Template Week 6 – Networking

Student number: 568681

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:

Screenshot website visit via IP address:

Assignment 6.3: subnetting

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

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

Check your two previous answers with this calculator:
https://www.calculator.net/ip-subnet-calculator.html

Explain the above calculation in your own words.

Assignment 6.4: HTML

Screenshot IP address Ubuntu VM:

Screenshot of Site directory contents:

Screenshot python3 webserver command:

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;

public class Application *implements Runnable* {

```
public static void main(String[] args) {
    SaxionApp.start(new Application(), 1000, 1000);
}

public void run() {
    SaxionApp.printLine("Enter IP Address: ");
    String ipAddress = SaxionApp.readString();

    SaxionApp.printLine("Enter Subnet Mask: ");
    String subnet = SaxionApp.readString();

    String ipBinary = ipToBinary(ipAddress);
    String subnetBinary = ipToBinary(subnet);
    String networkBinary = bitwiseAndIP(ipBinary, subnetBinary);

    SaxionApp.printLine("IP Address in binary: " + ipBinary);
    SaxionApp.printLine("Subnet Mask in binary: " + subnetBinary);
    SaxionApp.printLine("Network address in binary: " + networkBinary);
    String networkAddress = binaryToIp(networkBinary);
    SaxionApp.printLine("Network Address in decimal: " + networkAddress);

}
public String ipToBinary(String ip) {
    String[] octets = ip.split("\\.");
    StringBuilder binaryIp = new StringBuilder();

    for (String octet : octets) {
        int num = Integer.parseInt(octet);
        String binaryOctet = String.format("%8s", Integer.toBinaryString(num)).replace(' ', '0');
        binaryIp.append(binaryOctet).append(".");
    }

    return binaryIp.substring(0, binaryIp.length() - 1);
}

public String bitwiseAndIP(String ip1, String ip2) {
    String[] octets1 = ip1.split("\\.");
    String[] octets2 = ip2.split("\\.");
    StringBuilder networkBinary = new StringBuilder();

    for (int i = 0; i < 4; i++) {
        int part1 = Integer.parseInt(octets1[i], 2);
        int part2 = Integer.parseInt(octets2[i], 2);
        int result = part1 & part2; // Bitwise AND operation
        String binaryResult = String.format("%8s", Integer.toBinaryString(result)).replace(' ', '0');
        networkBinary.append(binaryResult).append(".");
    }
```

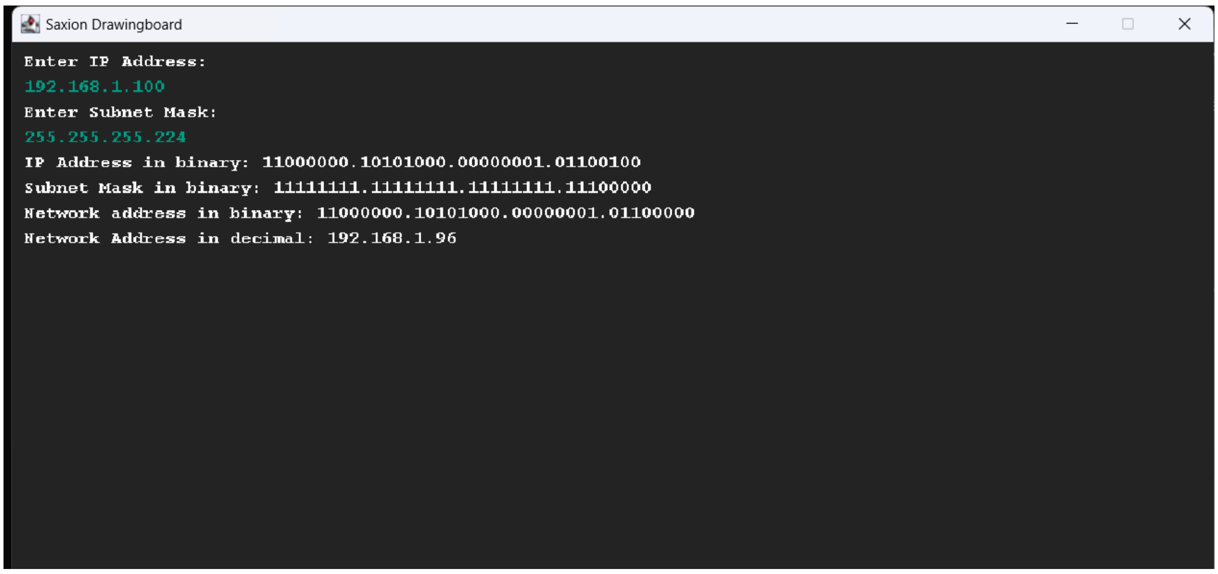
```
        return networkBinary.substring(0, networkBinary.length() - 1);
    }

    public String binaryToIp(String binary) {
        String[] octets = binary.split("\\.");
        StringBuilder ipAddress = new StringBuilder();

        for (String octet : octets) {
            int decimal = Integer.parseInt(octet, 2);
            ipAddress.append(decimal).append(".");
        }

        // Remove the last dot
        return ipAddress.substring(0, ipAddress.length() - 1);
    }

}
```


Saxion Drawingboard

```
Enter IP Address:
192.168.1.100
Enter Subnet Mask:
255.255.255.224
IP Address in binary: 11000000.10101000.00000001.01100100
Subnet Mask in binary: 11111111.11111111.11111111.11100000
Network address in binary: 11000000.10101000.00000001.01100000
Network Address in decimal: 192.168.1.96
```

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