

김쌤과 함께 쉽고 빠르게  
온라인 교육 만들기



## 부트스트랩을 활용한 반응형 웹 만들기



## STEP 02

# 서브라임텍스트 설치와 반응형 코딩

## 1. 프리웨어 코딩 프로그램 설치

<http://www.sublimetext.com/3>

## 2. 플러그인 설치를 할 수 있게 세팅

<https://packagecontrol.io/>

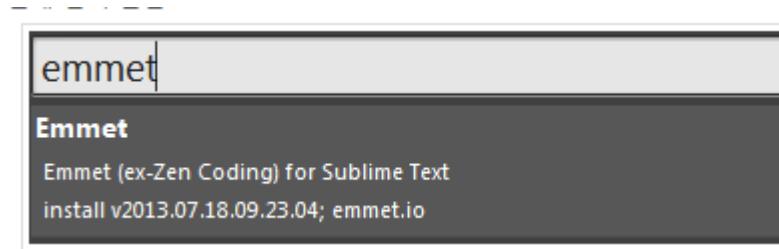
## 2. 플러그인 설치를 할 수 있게 세팅

<https://packagecontrol.io/>

- ① 메인페이지에 있는 install now 버튼을 클릭하고 sublime text 3 영역 안에 있는 내용을 복사한다.
- ② 서브라임에서 view>show console(ctrl+')클릭하여 콘솔창 열고 붙여 넣기
- ③ 서브라임 툴을 열고 tools>command palette(ctrl+shift+p)를 눌러 텍스트 입력창을 띄우고 install 을 입력하여 엔터후 설치하고 싶은 플러그인 이름을 입력!

## 2. 필요한 플러그인들을 차례대로 설치

- ✓ 각각의 플러그인 설치 시마다 **ctrl+shift+p**하여 입력창에 **install** 엔터 후 플러그인 입력 엔터



- ① emmet : 코딩 빠르게 해주는 플러그인
- ② bracket highlighter: 쌍으로 존재해야 하는 코드를 하이라이트로 처리해주는 플러그인
- ③ sidebar enhancement : 사이드 바의 기능을 강화해주는 플러그인
- ④ color picker : 색상 팔레트 설치하는 플러그인

### 3. 플러그인 설치 후 설정

- ✓ 사이드바의 기본 브라우저 설정  
(preference>package settings>side bar>settings user)

```
{  
  "default_browser": "chrome"  
}
```

### 3. 플러그인 설치 후 설정

- ✓ 사이드 바의 단축키 설정 변경

(preference>package settings>side bar>key binding- user)로 가서 아래코드 붙여넣기

```
[  
  { "keys": ["ctrl+b"], "command": "toggle_side_bar" },  
  { "keys": ["f12"], "command":  
"side_bar_open_in_browser" },  
  { "keys": ["ctrl+alt+n"], "command": "side_bar_new_file" },  
  { "keys": ["f2"], "command": "side_bar_rename" },  
  { "keys": ["ctrl+alt+m"], "command": "side_bar_move" },  
  { "keys": ["ctrl+alt+d"], "command": "side_bar_delete" }  
]
```

### 3. 플러그인 설치 후 설정

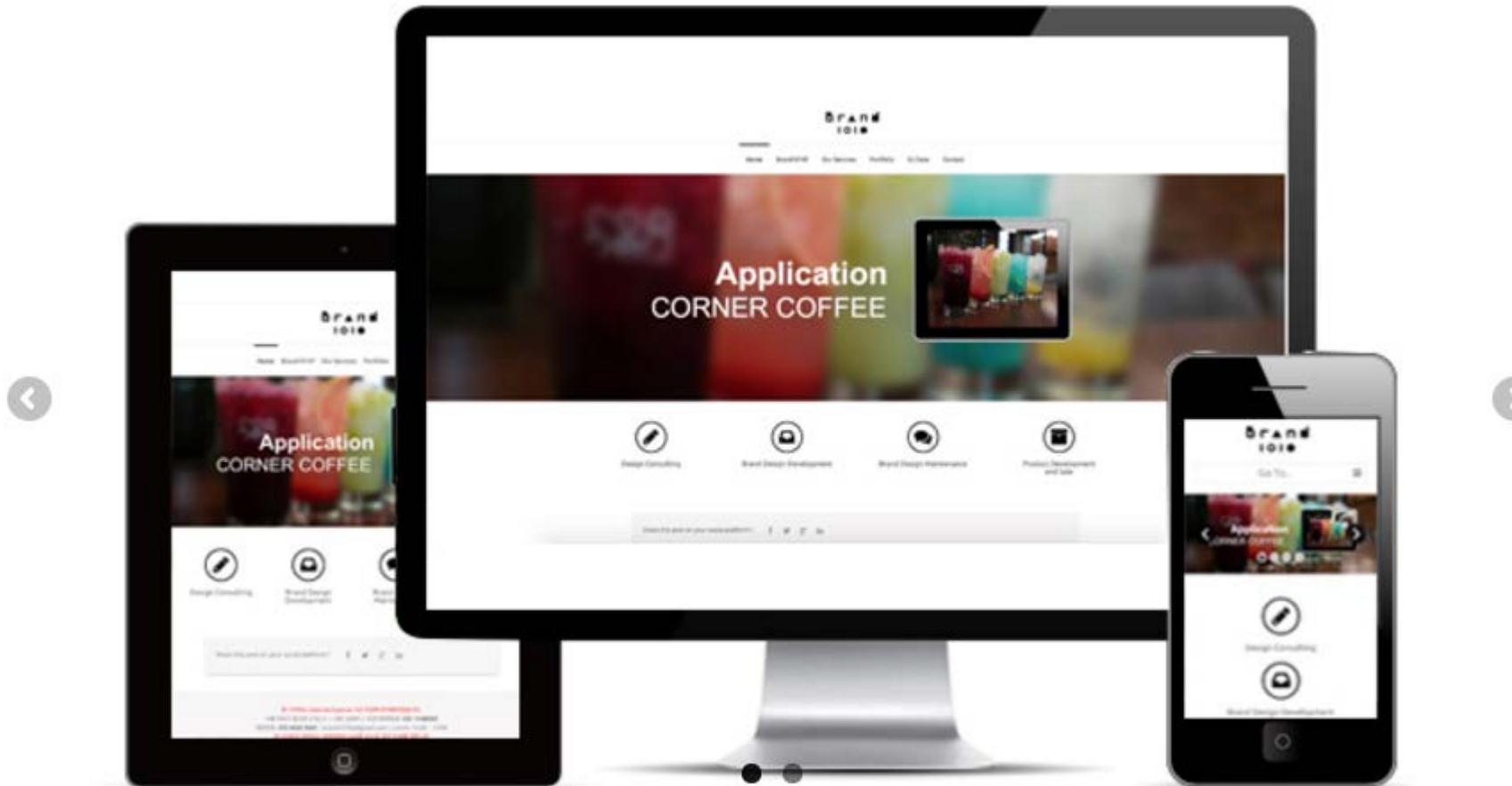
- ✓ 주 언어 설정을 한국어로 바꾸기

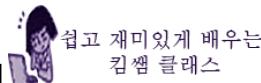
(preference>package settings>emmet>settings-user)로 가서 아래코드 붙여넣기

```
{  
  "snippets": {  
    "variables": {  
      "lang": "ko",  
      "locale": "ko-KR",  
      "charset": "UTF-8",  
      "indentation": "₩t",  
      "newline": "₩n"  
    },  
    "html": {  
      "abbreviations" : {  
        }  
    }  
  }  
}
```

## 4. CSS를 활용하여 반응형 웹 코딩하기 핵심포인트

- ✓ 디바이스 장치에 따라 레이아웃이 바뀌는 웹페이지





## 4. CSS를 활용하여 반응형 웹 코딩하기 핵심포인트 따라하기

## ✓ 미디어쿼리의 사용

@media 규칙을 이용하여 반응형 웹을 만들어 봅니다.

## 4. CSS를 활용하여 반응형 웹 코딩하기 핵심포인트 따라하기

- ① 아래 메타태그 꼭 포함하기

```
<meta name="viewport" content="width=device-width, initial-scale=1.0" />
```

## 4. CSS를 활용하여 반응형 웹 코딩하기 핵심포인트 따라하기

- ① 아래 메타태그 꼭 포함하기

```
<meta name="viewport" content="width=device-width, initial-scale=1.0" />
```

# 4. CSS를 활용하여 반응형 웹 코딩하기 핵심포인트 따라하기

## ② 다양한 디바이스 사이즈 별 미디어쿼리 예시

```
@media screen and (min-width: 1600px){  
    css 스타일  
}  
@media screen and (min-width: 1280px) and (max-width: 1599px){  
    css 스타일  
}  
@media screen and (min-width: 1024px) and (max-width: 1279px){  
    css 스타일  
}  
@media screen and (min-width: 780px) and (max-width: 1023px){  
    css 스타일  
}  
@media screen and (min-width: 640px) and (max-width: 779px){  
    css 스타일  
}  
@media screen and (min-width: 480px) and (max-width: 639px){  
    css 스타일  
}  
@media screen and (max-width: 479px){  
    css 스타일  
}
```

# 4. CSS를 활용하여 반응형 웹 코딩하기 핵심포인트 따라하기

## ③ 원하는 디바이스 선택하여 미디어 쿼리별 스타일 코딩하기

```
/*디바이스 장치가 최소 1200 이상일 때 :laptop컴퓨터 큰 모니터용*/
@media (min-width: 1200px){
    html, body{background-color: purple}
}

/*디바이스 장치가 최소 992 이상일 때 :laptop컴퓨터 작은 모니터용*/
@media (min-width: 992px){
    #wrap{width: 80%; }
}

/*디바이스 장치가 최소 768 이상일 때 :타블렛용*/
@media (min-width:768px {
    html, body{background-color: orange}
    #content_wrap #content{font-size: 1.5em;}
}

/*디바이스 장치가 최대 767 이하일 때 :스마트폰용*/
@media (max-width:767px) {
    html, body{background-color: pink}
    #content_wrap #content{font-size: 2em; }
    nav ul li{width: 100%;}
}
```

# 4. CSS를 활용하여 반응형 웹 코딩하기 핵심포인트 따라하기

- ④ 모니터 장치일 때의 스타일을 코딩해 봅시다.

/\*디바이스 장치가 최소 992 이상일 때 :laptop 컴퓨터용\*/

```
@media (min-width: 992px){  
    #wrap{width: 80%; }  
}
```

**반응형 웹**

menu1	menu2	menu3	menu4	menu5
-------	-------	-------	-------	-------

**반응형 웹은 미디어쿼리를 사용하여 스타일링합니다.**

Lorem ipsum dolor sit amet, consectetur adipisicing elit. Expedita perferendis porro aliquam dolorem inventore fuga, deleniti corporis quaerat quis vero recusandae laboriosam qui maiores blanditiis minus corrupti ipsum, possimus accusantium ex ducimus voluptate molestias sit, facere. Aliquam cumque accusamus maiores consectetur earum. Culpa sint earum, modi rem nobis! Modi asperiores architecto ratione illum adipisci repellat aspernatur quisquam, quas vel sunt deserunt nostrum error laudantium dicta dignissimos laborum placeat. Tenetur soluta amet, itaque dolor enim ea odit, fugit officiis delectus voluptas suscipit, in similique nam? Quaerat, iusto dicta facere quis porro voluptatem accusantium, consequuntur modi, enim repudiandae illo inventore quas accusamus.

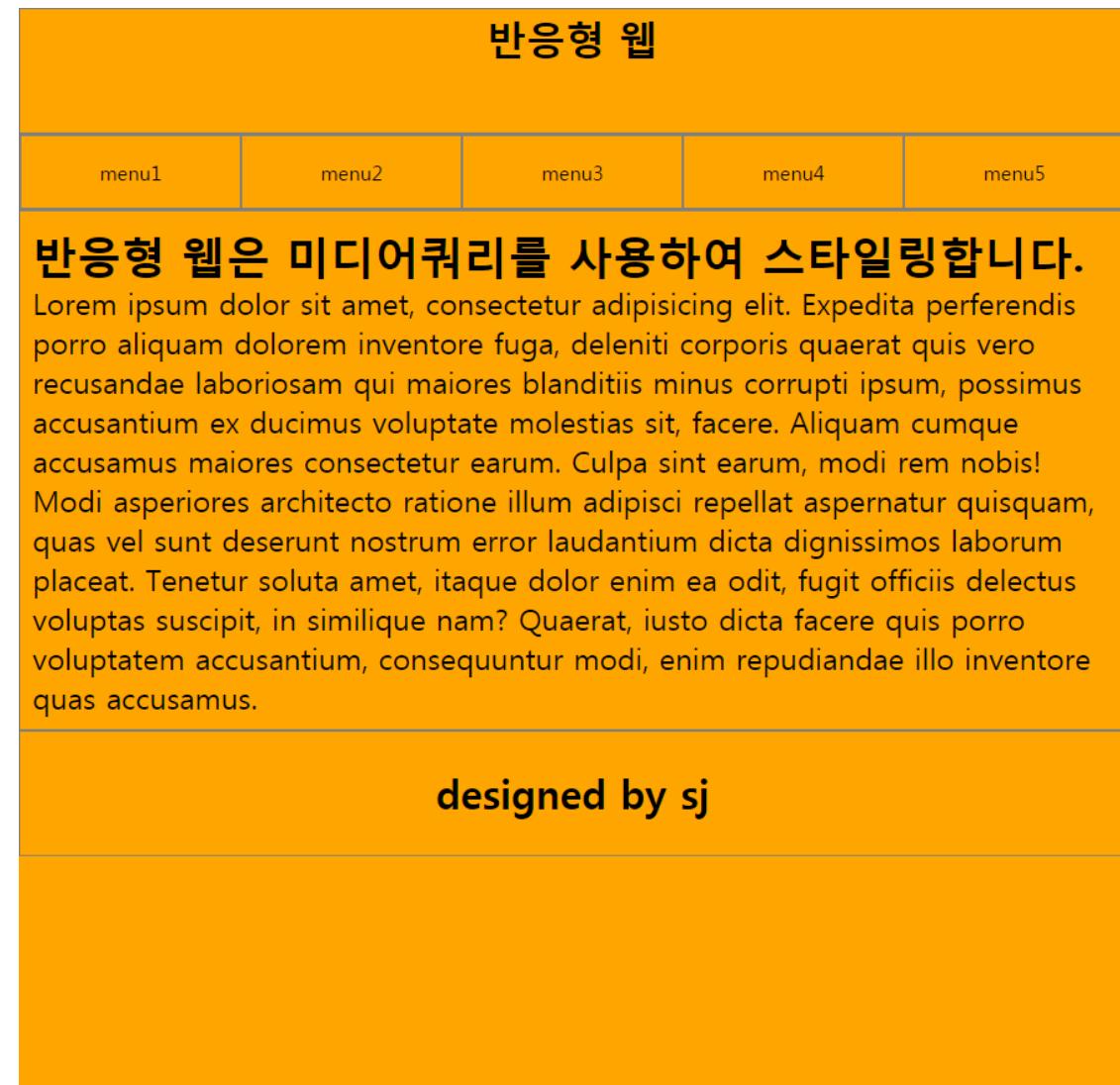
designed by sj

## 4. CSS를 활용하여 반응형 웹 코딩하기 핵심포인트 따라하기

- ⑤ 타블렛 장치일 때의 스타일을 코딩해 봅시다.

/\*디바이스 장치가 최소 768 이상일 때 :  
타블렛용\*/

```
@media (min-width:768px) {  
    html, body{background-color: orange}  
    #content_wrap #content{font-size:  
        1.5em;}  
}
```



반응형 웹

menu1 menu2 menu3 menu4 menu5

반응형 웹은 미디어쿼리를 사용하여 스타일링합니다.

Lorem ipsum dolor sit amet, consectetur adipisicing elit. Expedita perferendis porro aliquam dolorem inventore fuga, deleniti corporis quaerat quis vero recusandae laboriosam qui maiores blanditiis minus corrupti ipsum, possimus accusantium ex ducimus voluptate molestias sit, facere. Aliquam cumque accusamus maiores consectetur earum. Culpa sint earum, modi rem nobis! Modi asperiores architecto ratione illum adipisci repellat aspernatur quisquam, quas vel sunt deserunt nostrum error laudantium dicta dignissimos laborum placeat. Tenetur soluta amet, itaque dolor enim ea odit, fugit officiis delectus voluptas suscipit, in similique nam? Quaerat, iusto dicta facere quis porro voluptatem accusantium, consequuntur modi, enim repudiandae illo inventore quas accusamus.

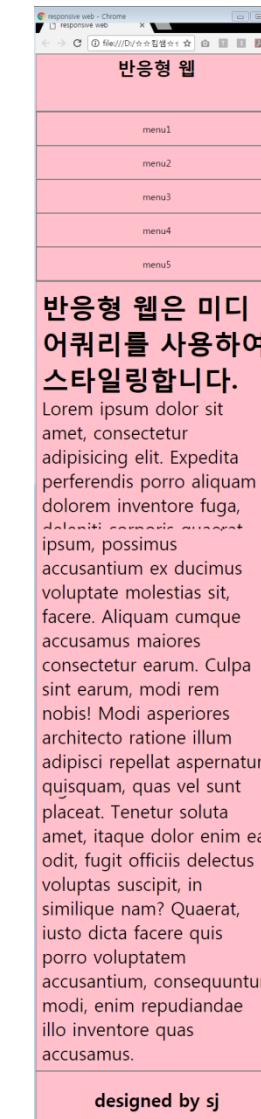
designed by sj

# 4. CSS를 활용하여 반응형 웹 코딩하기 핵심포인트 따라하기

⑥ 스마트폰 일때의 스타일을 코딩해 봅시다.

/\*디바이스 장치가 최대 767 이하일 때 :  
스마트폰 용\*/

```
@media (max-width:767px){  
    html, body{background-color: pink}  
    #content_wrap #content{font-size:  
        2em; }  
    nav ul li{width: 100%;}  
}
```



## ✓ 꼭 기억해야 할 포인트 정리

1. 디바이스 장치에서 레이아웃이 달라지는 반응형 웹페이지를 만들 때 기본적으로 꼭 포함해야 하는 meta태그에서 아래 빈칸에 들어가는 코딩은 무엇일까?

< meta [ ] , [ ], initial-scale=1.0" />

2. 미디어쿼리 규칙의 코딩은 [ ] (디바이스 사이즈){  
    css 스타일  
} 이다.

3. 부트스트랩에서는 디바이스 장치의 사이즈를 크게 3개로 나누는데 그 중에 laptop 컴퓨터용 사이즈는 디바이스 사이즈가 최소 [ ] px 이상일 때이고, 스마트폰은 디바이스 사이즈가 최대 [ ] px 이상일 때이다.

## ✓ 꼭 기억해야 할 포인트 정리

1. 디바이스 장치에서 레이아웃이 달라지는 반응형 웹페이지를 만들 때 기본적으로 꼭 포함해야 하는 meta태그에서 아래 빈칸에 들어가는 코딩은 무엇일까?

```
< meta name="viewport" content="width=device-width, initial-scale=1.0" />
```

2. 미디어쿼리 규칙의 코딩은  (디바이스 사이즈){  
 CSS 스타일  
} 이다.

3. 부트스트랩에서는 디바이스 장치의 사이즈를 크게 3개로 나누는데 그 중에 laptop 컴퓨터용 사이즈는 디바이스 사이즈가 최소  px 이상일 때이고, 스마트폰은 디바이스 사이즈가 최대  px 이상일 때이다.

## ✓ 꼭 기억해야 할 포인트 정리

1. 디바이스 장치에서 레이아웃이 달라지는 반응형 웹페이지를 만들 때 기본적으로 꼭 포함해야 하는 meta태그에서 아래 빈칸에 들어가는 코딩은 무엇일까?

```
< meta name="viewport" content="width=device-width, initial-scale=1.0" />
```

2. 미디어쿼리 규칙의 코딩은 @media (디바이스 사이즈){  
 CSS 스타일  
} 이다.

3. 부트스트랩에서는 디바이스 장치의 사이즈를 크게 3개로 나누는데 그 중에 laptop 컴퓨터용 사이즈는 디바이스 사이즈가 최소  px 이상일 때이고, 스마트폰은 디바이스 사이즈가 최대  px 이상일 때이다.

## ✓ 꼭 기억해야 할 포인트 정리

1. 디바이스 장치에서 레이아웃이 달라지는 반응형 웹페이지를 만들 때 기본적으로 꼭 포함해야 하는 meta태그에서 아래 빈칸에 들어가는 코딩은 무엇일까?

```
< meta name="viewport" content="width=device-width, initial-scale=1.0" />
```

2. 미디어쿼리 규칙의 코딩은 `@media` (디바이스 사이즈){  
    css 스타일  
} 이다.

3. 부트스트랩에서는 디바이스 장치의 사이즈를 크게 3개로 나누는데 그 중에 laptop 컴퓨터용 사이즈는 디바이스 사이즈가 최소 `992` px 이상일 때이고, 스마트폰은 디바이스 사이즈가 최대 `767` px 이상일 때이다.

