

Mappeelement 2

Intranettforside

I mappe 2 vil du finne exam2.html, exam2.css og exam2_oppgavetekst.pdf. Gjør deg kjent med filene i mappen. I dette mappeelementet behøver du kun gjøre endringer i exam2.html. I denne filen finner du HTML og noen startpunkter for JavaScript-koden som skal lages.

Målet (dersom man løser alle kravene) er å ha en intranettforside med fire seksjoner:

Regnskap

- Viser innkommende egenkapital i November måned (hvor mye penger som står på konto før November). Dette er allerede tilgjengelig i koden i exam2.html
- Viser bokføringer (innkjøp og salg) så langt i November måned, med kvitteringstekst, dato for bokføring og sum
- Regner ut en balanse (inntekter – utgifter) for alle bokføringer i November, og viser denne
- Regner ut ny egenkapital (innkommende egenkapital + balanse for November) og viser denne

Klokke

- Et enkelt område over kontakter-feltet som har en fungerende klokke

Kontakter

- En liste med kontakter, sortert alfabetisk på etternavn, som viser kontaktens navn, kontor og e-post. E-postadressen skal kunne klikkes på og åpne en ny e-post ved hjelp av mailto:-protokollen

Oppgaver

- En seksjon med en liste av oppgaver.
- Hver oppgave inneholder en oppgavetekst og en status.
- Hvis status er «To do», skal man kunne klikke på statusen. Når den klikkes, skal statusen for oppgaven oppdateres til «Done».
- PS: Det finnes klasser i exam2.css som kan brukes for å oppnå riktig farget bakgrunn for status-visningen. Se eksempel på bruk i HTML-koden i exam2.html.
- En seksjon med et skjema hvor brukeren kan skrive inn en ny oppgavetekst og lagre oppgaven i oppgavelisten.
- Hvis det ikke er skrevet noe i feltet for oppgavetekst, skal brukeren gis en beskjed om at det må fylles tekst i feltet for at oppgaven skal kunne lagres.
- Når en ny oppgave lagres, skrives den ut i listen sammen med de eksisterende oppgavene.

En video er tilgjengelig som viser hvordan nettsiden oppfører seg når man har løst alle kravene. Denne kan sees her:

<https://hiof.cloud.panopto.eu/Panopto/Pages/Viewer.aspx?id=421c6acc-bdfa-499e-b3ac-b22e00d6a3aa> .

Krav

Disse funksjonene er formulert som krav med stigende vanskelighetsgrad for å oppnå ulike karakternivåer. Alle kravene kan løses med de konseptene som er undervist og er tilgjengelig gjennom Webtricks LMS, med mindre annet er spesifisert.

Felles krav uavhengig av nivå:

1. All kode som lages skal kommenteres slik at studenten viser forståelse for hva som er kodet. Studenten kan velge å skrive dette i et eget dokument dersom det oppleves rotete med så mange kommentarer i koden.
2. Alle kilder og hjelp skal beskrives og linkes. Hvis det er kilder som løser enkeltproblemer kan dette legges som kommentarer på relevante steder i koden, ellers står studenten fritt til å lage et eget dokument som samler dette.

Minstekrav

1. Bytt ut de hardkodete oppgavene (linje 42-49 i exam2.html) med en dynamisk utskrift av oppgavene som ligger lagret i todos.
2. Gjør e-postene i kontakt-seksjonen til klikkbare linker som bruker mailto:-protokollen for å åpne en ny e-post når du klikker på en e-postadresse.

D-krav

1. Når knappen «Lagre» i Ny oppgave-seksjonen klikkes skal det lagres en ny oppgave i todos.
2. Bruk internett til og finne JavaScript som lager en fungerende digital klokke. Dette kan være kodeeksempler, tutorials eller github-kontoer. Implementer denne koden i exam2.html, og sørg for at klokke-seksjonen viser den fungerende klokken.
3. Klokken skal minimum vise timer og sekunder i klokkeslettet. Det er helt greit om den også viser sekunder, men dette er ikke et krav.
4. Inkluder en kommentar med link til kildene du har brukt for å lage klokken
5. Studenten viser god selvstendighet rundt løsningen av D-kravene

C-krav

1. Kontakter skal sorteres alfabetisk på etternavn før de skrives ut. (Her kan det hende dere må google litt!)
2. Lag kode som skriver ut bokføringene i books i listen med bokføringer. HTML-strukturen skal følge dette formatet:

```
<div class="bookline">  
  <span class="booktext">Kvitteringstekst</span>  
  <span class="bookdate">01.01.24</span>  
  <span class="bookamount">123 kr</span>  
</div>
```

3. Oppdater D-krav 1 slik at det ikke kan lagres en oppgave med mindre det står noe i input-feltet.
4. Gi brukeren en beskjed/tilbakemelding i henhold til C-krav 3.
5. Studenten viser god selvstendighet rundt løsningen av C-kravene

B-krav

1. Sørg for at listen med oppgaver blir oppdatert så fort brukeren legger til en ny oppgave
2. Bruk bokføringer for å regne ut balansen for november
3. Gjør nødvendige grep for å regne ut ny egenkapital
4. Vis balanse for november og ny egenkapital på riktige steder i intranettet
5. Studenten viser svært god selvstendighet rundt løsningen av B-kravene

A-krav

1. Oppgaver med status «To do» skal kunne endre status til «Done». Når de klikkes på. Oppgaver med status «Done» skal *ikke* endres tilbake til «To do» når de klikkes på.
2. Sørg for at grensesnittet (listen med oppgaver) oppdateres umiddelbart når en status endres fra «To do» til «Done»
3. Studenten viser svært god selvstendighet rundt løsningen av A-kravene