

10-3

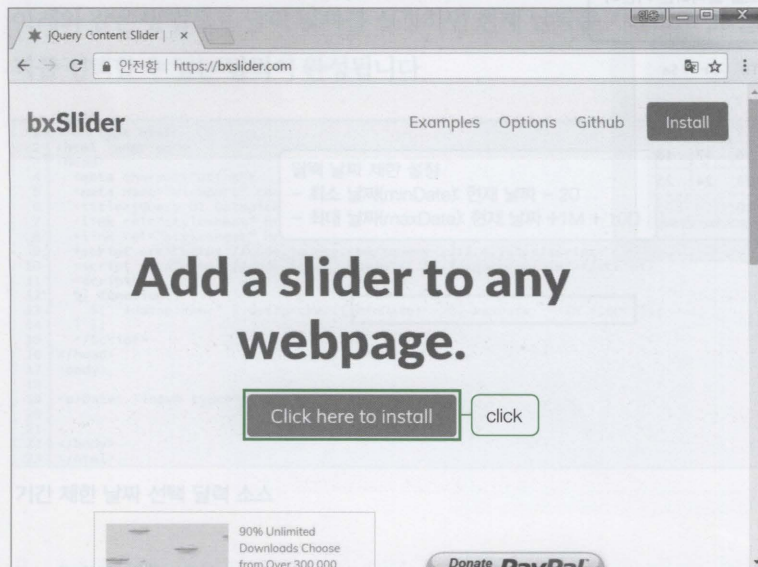
bxSlider 플러그인 활용하기

bxSlider 플러그인

bxSlider 플러그인은 여러 개의 배너에 슬라이드 기능을 간편하게 적용할 수 있는 플러그인입니다. 또한 bxSlider 플러그인은 모바일에도 최적화되어 있어 슬라이드 기능도 동일하게 동작하며 반응형 디자인에도 최적화되어 있습니다.

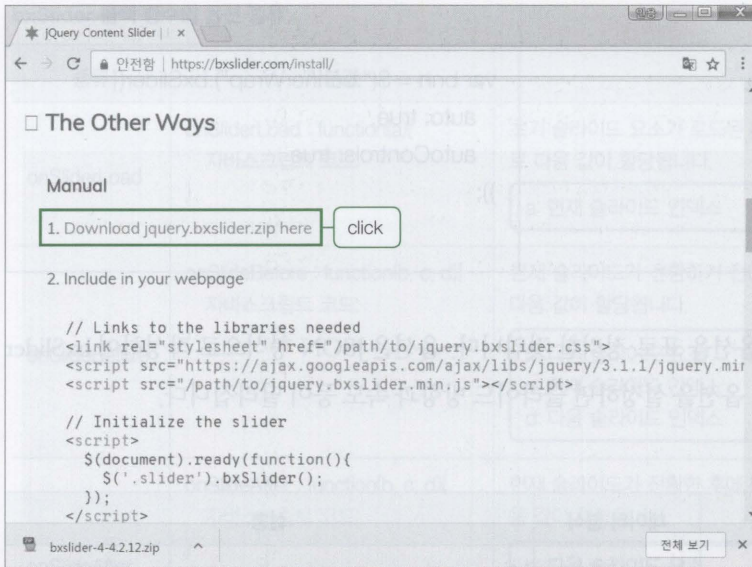
bxSlider 플러그인 설치하기

1. bxSlider(<http://bxslider.com/>) 홈페이지에서 [Click here to install] 버튼을 클릭합니다.



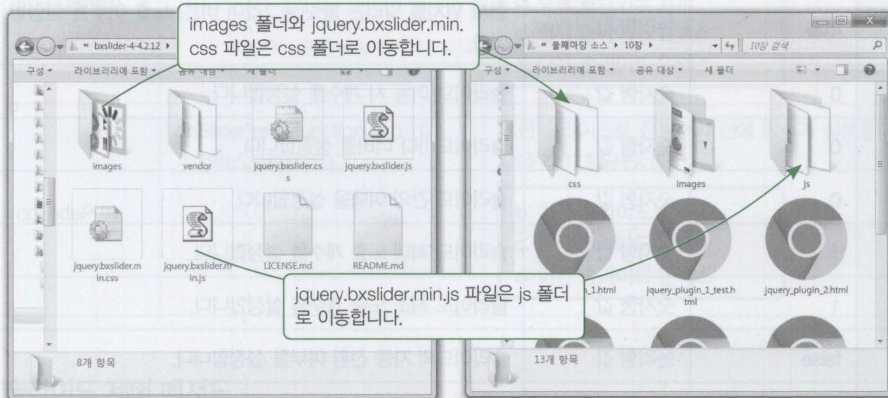
bxSlider(<http://bxslider.com/>)

2. 'install' 페이지에서 [Download jquery.bxslider.zip here]를 클릭하면 플러그인 파일을 내려받을 수 있습니다.



bxSlider 플러그인 내려받기

3. 내려받은 압축 파일의 압축을 풀어 'dist' 폴더로 옮깁니다. 압축용 css 파일과 'images' 폴더는 html 파일이 저장되어 있는 css 폴더로, js 파일은 js 폴더로 옮깁니다.



플러그인 파일 옮기기

다음은 bxSlider 플러그인의 기본형입니다.

기본형 참조 변수 = `$(“요소 선택”).bxSlider({`
 옵션 설정,
 `});`

사용 방법

<pre><ul class="bannerWrap"> 배너 콘텐츠 1 배너 콘텐츠 2 배너 콘텐츠 3 </pre>	<pre>var bnn = \$(".bannerWrap").bxSlider({ auto: true, autoControls: true });</pre>
--	--

다음은 bxSlider 플러그인 옵션을 표로 정리한 것입니다. 옵션은 JSON 형식으로 작성하여 bxSlider 메서드에 전달하면 됩니다. 옵션을 설정하면 슬라이드 방향과 속도 등이 달라집니다.

bxSlider 옵션 종류

종류	기본값	데이터 형식	설명
mode	"horizontal"	"horizontal" "vertical" "fade"	슬라이드 이동 방향을 설정합니다.
speed	500	숫자형 값	슬라이드 전환 시 속도를 설정합니다.
pause	4000	숫자형 값	슬라이드 자동 실행 전환 사이의 시간을 설정합니다.
pager	true	논리형 값	현재 위치를 알리는 볼릿(동그라미 버튼) 노출 여부를 설정합니다.
moveSlides	0	숫자형 값	슬라이드 이동 시 개수를 설정합니다.
sliderWidth	0	숫자형 값	슬라이드마다 너비를 설정합니다.
slideMargin	0	숫자형 값	슬라이드 간의 여백을 설정합니다.
maxSlides	1	숫자형 값	슬라이드 최대 노출 개수를 설정합니다.
minSlides	1	숫자형 값	슬라이드 최소 노출 개수를 설정합니다.
auto	false	논리형 값	슬라이드의 자동 전환 여부를 설정합니다.
autoHover	false	논리형 값	슬라이드에 마우스를 올렸을 때 자동 전환을 멈추게 할 것인지의 여부를 설정합니다.
controls	true	논리형 값	이전/다음 버튼의 노출 여부를 설정합니다.

슬라이드 콜백 함수

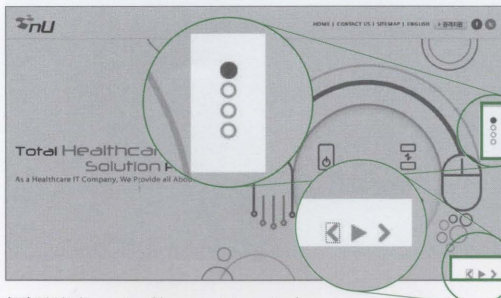
다음은 슬라이드에 콜백 함수를 설정하는 옵션입니다. 이 옵션은 슬라이드 요소가 완전히 로드된 다음 슬라이드 전환 전, 후에 콜백 함수를 호출하는 방식으로 적용할 수 있습니다.

bxSlider 콜백 함수의 옵션 종류

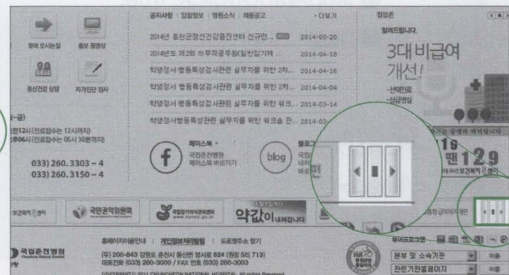
종류	기본값	설명
onSliderLoad	onSliderLoad : function(a){ 자바스크립트 코드; }	초기 슬라이드 요소가 로드된 다음 함수가 실행됩니다. 매개변수로 다음 값이 할당됩니다. a: 현재 슬라이드 인덱스
onSlideBefore	onSlideBefore : function(b, c, d){ 자바스크립트 코드; }	현재 슬라이드가 전환하기 전에 함수가 실행됩니다. 매개변수로 다음 값이 할당됩니다. b: 다음 슬라이드 요소 c: 현재 슬라이드 인덱스 d: 다음 슬라이드 인덱스
onSlideAfter	onSlideAfter : function(b, c, d){ 자바스크립트 코드; }	현재 슬라이드가 전환한 후에 함수가 실행됩니다. 매개변수로 다음 값이 할당됩니다. b: 다음 슬라이드 요소 c: 현재 슬라이드 인덱스 d: 다음 슬라이드 인덱스
onSlideNext	onSlideNext : function(b, c, d){ 자바스크립트 코드; }	다음 슬라이드로 전환하기 전에 함수가 실행됩니다. 매개변수로 다음 값이 할당됩니다. b: 다음 슬라이드 요소 c: 현재 슬라이드 인덱스 d: 다음 슬라이드 인덱스
onSlidePrev	onSlidePrev : function(b, c, d){ 자바스크립트 코드; }	이전 슬라이드로 전환하기 전에 함수가 실행됩니다. 매개변수로 다음 값이 할당됩니다. b: 이전 슬라이드 요소 c: 현재 슬라이드 인덱스 d: 이전 슬라이드 인덱스

슬라이드 제어 메서드

슬라이드 제어 메서드는 다음 그림과 같이 배너에 제어 버튼을 만들 때 사용합니다. 예를 들어 이전 버튼 또는 다음 버튼을 만들 때나 정지 버튼, 재생 버튼을 만들 때 사용할 수 있습니다.



(주)평화이즈: <http://www.phis.go.kr/>



국립춘천병원: <http://www.cnmh.go.kr/>

사용 방법

<pre><ul class="bannerWrap"> 배너 콘텐츠 1 배너 콘텐츠 2 배너 콘텐츠 3 </pre>	<pre>var bnn = \$(".bannerWrap").bxSlider({ auto: true, autoControls: true });</pre>
--	--

다음은 bxSlider 플러그인 옵션을 표로 정리한 것입니다. 옵션은 JSON 형식으로 작성하여 bxSlider 메서드에 전달하면 됩니다. 옵션을 설정하면 슬라이드 방향과 속도 등이 달라집니다.

bxSlider 옵션 종류

종류	기본값	데이터 형식	설명
mode	"horizontal"	"horizontal" "vertical" "fade"	슬라이드 이동 방향을 설정합니다.
speed	500	숫자형 값	슬라이드 전환 시 속도를 설정합니다.
pause	4000	숫자형 값	슬라이드 자동 실행 전환 사이의 시간을 설정합니다.
pager	true	논리형 값	현재 위치를 알리는 볼릿(동그라미 버튼) 노출 여부를 설정합니다.
moveSlides	0	숫자형 값	슬라이드 이동 시 개수를 설정합니다.
sliderWidth	0	숫자형 값	슬라이드마다 너비를 설정합니다.
slideMargin	0	숫자형 값	슬라이드 간의 여백을 설정합니다.
maxSlides	1	숫자형 값	슬라이드 최대 노출 개수를 설정합니다.
minSlides	1	숫자형 값	슬라이드 최소 노출 개수를 설정합니다.
auto	false	논리형 값	슬라이드의 자동 전환 여부를 설정합니다.
autoHover	false	논리형 값	슬라이드에 마우스를 올렸을 때 자동 전환을 멈추게 할 것인지의 여부를 설정합니다.
controls	true	논리형 값	이전/다음 버튼의 노출 여부를 설정합니다.

슬라이드 콜백 함수

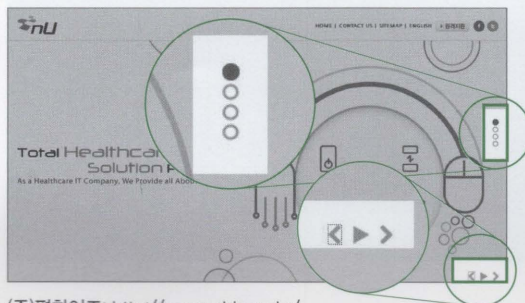
다음은 슬라이드에 콜백 함수를 설정하는 옵션입니다. 이 옵션은 슬라이드 요소가 완전히 로드된 다음 슬라이드 전환 전, 후에 콜백 함수를 호출하는 방식으로 적용할 수 있습니다.

bxSlider 콜백 함수의 옵션 종류

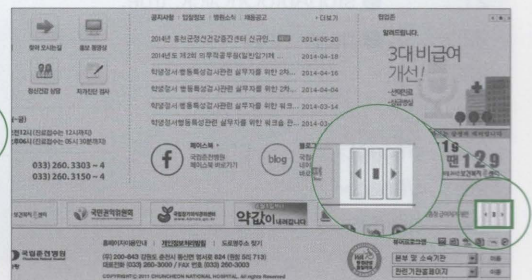
종류	기본값	설명
onSliderLoad	onSliderLoad : function(a){ 자바스크립트 코드; }	초기 슬라이드 요소가 로드된 다음 함수가 실행됩니다. 매개변수로 다음 값이 할당됩니다. a: 현재 슬라이드 인덱스
onSlideBefore	onSlideBefore : function(b, c, d){ 자바스크립트 코드; }	현재 슬라이드가 전환하기 전에 함수가 실행됩니다. 매개변수로 다음 값이 할당됩니다. b: 다음 슬라이드 요소 c: 현재 슬라이드 인덱스 d: 다음 슬라이드 인덱스
onSlideAfter	onSlideAfter : function(b, c, d){ 자바스크립트 코드; }	현재 슬라이드가 전환한 후에 함수가 실행됩니다. 매개변수로 다음 값이 할당됩니다. b: 다음 슬라이드 요소 c: 현재 슬라이드 인덱스 d: 다음 슬라이드 인덱스
onSlideNext	onSlideNext : function(b, c, d){ 자바스크립트 코드; }	다음 슬라이드로 전환하기 전에 함수가 실행됩니다. 매개변수로 다음 값이 할당됩니다. b: 다음 슬라이드 요소 c: 현재 슬라이드 인덱스 d: 다음 슬라이드 인덱스
onSlidePrev	onSlidePrev : function(b, c, d){ 자바스크립트 코드; }	이전 슬라이드로 전환하기 전에 함수가 실행됩니다. 매개변수로 다음 값이 할당됩니다. b: 이전 슬라이드 요소 c: 현재 슬라이드 인덱스 d: 이전 슬라이드 인덱스

슬라이드 제어 메서드

슬라이드 제어 메서드는 다음 그림과 같이 배너에 제어 버튼을 만들 때 사용합니다. 예를 들어 이전 버튼 또는 다음 버튼을 만들 때나 정지 버튼, 재생 버튼을 만들 때 사용할 수 있습니다.



(주)평화이즈: <http://www.phis.go.kr/>



국립춘천병원: <http://www.cnmh.go.kr/>

다음은 슬라이드 제어 메서드의 종류를 정리한 표입니다.

슬라이드 제어 메서드 종류

종류	설명
참조 변수.goToSlide(number);	지정한 슬라이드로 이동합니다.
참조 변수.goToNextSlide();	다음 슬라이드로 이동합니다.
참조 변수.goToPrevSlide();	이전 슬라이드로 이동합니다.
참조 변수.startAuto();	자동으로 슬라이드가 전환됩니다.
참조 변수.stopAuto();	자동으로 슬라이드가 전환되는 것을 정지시킵니다.
참조 변수.getCurrentSlide();	전환된 슬라이드의 인덱스값을 반환합니다.
참조 변수.getSlideCount();	전체 슬라이드의 개수를 반환합니다.
참조 변수.reloadSlider();	bxSlider() 메서드를 초기화한 다음 다시 적용합니다.
참조 변수.destroySlider();	적용된 bxSlider() 메서드를 제거합니다.

자동 슬라이드 갤러리 만들기

이번에는 갤러리 목록 태그에 bxSlider 플러그인을 적용하여 자동 슬라이드 갤러리를 만들어 보겠습니다.

코딩해 보세요!

• 실습 파일 jquery_plugin_3_test.html • 완성 파일 jquery_plugin_3.html

```

06: <script src="js/jquery.js"></script>
07: <script src="js/jquery.bxslider.min.js"></script>
08: <link rel="stylesheet" href="css/jquery.bxslider.min.css">
...
16: <script>
17: $(function() {
18:     $('.slide_gallery').bxSlider({
19:         ❶ auto: true,
20:         ❷ autoControls: true,
21:         ❸ stopAutoOnClick: true,
22:         ❹ pager: true
23:     });
24: });
25: </script>
26: </head>
27: <body>

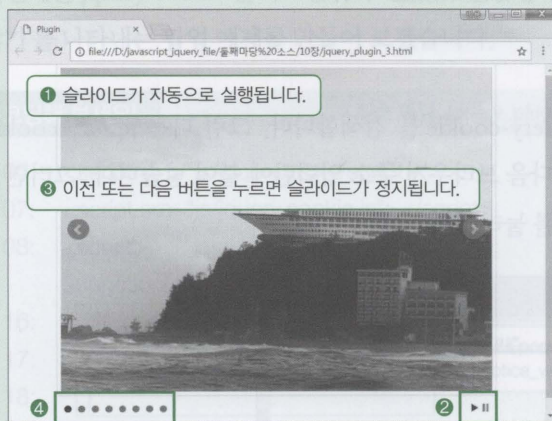
```

```

28: <div id="gallery_wrap">
29:   <ul class="slide_gallery">
30:     <li></li>
31:     <li></li>
32:     <li></li>
33:     <li></li>
34:     <li></li>
35:     <li></li>
36:     <li></li>
37:     <li></li>
38:   </ul>
39: </div>
40: </body>

```

:: 결과 화면



```

20: var $expOnk = $(expressOnk);
21: $("doPostBack").on("click", doPostBack);
22:
23: function doPostBack() {
24:   if($expOnk.is(":checked")) {
25:     $.cookie("doPostBack", "none", {expires: 3, path: "/"});
26:   }
27:   $("#gallery_wrap").slideToggle("fast");
28: }
29:
30: </script>
31: </body>

```