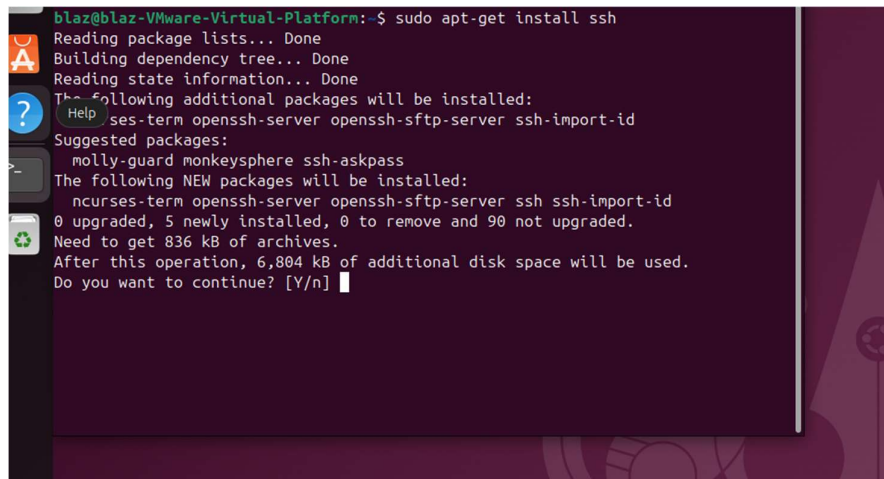


Template Week 6 – Networking

Student number: 569681

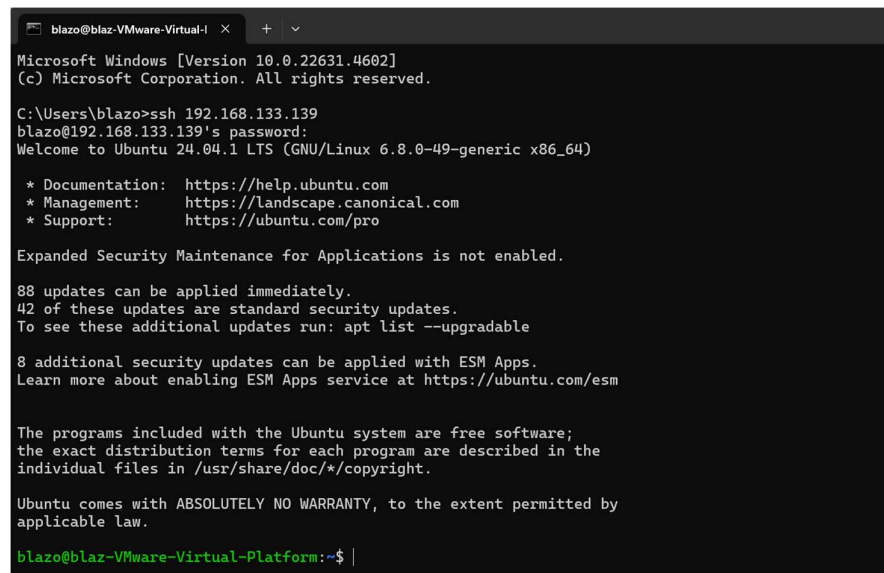
Assignment 6.1: Working from home

Screenshot installation openssh-server:



```
blaz@blaz-VMware-Virtual-Platform:~$ sudo apt-get install ssh
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
  ncurses-term openssh-server openssh-sftp-server ssh-import-id
Suggested packages:
  molly-guard monkeysphere ssh-askpass
The following NEW packages will be installed:
  ncurses-term openssh-server openssh-sftp-server ssh ssh-import-id
0 upgraded, 5 newly installed, 0 to remove and 90 not upgraded.
Need to get 836 kB of archives.
After this operation, 6,804 kB of additional disk space will be used.
Do you want to continue? [Y/n]
```

Screenshot successful SSH command execution:



```
blazo@blaz-VMware-Virtual-Platform:~$ ssh 192.168.133.139
Microsoft Windows [Version 10.0.22631.4602]
(c) Microsoft Corporation. All rights reserved.

C:\Users\blazo>ssh 192.168.133.139
blazo@192.168.133.139's password:
Welcome to Ubuntu 24.04.1 LTS (GNU/Linux 6.8.0-49-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.

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

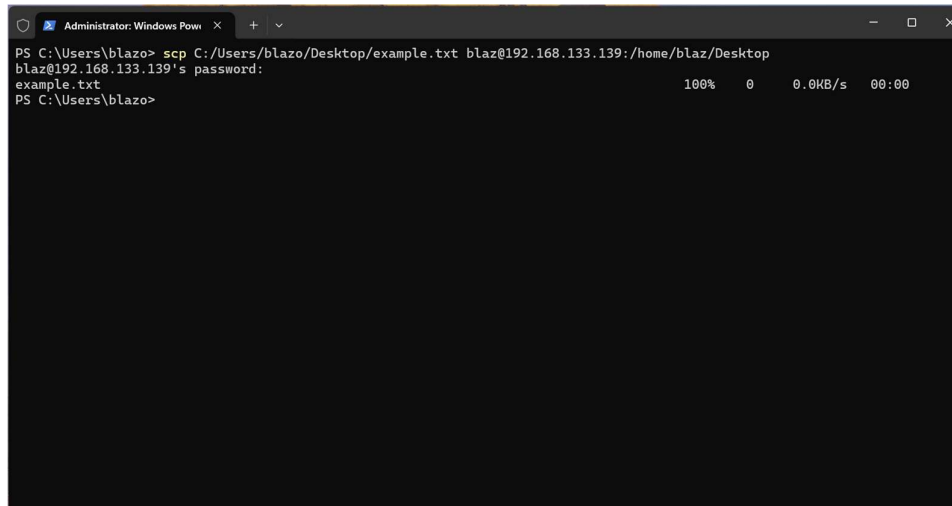
8 additional security updates can be applied with ESM Apps.
Learn more about enabling ESM Apps service at https://ubuntu.com/esm

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.

blazo@blaz-VMware-Virtual-Platform:~$
```

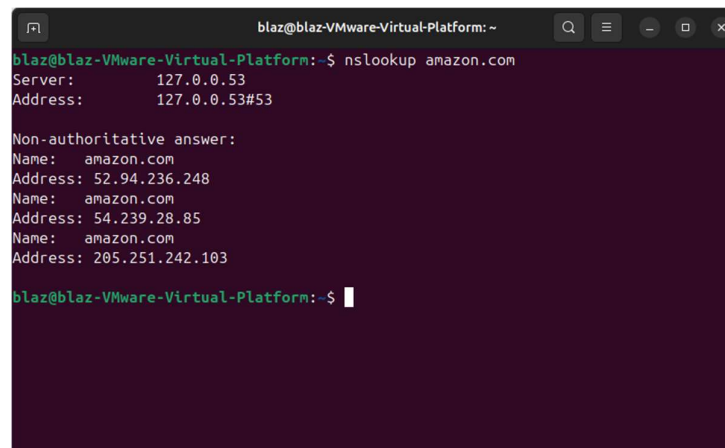
Screenshot successful execution SCP command:



```
Administrator: Windows PowerShell
PS C:\Users\blazo> scp C:/Users/blazo/Desktop/example.txt blaz@192.168.133.139:/home/blaz/Desktop
blaz@192.168.133.139's password:
example.txt                                100% 0 0.0KB/s 00:00
PS C:\Users\blazo>
```

Assignment 6.2: IP addresses websites

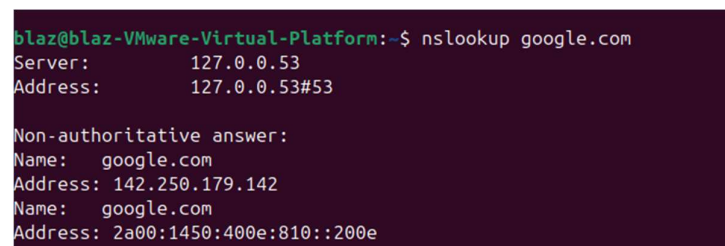
Relevant screenshots nslookup command:



```
blaz@blaz-VMware-Virtual-Platform: ~
blaz@blaz-VMware-Virtual-Platform:~$ nslookup amazon.com
Server:      127.0.0.53
Address:     127.0.0.53#53

Non-authoritative answer:
Name:   amazon.com
Address: 52.94.236.248
Name:   amazon.com
Address: 54.239.28.85
Name:   amazon.com
Address: 205.251.242.103

blaz@blaz-VMware-Virtual-Platform:~$
```



```
blaz@blaz-VMware-Virtual-Platform:~$ nslookup google.com
Server:      127.0.0.53
Address:     127.0.0.53#53

Non-authoritative answer:
Name:   google.com
Address: 142.250.179.142
Name:   google.com
Address: 2a00:1450:400e:810::200e
```

```
blaz@blaz-VMware-Virtual-Platform:~$ 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: 1.0.0.1
Name:   one.one.one.one
Address: 1.1.1.1
Name:   one.one.one.one
Address: 2606:4700:4700::1001
Name:   one.one.one.one
Address: 2606:4700:4700::1111
```

```
blaz@blaz-VMware-Virtual-Platform:~$ nslookup dns.google.com
Server:      127.0.0.53
Address:     127.0.0.53#53

Non-authoritative answer:
Name:   dns.google.com
Address: 8.8.8.8
Name:   dns.google.com
Address: 8.8.4.4
Name:   dns.google.com
Address: 2001:4860:4860::8844
Name:   dns.google.com
Address: 2001:4860:4860::8888
```

```
blaz@blaz-VMware-Virtual-Platform:~$ nslookup bol.com
Server:      127.0.0.53
Address:     127.0.0.53#53

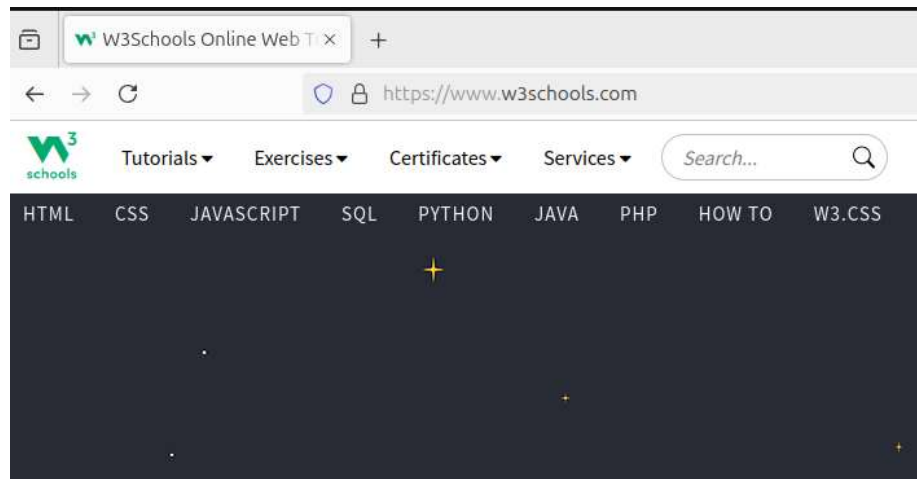
Non-authoritative answer:
Name:   bol.com
Address: 34.36.121.47

blaz@blaz-VMware-Virtual-Platform:~$
```

```
blaz@blaz-VMware-Virtual-Platform:~$ nslookup w3schools.com
Server:      127.0.0.53
Address:     127.0.0.53#53

Non-authoritative answer:
Name:   w3schools.com
Address: 156.234.234.233
```

Screenshot website visit via IP address:



Assignment 6.3: subnetting

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

The number of IP addresses available in this network configuration is 128.

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

The number of IP addresses available in this network configuration is 126.

Check your two previous answers with this calculator:

<https://www.calculator.net/ip-subnet-calculator.html>

Explain the above calculation in your own words.

The total length of an IPv4 address is 32 bits. Since 25 bits are used for the network, there are:

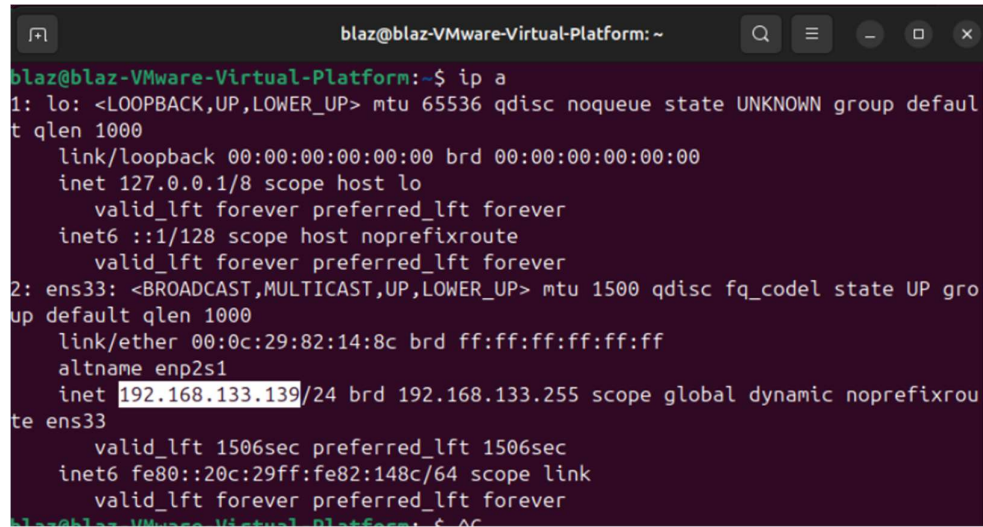
$32 - 25 = 7$ host bits

The number of available IP addresses is calculated as:

number of host bits $= 2^7 = 128$ IP addresses

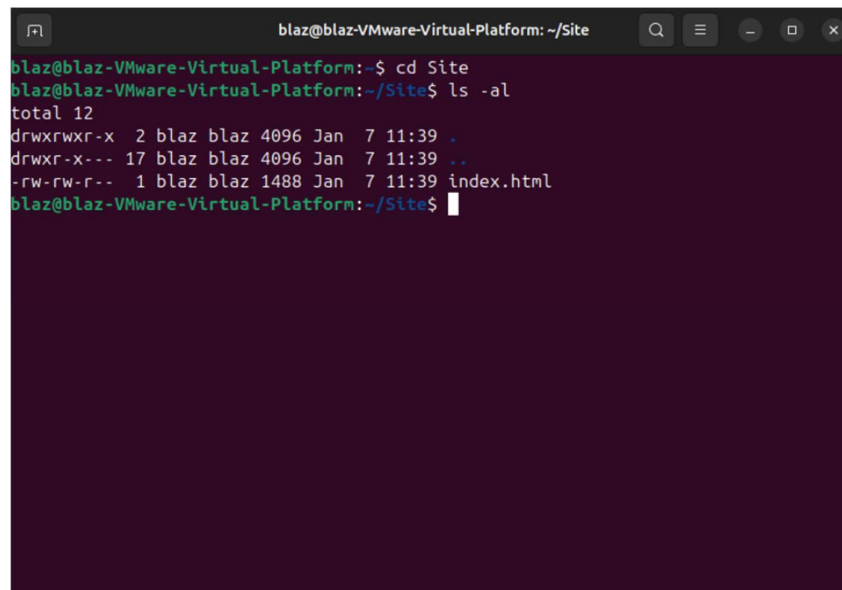
Assignment 6.4: HTML

Screenshot IP address Ubuntu VM:

A terminal window titled 'blaz@blaz-VMware-Virtual-Platform: ~' showing the output of the 'ip a' command. The output lists network interfaces: 'lo' (loopback) with IP 127.0.0.1 and 'ens33' (Ethernet) with IP 192.168.133.139. The IP address 192.168.133.139 is highlighted with a yellow box.

```
blaz@blaz-VMware-Virtual-Platform:~$ ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default
    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:29:82:14:8c brd ff:ff:ff:ff:ff:ff
    altname enp2s1
    inet 192.168.133.139/24 brd 192.168.133.255 scope global dynamic noprefixroute
        valid_lft 1506sec preferred_lft 1506sec
    inet6 fe80::20c:29ff:fe82:148c/64 scope link
        valid_lft forever preferred_lft forever
blaz@blaz-VMware-Virtual-Platform:~$
```

Screenshot of Site directory contents:

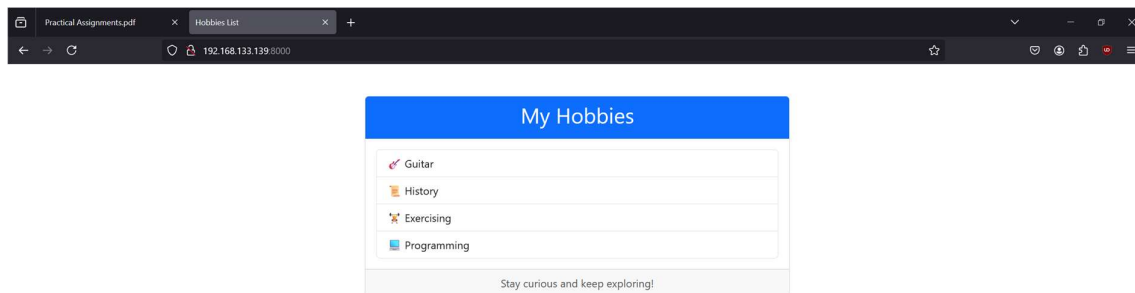
A terminal window titled 'blaz@blaz-VMware-Virtual-Platform: ~/Site' showing the output of the 'cd Site' and 'ls -al' commands. The 'ls -al' output shows three entries: '.', '..', and 'index.html'.

```
blaz@blaz-VMware-Virtual-Platform:~$ cd Site
blaz@blaz-VMware-Virtual-Platform:~/Site$ ls -al
total 12
drwxrwxr-x  2 blaz blaz 4096 Jan  7 11:39 .
drwxr-x--- 17 blaz blaz 4096 Jan  7 11:39 ..
-rw-rw-r--  1 blaz blaz 1488 Jan  7 11:39 index.html
blaz@blaz-VMware-Virtual-Platform:~/Site$
```

Screenshot python3 webserver command:

```
blaz@blaz-VMware-Virtual-Platform: ~/Site
blaz@blaz-VMware-Virtual-Platform:~$ mkdir Site
blaz@blaz-VMware-Virtual-Platform:~$ cd Site
blaz@blaz-VMware-Virtual-Platform:~/Site$ touch index.html
blaz@blaz-VMware-Virtual-Platform:~/Site$ sudo nano index.html
[sudo] password for blaz:
blaz@blaz-VMware-Virtual-Platform:~/Site$ python3 -m http.server 8000
Serving HTTP on 0.0.0.0 port 8000 (http://0.0.0.0:8000/) ...
127.0.0.1 - - [07/Jan/2025 11:40:52] "GET / HTTP/1.1" 200 -
127.0.0.1 - - [07/Jan/2025 11:40:53] code 404, message File not found
127.0.0.1 - - [07/Jan/2025 11:40:53] "GET /favicon.ico HTTP/1.1" 404 -
192.168.133.1 - - [07/Jan/2025 11:41:54] "GET / HTTP/1.1" 200 -
192.168.133.1 - - [07/Jan/2025 11:41:54] code 404, message File not found
192.168.133.1 - - [07/Jan/2025 11:41:54] "GET /favicon.ico HTTP/1.1" 404 -
192.168.133.1 - - [07/Jan/2025 11:42:12] "GET / HTTP/1.1" 200 -
192.168.133.1 - - [07/Jan/2025 11:42:14] "GET / HTTP/1.1" 200 -
192.168.133.1 - - [07/Jan/2025 11:42:15] "GET / HTTP/1.1" 304 -
192.168.133.1 - - [07/Jan/2025 11:42:15] "GET / HTTP/1.1" 304 -
192.168.133.1 - - [07/Jan/2025 11:42:15] "GET / HTTP/1.1" 304 -
192.168.133.1 - - [07/Jan/2025 11:42:15] "GET / HTTP/1.1" 304 -
192.168.133.1 - - [07/Jan/2025 11:42:15] "GET / HTTP/1.1" 304 -
192.168.133.1 - - [07/Jan/2025 11:42:15] "GET / HTTP/1.1" 304 -
192.168.133.1 - - [07/Jan/2025 11:42:15] "GET / HTTP/1.1" 304 -
192.168.133.1 - - [07/Jan/2025 11:42:16] "GET / HTTP/1.1" 304 -
```

Screenshot web browser visits your site



Bonus point assignment – week 6

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;

import java.awt.*;

public class Application implements Runnable {

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

    public void run() {
        showMenu();
    }

    private void showMenu() {
        int chosenOption;

        SaxionApp.clear();

        SaxionApp.println("Choose a menu option:");
        SaxionApp.println("1. Is number odd?");
        SaxionApp.println("2. Is number a power of two?");
        SaxionApp.println("3. Print two's complement of a number");
        SaxionApp.println("4. Calculate the network address");

        chosenOption = SaxionApp.readInt();

        while (!checkInputBounds(chosenOption, 0, 4)) {
            SaxionApp.println("Choose valid option!", Color.RED);
            chosenOption = SaxionApp.readInt();
        }
    }
}
```

```

        switch (chosenOption) {
            case 1:
                isNumberOdd();
                break;
            case 2:
                isPowerOfTwo();
                break;
            case 3:
                printTwoComplement();
                break;
            case 4:
                calculateNetworkSegment();
                break;
        }
    }

    private boolean checkInputBounds(int input, int lowerBound, int
upperBound) {
        return input >= lowerBound && input <= upperBound;
    }

    private void isNumberOdd() {
        int number;

        SaxionApp.clear();
        SaxionApp.println("Input your number");
        number = SaxionApp.readInt();

        if ((number & 1) == 1) {
            SaxionApp.println(number + " is odd.");
        } else {
            SaxionApp.println(number + " is even.");
        }

        SaxionApp.pause();
        showMenu();
    }

    private void isPowerOfTwo() {
        int number;

        SaxionApp.clear();
        SaxionApp.println("Input your number");
        number = SaxionApp.readInt();

        if (number > 0 && (number & (number - 1)) == 0) {
            SaxionApp.println(number + " is a power of two.");
        } else {
            SaxionApp.println(number + " is not a power of two.");
        }

        SaxionApp.pause();
        showMenu();
    }

    private void printTwoComplement() {
        int number;

```



```

SaxionApp.clear();
SaxionApp.println("Input your number");

number = SaxionApp.readInt();
number = ~number + 1;

SaxionApp.println("Two's complement is: " + number);

SaxionApp.pause();
showMenu();

}

private void calculateNetworkSegment() {
    String ip, subnet;
    String result = "";
    String[] ipValues, subnetValues;

    SaxionApp.clear();
    SaxionApp.println("Input your IP address:");

    ip = SaxionApp.readString();

    while (!checkAddress(ip)) {
        ip = SaxionApp.readString();
    }

    SaxionApp.println("Input your subnet mask:");

    subnet = SaxionApp.readString();

    while (!checkAddress(subnet)) {
        subnet = SaxionApp.readString();
    }

    ipValues = ip.split("\\.");
    subnetValues = subnet.split("\\.");

    for (int i = 0; i < ipValues.length; i++) {
        int currentIP = Integer.parseInt(ipValues[i]);
        int currentSubnet = Integer.parseInt(subnetValues[i]);

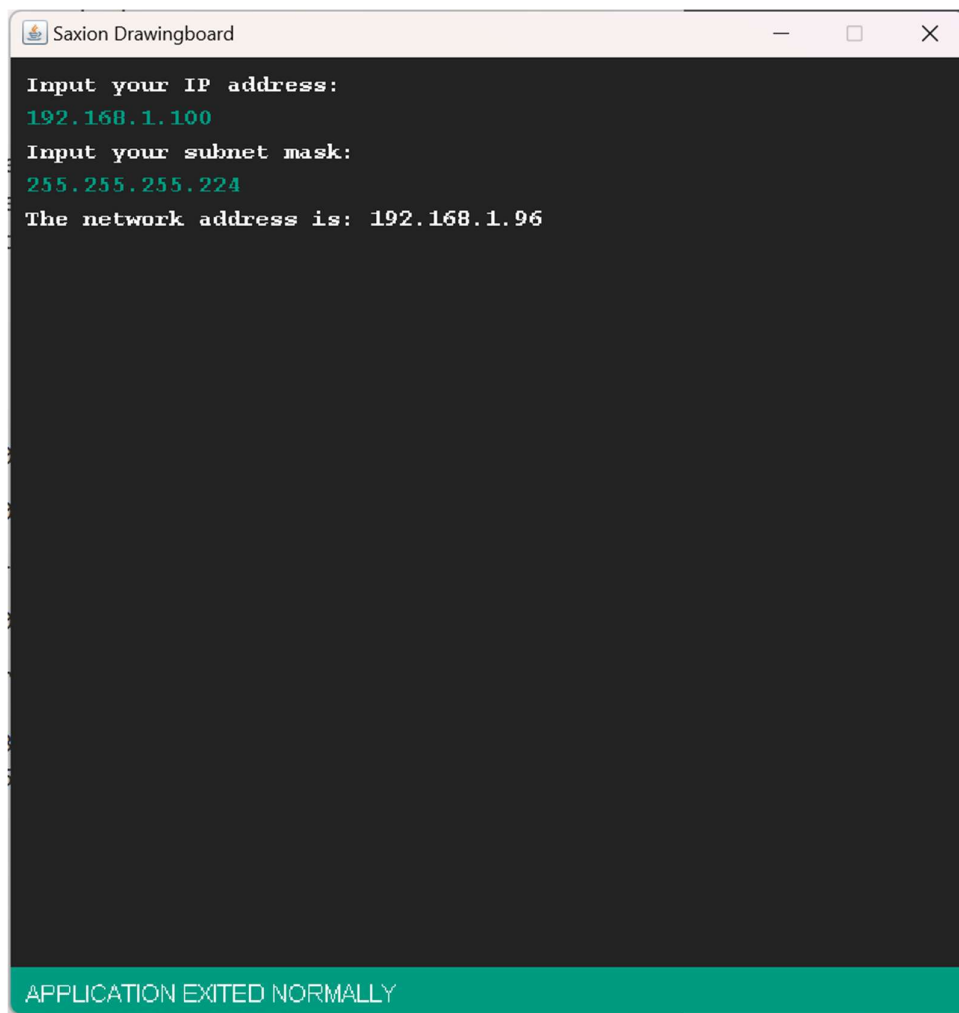
        if (i != ipValues.length-1) {
            result += (currentIP & currentSubnet) + ".";
        } else {
            result += String.valueOf(currentIP & currentSubnet);
        }
    }

    SaxionApp.println("The network address is: " + result);
}

private boolean checkAddress(String ip) {
    for (int i = 0; i < ip.length(); i++) {
        char c = ip.charAt(i);

```

```
        if (Character.isAlphabetic(c)) {  
            return false;  
        }  
    }  
    return true;  
}
```



```
Saxion Drawingboard  
Input your IP address:  
192.168.1.100  
Input your subnet mask:  
255.255.255.224  
The network address is: 192.168.1.96  
  
APPLICATION EXITED NORMALLY
```