

1 **DAT310 Info Trial 2019**

Prøveeksamen inneholder 14 oppgaver. Du kan samle 79 poeng totalt.
Prøveeksamen har ikke omfang til et helt eksamen men skal besvares innen **3 timer**.

Alle hjelpemidler er tillat. Dette inkluderer både programmer på PCen og nettsider.
Husk at om du bruker mye tid for å søke løsninger på nettet blir du kanskje ikke ferdig.

Eksamen inneholder to deler.

Del 1 inneholder oppgaver og spørsmål der du skal svare i de feltene i oppgaven.
Flervalgsoppgaver gir poeng på følgende måte:

- Riktig svar gir 2 poeng
- Feil svar gir -1 poeng

Del 2 inneholder flere programmeringsoppgaver relatert til en eksempel applikasjon.
Start filer er gitt som en zip arkiv.
Last ned start filene og bruk din IDE for å løse oppgavene.

Oppgavene inkluderer også svarfelder.
Om du kopierer kode fra net resurser må du oppgi URLen her.

For å levere svar for del 2 skal du last opp et enkelt zip arkiv, som inneholder både start filene og din kode.
Mulighet for opplastning er i slutten av eksamen.
Kode som er lagt til i svar feltene til enkelte oppgaver vil kun bli sett på når ingen zip arkiv er tilstede. Dette skylle være en siste utvei.

Hvis du har kommentarer, kan du skrive dem her

Maks poeng: 0

2 **CSS guess**

Hvilken CSS-attributt kan brukes til å permanent vise en ”sidebar” på en nettside? (2 poeng)
Velg ett alternativ

- ☐ relative
- ☐ fixed
- ☐ static
- ☐ absolute



Maks poeng: 2

3

CSS guess

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <title>CSS guess</title>
  <style>
    * {color: yellow;}
    .container p {color: red;}
    #main {color: blue;}
  </style>
</head>
<body>
  <div id="main" class="container">
    <p id="foo" class="bar boo">Something clever goes here</p>
  </div>
</body>
</html>
```

Hvilken farge vil teksten være? (2 poeng)

Velg ett alternativ

- ☐ black
- ☐ yellow
- ☐ blue
- ☐ red



Maks poeng: 2

4

JS what

```
var a = ["Stef", "Amit"];
a.push("Brian");
a.unshift("Morgan");
a.pop();
a.shift();
a.sort();
a.splice(1, 0, "Victoria");
```

Hva er innholdet av array "a" etter å ha kjørt JavaScript-koden? (3 poeng)

a = [(Amit, Victoria, Stef, Amit", "Victoria", "Stef)]

Maks poeng: 3

5 HTTP request guess

What is sent along as part of the HTTP request? (2 points)
Velg ett alternativ

- ☒ Cookie
- ☐ Session
- ☐ Both of them
- ☐ Neither of them



Maks poeng: 2

6 JS guess

What will be the output? (2 points)

```
var x="5", y="10";  
console.log(x+y);
```

Velg ett alternativ

- ☐ 15
- ☒ 510
- ☐ 50
- ☐ 105



Maks poeng: 2

7 HTML guess

Hvorfor er "hidden" formfelter nyttige? (2 poeng)

Velg ett alternativ

- ☒ fordi de kan holde på data til senere bruk
- ☐ fordi deres verdier er unike
- ☐ fordi de holder seg ute av veien
- ☐ fordi det du ikke vet har du ikke vondt av



Maks poeng: 2

8

AJAX guess

Hvilken av disse er sant om AJAX (2 poeng)
Velg ett alternativ

- ☐ Flere kall kan ikke gjøres samtidig.
- ☒ Låser ikke siden mens kallet foregår.
- ☐ Svarene til flere kall returnerer i rekkefølgen kallene ble sent.
- ☐ Svarene blir alltid gitt i JSON format.

Maks poeng: 2

9

POST guess

Hvilken av disse gjelder for **POST** requests
Velg ett alternativ

- ☒ Bør brukes når man håndterer sensitive data
- ☐ Kan bokmerkes
- ☐ Har lengdebegrensninger
- ☐ Data vises i nettadressen

Maks poeng: 2

10

Session example

Gi et eksempel for hva slags data som gjerne lagres i en session og hva dette brukes til. (4 poeng)
Skriv ditt svar her...

Maks poeng: 4

11 HTML coding

Subscription

Period:

Half year

Comment:

Subscribe

Lag formen ovenfor i HTML. (8 poeng)

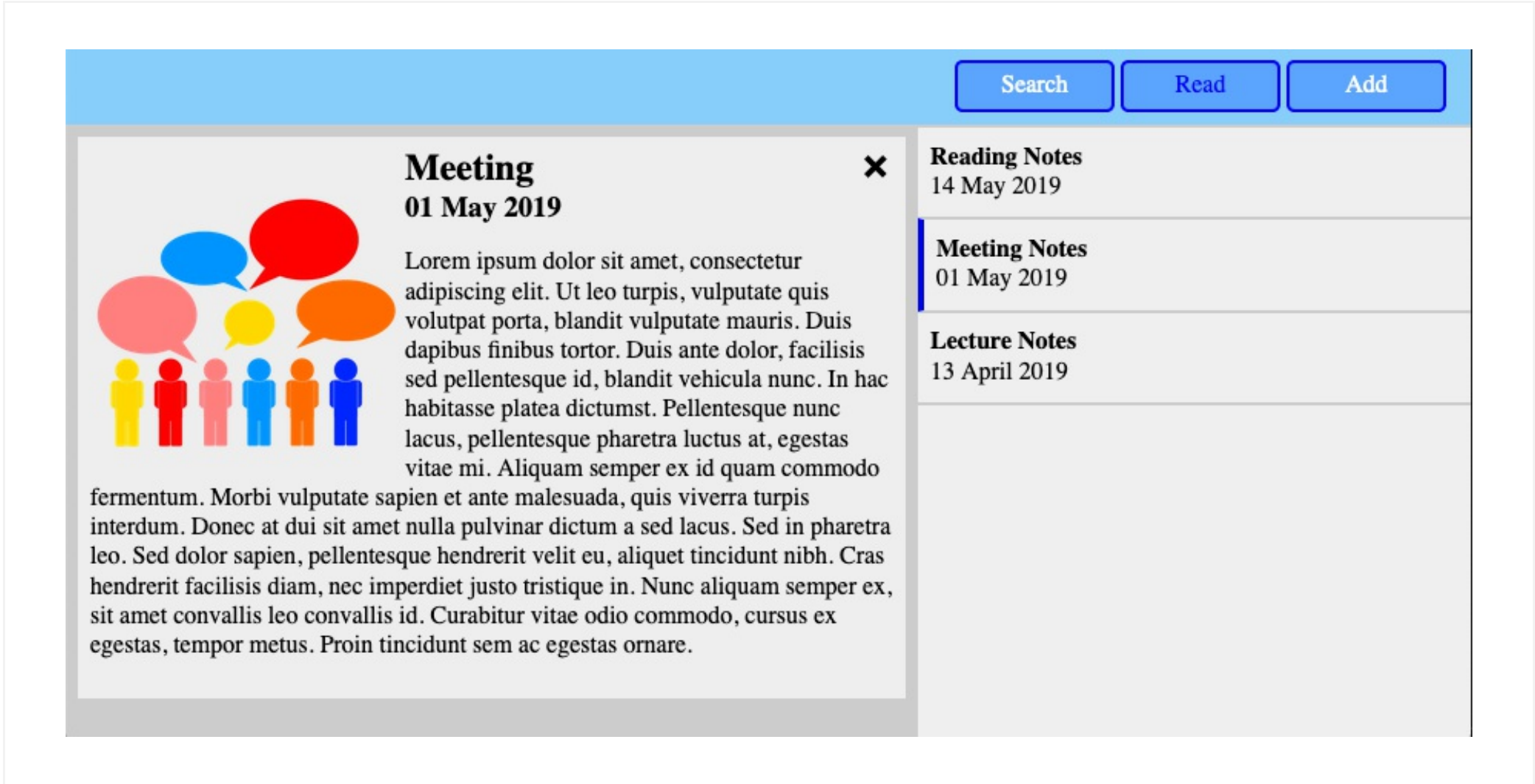
- Grupper feltene sammen i en fieldset kalt "Subscription".
- Bruk <label> til å tildele feltnavn (labels) til input feltene.
- Period er en dropdown liste med tre valg: "Half year", "One year", og "Two years". De samsvarende verdiene er lengden av perioden i måneder, altså, 6, 12, and 24. Standard valget er "Half year".
- Kommentar-feltet har navnet "comment". Den er 20 karakterer bred og 3 linjer høy.
- Submit knappen har label "Subscribe".
- Du trenger ikke bekymre deg om justering av feltene eller label.
- Du trenger ikke å lage hele HTML siden, bare form delen av den.

Write HTML code here

Maks poeng: 8

i

My Notes



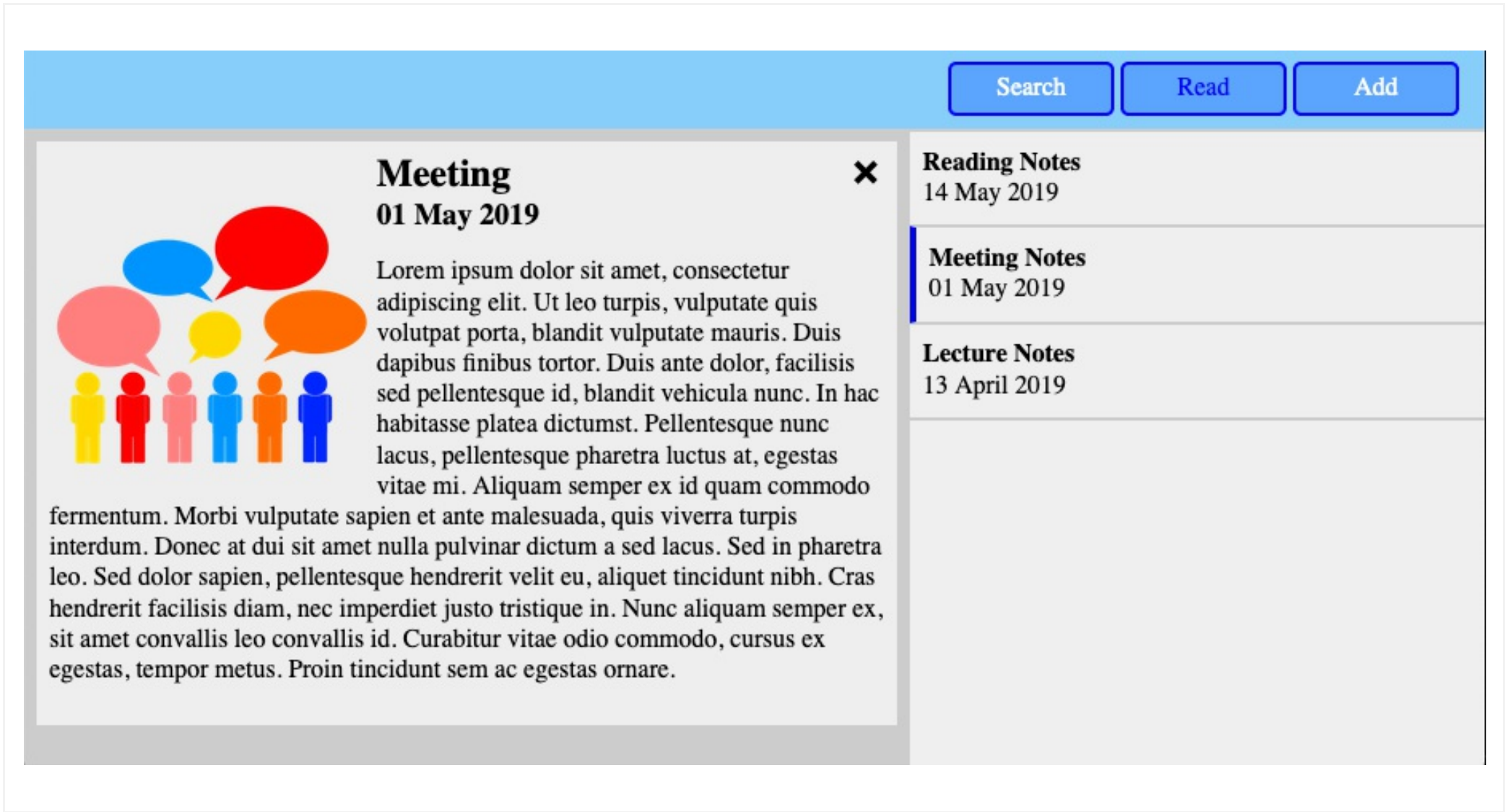
Del 2: I de neste oppgaver skal du fullføre noe kode i en eksempel applikasjon. Applikasjonen lar brukeren lese og opprette notater, e.g. forelesningsnotater, møtenotater, ...

Last ned startfilene under.

[starterFiles](#)

12

CSS



Skriv CSS for å stile siden som vist ovenfor. (15 poeng)
HTML filen finner du i exam.zip under files/static/index_static.html.
Skriv CSS coden i files/static/style.css og last denne opp som del av zip filen.

Vær sikkert å få med følgende.
Instruksjoner som er *kursivt* er allerede implementert i files/static/index_static.html

Baggrunnsfarger:

- 1. *Hele siden har baggrunnsfarge #cccccc.*
- 2. *Innhold i artikkelen (article) og aside med listen over notater (note-list) har baggrunnsfarge #efefef.*
- 3. *Headeren har baggrunnsfarge #88baf9.*
- 4. *List elementene i nav-list har en baggrunnsfarge #5ca5ff.*

Header:

- 1. *Headeren er alltid minst 50px høy*
- 2. *Nav listen har 1em avstand fra høyre siden.*
- 3. *List elementene har en mørkeblå (#0000ff) ramme (boarder).*
- 4. *Headeren spanner hele bredden og er i toppen av siden.*
- 5. *Elementene i nav listen er plassert ved siden av hverandre.*
- 6. *Rammen på nav list elementene har runde hjørner.*
- 7. *Teksten i nav-list elementene er horisontal sentrert i rammen.*
- 8. *Nav-list elementene er vertikal sentrert i headeren.*
- 9. *Teksten i lenkene <a> har ikke en linje under.*
- 10. *Teksten i nav-list elementet med klasse "showing" er mørkeblå, ikke hvit.*
- 11. *På hele nav-list elementet viser cursoren en hånd, slik den vanslighvis gjør for lenker.*

Posisjonering:

- 1. *Aside listen har alltid bredden 360px.*
- 2. *Aside listen er lokalisert på høyre side og strekker seg helt fra headeren til bunnan av siden.*
- 3. *Artikkelen er lokalisert til venstre av aside, såm på bildet.*
- 4. *Om skjermen er for liten er det mulig å scrolle i artikkel og i listen uavhengig. (Se flere screenshots i files/statis/screenshots).*

Artikkelen:

- 1. *Bildet i artikkelen med klasser "icon" og "main" vises oppe til venstre og teksten flyter rundt bildet.*
- 2. *Det er 5px avstand mellom bildet og teksten.*
- 3. *Det lille krusset (.icon.exit) er lokalisert oppe til høyre i artikkelen.*
- 4. *Når musa ligger over det lille krusset (.icon.exit) har dette en lyseblå baggrunnsfarge (e.g. #87cefa).*

Notat listen:

- 1. *List elementene i note-list har ingen punkter forran.*
- 2. *Under siste og over første elementet i listen er den gråe rammen (border) 2px tykk, ikke bare 1px.*
- 3. *List elementet med klassen "showing" har i tillegg en 2px ramme (boarder) på venstre siden med fargen #0000ff.*
- 4. *Når musa ligger over et element i note-listen forandres bakgrunnsfargen til lyseblått (e.g. #e1ebf7)*

List kilder du bruker her.

Maks poeng: 15

13

Jinja

Fullfør files/template/index.html. (5 poeng)

I files ligger en flask applikasjon app.py.
Denne inneholder en funksjon index() som bruket er template index.html.
Du skal lage templatene index.html slik at

- 1. Den utvider templatet layout.html
- 2. Den bruker en for løkke for å vise notatene "notes".
- 3. Resultater er lik som static/index_template_goal.html (inkludert id'er).

List kilder her

Maks poeng: 5

14

JS programming

Fullfør filen files/static/script.js (15 poeng)

Initialization:

- 1. Call funksjonen *init* når HTML dokumentet er lastet inn.
- 2. Fullfør funksjonen *init*:
 - For hvert list element i note-list legg til en event listener som utfører *listClick* funksjonen på click.
- 3. Fullfør funksjonen *listClick*.
 - Den eksisterende kode legger til klassen "showing", finner note_id fra list-elementets id og oppretter et artikkel element med correct id. (Det er ikke nødvendig, men du får gjerne forandre eller flytte coden i *listClick*.)
 - 1. Gjør et AJAX kall til /note/<note_id> for å få innhold til artikkel elementet.
 - 2. Legg til innholdet i artikkel elementet og legg til artikkelen som første barn i container div elementet.
 - Det skal være mulig å vise flere artikler. Da den som ble lagt til sist skal vises øverst.
- 4. Fullfør funksjonen *removeArticle(note_id)*.
 - Denne funksjonen utføres når brukere clicker på krysset i artikkelen.
 - 1. Fjern artikkelen som hører til dette krysset fra DOMen.
 - 2. Fjern klassen "showing" fra listNote elementet som hører til denne artikkelen.

List kilder du har brukt.

Maks poeng: 15

15

Flask programming

Programmer funksjonalitet i Flask appen for å legge til notater (15 poeng)

Du skal implementere funksjonen `add()` i `/files/app.py`.
Den eksisterende funksjonen returnerer en html side som viser et enkelt sjema (form) for å lage et nytt notat.
Denne formen sendes til samme route men med en POST request.
Du skal implementere følgende funksjonalitet:

1.

På en GET request skal funksjonen fortsatt returnere siden med sjemaet.
2.

På en POST request skal du prosessere formen på følgende måte:

1.

Sjekk om feltene *title*, *date*, og *imageUrl* er fylt. Om et av disse mangler returner igjen formen.

2.

Lag et nytt dictionary fra formen som representerer det nye notatet og ligner på dictionaries som brukes i `index` funksjonen.

■

Legg til *content*, *imageUrl*, og *title* fra formen.

■

Legg til *date* ved å konvertere form verdien til et *datetime* objekt.

3.

Lagre det nye notatet (dictionary) i en session. Ikke glem:

■

Still sikkert at appen kan bruke en session.

■

Før du legger til notatet i session, legg til et *id* felt med en ny unik id.

4.

Til slutt oversend brukeren til hovedsiden.

5.

Oppdater funksjonene *index* og *getNote(note_id)* slik at de også returnerer/viser notater som er lagret i sessionen.

List kilder du har brukt.

Maks poeng: 15

16

Upload

Lag et zip arkiv av hele *files* mappen. Last opp arkivet her:

Maks poeng: 0

9/9