

- 목차
 - 1. CSS
 - 1.1. CSS 기본설정
 - 1.2. 모바일 우선기능
 - 1.3. CSS 사용
 - 2. bootstrap CSS 기반
 - 2.1. LESS
 - 3. Bootstrap 프레임워크 CSS 기능
 - 4. 그리드 시스템
 - 4.1. Bootstrap의 그리드 시스템의 작동 방식

1. CSS

CSS 기능을 이용하기 위해서는 기본적으로 **HTML5**를 사용해야 한다.

1.1. CSS 기본설정

그러기 위해서는 **DOCTYPE**에 대해서 **HTML5**로 정의해야 한다. 가장 위에 문서 양식은 아래와 같이 정의하면 사용할 수 있다..

```
<!DOCTYPE html>
```

1.2. 모바일 우선기능

그리고 **Bootstrap 2**에서는 모바일은 옵션으로 다루고 있었지만, **Bootstrap 3**에서는 모바일 우선 (**mobile first**)인 프레임워크로 작성이 되었기 때문에 뷰포트에 대한 메타태그 설정을 하면 좋다. 물론 사이트에 따라서 이러한 뷰포트는 다르게 설정을 해도 되며, 기존의 사이트에 적용하는 경우에는 모바일에 대한 화면의 크기가 달라지기 때문에 예상하지 못하는 문제가 발생할 수 있으므로 조심해야 한다.

```
<meta name="viewport" content="width=device-width, initial-scale=1">
```

1.3. CSS 사용

이후에 **<head>** 태그 안에서 **CSS** 파일을 **link** 시키면 기본적인 **CSS** 기능들은 사용할 수 있다. 아래는 **CDN**에 있는 **Bootstrap CSS** 파일을 사용하는 것으로, 필요하면 개별적으로 서버에서 다운 받아서 사용하면 된다.

```
<link rel="stylesheet"
href="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.6/css/bootstrap.min.css"
integrity="sha384-
1q8mTJOASx8j1Au+a5WDVnPi2lkFfwwEAa8hDDdjZlpLegxhjVME1fgjWPGmkzs7"
crossorigin="anonymous">
```

2. bootstrap CSS 기반

부트스트랩의 **CSS**는 **Less**를 기반으로 만들어졌습니다. **Less**는 **CSS** 컴파일링을 위한 변수, 믹스인, 함수 같은 추가 기능을 갖춘 프리프로세서입니다. 컴파일된 **CSS** 파일들 대신 우리가 프레임워크 내에서 사용한 수많은 변수들과 믹스인들을 활용하는 **Less** 파일들을 사용하는 것을 고려해 보아야 합니다.

2.1. LESS

css는 **html**을 꾸며주는 언어입니다. 하지만 단점도 있는데요. 이를테면 작성하기는 쉽지만, **firebug**와 같은 도구 없이는 유지보수하는 것이 매우 어렵습니다. 또 동적인 언어의 특징인 변수나 함수와 같은 특성을 가지고 있지 않기 때문에 많은 양의 코드가 동원되기도 합니다. 이러한 문제를 해결하기 위해서 기술 중의 하나가 **lesscss**입니다. **lesscss**를 보다 간결하고 유지보수하기 쉬운 **css**를 만들 수 있습니다.

LESS는 NodeJS를 엔진으로 사용한다

3. Bootstrap 프레임워크 CSS 기능

Bootstrap 프레임워크의 **CSS** 기능은 웹페이지에 적용하게 되는 **CSS** 규칙들을 기본적으로 설정하고, 기본 **HTML** 요소들에 대한 스타일을 적용하는 기능들을 포함하고 있다. 특히, **Bootstrap** 프레임워크에서 제공해주고 있는 반응형 웹페이지의 가장 큰 기능인 그리드 형식으로 여러 칼럼으로 보여주다가 브라우저의 크기가 작아지면 세로로 나열시키는 기능이 **CSS** 기능에서 적용하면 된다. 양식이나 버튼들에 대해서도 다양한 **CSS** 설정을 기본적으로 해주니 기본 **UI** 디자인에서 조금 더 향상된 **CSS**를 체험할 수 있다.

4. 그리드 시스템

Bootstrap은 기기나 뷰포트 크기가 증가함에 따라 **12** 열이 적절하게 확대되는 반응형, 모바일 우선 유동 그리드 시스템입니다. 그것은 쉬운 레이아웃을 위해 미리 정해진 클래스들 뿐만 아니라 강력한 더 시멘

택한 레이아웃을 생성하기 위한 믹스인 을 포함하고 있습니다.

그리드 시스템은 콘텐츠를 보관할 행과 열 시리즈를 통해 페이지 레이아웃들 만드는데 사용되어집니다.

4.1. Bootstrap의 그리드 시스템의 작동 방식

- 행은 반드시 적절한 정렬과 패딩을 위해서 `.container (fixed-width)` 나 `.container-fluid (full-width)` 안에 위치해야 합니다.
- 열들의 수평그룹을 만드는데 행을 이용하세요.
- 콘텐츠는 열안에 위치해야 합니다. 그리고 열들만이 행의 바로 아래에 올 수 있습니다.
- `.row` 과 `.col-xs-4` 같은 사전정의된 그리드 클래스들은 간편하게 그리드 레이아웃 만드는 것을 가능하게 합니다. Less 믹스인은 좀 더 시멘틱한 레이아웃을 위해 사용되어질 수 있습니다.
- 열은 `padding` 으로 사이 간격을 만듭니다. 패딩은 행 내에서 첫열과 마지막열을 위해 `.row` 내에 음수 마진으로 `offset` 되어 있습니다.
- 음수 마진은 아래의 예제들이 내어쓰기가 되어 있는 이유입니다. 그것은 그리드 열 내의 콘텐츠는 비그리드 콘텐츠와 정렬되기 위함입니다.
- 그리드 열은 12개의 가능한 열들을 원하는 만큼 명시하는 것으로 만들어집니다. 예를 들면, 같은 크기의 3개 열은 `.col-xs-4` 를 3개 사용할 수 있습니다.
- 만약 한 행에 12열보다 더 많이 배치된다면, 남은 열들은, 하나의 유닛으로, 새로운 라인에 감싸집니다.
- 그리드 클래스는 분기점 크기보다 크거나 같은 너비의 화면을 가진 기기에 적용됩니다. 그리고 보다 작은 기기의 그리드 클래스가 오버라이드 됩니다. 그리하여, 예를 들어 요소에 `.col-md-*` 클래스를 적용하는 것은 중간 기기에 스타일이 효과가 있는 것뿐만 아니라 `.col-lg-*` 클래스가 없다면 큰 기기에도 효과가 있게 됩니다.