**CSS3 스타일 시트 구성**

span { color : blue; font-size ; 20px; } /\*주석문\*/

-span : 셀렉터 / CSS3 스타일시트를 HTML 페이지에 적용하도록 만든 이름

-color : 프로퍼티 / 스타일 속성 이름. 약 200개 정도의 프로퍼티 존재

-blue : 값 / 프로퍼티의 값

-세미콜론으로 프로퍼티 구분.

-대소문자 구분 없음.

**CSS3 스타일 시트 만들기**

-<body>태그 내부 삽입형 : in-line style

문서 내의 각 태그 요소에 직접 스타일 시트를 정의

형식 : <태그 style=“프로퍼티:값; 프로퍼티:값;”>

-문서 <head> 내 지정형 : embedded/internal/global style

문서의 <head>부분에 <style>~~</style> 태그로 스타일 시트 정의

형식 : <style type=“text/css”>스타일시트작성</style>

-외부 문서삽입형 : linked/external style

스타일 시트를 별도의 파일로 저장하여, 다른 문서에서도 사용가능하게 한 것.

형식 :

1. <head>에 <link rel=“stylesheet” href=“스타일 파일 전체경로” type=“text/css”>
2. <style>에 @import 명령어를 사용하여 연결

@import url(stylelink.css);

-우선순위 : inline > global/external

global과 external은 가장 나중에 지정한 것이 적용된다.

**스타일지정방법 – 선택자**

-선택자들을 콤마를 이용해 그룹화 가능 (ex : h1, h2, h3 {color : red})

-태그선택자/전체선택자/속성선택자

태그선택자 : HTML요소 이름을 사용

h1 {color : green} //모든 h1요소를 선택함

전체선택자 : 페이지 안의 모든 요소를 선택

\* {color : green} //전체요소를 선택함.

속성선택자 : HTML 태그의 특정 속성에 대해 값이 일치하는 태그에만 스타일을 적용.

input[type=text] {color : red} //<input type=“text”>인 태그만 선택

선택자의 속성 상속/계승 -> 서로 다른 태그를 해도 된다. div > img

자손선택자 : h1 em {color : red} //h1의 하위로 나오는 em만 선택함.

자식선택자 : h1 > em {color:red} //h1의 직계하위로 나오는 em만 선택함 하하위x

-class 속성을 이용한 스타일 정의 : 동일한 태그에 서로 다른 스타일을 지정할 때 사용.

클래스를 사용하면 원래의 태그 스타일이 있더라도, 클래스 스타일로 정의한 속성이 적용.

선택자.클래스명 {속성:속성값} //<태그명 class=“클래스명”>~~</태그명>

.클래스명 {속성 :속성값} //모든 태그에 사용가능.

-ID속성을 이용한 스타일 정의

이미 지정된 스타일 속성 이외에 또 다른 속성을 지정하여 사용가능.

선택자#ID이름{속성: 속성값} //<태그명 id =“ID이름”>~~</태그명>

id이름은 대소문자를 구분하여 사용

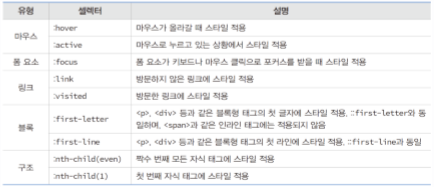
하나의 id이름은 하나의 태그에서만 사용가능하다. 그룹화가 불가능.

-가상선택자 : 어떤 상황이 발생하였을 때만 적용하는 셀렉터

특수한 상황에서만 발생하는 동적인 요소를 CSS명령으로 제어하는 기법.

class선택자는 아니지만 엘리먼트의 상태에 따라 여러 엘리먼트를 선택할 수 있다.

선택자:가상요소 {속성 : 속성값}



가상선택자를 사용할 때, 콜론(:)앞 뒤에 빈칸을 두면 안된다.

**CLASS VS ID선택자**

-class선택자

태그의 종류에 관계없이 class 셀렉터 활용가능

여러 태그를 하나의 그룹으로 묶어 단체로 동일한 CSS스타일을 적용할 때 적합

class 속성 값이 같은 태그에 모두 CSS스타일 적용

-id선택자

오직 하나의 태그에서만 사용. 그룹화하여 사용이 불가능.

여러 태그 중 특정 태그에만 CSS스타일을 적용할 때 적합

class와 동일하지만, 한문서에서 하나의 id이름은 하나만 있어야한다.

자바스크립트 코드에서 id값을 가진 태그 객체를 찾을 때 문제됨.

class보다 우선적으로 적용된다.

**!important 선언**

-!important선언은 모든 style규칙에 우선하며 inline style, id, class 등으로 같은 style속성을

선언했더라도 그 값을 모두 무시하고 !important로 선언한 속성값을 적용.

-사용법 : {속성 : 속성값 !important} //공백 있어도 됨.

{속성 : 속성값 !important; 속성2:속성값}//속성값 뒤에 붙임.

-주로 프린트 적용을 위해 사용.

출력할 땐 모두 검정글씨로 통일하는 게 편하기 때문.

!important선언으로 하나하나 수정하기보다 한번에 할 수 있다.

**우선순위**

1.!important

2.in-line으로 지정한 속성

3.#id

4.class 또는 가상선택자로 지정한 속성

5.태그이름으로 지정한 속성

6.상속된 속성.

-같은 우선순위에 있는 경우, 나중에 선언한 것이 우선되어 적용된다.

head삽입 class하고

body삽입 class중에선 body가 우선. -> 밑줄 친 부분이 적용됨.

**CSS에서 색표현**

-16진수 코드로 표현 #8A2BE2 : 2칸씩 red, green, blue

-10진수 코드와 rgb로 표현 rgb(138,43,226) : red, green, blue

-색이름 표현 blueviolet

**색관련 프로퍼티**

-color : 글자색

-background-color : 배경색

-border-color : 테두리색

**텍스트**

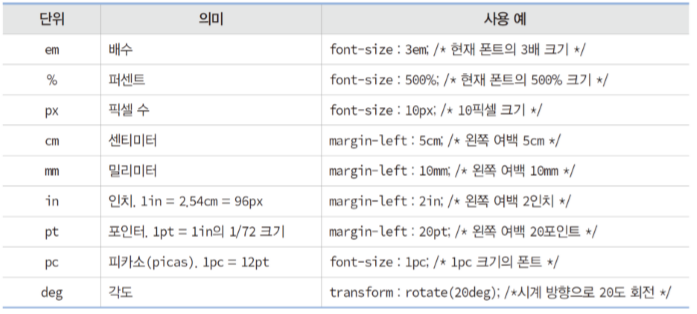
-text-indent : <length>|<percentage> //들여쓰기

-text-align : left|rigth|center|justify //정렬

-text-decoration : none|underline|overline|line-through //텍스트꾸미기

none하면 하이퍼링크에 밑줄을 제거할 수 있음.

**CSS3의 표준단위**



**폰트**

-font-style : normal | italic | oblique 중 하나 선택

-font-weight : 폰트 굵기

font-weight : 300; //100~900의 범위에서 굵기 지정.

font-weight : bold; //normal, bold, bolder, lighter 중 하나 선택. normal(400)

-font-size

font-size : 40px; | font-size : medium; | font-size ; 1.6em

medium은 브라우저마다 다르다.

**-**font-family : 구체적인 폰트이름 지정

font-family : Arial, “Times New Roman”, Serif;

Arial이 없으면 “Times new Roman”을 적용하는 방식

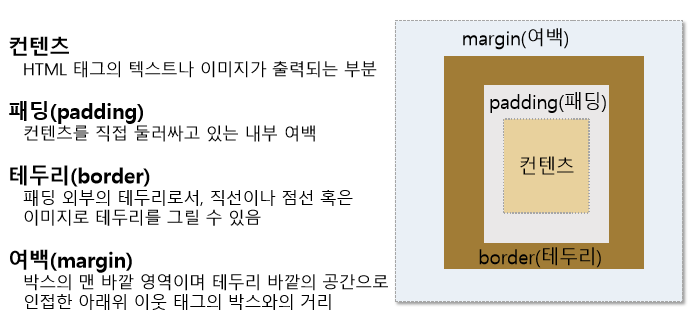
-font : font-style font-weight font-size font-family

font 프로퍼티를 한번에 지정할 때 사용

font-size, font-family는 생략불가

**박스모델**

-구성



-프로퍼티



-box model의 색

패딩 : 태그의 배경색(background)이 적용됨.

테두리(border) : 개발자가 선정가능

여백(margin) : 투명하게 처리

-박스 테두리(border)

-border-style : dotted | solid

-border : border-width border-style border-color / 1px solid blue;

-border-radius : 테두리의 네 모서리를 둥글게 만듦.

border-radius : 50px;

border-radius : 0px 20px 40px 60px; //왼쪽 위부터 시계방향

**배경다루기**

-background-color : 색상 //배경색 지정

-background-image : url(주소) //배경이미지

둘다 지정되면 배경이미지가 없는 곳에 배경색 출력

-background-position : [x축] [y축];

left | top | center | right | bottom 사용.

배경 이미지 시작위치 지정

-background-size : 100px 100px //배경 이미지 크기

-background-repeat : no-repeat/repeat/repeat-x/repeat-y //배경 이미지반복출력

-background : color image position/size repeat

background : skyblue url(“.jpg") center center/100px 100px repeat-y;

**그림자효과**

-text-shadow : h-shadow v-shadow blur-radius color |none

h-shadow. v-shadow : 원본 텍스트와 그림자 텍스트 사이의 수평/수식 거리 (필수)

blur-radius : 흐릿하게 번지는 길이(선택)

color : 그림자 색

none : 그림자효과 없음.

-box-shadow : h-shadow v-shadow blur-radius spread-radius color |none|inset

spread-radius : 그림자크기(선택, 디폴트 0)

inset : 음각 박스로 보이게 박스 상단 안쪽에 그림자형성

**마우스 커서 제어**

-cursor : value

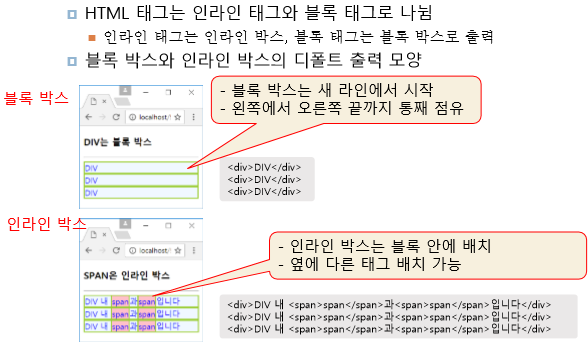
태그 위에 마우스가 올라갈 때 마우스의 커서 모양 지정



**박스**

-블록박스 : 항상 새라인에서 시작. 화면의 한줄을 전부 차지함.

**-인라인박스 : 한 줄에 차례대로 배치. 현재 줄에서 필요한 만큼의 너비만을 차지함.**



**배치**

-CSS3로 HTML 태그가 출력되는 위치 지정.

-display : 박스의 유형 제어

display : block | inline | inline-block

블록박스를 인라인처럼 인라인박스를 블록처럼 이용가능.

단, 인라인박스 안에 display:block을 넣으면 문제발생

-top | bottom | left | right : 크기 - ‘크기’만큼 선택한 속성에서 떨어짐.

-normal flow : 웹 페이지에 나타난 순서대로 HTML 태그 배치. position으로 무시가능.

-position : static(디폴트) | relative | absolute | fixed / float : left | right

top, bottom left, right, width, height로 태그의 위치와 크기 지정.

static : 문서의 정상적인 흐름에 따라 배치. top, bottom, left, right의 영향을 받지 않음

relative : 정상적인 위치에서 상대적으로 요소가 배치. static으로 지정된 정상적인 위치에서

top, bottom, left, right의 값만큼 이동한 ‘상대 위치’에 배치

absolute : 페이지의 시작위치에서 top, bottom, left, right의 값만큼 이동한 위치에 배치

fixed : 브라우저 윈도우에 상대적으로 요소의 위치를 잡는 것. 페이지를 스크롤하여도

같은 위치에 계속 있다.(광고를 생각.)

float : left | right – 유동배치

항상 브라우저의 왼쪽 또는 오른쪽에 배치.

**fixed와 다른 건 top, bottom, left, right로 위치지정안함. 크기 지정만 해줌.**

clear : both – float 속성을 중단할 때 사용됨.

-z-index : 양수 | 음수

position이 relative, absolute, fixed로 설정되어 있어야 작동함.

position이 없으면 z-index가 0인 것처럼 작동.

**visibility 프로퍼티**

**-**visibility : visible | hidden

hidden – 태그 공간만 할당되고 보이지는 않는다. / 이미지도 안보임.

display : none – 태그 공간이 할당 안 되고 보이지도 않는다.

**overflow 프로퍼티**

**-HTML 콘텐츠가 width와 height 프로퍼티에 주어진 태그의 크기를 넘어가는 경우 콘텐츠를 자를 것인지**

**를 지정하는 프로퍼티(글 내용이 설정한 크기보다 클 경우)**

-overflow : visible | hidden | scroll | auto

visible : 콘텐츠가 잘리지 않고 태그 영역을 넘어 출력(디폴트)

hidden : 콘텐츠를 태그 크기로 잘라 넘어가는 것은 보이지 않음. 스크롤바 없음.

scroll : 스크롤바를 항상 부착하여 콘텐츠 출력

auto : 콘텐츠가 박스의 높이(height 프로퍼티)보다 넘치면 스크롤바 자동생성

**동적변화**

-애니메이션

@keyframes name { @keyframes color {

시간비율 {프로퍼티; 값;} 0% {color : blue;}

시간비율 {프로퍼티; 값;} 30% {color : green;}

시간비율 {프로퍼티; 값;} } 100% {color : red;} }

애니메이션 스타일시트

animation-name : 애니메이션 이름; //@keyframes의 name지정

animation-duration : 시간; //1회 애니메이션 시간 3s 5s등

animation-iteration-count : 애니메시션 반복횟수 //infinite로 주면 무한 반복.

-전환

HTML태그에 적용된 CSS3 프로퍼티 값의 변화를 서서히 진행시켜 애니메이션 효과생성.

span { transition : font-size 5s; } //글자크기가 변하면 5초동안 transition 작동

span:hover { font-size ; 500% } //마우스 올리면 글자크기 500%로 확대

@keyframes size { 100% {font-size : 500%} }

span { transition : font-size 3s; font-size: 100%; animation-name : size;}

-변환

텍스트나 이미지를 회전, 확대 등 다양한 기하학적인 모양으로 출력

회전각도의 단위는 deg이며 시계방향으로 회전.



img#a {width: 300px; height: 300px}

img#a:hover {transform: scale(2,2) translate(260px, -80px);}

이런 식으로 사용가능