**2022학년도 1학기 중간과제물(온라인제출용)**

**교과목명 : HTML 웹프로그래밍**

**학 번 : 202034-153746**

**성 명 : 이동열**

**연 락 처 : 010-5264-5565**

**과제유형(공통형/지정형) : 공통형**

1.

(1) header

header 태그는 HTML에서 레이아웃을 나눌 때 페이지 소개 및 정보 검색을 위해 상단에 배치하는 태그이다. 이름이 header인 만큼 태그들을 나열 할 때 최상단에 배치되는게 일반적이며, 페이지의 제목이나 전체 검색 창, 로고, 로그인 버튼 등을 배치하기 때문에 페이지 상에서도 보통은 최상단에 출력시킨다. 이 태그는 전역 속성만을 가지고 있으며, 필수적 하위 태그가 존재하지는 않지만 주로 h태그를 감싸는데 사용 된다. 사용 예시는 아래와 같다.

<header>

<h2>HTML 웹프로그래밍</h2>

</header>

(2) hgroup

hgroup 태그는 이름 그대로 h태그들을 그룹화 해줄 때 사용되는 태그이다. 단순히 그룹화하는 태그는 아니고 제목-부제목 형식으로 h태그를 그룹화한다. 따라서 hgroup에 여러 h태그가 하위에 포함되더라도 구획을 가지는 h태그는 하나이며 나머지는 자동으로 부제목이 된다. 이 태그는 하위 요소로 <h1>~<h6> 태그만을 가질 수 있으며, 전역 속성만을 제공한다. 사용 예시는 아래와 같다.

<hgroup>

<h1>HTML 웹프로그래밍</h1>

<h2>HTML WEBPROGRAMMING</h2>

</hgroup>

(3) nav

nav 태그는 현재 페이지의 특정 구간에 대한 링크나 다른 페이지로 링크를 안내해줄 때 사용되는 태그이다. 주로 메뉴나 목차 등에 주로 사용되며 전역 속성만을 사용 할 수 있고 하위 요소로 플로우 콘텐츠만을 가진다. 사용 예시는 아래와 같다.

<nav>

<div><a href=”#section1”>1번</a></div>

<div><a href=”#section2”>2번</a></div>

</nav>

(4) section

section 태그는 한 구획을 나타내는 태그로 다른 Semantic 태그로 의미를 지정하기 힘든 구획을 구성할 때 사용하는 태그이다. 하위 요소로 플로우 콘텐츠들을 가지기 때문에 자유롭게 사용 할 수 있으나 일반적으로 해당 section을 식별하기 위해 h태그를 사용한다. 사용 예시는 아래와 같다.

<section>

<h3>000-section</h3>

<p>contents…</p>

</section>

(5) article

article 태그는 일종의 모듈같은 개념으로, 내부적으로 구획을 나누는 용도가 아닌 독립적으로 출력시킬 수 있는 영역을 나타낸다. 이 태그는 하위 요소로 플로우 콘텐츠를 가지며 전역 속성만을 가지고 있다. 이 태그 자체가 하나의 의미있는 영역을 나타내기 때문에 주로 h태그를 사용해 식별한다. 사용 예시는 아래와 같다.

<article>

<header>

<h2>HTML이란?</h2>

</header>

<section>

<p>

HTML은 하이퍼 텍스트 마크업 언어의 약자로…

</p>

</section>

</article>

(6) aside

aside 태그는 문서와 간접적으로 연관된 부분을 표현 할 때 사용하는 태그이다. 부연 설명 같은 느낌으로 사용한다. 이 태그는 하위요소로 플로우 콘텐츠를 가지며 부모로 address태그를 가질 수 없다. 속성은 전역 속성만을 가지고 있다. 사용 예시는 아래와 같다.

<section>

<h3> HTML이란 </h3>

<p>

HTML은 하이퍼 텍스트 마크업 언어의 약자로…

</p>

<aside>

마크업 언어란 태그를 이용한 데이터 구조…

</aside>

</section>

(7) footer

footer 태그는 영역의 하단 영역을 표현 할 때 사용된다. 이 하단 영역에는 사이트의 정보가 표시되는데 문서 관련자(사업자나 작성자) 정보, 사이트 맵, 저작권 등 다양한 정보가 표시된다. 이 태그는 하위 요소에 header와 footer를 둘 수 없으며 부모 요소로 address 가질 수 없다. 속성은 전역 속성만 가진다. 사용 예시는 아래와 같다.

<footer>

<div>

대표자 : 이동열 <br>

이메일 : ldy@email.com

</div>

</footer>

(8) mark

mark 태그는 내용 중 중요한 부분을 표시한 효과를 내는 태그이다. 실제 책에서 형광펜으로 칠하는 것과 비슷한 의미로 사용된다. 하위 요소로는 구문 콘텐츠만을 가지며 부모 요소 또한 구문 콘텐츠를 허용해야 한다. 속성은 전역 속성만 가지고 있고 사용 예시는 아래와 같다.

<p>

HTML은 <mark>하이퍼 텍스트 마크업 언어</mark>로..

</p>  
(9) time

time 태그는 특정 시간이나 시간대를 표시 할 수 있는 태그이다. 0~24시를 표현하거나 그래고리력 날짜를 표기 할 수 있다. 이 태그는 하위에 구문 콘텐츠를 포함 할 수 있고 속성으로 datetime을 지정해 상세한 날짜나 시간을 지정 할 수 있다. 사용 예시는 아래와 같다.

<time datetime=”2022-03-16 22:12”>2022년 3월 16일 22시 12분</time>  
(10) meter

meter 태그는 범위를 지정해 해당 범위 내에서 스칼라 값이나 백분율을 표현 할 수 있는 태그이다. 이 태그는 하위 요소로 구문 콘텐츠를 포함 할 수 있지만 meter태그는 포함 시킬 수 없다. 이 태그의 속성으로는 현재 값을 지정하는 value, 측정 가능한 범위 중 최솟 값을 나타내는 min, 특정 가능한 범위 중 최댓값을 나타내는 max, 값의 범위 중 낮은 값의 기준이 되는 low, 값의 기준 중 높은 값의 기준이 되는 high, 적정 값의 기준이 되는 optimum, form 태그와 연결해 input 중 number타입의 값 범위를 보여주는 form 속성이 존재하며 여기서 min과 max는 말 그대로 value로 지정 가능한 값의 최소/최대 범위를 말하고 low와 high는 value로 지정한 값이 낮은 값인지 높은 값인지 판단해주는 기준 값을 말한다. optimum의 값을 기준으로 low, high 기준 값을 넘어 갈때마다 meter 막대의 색깔이 초록->노랑->빨강 색으로 변한다. 색의 변경 기준은 optimum에서 얼마나 멀리 떨어져 있는지를 기준으로 한다. 또한 멀리 떨어져 있는지의 여부는 low, high 속성을 넘어 갔느냐를 기준으로 한다. 사용 예시는 아래와 같다.

<meter value=”2” min=”0” max=”20” low=”4” high=”15” optimum=”7” form=”form\_1”>

2/20

</meter>  
(11) progress

progress 태그는 진행도를 나타내는 태그로 meter 태그처럼 막대 형태로 표시 된다. 이 태그의 하위 요소로는 progress 태그를 제외한 구문 콘텐츠가 올 수 있으며, 속성으로는 총 작업량을 나타내는 max, 현재 진행 상태를 나타내는 value가 존재한다. 사용 예시는 아래와 같다.

<progress max=”300” value=”60”>20%</progress>  
(12) ruby

ruby 태그는 루비 주석을 표현 할 때 쓰는 태그로, 루비 주석은 일본어 같은 동아시아 언어의 발음 표현 할 때 사용되는 주석이다. 이 태그의 하위 요소로는 rp 태그와 rt태그, 그 외에 구문 콘텐츠가 올 수 있는데 rp 태그는 루비 주석을 지원하지 않는 브라우저에 표시하기 위해 사용되는 태그이며, rt태그는 루비 주석의 윗첨자를 표현 할 때 사용되는 태그이다. 속성은 전역 속성만을 가지고 있다. 사용 예시는 아래와 같다.

<ruby>

東<rp>(</rp><rt>Dong</rt><rp>)</rp>

列<rp>(</rp><rt>Yeol</rt><rp>)</rp>

</ruby>  
(13) wbr

텍스트가 길어지면 자동으로 줄바꿈이 되어 원치 않은 줄바꿈이 되거나 줄바꿈이 아예 안될 수도 있다. 이 때 wbr 태그를 사용하면 줄바꿈이 되는 위치를 직접 지정 할 수 있다. 이 태그는 내용이 없는 빈 요소이기 때문에 하위 요소를 포함 할 수 없으며 속성도 전역 속성만 가지고 있다. 사용 예시는 아래와 같다.

<p>HTML은<wbr> 하이퍼 텍스트 마크업<wbr> 언어로..</p>  
(14) details

details 태그는 드랍다운 같이 클릭하면 나열되는 콘텐츠를 표현 할 때 사용되는 태그이다. 이 태그의 하위 요소에는 summary 태그가 포함 될 수 있는데 이 태그를 사용하면 이 콘텐츠의 라벨을 표시 할 수 있게 된다. 속성으로는 콘텐츠가 보이는지 여부를 설정 할 수 있는 open 속성을 가지고 있으며 true면 보이는 상태로 출력되고 false면 숨겨진 상태로 출력된다. 사용 예시는 아래와 같다.

<details>

<summary>눌러보시오</summary>

<p>내용내용~</p>

</details>   
(15) datalist

datalist 태그는 다른 입력창에서 특정 데이터들을 선택 할 수 있게 해주는 태그이다. 비슷한 예로 검색창을 클릭하면 하위에 추천 검색목록이나 검색 내역이 표시되는 것 같다. 이 태그는 하위에 option 태그를 가질 수 있는데 이 태그가 표시되는 데이터 목록이 된다. 속성은 전역 속성만 가지며 사용 예시는 아래와 같다.

<datalist>

<option value=”hyper”>

<option value=”text”>

<option value=”markup>

<option value=”language”>

</datalist>   
(16) output

output 태그는 계산된 값을 표시 할 수 있는 태그로 for 속성을 사용해 계산에 사용되는 요소를 표시하고, form 속성을 사용해 이 태그가 포함될 form 태그를 지정 하거나 name 속성을 사용해 이 태그의 이름을 지정 할 수 있다.

<form id=”form\_1” oninput=”output\_1.value=parseInt(input\_1.value)+parseInt(input\_2.value)”>

<input type=”number” id=”input\_1” name=”input\_1” value=”20”>

<input type=”number” id=”input\_2” name=”input\_2” value=”5”>

<output name=”output\_1” for=”input\_1 input\_2” form=”form\_1”>15</output>

</form>

2.

전환(transition)

1. transition

transition 속성은 한 요소에 대해 상태 변화를 지정하는 속성으로, transition-delay, transition-duration, transition-property, transition-function의 속성을 약식으로 포함하고 있다. 이 속성은 Javascript를 사용해서 동적으로 변환 시키거나 css로 의사 클래스를 사용해 변환시킬 수 있다. 사용 방법은 아래와 같다.

transition : <property name> <duration> <function> <delay>

1. transition-delay

transition-delay는 전환하기 전에 대기 하는 시간을 지정 하는 속성이다. 값으로 s또는 ms를 지정할 수 있으며, 0으로 지정하면 바로 전환을 시작한다. 또한 property가 여러 개라면 이 값도 여러 개 지정 할 수 있다. 사용 예시는 아래와 같다.

transition-delay : 4s

1. transition-duration

transition-duration은 전환이 지속되는 시간을 지정 하는 속성이다. 값으로 s또는 ms를 지정 할 수 있으며, 요소가 전환을 시작하면 여기서 지정한 시간에 맞춰서 전환이 진행된다. 따라서 이 값이 크면 전환이 느리게 진행되고 작으면 빠르게 진행된다. 사용 예시는 아래와 같다.

transition-duration : 4s

1. transition-property

transition-property는 전환이 될 css의 속성을 지정하는 속성이다. 값으로 none, all, custom-ident를 지정 할 수 있다. none은 전환 할 속성을 지정하지 않는 것이고 all은 모든 속성을 지정하는 것이다. custom-ident는 속성을 직접 지정하는 것인데, 예를 들어 font-size를 지정하면 font-size가 전환되고 margin을 지정하면 margin이 전환 된다. 사용 예시는 아래와 같다.

transition-property : font-size

1. transition-timing-function

transition-timing-function은 다양한 함수를 지정해 가속을 조절 할 수 있다. 지정 가능한 함수는 cubic-bezier, steps가 있다. cubic-bezier 함수는 bezier곡선을 정의 할 수 있는데 직접 값을 정의 할 수도 있고 사전 정의된 값들(ease,linear,ease-in,ease-out,ease-in-out)을 지정 할 수도 있다. steps 함수는 step(n, <jumpterm>) 형태로 사용되며 n은 단계를 뜻하며, jumpterm은 어떤 방식으로 점프 할 지를 지정 할 수 있다. n은 숫자이며, jumpterm은 전환이 시작되면 점프가 시작되는 jump-start(start와 동일), 전환이 끝나면 점프가 시작되는 jump-end(end와 동일), 전환 중간에서 점프가 시작되는 jump-none, 0%, 100%에서 정지해 단계를 추가하는 jump-both 중 선택하면 된다. steps 함수의 경우도 사전 정의된 값인 step-start, step-end를 사용 할 수 있다. 사용 예시는 아래와 같다.

transition-timing-function: ease;

애니메이션

1. animation

animation 속성은 해당 속성과 같은 부류인 다른 모든 속성에 대한 값을 약식으로 포함하고 있는 속성이다. 이 속성의 값으로는 애니메이션 재생 횟수인 single-animation-iteration-count, 재생 방향을 뜻하는 single-animation-direction, 실행 전후를 기준으로 스타일 적용 방식을 정하는 single-animation-fill-mode, 재생 여부를 뜻하는 single-animation-play-state가 존재한다. 사용 예시는 아래와 같다.

animation: <@keyframes duration> <easing-function> <delay> <iteration-count> <direction> <fill-mode> <play-state> <name>

1. animation-delay

animation-delay 속성은 애니메이션이 시작하기 전에 대기 하는 시간을 지정하는 속성이다. 값으로는 시간(m, ms)을 지정 할 수 있으며 양수를 지정하면 지정한 시간만큼 대기했다가 시작되고 음수를 지정하면 지정한 초의 절대값 만큼 진행된 상태로 출력되고 이어서 진행된다. 사용 예시는 아래와 같다.

animation-delay : 2s

1. animation-direction

animation-direction 속성은 애니메이션의 방향을 지정하는 속성이다. 값으로는 앞으로 재생되는 nomarl, 뒤로 재생되고 기능도 반대로 적용되는 reverse, 첫 시작을 앞으로 해서 매번 앞뒤로 한 번씩 방향이 바뀌는 alternate, 첫 시작을 뒤로 해서 매번 앞뒤로 방향이 바뀌는 alternate-reverse가 존재한다. 사용 예시는 아래와 같다.

animation-direction: alternate

1. animation-duration

animation-duration은 애니메이션이 한 번 완료하는데 걸리는 시간을 지정하는 속성이다. 값으로는 시간을 지정 할 수 있는데 음수는 지정 할 수 없으며 양수 또는 0만을 지정 할 수 있다. 양수를 지정하면 그 시간만큼 한 주기를 수행하고 0으로 지정하면 애니메이션이 발생하지 않는다. 사용 예시는 아래와 같다.

animation-duration: 320ms

1. animation-fill-mode

animation-fill-mode 속성은 실행 전후를 기준으로 스타일을 지정하는 속성이다. 값으로는 스타일을 적용하지 않는 none, 마지막 키프레임을 유지하는 forwards, 스타일을 애니메이션이 시작되기 전에 적용 시키는 backwards, forwards와 backwards를 동시 적용시킨 효과를 보이는 both가 존재한다. 사용 예시는 아래와 같다.

animation-fill-mode: forwards

1. animation-iteration-count

animation-iteration-count 속성은 애니메이션이 몇 번 반복 될 지를 결정하는 속성이다. 값으로는 무한 반복을 뜻하는 infinite, 반복 횟수를 지정하는 number가 존재한다. number의 경우 양수를 지정하면 된다. 사용 예시는 아래와 같다.

animation-iteration-count : 2

1. animation-name

animation-name은 해당 요소에 적용할 애니메이션을 정의해 놓은 @keyframes 이름을 지정하는 속성이다. 값으로는 keyframes가 없음을 나타내는 none, keyframes 이름을 지정하는 custom-ident가 존재한다. 사용 예시는 아래와 같다.

animation-name: custom-keyframes

1. animation-play-state

animation-play-state 속성은 애니메이션 실행 여부를 설정하는 속성이다. 값으로는 실행 중임을 뜻하는 running, 정지 중임을 뜻하는 paused가 존재한다. 사용 예시는 아래와 같다.

animation-play-state: running

1. animation-timeline

animation-timeline 속성은 스크롤 애니메이션을 정의한 @scroll-timeline 이름을 지정하는 속성이다. 값으로는 문서의 기본 timeline을 따라가는 auto, timeline이 없는 none, scroll-timeline 이름을 지정하는 timeline-name이 존재한다. 사용 예시는 아래와 같다.

animation-timeline: custom-timeline

1. animation-timing-function

animation-timing-function 속성은 애니메이션이 진행되는 방식을 지정하는 속성이다. 값으로는 transition-timing-function처럼 cubic-bezier 함수나 steps 함수를 지정해 줄 수 있으며, 사전 정의된 값을 지정해 줄 수도 있다. 사용 예시는 아래와 같다.

animation-timing-function: steps(4, jump-end);

1. @keyframes

이 속성은 진행률에 맞춰 애니메이션 스타일을 지정 할 수 있는 속성이다. 값으로는 keyframes의 이름을 뜻하는 custom-ident, 0%에서의 시작을 뜻하는 from, 100%의 끝을 뜻하는 to, 중간 지점을 지정 할 수 있는 percentage가 존재한다. 사용 예시는 아래와 같다.

@keyframes custom {

from { padding-top : 10rem; padding-bottom: 5rem;}

25% { padding-top : 2rem; padding-bottom: 2rem;}

to { padding-top : 20rem; padding-bottom: 10rem;}

}

1. @scroll-timeline

이 속성은 스크롤 진행에 맞춰 애니메이션 timeline을 지정 할 수 있는 속성이다. 값으로는 timeline의 이름을 뜻하는 custom-ident, 스크롤 진행의 대상이 되는 source, 스크롤의 방향을 뜻하는 orientation, 스크롤의 offset을 결정하는 scroll-offsets가 존재한다. 사용 예시는 아래와 같다.

@scroll-timeline custom {

source: auto;

orientation: auto

scroll-offsets: none

}

다단(multiple column)

1. column-width

column-width 속성은 한 영역에서 콘텐츠의 너비를 지정해 열을 나눌 수 있는 속성이다. 값으로는 너비를 지정 할 수 있는 length, 다른 속성의 영향을 받아 너비를 결정하는 auto가 존재한다. 사용 예시는 아래와 같다.

column-width: 30px

1. column-count

이 속성은 콘텐츠의 열을 너비가 아닌 열의 수를 지정 해 나눌 수 있는 속성이다. 값으로는 열의 개수를 뜻하는 count가 존재한다. 사용 예시는 아래와 같다.

column-count: 3;

1. columns

columns 속성은 column-width 속성과 column-count 속성을 약식으로 포함하고 있는 속성이다. 값으로는 너비인 width와 열 개수인 count를 지정해 줄 수 있다. 사용 예시는 아래와 같다.

columns: 11em auto

1. column-rule-color

column-rule-color 속성은 각 열 사이에 생성되는 선의 색깔을 지정 할 수 있는 속성이다. 값으로는 색상을 뜻하는 color를 지정해 줄 수 있다. 사용 예시는 아래와 같다.

column-rule-color: rgb(255, 255, 255);

1. column-rule-style

이 속성은 각 열 사이에 생성되는 선의 스타일을 지정 할 수 있는 속성이다. 값으로는 선의 스타일을 뜻하는 border-style을 지정 할 수 있다. 사용 예시는 아래와 같다.

column-rule-style: outset;

1. column-rule-width

이 속성은 각 열 사이에 생성되는 선의 너비를 지정 할 수 있는 속성이다. 값으로는 선의 너비를 뜻하는 border-width를 지정해 줄 수 있으며, 사전 정의된 값인 thin, medium, Thick를 지정 할 수도 있다. 사용 예시는 아래와 같다.

column-rule-width: medium;

1. column-rule

이 속성은 column-rule-color, column-rule-style, column-rule-width 속성을 약식으로 포함하고 있는 속성이다. 값으로는 선의 색상을 뜻하는 color, 선의 스타일을 뜻하는 style, 선의 너비를 뜻하는 width를 가지고 있다. 사용 예시는 아래와 같다.

column-rule: dotted 1px rgb(255,255,255);

1. column-span

이 속성을 all로 설정하면 해당 속성을 지정한 요소는 모든 열에 걸쳐서 출력 된다. 값으로는 모든 열에 걸치는 설정인 all, 걸치지 않게 하는 none이 존재한다. 사용 예시는 아래와 같다.

column-span: all;

1. column-fill

이 속성은 해당 요소의 콘텐츠가 각 열에 균형있게 배치되는 지를 설정하는 속성이다. 값으로는 한 열씩 순서대로 채워가는 auto, 각 열에 균등하게 배분되지만 페이지 미디어에서는 마지막만 균등하게 배분되는 balance, 페이지 미디어를 포함한 모든 콘텐츠가 균등하게 배분되는 balance-all을 지정 할 수 있다. 사용 예시는 아래와 같다.

column-fill: balance-all

1. column-gap

이 속성은 각 열 사이의 간격을 지정하는 속성이다. 값으로는 브라우저의 기본 간격을 뜻하는 normal, 간격을 직접 지정하는 length, 간격을 백분율로 지정하는 percentage가 존재한다. 사용 예시는 아래와 같다.

column-gap: 11%