

Seksjon 1

Question	Question title	Marks	Question type
1	JS coding - todolist	24	Programming
2	CSS My Notes	15	Programming
3	jQuery coding timer	10	Programming

1 **JS coding - todolist**

Implement additional functionality for a todo-notes application. (24 points)

In this exercise you have to implement additional functionality for a JS application for creating a todo-list. The current application allows the user to add new entries with title, date and content to a TODO list.

You can find the todo-list application in the zip arkive in folder **todolist/**.
You have to add code to both index.html, todos.js and possibly a new CSS rule.

Implement the following additional functionality:

Parse multiple todo point from the input: (See Image 4 and 5 below.)

- Add the following placeholder to the *Todo* textarea in the addentry form:
 - "Separate todos by semicolon: todo1; todo2; ..."
- The instruction should show when the textarea is empty.
- Update the function *Entry()* constructor in **todos.js** to split the *content* by semicolon, resulting in an array of todo points.
- Update the function *displayEntry*(idx) in **todos.js** to display the different elements from the *content* array as an unordered list.
- Make sure no empty list elements are displayed.

Add Categories: (See Image 1 and 2 below for an example)

- Add a dropdown select to the AddTodo form. The select should have a label *Category* and allow to select one out of the categories *home*, *work*, *other*.
- When adding a new entry, the category should be stored in the *category* of the new object.
- The category should be displayed as a div element with class *todo_category*.
- Todo entries with category *home* should have a yellow background and border.
- Todo entries with category *work* should have a blue background and border.

Tip: add classes *work* and *home* to CSS.

Add data:

- Update the **data.js** file. Add two more entries to the *todos* array with categories *work* and *other*.

List sources here

Image 4: Adding a Todo with bullet points

Image 1: *category* select dropdown



Image 2: colored todo elements

Add Todo

Title:

Category:

Other

Date:

mm / dd / yyyy

Todo:

Separate todos by semicolon: todo1; todo2; ...

Add

Cleaning

16/05/2019

#home

• clean living room

• clean kitchen

A work todo

29/05/2019

#work

Another todo

30/05/2019

#other

Add Todo

Title:

Todo with bullet list

Category:

Other

Date:

05 / 30 / 2019

Todo:

Point 1; point2; point3;

Add

Image 5: The Todo with bullet points

Todo with bullet list

30/05/2019

#other

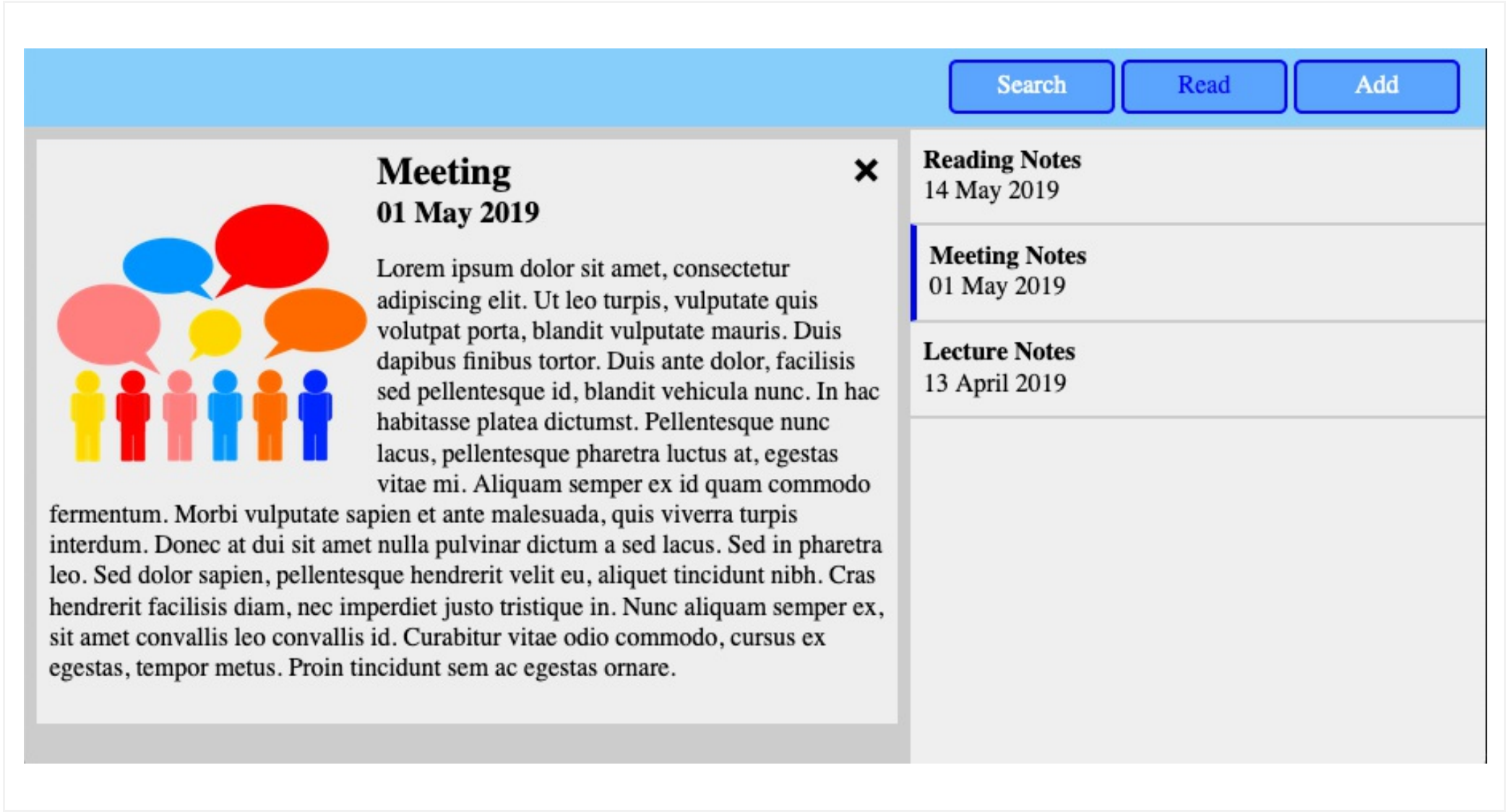
• Point 1

• point2

• point3

Maximum marks: 24

2 CSS My Notes



Write CSS rules to stile the page shown above. (15 points)

You find the HTML file in my_notes_css/index.html.

Write the CSS code in my_notes_css/style.css.

Be sure to include the following:

Instructions that are *italic* are already implemented.

Background color:

- 1. *The complete page has background color #cccccc.*
- 2. *The content in the article and aside with the list of notes (note-list) has background color #efefef.*
- 3. *The header has background color #88baf9.*
- 4. *List elements in the nav-list have a background color #5ca5ff.*

Header:

- 1. *The header is always at list 50px high.*
- 2. *The Nav list has 1em distance from the right side of the page.*
- 3. *List elements in the nav-list have a blue (#0000ff) boarder.*
- 4. The header should be in at the very top of the page and reach from left end to right end.
- 5. List elements in the nav list are placed horizontally.
- 6. The border on nav list elements has rounded edges.
- 7. The text in nav list elements is horizontally centered.
- 8. Nav-list elements are vertically centered in the header.
- 9. The text in links (<a>) is not underlined.
- 10. The text in the nav-list element with class "showing" is blue (#0000ff), not white.

Positioning:

- 1. *The aside list has always a width of 360px.*
- 2. The aside list is located on the right side and stretches from the header to the bottom of the page.
- 3. The article is positioned to the left of aside, as in the image.

Article:

- 1. The image in the article that has class "main" is displayed in the upper left corner, with text and heading flowing around it.
- 2. There is a 5px distance between the image and the text.
- 3. The small X (with class "exit") is located in the upper right corner of the article.
- 4. When the mouse pointer is located over the X then it has a blueish background color (#87cefa).

Note list:

- 1. List elements in the note-list have not bullet point before.
- 2. Under the last and above the first element in the list, the grey borrderr is 2px thick, not just 1px.
- 3. List elements with the class "showing" have an additional 2px border on the left side with blue color (#0000ff).
- 4. While the mouse pointer lies above an element in the note-list, this element's background color is changed to #e1ebf7.

Fill in your answer here

Maximum marks: 15

3 **jQuery coding timer**

Implement additional functionality for the jQuery timer. *(10 points)*
In this exercise you have to implement additional functionality for a jQuery timer.
You can find the timer application in the zip arkive in folder **timer/**.

Add the following functionality:

Buttons: *(see images 1 and 2 below)*

- Add a *cancel* button to the timer. When clicked, the timer disappears and the countdown form appears, allowing to specify a new timer.
- Add a *stop* and *continue* buttons to the timer. When clicked the *stop* button halts the countdown. The *continue* button allows to continue the countdown.
- When the countdown is running, the *continue* button should be hidden.
- When the countdown is stopped, the *continue* button is visible but the *stop* button is hidden.

Countdown title *(see images 3, 4 and 5 below)*

- Add a form field to the countdown form that allows to specify a title for the countdown.
- Convert the title to only capital letters (upper case) and display it as part of the countdown.
- When the countdown reaches 00:00 display the title to the user in an alert.

You can but you do not have to use jQuery.

List kilder her

Image 1: *stop* and *cancel* buttons

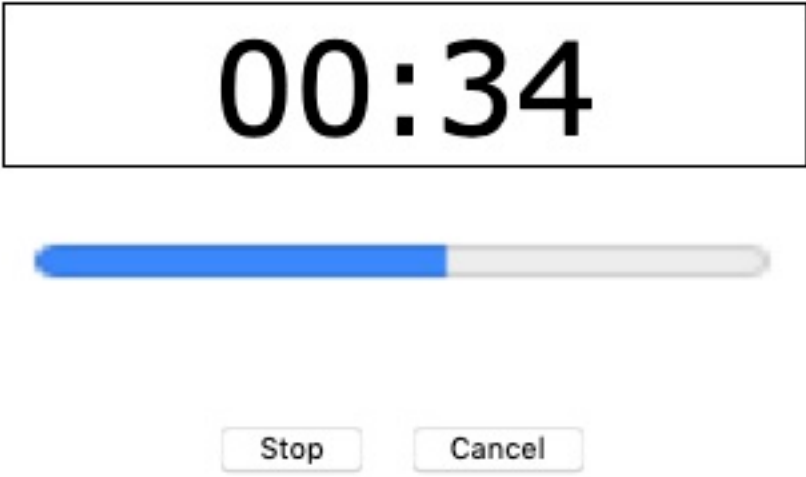


Image 2: *continue* and *cancel* buttons

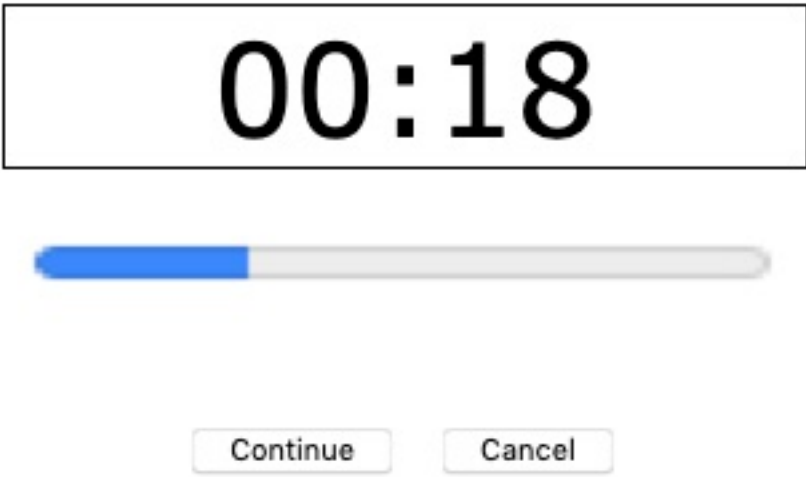


Image 3: *Countdown form* with title form field

1 minute 

Go!

Image 4: *Countdown title* displayed



Image 5: Alert



Maximum marks: 10