**OPPDRAG 1 – Dokumentasjon**

**OPPGAVE 1**

**1. Koble til nettverket**

Jeg koblet meg først til 2IMD-ruteren.

**2. Statisk IP-adresse** Jeg satte en statisk IP-adresse på Pi-en ved å kjøre:

sudo ip addr add 192.168.13.16/16 dev wlan0

**3. Se adresser**

Ved å kjøre ip a kunne jeg se de forskjellige adressene, men senere lærte at « hostname -I» Bare viser ip addressen.

**4. Feilmeldinger**

Jeg fikk en feilmelding om at jeg ikke fikk kontakt. Dette var fordi PC-ene mine ikke var på samme nettverk eller ikke hadde riktig DNS/GATEWAY.

**5. Flere IP-adresser**

Under konfigurasjonen klarte jeg å sette 6 IP-adresser på Pi-en. Jeg måtte derfor søke hvordan man sletter en adresse. Jeg fant i en cheatsheet at kommandoen sudo ip addr del

(ip-adresse) dev eth0/wlan0 sletter den.

**6. Til slutt** Til slutt klarte jeg å pinge mellom PC-ene.

**Bilder**

Her kan du sette inn skjermbilder av:

* **Kommandoen for å sette IP-adresse**
* **Utdata fra ip a**

A computer screen with white text

AI-generated content may be incorrect.

* 
* **Ping mellom PC-ene**

A computer screen shot of a number

AI-generated content may be incorrect.

A computer screen with white text

AI-generated content may be incorrect.

**Oppsummering/det jeg lærte**

Wlan0/Eth0 står får wireless og ethernet nettverk.

Sette statisk ip-addresse på pi-en gjennom terminal og manuelt.

**OPPGAVE 2**

**OPPGAVE 2**

**1. Oppdater systemet**  
Jeg oppdaterte programvaren og sikkerheten på Pi-en ved å kjøre:

sudo apt update

**Kommando for å oppdatere pakkelisten med updates**

sudo apt upgrade -y

**Kommando for å laste ned updates som blir funnet av «sudo apt update»**

Dette sørget for at systemet hadde de nyeste pakkene.

**2. Installer en webserver**  
Jeg valgte å installere en enkel webserver med navn «Apache» ved å bruke komandoen.

sudo apt install apache2 -y

**3. Finn IP-adressen til Pi-en**  
Jeg fant IP-adressen ved å kjøre:

hostname -I

Deretter skrev jeg denne adressen inn i en nettleser på en enhet på samme nettverk og fikk opp standard “It works”-siden fra Apache. Jeg fant ut at Apache starter som en tjeneste med en gang du laster den ned. Som betyr at webserveren starter med en gang.

**4. Dokumentrot (hvor nettsiden ligger)**

Her begynner jeg å ekperimentere fordi oppgaven ba om en webserver, så jeg tenkte «kan jeg legge inn html?»

Ja, jeg gjorde litt søk og fant ut hvordan veldig raskt.

Apache blir lagret i et standard dokument som er:

/var/www/html

Ved å bruke «nano» komandoen kunne jeg legge mine egne HTML-filer inn.

**5. Gi brukeren min tilgang til webmappen**  
Som standard eies denne mappen av “root”. For å kunne redigere uten sudo kjørte jeg:

sudo chown -R pi:pi /var/www/html

Men I mitt tilfelde måtte jeg bruke brukernavnet mitt på pi-en.

Sudo er som en admin fant jeg ut senere.

og å kjøre «sudo» foran en komando så sier du til systemet «Kjør denne kommandoen som om jeg var admin eller superbruker.

(samme prinsipp som i Oppgave 1: sudo = kjør som administrator, chown = endre eier, pi:pi = bruker og gruppe “pi”, -R = rekursivt).

**6. Lag din egen nettside**  
Jeg redigerte/lage index.html ved hjelp av chatgpt fordi jeg ikke har tid til å redigere html nå.

nano /var/www/html/index.html

Limte inn HTML, lagret (Ctrl+O) og lukket (Ctrl+X).  
Oppdaterte nettleseren på http://<Pi-IP> – og så min egen side.

**9. Oppsummering av viktige begreper**

* Webserver = program som sender filer til nettleseren
* Dokumentrot = mappen der nettsiden ligger
* sudo = kjør som administrator
* chown = endre hvem som eier filer/mapper
* IP-adresse = adressen du bruker i nettleseren for å nå Pi-en