

# Mappedokument

## Forutsetninger

**Oppgave 1:** Dette gikk ut på at vi skulle jukse så mye vi ville, derfor tolket jeg dette som at vi kan kopiere all kode fra AI, og få AI til å gjøre oppgaven «for deg». Prompts og resultat kommer senere i dokumentet. Jeg syntes det var litt vanskelig å forstå hva som faktisk skulle inkluderes i videoen. Jeg tolket dette som at vi skal vise resultatene våres og hvordan vi fikk resultatene. Jeg kunne også forklart all kode som jeg fikk av AI i videon, men 5 minutter hadde ikke vært nok å forklart alt, så jeg tolket som at det ikke var det som var ment å ha med i videon.

**Oppgave 2:** Hvor vi skal refaktorere javascript-kode at man kan bruke prinsipper fra webtricks og undervisningen. Jeg tolket det som at ting fra undervisning er også fra arbeidskrav. Mer om dette på hjelpemdirler seksjonen.

**Oppgave 3:** Det jeg syntes ble litt utydelig i denne oppgaven var hvor viktig CSS var. Vi har ikke brukt mye tid på CSS i dette faget, så jeg syntes i utgangspunktet det var rart at vi skulle lage våres egen CSS her. Her tok jeg også i bruk av AI, men redigerte noe i CSS-filen slik at det ble litt mer som jeg ønsket. Det ble også fortalt når vi fikk oppgaven at selve CSSen IKKE skulle bli vurdert, men i oppgave teksten så står det at det vil bli vurdert. Dette syntes jeg ble litt forvirrende og motsigende. Derfor har jeg gjort litt endringer selv også i CSSen. Jeg har ikke kommentert i selve CSS koden hva jeg har endret på fordi funksjonaliteten var prioriteringen.

## Hjelpemdirler

**Oppgave 1:** Som nevnt tidligere, så brukte jeg AI for å løse dette. Jeg nevner det en gang til, men jeg tolket dette som at her var «alt» lov, inkludert å kopiere alt AI har laget til deg. Her brukte jeg Microsoft CoPilot. De to første bildene er prompts som jeg sendte til CoPilot.

Første bilde er for register.js og andre bilde er for actions.js. De andre bildene er resultatene jeg fikk av CoPilot. Bilder av prompt og resultat:



```

// Javascript
// register.js

// Liste over stillinger
const stillinger = [
    "rektor", // Index 0
    "vaktmester", // Index 1
    "vaktmester", // Index 2
    "professor", // Index 3
    "lektor" // Index 4
];

// Liste over ansatte
const ansatte = [
    {
        "Rektor": {
            etternavn: "Olsen",
            fornavn: "Per",
            kontor: "B201",
            stilling: 1,
            epost: "per.olsen@høyskole.no",
            kursansvar: []
        }
    },
    {
        "Dekaner": {
            etternavn: "Selberg",
            fornavn: "Henrik",
            kontor: "C104",
            stilling: 1,
            epost: "henrik.selberg@høyskole.no",
            kursansvar: []
        }
    },
    {
        "Lektorer": {
            etternavn: "Dahl",
            fornavn: "Camilla",
            kontor: "C305",
            stilling: 3,
            epost: "camilla.dahl@høyskole.no",
            kursansvar: ["Økonomi"]
        }
    },
    {
        "Vaktmestre": {
            etternavn: "Andreasen",
            fornavn: "Morten",
            kontor: "Vaktmesterbod",
            stilling: 2,
            epost: "morten.andreasen@høyskole.no",
            kursansvar: []
        }
    },
    {
        "Professorer": {
            etternavn: "Hansen",
            fornavn: "Eva",
            kontor: "A101",
            stilling: 1,
            epost: "eva.hansen@høyskole.no",
            kursansvar: []
        }
    },
    {
        "Professorer": {
            etternavn: "Hansen",
            fornavn: "Jens",
            kontor: "A102",
            stilling: 3,
            epost: "jens.hansen@høyskole.no",
            kursansvar: ["Fysikk 1"]
        }
    },
    {
        "Professorer": {
            etternavn: "Pettersen",
            fornavn: "Maria",
            kontor: "A103",
            stilling: 3,
            epost: "maria.pettersen@høyskole.no",
            kursansvar: ["Fysikk 1", "Biofysikk"]
        }
    },
    {
        "Lektorer": {
            etternavn: "Selberg",
            fornavn: "Rune",
            kontor: "C103",
            stilling: 4,
            epost: "rune.selberg@høyskole.no",
            kursansvar: ["Informatikk"]
        }
    },
    {
        "Lektorer": {
            etternavn: "Dahl",
            fornavn: "Nina",
            kontor: "C306",
            stilling: 3,
            epost: "nina.dahl@høyskole.no",
            kursansvar: ["Historie"]
        }
    },
    {
        "Lektorer": {
            etternavn: "Hansen",
            fornavn: "Knut",
            kontor: "C307",
            stilling: 4,
            epost: "knut.hansen@høyskole.no",
            kursansvar: []
        }
    },
    {
        "Lektorer": {
            etternavn: "Hansen",
            fornavn: "Knut",
            kontor: "C308",
            stilling: 4,
            epost: "knut.hansen@høyskole.no",
            kursansvar: ["Geografi"]
        }
    },
    {
        "Lektorer": {
            etternavn: "Hansen",
            fornavn: "Knut",
            kontor: "C309",
            stilling: 4,
            epost: "knut.hansen@høyskole.no",
            kursansvar: ["Fysikk 2"]
        }
    },
    {
        "Lektorer": {
            etternavn: "Hansen",
            fornavn: "Knut",
            kontor: "C310",
            stilling: 4,
            epost: "knut.hansen@høyskole.no",
            kursansvar: ["IT-Grundkurs"]
        }
    }
];

// Funktion som lager HTML for én ansatt
// Tar imot index (plassering i ansatte-listen)
function lagAnsattkort(index) {
    const person = ansatte[index]; // Henter personen fra listen
    if (!person) return "<p>Ingen ansatt funnet</p>"; // Hvis index er feil
    const stilling = stillinger[person.stilling]; // Henter stilling fra sti
    // Lager kursliste (hvis personen har kursansvar)
    let kursTekst = "";
    if (person.kursansvar.length > 0) {
        kursTekst = "<ul>";
        for (let kurs of person.kursansvar) {
            kursTekst += "<li>" + kurs + "</li>";
        }
        kursTekst += "</ul>";
    } else {
        kursTekst = "<p>Ingen kursansvar</p>";
    }
}

// Returnerer HTML-struktur for ansattkortet
return `

<h3>${person.fornavn} ${person.etternavn}</h3>
<p>Stilling: ${stilling}</p>
<p>Kontor: ${person.kontor}</p>
<p>E-post: ${person.epost}</p>
<p>Kursansvar:</p>
${kursTekst}
`;
```

```


    /**
 * 6. Funksjon som henter administrasjonen (rektor, dekan, valtmester)
 */
function hentAdministrasjon() {
    let resultat = [];
    for (let i = 0; i < ansatte.length; i++) {
        if (ansatte[i].stilling === 0 || ansatte[i].stilling === 1 || ansatte[i].stilling === 2) {
            resultat.push(lagAnsattKort(i));
        }
    }
    return resultat;
}

function lagAnsattKort(index) {
    const ansatt = ansatte[index];
    return {
        fornavn: ansatt.fornavn,
        etternavn: ansatt.etternavn,
        kontor: ansatt.kontor,
        stilling: ansatt.stilling,
        epost: ansatt.epost,
        kursansvar: ansatt.kursansvar
    };
}

function leggTilUnderviser(fornavn, etternavn, kontor, stilling, epost, kursansvar) {
    ansatte.push({
        fornavn,
        etternavn,
        kontor,
        stilling,
        epost,
        kursansvar
    });
}

function slettAnsatt(index) {
    if (index >= 0 && index < ansatte.length) {
        ansatte.splice(index, 1); // Fjerner ansatt fra listen
        return true;
    }
    return false;
}


```

**Oppgave 2:** I tolking av python koden brukte jeg W3 Schools fordi Python er ikke noe jeg er godt kjent med. Selv om koden vi fikk var lett å tolke, var det noe jeg var usikker på som append, split og len. Se litteraturliste for kilde. I refaktorerering av javascript-kode så brukte jeg arbeidskrav 2 som hjelp for å finne ut av gjennomsnittet siden problemet var helt likt, bare at på denne oppgaven var det navn inkludert. Se litteratur liste for referanse til dette.

**Oppgave 3:** I denne oppgaven har jeg brukt en liten blanding av hjelpe midler. Jeg har brukt internett og CoPilot. Jeg brukte CoPilot og internett til å finne ut av hvordan man kan få «slett» knapp på den ene html-filen (admin) og ikke den andre (register).

Hvordan kan jeg få slett knappen til å dukke opp på admin.html, og ikke register.html?

```


// sjekk hvilken side vi er på
const erAdmin = location.pathname.includes("admin.html");


```

Det denne koden gjør er at den sjekker om den er på admin.html eller ikke. Dette gjør man ved å bruke location.pathname. Det denne gjør er at den finner URL som den blir gitt. Derfor legger man til includes for at den skal vite hvilken URL (eller i dette tilfelle, filen) som den skal finne. Og deretter filnavnet i parentes.

Jeg trengte også hjelp til hvordan man skulle legge til undervisere. Jeg visste at man må hente verdiene fra skjemaet. Jeg la inn disse ved å bruke document.getElementById, navnet og med value. Men når fylte inn feltene og fikk laget et ansattkort, la kursansvar seg nedover hverandre. Jeg lærte senere at jeg måtte bruke value.split(','). Dette gjør (sopm i likhet med python) lukker tomrom og gjør at bokstavene kommer ved siden av hverandre. Jeg ville også at skjemaet skulle tømme seg etter jeg hadde lagt til et ansattkort. Jeg fant ut at man kan bruke .reset() for at dette skal fungere.

Det siste jeg trengte hjelp med var å få til å slette ansattkortene. Her brukte jeg igjen CoPilot for å spørre hva jeg måtte gjøre. Funksjonen var allerede laget, så det var bare en linje med kode som jeg trengte. Det jeg trengte er det som står etter innerHTML.

Hvordan kan jeg få en knapp til å slette en artikkel/objekt?

```
document.getElementById("ansattListe").innerHTML =  
ansatte.map(_ , i) => lagAnsattKort(i)).join("")
```

Først og fremst bruker denne linjen .map. Det den gjør er at den kan «transformere» et objekt i en array. I dette tilfelle er det å fjerne/slette dette objektet. I parantesen er det en understrek. Dette er for å lage et element som ikke betyr noe i indeksen, som «i» står for. Deretter returnerer den funksjonen lagAnsattKort, slik at ikke alle ansattkortene forsvinner. Deretter .join som setter objektene sammen.

Som nevnt tidligere, så brukte jeg AI når det kom til CSS. Jeg måtte endre på noen små detaljer og noen av de ulike navnene jeg fikk av AI fordi ellers hadde ikke disse funket fordi det ikke hadde vært riktig navn på id/class.



## Referanseliste

*HTMLFormElement: Reset() method - Web APIs | MDN.* (2025, april 10). MDN Web Docs.  
<https://developer.mozilla.org/en-US/docs/Web/API/HTMLFormElement/reset>

*JavaScript String split() Method.* (u.å.). Hentet 11. desember 2025, fra  
[https://www.w3schools.com/jsref/jsref\\_split.asp](https://www.w3schools.com/jsref/jsref_split.asp)

*Jesper Østli Simensen, "Arbeidskrav 2", Oktober, 2025*

*Location: Pathname property - Web APIs | MDN.* (2023, november 26). MDN Web Docs.  
<https://developer.mozilla.org/en-US/docs/Web/API/Location pathname>

*Microsoft.* (2025). Copilot [AI-assistent]. <https://copilot.microsoft.com/>

*Python Dictionary keys() Method.* (u.å.). Hentet 11. desember 2025, fra  
[https://www.w3schools.com/python/ref\\_dictionary\\_keys.asp](https://www.w3schools.com/python/ref_dictionary_keys.asp)

*Python len() Function.* (u.å.). Hentet 11. desember 2025, fra  
[https://www.w3schools.com/python/ref\\_func\\_len.asp](https://www.w3schools.com/python/ref_func_len.asp)

*Python List append() Method.* (u.å.). Hentet 11. desember 2025, fra  
[https://www.w3schools.com/python/ref\\_list\\_append.asp](https://www.w3schools.com/python/ref_list_append.asp)

*Python String split() Method.* (u.å.). Hentet 11. desember 2025, fra  
[https://www.w3schools.com/python/ref\\_string\\_split.asp](https://www.w3schools.com/python/ref_string_split.asp)

Link til repositoriet: [https://github.com/StormyDoom/Eksamensoppgaver\\_2025\\_Programmering](https://github.com/StormyDoom/Eksamensoppgaver_2025_Programmering)