

WEB STANDARD



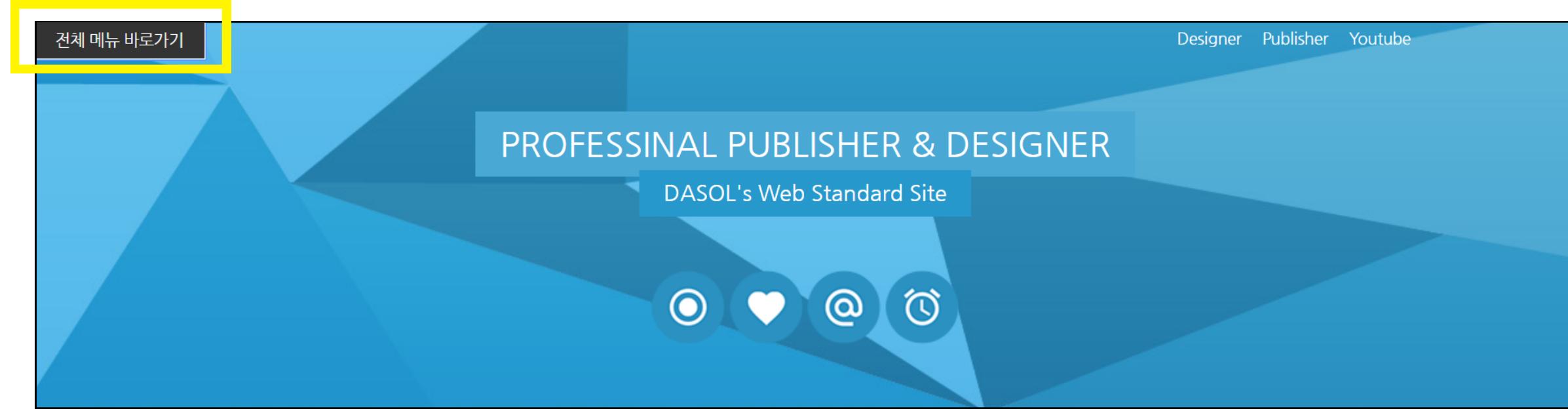
웹표준 지침서를 준수한 사이트입니다.

대체 텍스트를 사용하여, 모든 사용자의 접근성을 고려했습니다.

아코디언메뉴, 탭메뉴 등을 jQuery로 코딩하여 만들었습니다. **slick, light box** 제이쿼리 플러그인을 사용하여 배너와 이미지에 동적 효과를 주었습니다.

Contents

- 1. Skip 메뉴
- 2. IR 기법 -Phark Method
- 3. Image Sprite
- 4. 아코디언 메뉴 -jQuery
- 5. IR 기법
- 6. slick.js -jQuery
- 7. float: left;
- 8. 한 줄 효과
- 9. Position
- 10. 팝업 효과 -jQuery
- 11. Mouse Hover 효과
- 12. Tab 메뉴
- 13. :before
- 14. W3C 검사



HTML

```
<!-- 스킵 네비게이션 -->
<div id="skip">
    <a href="#cont_nav">전체 메뉴 바로가기</a>
    <a href="#cont_ban">배너 메뉴 바로가기</a>
    <a href="#cont_cont">컨텐츠 메뉴 바로가기</a>
</div>
<!-- //스킵 네비게이션 -->
```

CSS

```
/* 스킵 네비게이션 */
#skip { position: relative; }
#skip a { position: absolute; left: 0px; top: -35px;
border: 1px solid #fff; color: #fff; background:
#333; line-height: 30px; width: 140px; text-align:
center; }
#skip a:active, #skip a:focus { top: 0px; }
```

Skip 메뉴

skip 네비게이션을 추가하여 접근성을 높였습니다. 여러 링크를 거치지 않고 원하는 영역의 본문으로 바로 이동할 수 있도록 했습니다. position top 값을 -35px에서 0px로 이동하도록 조절하여 화면에 나타냈습니다.

02

IR 기법 - Phark Method



HTML

```
<div class="header-icon">
    <a href="#" class="icon1"><span class="ir_pm">icon1</span></a>
    <a href="#" class="icon2"><span class="ir_pm">icon2</span></a>
    <a href="#" class="icon3"><span class="ir_pm">icon3</span></a>
    <a href="#" class="icon4"><span class="ir_pm">icon4</span></a>
</div>
<!-- //헤더 아이콘 --&gt;</pre>

```

CSS

```
/* IR 효과 */
.ir_pm {display:block; overflow:hidden; font-size:0; line-height:0; text-indent:-9999px;}
.ir_wa {display:block; overflow:hidden; position:relative; z-index:-1; width:100%; height: 100%;}
.ir_su {overflow: hidden; position:absolute; width:0; height:0; line-height:0; text-indent:-9999px;}
```

CSS

```
.header .header-icon { text-align: center; margin-top: 30px;}
.header .header-icon a { display: inline-block; width: 60px; height: 60px;
background: url(..../img/icon.png);}
.header .header-icon a.icon1 { background-position: 0 0; }
.header .header-icon a.icon2 { background-position: 0 -60px; }
.header .header-icon a.icon3 { background-position: 0 -120px; }
.header .header-icon a.icon4 { background-position: 0 -180px; }
.header .header-icon a.icon1:hover { background-position: -60px 0; }
.header .header-icon a.icon2:hover { background-position: -60px -60px; }
.header .header-icon a.icon3:hover { background-position: -60px -120px; }
.header .header-icon a.icon4:hover { background-position: -60px -180px; }
```

의미있는 이미지의 대체 텍스트를 제공하고자 Phark Method를 사용했습니다. **text-indent**를 이용하여, 텍스트를 화면 바깥으로 -9999px 만큼 빼내어 보이지 않게 했습니다.

03

Image Sprite

여러 이미지가 모인, 하나의 이미지만을 사용했습니다. class name과 mouse hover에 따라 **background-position** 값을 달리했습니다.

04

아코디언 메뉴- jQuery



jQery

```
//아코디언 메뉴
$(".tit .btn").click(function(e){
    e.preventDefault();
    $("#cont_nav").slideToggle(200);
    $(this).toggleClass("on");
});
```

HTML

```
<div class="container">
    <h2 class="ir_su">전체 메뉴</h2>
    <div class="nav">...
    </div>
</div>
```

CSS

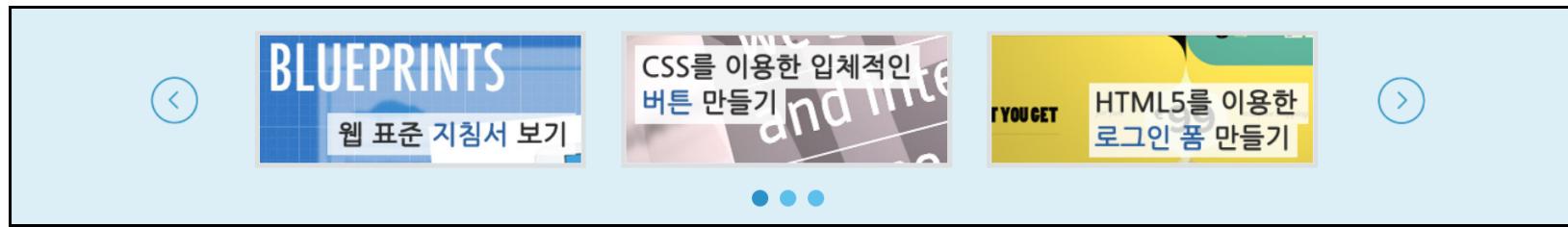
```
/* IR 효과 */
.ir_pm {display:block; overflow:hidden; font-size:0; line-height:0; text-indent:-9999px;}
.ir_wa {display:block; overflow:hidden; position:relative; z-index:-1; width:100%; height: 100%;}
.ir_su {overflow: hidden; position:absolute; width:0; height:0; line-height:0; text-indent:-9999px;}
```

버튼을 클릭했을 때 메뉴가 자연스럽게 내려오도록 **slideToggle**을 사용했습니다. 메뉴가 내려왔을 때, **toggleClass**로 **on**을 추가하여 버튼의 모양을 바꿨습니다. 버튼은 **image sprite**를 사용했습니다.

05

IR 기법

숨김 텍스트를 제공하여 접근성을 높였습니다. **width**, **height**를 0, **text-indent** 값을 -9999px로 주어, 화면에 보이지 않게 했습니다.



slick.js- jQuery 06

Notice
웹 페이지에서 가장 기본이 되는 게시판 유형입니다.

jQuery Plugin

포트폴리오

```
//배너
$(".ban").slick({
    infinite: true,
    slidesToShow: 3,
    slidesToScroll: 3,
    autoplay: true,
    autoplaySpeed: 3000,
    dots: true
});
```

```
//갤러리
$(".gallery_img").slick({
    arrows: false,
    fade: true,
    pauseOnHover: true,
    infinite: true,
    autoplay: true,
    autoplaySpeed: 3000,
    speed: 300
});

$(".play").click(function(){
    $(".gallery_img").slick("slickPlay");
});
$(".pause").click(function(){
    $(".gallery_img").slick("slickPause");
});
$(".prev").click(function(){
    $(".gallery_img").slick("slickPrev");
});
$(".next").click(function(){
    $(".gallery_img").slick("slickNext");
});
```

jQuery 플러그인, slick.js를 각 스타일에 맞게 배너와 6번째 게시판에 사용했습니다.

1) 배너

3개의 사진이 보이도록 slidesToShow: 3으로 설정했고 dots를 true로 살렸습니다.

2) 6번째 게시판(갤러리)

arrows를 false로, pauseOnHover를 true로 설정하여 play 버튼을 살렸습니다. 각 버튼을 클릭했을 때 해당 기능이 실행되도록 함수를 추가했습니다.

각 버튼은 image sprite 기법을 사용했으며, mouse hover에 따라 색을 달리했습니다.

07

float: left;

The screenshot displays a website layout with six columns. From left to right: 1. A 'Notice' section with a blue icon and text. 2. A 'Web Publisher Notice' section with a blue icon and a list of tips. 3. A 'Web Designer Notice' section with a blue icon and a list of tips. 4. A 'Notice' section with a blue icon and text. 5. A 'Mouse Hover' section with a blue icon and a grid of images for '와이어 프레임', '스케치 작업', 'UI 디자인', 'HTML', 'CSS', and 'jQuery'. 6. A 'Notice' section with a blue icon and text. Below these columns are two additional sections: 'HTML Reference' and 'Advertisement'.

HTML

```
<div class="cont">
  <div class="column col1"> ...
  </div>
  <!-- //col1 -->
  <div class="column col2"> ...
  </div>
  <!-- //col2 -->
  <div class="column col3"> ...
  </div>
  <!-- //col3 -->
  <div class="column col4"> ...
  </div>
  <!-- //col4 -->
  <div class="column col5"> ...
  </div>
  <!-- //col5 -->
  <div class="column col6"> ...
  </div>
  <!-- //col6 -->
</div>
```

가장 기본적인 게시판 유형 6가지입니다. 각 게시판에 공통의 class name인 column과 비공통의 col1 ~ 6을 주어 CSS 코딩을 용이하게 했습니다. 게시판의 배열을 위해 .column에 float: left를 사용하였고 상위 요소에 overflow: hidden을 주었습니다.

css

```
/* 컨텐츠 */
.cont { overflow: hidden; padding-top: 30px; }
.cont .column { position: relative; float: left; margin: 0 30px 30px 0;
padding-right: 30px; width: 289px; height: 363px; }
```

08 한 줄 효과



CSS

```
.notice ul li { overflow: hidden; text-overflow: ellipsis; white-space: nowrap;
background: url(..../img/dot.gif) no-repeat 0 8px; padding-left: 8px;}
.notice ul li a { font-size: 12px;}
```

CSS

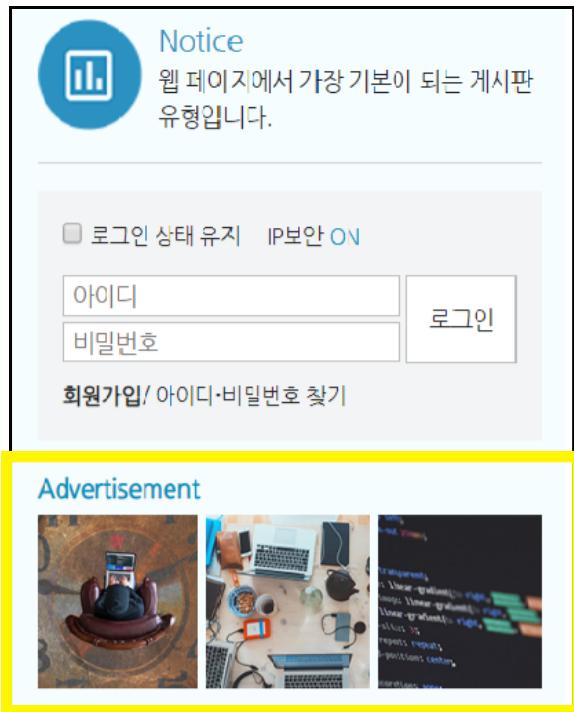
```
/* 게시판3 */
.notice3 { position: relative; }
.notice3 h4 { font-size: 14px; color: #0093bd; padding-bottom: 3px; font-weight: bold; }
.notice3 ul li {position: relative; padding: 8px 0 14px 60px; }
.notice3 ul li a img { position: absolute; width: 50px; left: 0; top: 0; border: 1px solid #0093bd; }
.notice3 ul li a strong { display: block; overflow: hidden; text-overflow: ellipsis;
white-space: nowrap; }
.notice3 ul li a span { display: block; overflow: hidden; text-overflow: ellipsis;
white-space: nowrap; font-size: 12px; }
.notice3 a.more { display: block; position: absolute; right: 0; top: 0; background: url(..../img/icon.png) no-repeat -150px -90px; width: 17px; height: 17px; }
```

overflow: hidden으로 넘치는 텍스트를 숨기고 **text-overflow: ellipsis**를 사용하여 말 줄임표를 나타냈습니다. 공백을 병합하고 자동 줄 바꿈을 방지하기 위해 **white-space: nowrap**을 사용하여 한 줄 효과를 냈습니다.

09 position

position: absolute를 이용하여 사진, 소제목과 텍스트의 위치를 잡았습니다. 기준이 되는 상위 요소에 **position: relative**를 주었습니다.

팝업 효과- jQuery 10



jQuery

```
//레이어 팝업
$(".layer").click(function(e){
    e.preventDefault();
    $("#layer").slideDown();
});

$("#layer .close").click(function(e){
    e.preventDefault();
    $("#layer").slideUp();
});
```

1) 레이어 팝업

사진을 클릭했을 때 페이지 상단으로 이동하지 않도록 `e.preventDefault()`를 함수에 포함했습니다. 레이어가 자연스럽게 보이도록 `slideDown`과 `slideUp`을 이용했습니다.

jQuery

```
//윈도우 팝업
$(".window").click(function(e){
    e.preventDefault();
    window.open("popup.html", "popup01", "width=800, height=590,
    left=50, top=50, scrollbars=0, toolbar=0, menubar=0");
});
```

jQuery Plugin

```
//라이트 박스
$(".lightgallery").lightGallery({
    thumbnail: true,
    autoplay: true,
    pause: 3000,
    progressBar: true
});
```

2) 윈도우 팝업

`window.open()`을 사용하여 클릭 후 새창이 열리며 팝업창이 뜨도록했습니다.

3) lightgallery.js 팝업

jQuery 플러그인, `lightgallery.js`를 사용했습니다. 사진이 3초마다 자동으로 넘어가도록 설정했습니다.

11 Mouse Hover 효과



HTML

```
<li>
  <a href="#">
    <span><em>바로가기</em></span>
    <strong>와이어 프레임</strong>
  </a>
</li>
```

CSS

```
.notice_hover ul li a span img { width: 100%; }
.notice_hover ul li a span em { visibility: hidden; position: absolute; left: 0; top: 0;
background: rgba(0,0,0,0.7); color: #fff; width: 100%; height: 100%; }
.notice_hover ul li a span:hover em { visibility: visible; }
```

CSS

```
.notice_hover2 ul li a span img { width: 100%; }
.notice_hover2 ul li a span em { visibility: hidden; position: absolute;
left: 0; bottom: 0; width: 100%; height: 25px; background: rgba(0,0,0,0.7);
}
.notice_hover2 ul li a span:hover em { visibility: visible; color: #fff; }
```

mouse hover시 나타낼 효과를 em 태그에 설정한 후 visibility: hidden을 주었습니다. span에 :hover를 붙이고 visibility: visible을 주어, mouse hover시 설정한 효과가 보이도록 했습니다.

Tab 메뉴



HTML

```
<div class="tab_menu">
  <h4 class="ir_su">탭 메뉴</h4>
  <ul>
    <li class="active"><a href="#">공지사항1</a>
      <ul> ...
        </ul>
    </li>
    <li><a href="#">공지사항2</a>
      <ul> ...
        </ul>
    </li>
    <li><a href="#">공지사항3</a>
      <ul> ...
        </ul>
    </li>
  </ul>
</div>
<!-- //탭메뉴 --&gt;</pre>
```

CSS

```
.tab_menu ul li { float: left; border: 1px solid #ccc; margin-right: -1px; margin-bottom: -1px; }
```

제목에 해당하는 ul과 내용의 ul을 동등하게 놓지 않고 `ul>li*3>ul>li*4`로 emmet 기능을 사용하여 코드를 작성했습니다. 이를 통해, 스크린리더가 올바른 순서대로 읽도록 하여 접근성을 높였습니다.

jQuery

```
//탭메뉴
var $tab_list = $(".tab_menu");

$tab_list.find("ul ul").hide();
$tab_list.find("li.active > ul").show();

function tabMenu(e){
  e.preventDefault();
  var $this = $(this);
  $this.next("ul").show().parent("li").addClass("active").siblings("li").removeClass("active").find(">ul").hide();
}

$tab_list.find("ul>li>a").click(tabMenu).focus(tabMenu);
```

제목 li에 `margin-right: -1px`을 주어 border값이 서로 겹치도록 했습니다.

tab 메뉴의 내용이 포함된 ul을 숨기고 `.active class`가 붙은 ul은 보이도록 설정했습니다. 클릭한 부분의 다음 ul을 노출하고 부모 li에 `.active class`를 추가했습니다. 형제 li의 `class`를 제거하고 하위 ul을 숨겼습니다.



```
css .footer li:before { content: ""; position: absolute; left: 0; top: 2px; width: 1px; height: 12px; background: #ccc; } .footer li:first-child:before { width: 0; height: 0; }
```

```
HTML <div class="w3c mt15"> <a href="http://validator.w3.org/check?uri=referer">  </a> <a href="http://jigsaw.w3.org/css-validator/check/referer">  </a> </div>
```

13
:before

li:before로 가상요소를 추가했습니다. 위치와 width, height를 적절하게 설정하여 웹의 디테일을 살리고 완성도를 높였습니다.
li:first-child:before를 사용하여 첫번째 li의 가상요소를 없애주었습니다.

14
W3C 검사

W3C Validator를 이용하여 HTML, CSS를 검사했습니다. 웹 페이지에 에러가 없음을 확인했습니다.