

## CSS셀렉터란 무엇인가?

CSS의 구조



CSS는 “스타일을 지정할 대상(셀렉터 : selector)” 과 “적용할 내용 (선언블록 : property : value ;)” , 이렇게 두가지로 구성되어 있다. 선언블록은 다시 프로퍼티(property)와 값(value)으로 나누어 진다. CSS셀렉터는 스타일을 지정할 대상을 일컫는다.

## 유니버살 셀렉터(Universal selector)

유니버살 셀렉터는 모든 요소를 대상으로 하는 셀렉터이다. CSS2에 정의되어 있다.

```
* {  
  margin:0;  
  padding:0;  
}
```

## 타입 셀렉터(Type selector)

E

타입 셀렉터는 가장 단순한 셀렉터이다. 요소명을 지정하여 그 요소에 스타일을 적용시킨다.

```
p {  
  margin:0;  
  padding:0;  
}
```

## 자손 셀렉터(Descendant combinator)

E F

복수의 실렉터를 조합해 사용하는 방법이다. 자손 셀렉터는 부모요소에 포함된 모든 자손 요소에 스타일을

```
p span.note {  
  color:red;  
}
```

```
ul li * li {  
  margin-left:2em;  
}
```

## 자식 선택터(Child combinator)

### E > F

자식 선택터는 부모 요소의 직접적인 자식요소에 스타일을 적용한다. 손자 요소 이하에는 적용 되지 않는다. CSS2에 정의되어 있다. 적용한다. 유니버설 선택터나 속성 선택터등과 조합해서 사용할 수 있다. CSS1에 정의되어 있다.

## 인접 선택터(Adjacent sibling combinator)

### E + F

E라고 하는 요소 중에서 직접적인 동생 요소인 F라고 하는 요소에 스타일을 적용한다. CSS2에 정의되어 있다.

형제 요소란 어떤 부모 요소안에서 동렬로 존재하는 요소 가운데 앞(형), 뒤(동생)에 존재하는 요소를 가리킨다. 직접적인 동생 요소란 E라고 하는 요소와 동렬로, E라고 하는 요소의 바로 뒤에 인접해 존재하는 요소를 가리킨다. 예를 들면 다음과 같은 경우 h2요소는 h1요소의 직접적인 동생 요소이다. CSS2에 정의되어 있다. 적용한다. 유니버설 선택터나 속성 선택터등과 조합해서 사용할 수 있다. CSS1에 정의되어 있다.

## 유사 클래스(Structural pseudo-classes)

### E:first-child

E라고 하는 요소 중에서 부모요소에 처음으로 출현하는 자식요소인 E요소에 스타일을 적용한다. CSS2에 정의되어 있다.

```
div p:first-child {  
  margin:0;  
  padding:0;  
}
```

## 링크 유사클래스(The link pseudo-classes)

### E:link E:visited

문서안의 하이퍼 링크 중에서 유저가 아직 방문하지 않은 링크인가(:link), 또는 방문을 한 링크인가(:visited)에 따라 각각의 스타일을 적용한다. CSS1부터 존재한다.

유저 액션 유사클래스(The user action pseudo-classes)

**E:active**

**E:hover**

**E:focus**

유저의 조작에 따라 스타일을 적용한다. :active는 유저에 의해서 대상 요소가 액티브하게 되었을 때, 예를 들어 마우스로 클릭하고 버튼에서 손을 뗐을때 적용된다. :hover는 유저에 의해서 대상 요소가 가리켜지고 있을 때, 롤 오버 등. 이 두가지는 CSS1부터 존재. :focus는 대상 요소가 포커스 되었을때 스타일이 적용된다. 예를 들면 입력 대기가 되어 있는 input요소 등. :focus는 CSS2에 정의되어 있다.

속성 선택터(Attribute selectors)

**E[foo]**

어떤 요소 중에서 지정한 속성을 가지고 있는 요소에 스타일을 적용한다. CSS2에 정의되어 있다.

```
p[class] {  
  margin:0;  
  padding:0;}
```

**E[foo=" bar" ]**

같은 속성 선택터이지만 속성명뿐만이 아니라 그 값도 포함해서 지정한다. CSS2에 정의되어 있다.

```
a[rel=" external" ] {  
  padding-right:15px;  
  background:(image/external-icon.png) no-repeat right center;  
}
```

클래스 선택터(Class selectors)

**E.warning**

class속성 값에 따라 스타일을 적용한다. CSS1에 정의되어 있다.

| ID 셀렉터(ID selectors)

**E#myid**

id속성 값에 따라 스타일을 적용한다. CSS1에 정의되어 있다

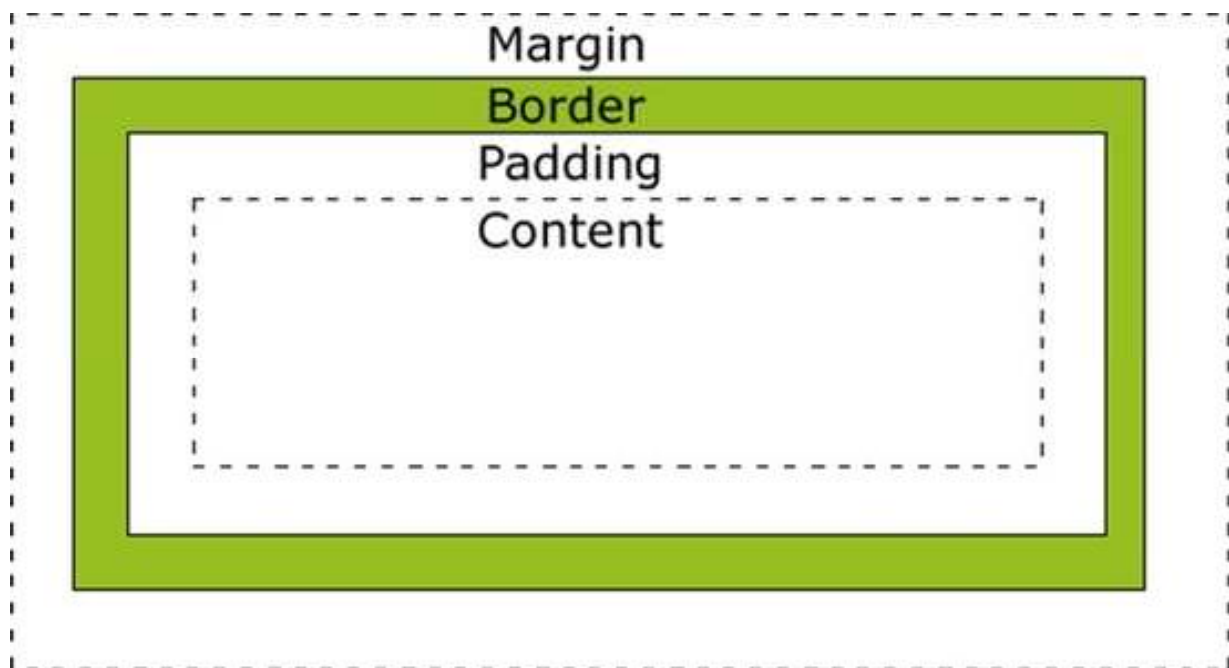
## The CSS Box Model

모든 HTML elements는 box로서 고려될 수 있다. CSS에서, 용어 “box model” 은 desing 과 layout 에 대해 이야기할 때 사용되어 진다.

CSS box model은 HTML elements 주위를 둘러싸는 box이고, margins, borders, padding, 그리고 실제적인 content로 구성되어 진다.

box model은 elements 주위에 border를 넣고 다른 elements와의 공백을 넣는 것을 허락한다.

아래의 image는 box model을 보여준다.



각각에 대한 설명이다.

### Margin

– border 주위의 영역을 지운다. margin은 background color를 가질 수 없고 투명하다.

### Border

– border는 padding 과 content 주위에 놓인다. border는 box의 background color 에 의해 영향을 받는다.

### Padding

– content 주위의 영역을 지운다. padding은 box의 background color에 의해 영향을 받는다.

**Content** – text 와 images 가 나타나는 box의 content.

모든 브라우저에서 정확하게 element의 width 와 height 를 설정하기 위해, box model이 작동하는 방법을 알아야한다.

## Width and Height of an Element

**Important** : CSS를 사용하여 element의 width 와 height properties를 명시할 때, **단지 content 영역의 width 와 height 만 설정한다.**

element의 full size를 알기 위해, 반드시 padding, border, 그리고 margin 을 포함해야 한다.

element의 총 width는 항상 다음과 같이 계산된다.

**Total element width** = width + left padding + right padding + left border + right border + left margin + right margin

element의 총 height는 항상 다음과 같이 계산된다.

**Total element height** = height + top padding + bottom padding + top border + bottom border + top margin + bottom margin

## Browsers Compatibility Issue

IE는 DOCTYPE 이 선언되지 않는다면, width property 가 설정될 때 **width 안에 padding 과 border를 포함한다.**

이 문제를 고치기 위해서는 DOCTYPE를 코드에 추가하면 된다.

## box-sizing property (CSS3)

box-sizing는 content-box로 계산되는 방식을 border-box로 바꾸어서 계산 할 수 있다.  
width안에 padding과 border의 수치를 포함 시킨 방식이다.

절대적 수치와 상대적 수치를 함께 쓸때 사용된다.

## 상속 (Inherit)

하위 요소가 상위 요소의 사용값을 물려 받느냐 아니냐를 나타내는 것이다.  
이러한 값의 상속은 상속되는 것이 있고 안 되는 것이 있다. property마다 다르기때문에 확인해야한다.  
값이 자동으로 자식 요소에 상속되지 않는 경우 “**inherit**” 키워드를 이용하여 강제로 상속시킬 수 있다.

## 겹침 (Cascading)

css의 property의 우선순위를 말한다.  
스타일의 충돌 : 두 개 이상의 규칙이 동일한 한 개의 요소에 적용시킬때 적용된다.  
스타일의 우선순위 결정 규칙 : **가장 마지막에 지정된 스타일을 우선적으로 적용**

## 특정도(specificity)

css의 property의 우선순위를 말한다.  
스타일의 충돌 : 두 개 이상의 규칙이 동일한 한 개의 요소에 적용시킬때 적용된다.  
스타일의 우선순위 결정 규칙 : **가장 마지막에 지정된 스타일을 우선적으로 적용**

선택자	사용 예	특정도
Type 선택자	p	1
Class 선택자	.note	10
Id 선택자	#gnb	100
Inline 스타일	style = “color:red”	1000
가상요소	:first_line	1
가상 클래스	:hover	10

## 중요 (!important)

css는 위에서 아래로 차례로 읽혀지며 적용되기때문에 중복된 속성이 있을경우 뒤에 있는것이 최종적용된다.  
이들 중복된 속성중에 **!important** 를 사용하면 뒤에 있지 않더라도 중복된속성에서 우선적용시킬 수 있게 된다.

```
p span {  
  color: #ff0000;  
  border: 1px solid red !important;  
}
```