Lesson 5
Apply Intelligence with JavaScript

Daniel Ren

2020-08-18

Outline

- Review
- Thinking Time
- Knowledge
- Exercise
- Assignment

Review

Work with CSS:

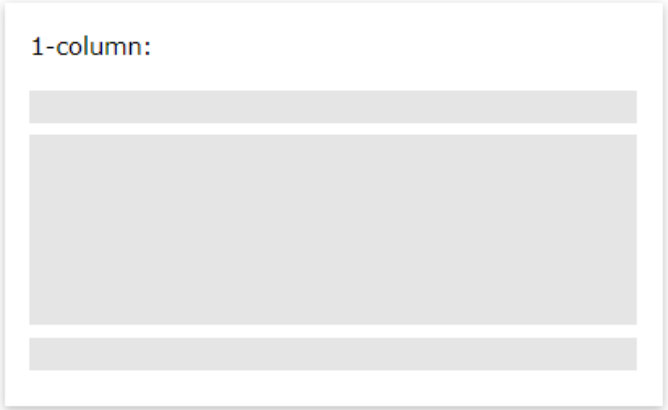
- Position of elements
- Layout of page

Review

- **1-column** (often used for mobile browsers)
- **2-column** (often used for tablets and laptops)
- **3-column layout** (only used for desktops)

Review

1-column:



```
<div class="Header">
    Header
</div>
<div class="SingleColumn">SingleColumn</div>
<div class="Footer">
    Footer
</div>
```

```
.SingleColumn{
    background-color: teal;
    text-align: center;
    padding: 20px;
    border: 4px solid white;
    height: 500px;

}
```

Review


```
<div class="Header">
    Header
</div>
<div class="ContentRow">
<div class="ContentColumnLeft">ContentColumnSide</div>
<div class="ContentColumnRight">ContentColumnSide</div>
</div>
<div class="Footer">
    Footer
</div>
```

```
.ContentColumnLeft{
    display: table-cell;
    text-align: center;
    background-color: teal;
    padding: 20px;
    border: 4px solid white;
    width: 50%;
}
```

```
.ContentColumnRight{
    display: table-cell;
    text-align: center;
    background-color: teal;
    padding: 20px;
    border: 4px solid white;
    width: 50%;
}
```

Knowledge

We have two types of CSS.

- External CSS

CSS is put in an external file and linked it to the HTML

- Internal CSS

CSS is put in the <head> element of HTML.

Note: External CSS is the best practice.

Knowledge

Selectors: link the style to the elements in HTML.

Selector	Example	Description
.class	. SingleColumn	Selects all elements with class="SingleColumn"
#id	#FirstLine	Selects the element with id="FirstLine“
element	p	Selects all <p> elements

Exercise

Let’s try the selectors.

Thinking Time

How can we make our page more interactive or intelligent?

Knowledge

JavaScript or ECMAScript is a programming language that helps you add interactivity to your webpage.

Example: When you select a button, JavaScript is the code that defines the event or behaviour that will happen, such as open a popup window.

Exercise

Write a hidden message:

1. Create a folder “JavaScripts” in your web folder
2. Create app.js in the folder “JavaScripts”
3. Link app.js to our HTML page.
4. Add code in your app.js
5. Test the page

Exercise

Write a hidden message:

3. Link app.js to our HTML page by adding the following code in you index.html

Exercise

Create your first magic button:

4. Add code in your app.js

JavaScripts > JS app.js

```
1   'use strict'
2
3   // a hidden console message
4   console.log('Here\'s a hidden message.')
5
```

By activate strict mode, it reduces silent errors, improves performance, provides more warnings, and fewer unsafe features

A hidden message that won't appear on your webpage. However, what you write in the console will show up in the browser developer tools. Using *console messages* can be really helpful for seeing the result of our code.

Exercise

Write a hidden message:

5. Test the page

Assignment

Use different browsers to test your script.

The attached file “Finding Your Browser's Developer Console.pdf” can help you.