

2020-2

웹 시스템 설계 학습과제

Week 3

HTML & CSS(2)

Composed by:

WISE Research Lab Ajou University



AJOU UNIVERSITY

WISE

학습목표

1. HTML5 의 semantic element 를 이해하고 활용한다.
2. CSS 의 속성(property) 및 media query 를 이해하고 활용한다.

HTML(2주차에 이어서)

1. Semantic Elements in HTML5 ¹

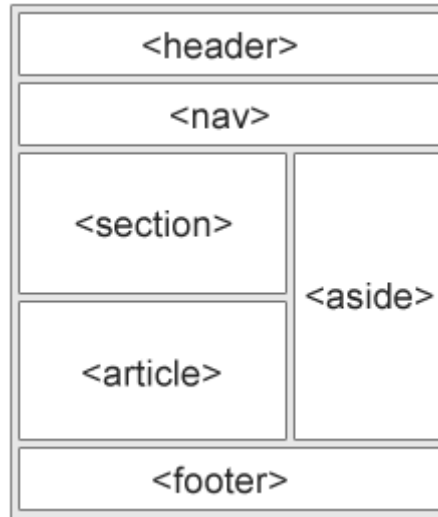


그림 1 Semantic Element를 사용한 HTML의 구조의 예

- Semantic Element란 특정한 의미를 가진 element를 의미함
- Semantic Element를 사용하는 이유
 - ✓ HTML5 이전에는 문서의 구조를 정의할 때, element의 종류가 다양하지 않아서 용도와 의미에 따라 다양한 태그를 사용하지 않았음 (주로 div 만을 사용하여 구조화)
 - ✓ Element의 종류에 따라서 각 Element의 태그가 구조적으로 어떤 의미를 가지는지 브라우저와 개발자 모두에게 명시하기 위해 Semantic Element가 도입됨
- Semantic Element의 예
 - ✓ <header> : 사이트의 소개와 관련된 콘텐츠 (로고, 메뉴 등을 포함)
 - ✓ <nav> : 사이트를 탐색할 수 있는 각종 링크의 세트를 정의하는 네비게이터 영역
 - ✓ <section> : 제목이 있는 주제별 콘텐츠 그룹을 구분
 - ✓ <article> : 문서의 다른 내용과 독립적으로 읽을 수 있는 독립적인 콘텐츠
 - ✓ <aside> : 문서의 주된 내용이 아닌 일부 콘텐츠의 영역 (사이드 바)
 - ✓ <footer> : 문서 또는 섹션의 끝 내용을 정의함 (작성자/저작권 정보, 이용 약관 등)

¹ https://www.w3schools.com/html/html5_semantic_elements.asp

CSS(2주차에 이어서)

1. CSS 기본 속성

- Font

```
body {  
  color: red;  
  font-size: 16px;  
  text-align: center;  
}
```



그림 2 body 내 font 속성 설정 예시

- ✓ color 속성: 해당 Element 하위에 있는 텍스트의 색상 지정
 - red, blue 등 미리 정의된 표준 색상을 사용할 수 있으며, 이외의 색상은 RGB 컬러코드를 직접 작성해야 함 (#FF00FF, #FFFFFF, #000000 ...)
 - 표준 색상의 종류는 Figure 2 의 표 참고
- ✓ font-size: 폰트의 크기 지정.
- ✓ text-align: 폰트의 정렬 지정 (right, left, center)

aqua	black
blue	fuchsia
gray	green
lime	maroon
navy	olive
purple	red
silver	teal
white	yellow

그림 3 표준으로 지정된 색상 표²

- Element의 크기(width, height)와 정렬 위치(float) 지정
 - ✓ width(너비)와 height(높이) 속성값으로 element의 크기를 정의할 수 있음
 - ✓ CSS에서 크기를 정의할 때 여러 가지의 단위를 사용할 수 있도록 지원함

² http://tcpschool.com/css/css_basic_colors

- 고정 크기 단위
 - px: 모니터의 해상도에 따라 화소(픽셀) 단위로 크기 지정
 - pt: point(1/72 inch) 단위로 크기 지정
- 상대적 크기 단위
 - em: 부모 Element 에서 지정한 글자 크기를 기준으로 배율 조정
 - rem: Root element 인 <html>에서 지정된 크기를 기준으로 배율 조정
- ✓ float: 좌측 정렬 또는 우측 정렬로 Element 를 배치함 ³
 - float:left
 - float:right
 - 지정한 값이 없을 경우 none 이 기본값으로 설정 됨
- ✓ clear: float 를 적용한 이후에 float 의 영향을 받지 않는 Element 를 배치할 때 사용 ⁴
 - clear:left
 - clear:right
 - left, right 을 모두 제거하려면 clear:both 를 사용함

³ <https://developer.mozilla.org/ko/docs/Web/CSS/float>

⁴ <https://developer.mozilla.org/ko/docs/Web/CSS/clear>

```

<!DOCTYPE html>
<html>
  <head>
    <style>
      .small-box {
        width : 120px;
        height : 30px;
      }
      .large-box {
        width : 240px;
        height : 60px;
      }
      #h {
        float : left;
        background-color : red;
      }
      #f {
        float : right;
        background-color : blue;
      }
    </style>
  </head>
  <body>
    <header id="h" class="large-box"></header>
    <footer id="f" class="small-box"></footer>
  </body>
</html>

```

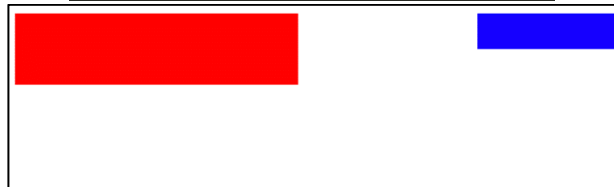


그림 4 width, height와 float 지정 예제

- Margin
 - ✓ Element 의 외부 여백(다른 element 와의 간격)을 지정
 - ✓ margin: auto
 - 해당 Element 가 속한 컨테이너 영역에서 Element 를 중앙으로 정렬
 - ✓ margin: 10px
 - Element 의 상하좌우의 간격을 일정하게 지정
 - ✓ margin: 10px 5px 10px 5px
 - Element 의 상하좌우 간격을 각각 지정함 (top right bottom left 순서)
 - ✓ margin-top, margin-right, margin-bottom, margin-left
 - 특정 방향에 대한 margin 만을 지정함
- Padding
 - ✓ Element 의 내부 여백을 지정
 - ✓ padding 의 지정 방법은 margin 과 동일한 방식

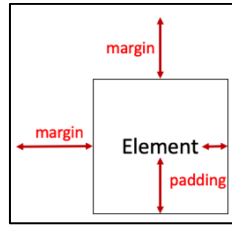


그림 5 margin과 padding의 차이점

```
<!DOCTYPE html>
<html>
<head>
<style>
body, .box {
margin : 0;
padding : 0;
}
.box {
float : left;
text-align : center;
width : 280px;
height : 200px;
border : 2px solid black;
}
.box-margin {
/*표기순서 (top right bottom left)*/
margin : 40px 20px 0 20px;
}
.box-padding {
padding-bottom : 40px;
}
</style>
</head>
<body>
<section class="box"><h1> Box without margin and padding</h1></section>
<section class="box box-margin"><h1>Box with margin</h1></section>
<section class="box box-padding"><h1>Box with padding</h1></section>
</body>
</html>
```

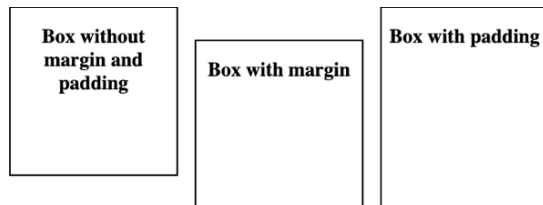


그림 6 margin, padding 활용 예제

2. Media Query ⁵

- 해당 문서를 열람하는 미디어 장치의 종류 또는 해상도 정보에 따라 레이아웃을 다르게 작성할 수 있음
 - ✓ 단일 웹페이지만으로 PC, 스마트폰, 태블릿 등 다양한 해상도를 가진 디스플레이에 대해 여러 형태의 레이아웃을 제공할 수 있음
 - ✓ 즉, Media query 를 활용하여 반응형 웹(Responsive Web)의 구현이 가능함
- Media query를 작성하는 방법
 - ✓ Desktop First: 큰 화면->작은 화면 순서로 Media query 를 작성하는 방식
 - 데스크탑(큰 해상도를 가진 장치)의 접속이 더 많은 경우 유리함

⁵ https://developer.mozilla.org/ko/docs/Web/Guide/CSS/Media_queries

```
/* 1024px 이상의 해상도를 가진 장치를 위한 CSS*/
.grid-example {
  width : 25%;
  height : 160px;
  text-align : center;
  float:left;
  color : white;
}

/* 1024px 미만, 768px 이상 해상도를 가진 장치를 위한 CSS*/
@media(max-width : 1023px) {
  .grid-example {
    width : 50%;
    height : 120px;
  }
}

/* 768px 미만 해상도를 가진 장치를 위한 CSS*/
@media(max-width : 767px){
  .grid-example {
    width : 100%;
    height : 80px;
  }
}
```

- ✓ Mobile First: 작은 화면->큰 화면 순서로 Media query 를 작성하는 방식
 - 모바일 장치(작은 해상도를 가진 장치)의 접속이 더 많은 경우 유리함

```
/* 768px 미만 해상도를 가진 장치를 위한 CSS*/
.grid-example {
  width : 100%;
  height : 80px;
  text-align : center;
  float:left;
  color : white;
}

/* 1024px 미만, 768px 이상 해상도를 가진 장치를 위한 CSS*/
@media(min-width : 768px) {
  .grid-example {
    width : 50%;
    height : 120px;
  }
}

/* 1024px 이상의 해상도를 가진 장치를 위한 CSS*/
@media(min-width : 1024px){
  .grid-example {
    width : 25%;
    height : 160px;
  }
}
```



그림 7 1024px 이상의 화면에서 보이는 레이아웃

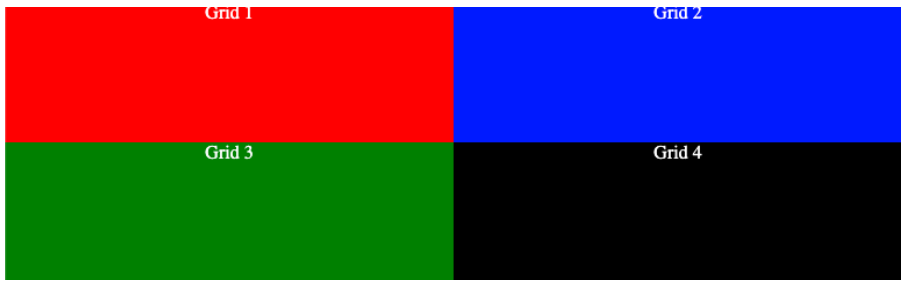


그림 8 1024px 미만, 768px 이상의 화면에서 보이는 레이아웃

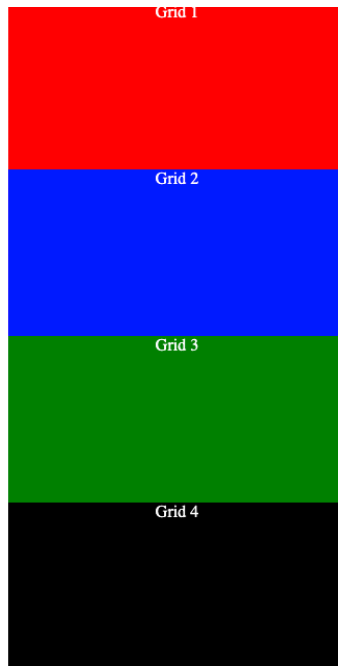


그림 9 768px 미만의 화면에서 보이는 레이아웃

학습과제

1. 과제 개요

- Semantic 태그들을 사용하여 HTML 문서를 구성하고, 문서에 CSS를 적용시켜 반응형 웹 페이지를 만들어본다.

2. 학습과제 파일 구성 및 과제 제출 양식

- Lab3.html : HTML 구조를 만들고, Lab3.css를 불러와서 적용시킨 HTML 문서
- Lab3.css : Desktop화면과 Mobile 화면에 대응하는 각 Element의 크기, 색상, 폰트 등의 style 속성값을 정의한 파일
- 위의 파일을 하나의 폴더내에서 생성하고, 폴더(프로젝트)의 이름을 "Lab3_자신의학번"으로 생성한다.
- 압축 파일의 이름을 "Lab3_자신의학번.zip"으로 지정하여 폴더(프로젝트)를 압축한다.

3. 제출방법 및 주의사항

- 학습과제와 같이 업로드되는 학습보고서의 답안을 작성한다.
- 프로젝트(폴더)의 압축파일을 학습보고서와 함께 제출한다.
- 지각 제출 시, 0점으로 처리한다.
- 채점 시 완성도의 평가는 Google의 [Chrome 브라우저](#)에서 렌더링 된 화면을 기준으로 함

※ "4. 프로그램 설계"를 참고하여 프로그램이 올바르게 동작하도록 구현한다.

4. 프로그램 설계

4.1 HTML 문서(Lab3.html)

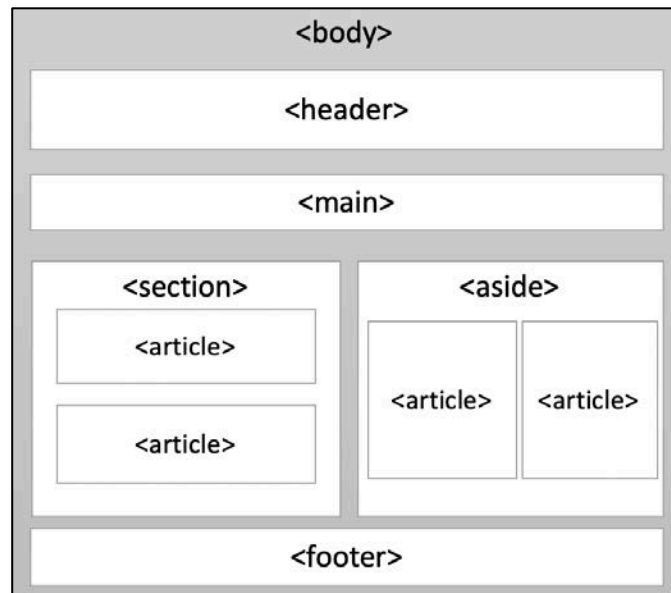


그림 10 HTML의 element 구조

- Element의 구조는 그림 10과 동일하게 구성하도록 한다.
- HTML 문서의 제목은 "Lab3-자신의 학번"으로 지정한다.
 - ✓ <head> 내에서 <title>Lab3-201912345</title> 와 같이 작성
- 같은 폴더 내에 위치한 Lab3.css를 불러오도록 한다.
- <header>, <main> 및 모든 <article> 내에 그림 12와 같은 텍스트를 작성하도록 한다.
- <footer>의 내용은 그림 12와 같이 "Implemented by [Your name], [Your student id]"가 되도록 작성한다. (자신의 이름과 학번)

4.2 CSS 문서(Lab3.css)

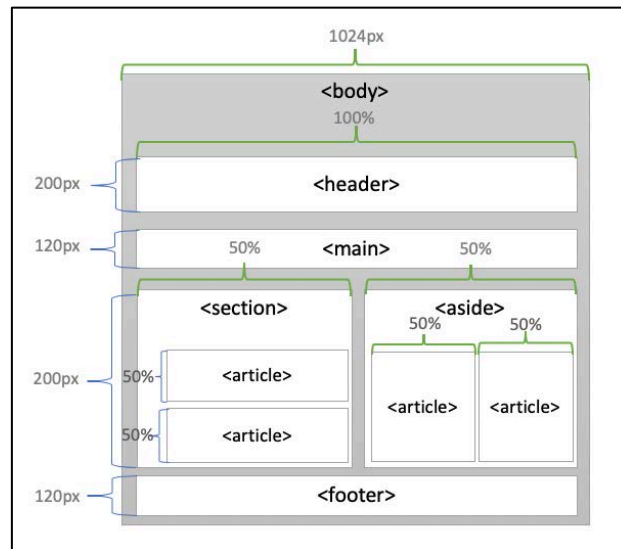


그림 11 레이아웃 규격

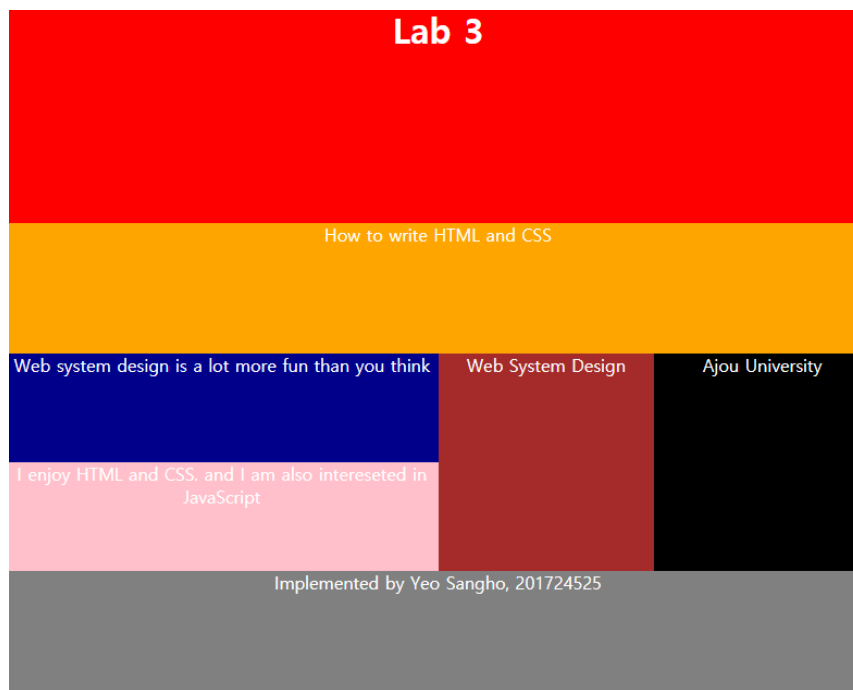


그림 12 HTML 문서 예시

- 각 Element의 크기 속성을 정의한다. (그림 11, 그림 12 참고)
 - ✓ 각 Element 의 width
 - **<body>**: max-width 를 1024px 로 지정 (Mobile, Desktop 에 관계 없이 지정)
 - **<header>**, **<main>**, **<footer>**: 100%
 - **<section>**, **<aside>**: 50%

- <section>내의 <article>은 section 의 width 와 동일함 (100%)
 - <aside>내의 <article>은 aside 내에서 좌우으로 이등분 된다. (width 50%)
- ✓ 각 Element 의 height
 - <header>, <section>, <aside>: 200px
 - <section>내의 <article>은 section 내에서 상하로 이등분 된다. (height 50%)
 - <main>, <footer>: 120px
- Header, main, section, article, aside, footer (즉, 모든 Element)는 인접한 다른 Element와 구분되는 고유한 배경 색상(background-color)을 지정한다.
- 각 Element 사이에 여백(빈 공간)이 생기지 않아야 함
- 모든 텍스트는 중앙 정렬되어야 함
- 요구사항에 따라 레이아웃을 구성하되 padding, margin 및 폰트 크기와 관련된 속성값은 자유롭게 지정한다.