.start(new Application(), 800, 800);
    }

    public void run() {

        // Input IP and subnet mask using SaxionApp
        SaxionApp.print("Enter IP address: ");
        String ipStr = SaxionApp.readString();
        SaxionApp.print("Enter Subnet mask: ");
        String subnetStr = SaxionApp.readString();

        // Split into octets
        String[] ipParts = ipStr.split("\\.");
        String[] subnetParts = subnetStr.split("\\.");

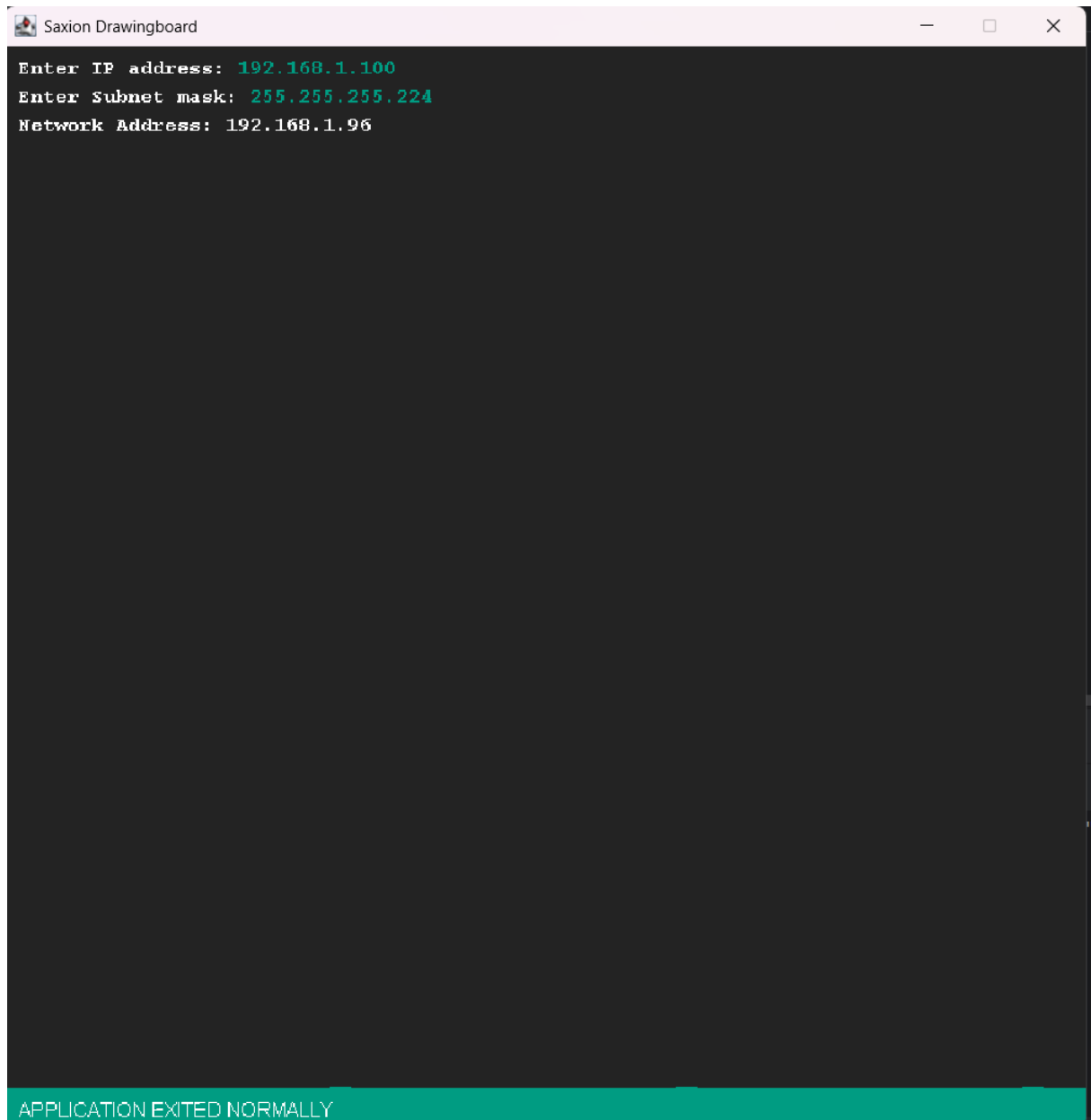
        int[] ip = new int[4];
        int[] subnet = new int[4];
        int[] network = new int[4];

        // Calculate network address
        for (int i = 0; i < 4; i++) {
            ip[i] = Integer.parseInt(ipParts[i]);
```

```

        subnet[i] = Integer.parseInt(subnetParts[i]);
        network[i] = ip[i] & subnet[i]; // bitwise AND
    }
    // Print network address
    SaxionApp.println("Network Address: " + network[0] + "." + network[1] + "." + network[2] + "."
+ network[3]);
    }
}

```



```

Saxion Drawingboard
Enter IP address: 192.168.1.100
Enter Subnet mask: 255.255.255.224
Network Address: 192.168.1.96

APPLICATION EXITED NORMALLY

```

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