

# FILA

## PORTFOLIO

장형우

# 목차

1) 기획 의도 및 제작 기간

2) 웹 접근성 확인 결과

3) 페이지 소개

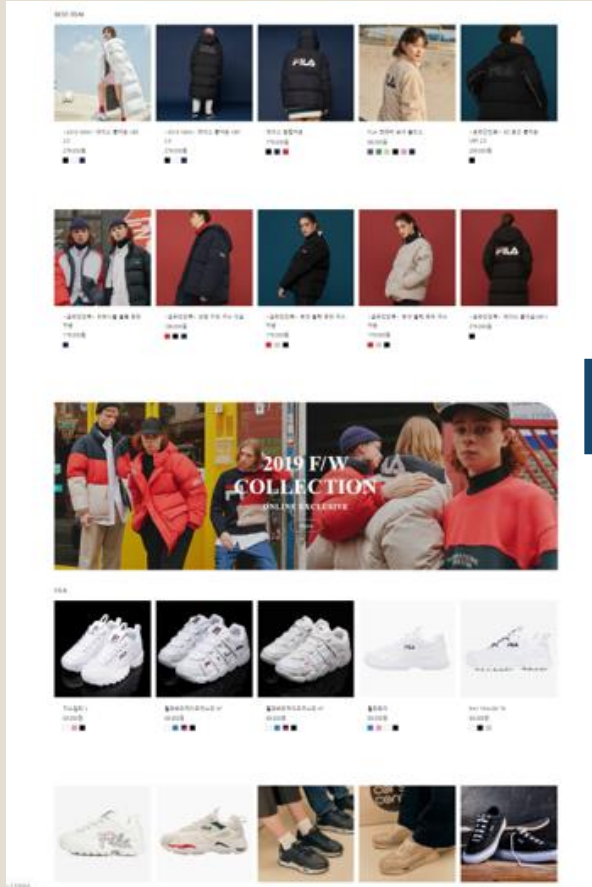
4) 기능 소개

# 제 1 장

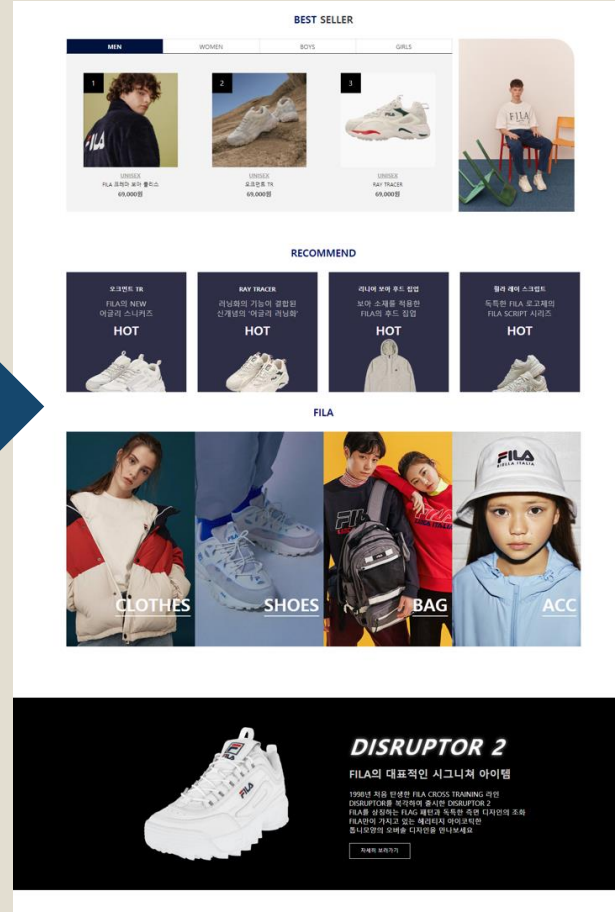
## 기획 의도 및 제작 기간



### 기존의 FILA index페이지



### 리뉴얼한 FILA index페이지



### 기획 의도

기존의 단순 상품 나열식의 FILA index페이지에 한눈에 봤을 때 보다 다양한 정보에 접근할 수 있도록 추천 영역과 카테고리 영역을 추가하였습니다.

다양한 볼거리를 제공해 자연스럽게 사용자가 보다 메인 페이지에 오래 머물고 FILA의 상품에 대한 홍보를 높일 수 있게 의도하였습니다.

또한 반응형 페이지를 제작해 보다 다양한 환경에서 FILA 사이트를 볼 수 있도록 제작하였습니다.

### 제작 기간

2019.10.05 ~ 2019.11.26

## 제 2 장

## 웹 접근성 확인 결과



W3C<sup>®</sup>

Markup Validation Service

Check the markup (HTML, XHTML, ...) of Web documents

Jump To:

[Congratulations](#) - [Icons](#)

This document was successfully checked as XHTML 1.0 Transitional!

Result:

Passed

File :

파일 선택

선택된 파일 없음

Use the file selection box above if you wish to re-validate the uploaded file index.html

Encoding :

utf-8

(detect automatically)

Doctype :

XHTML 1.0 Transitional


(detect automatically)

Root Element:

html

Root Namespace:

<http://www.w3.org/1999/xhtml>



Interested in "developing" your developer skills? In W3C's hands-on Professional Certificate Program, learn how to code the right way by creating Web sites and apps that use the latest Web standards.  
[Find out more!](#)

[Donate](#) and help us build better tools for a better web.

Options

☐ Show Source

☐ Validate error pages

☐ Show Outline

☐ Verbose Output

☒ List Messages Sequentially

☐ Group Error Messages by Type

☐ Clean up Markup with HTML-Tidy

[Revalidate](#)

[Help](#) on the options is available.

Congratulations

The uploaded document "index.html" was successfully checked as XHTML 1.0 Transitional. This means that the resource in question identified itself as "XHTML 1.0 Transitional" and that we successfully performed a formal validation of it. The parser implementations we used for this check are based on [OpenSP](#) (SGML/XML) and [libxml2](#) (XML).

"valid" Icon(s) on your Web page

To show your readers that you have taken the care to create an interoperable Web page, you may display this icon on any page that validates. Here is the HTML you could use to add this icon to your Web page:

W3C XHTML 1.0 Transitional icon

</img>  
</img>


A [full list](#) of icons, with links to alternate formats and colors, is available. If you like, you can download a copy of the icons to keep in your local web directory, and change the HTML fragment above to reference your local image rather than the online icons server.  
See also our help items related to documents transferred over secure protocols for these [icons](#) and the ["uri=referer" feature](#).

Validating CSS Style Sheets

If you use [CSS](#) in your document, you can check it using the W3C [CSS Validation Service](#).

Home About... News Docs Help & FAQ Feedback Contribute

W3C<sup>®</sup>



open source

COPYRIGHT © 1994-2013 W3C<sup>®</sup> ([MIT](#), [ERCIM](#), [KEIO](#)). ALL RIGHTS RESERVED. W3C LIABILITY, TRADEMARK, DOCUMENT USE AND SOFTWARE LICENSING RULES APPLY. YOUR INTERACTIONS WITH THIS SITE ARE IN ACCORDANCE WITH OUR PUBLIC AND MEMBER PRIVACY STATEMENTS.

This service runs the W3C Markup Validator, v1.3.14g

xhtml 문법 기준  
validator  
에러없음 확인!

## 웹 접근성 자동점검 보고서



웹사이트	http://janghwoo.dothome.co.kr/fila/index.html	수집 페이지	1 페이지
평가기준	한국형 웹 콘텐츠 접근성 지침 2.0	검사 일자	2019년 10월 10일

## 자 동 점 검 결 과

평가 항목	항목수	오류수	준수율
1. 대체 텍스트 제공	76	0	100.0
<img> 대체텍스트 없음	76	0	100.0
<area> 대체텍스트 없음	-	-	-
<input type=image> 대체텍스트 없음	-	-	-
<applet> 대체텍스트 없음	-	-	-
2. 제목 제공	1	0	100.0
페이지 제목 없음	1	0	100.0
<frame> 제목 없음	-	-	-
<iframe> 제목 없음	-	-	-
3. 기본언어 정의	1	0	100.0
기본언어 미정의	1	0	100.0
4. 새창열림 사전공지	-	-	-
<a> 알려지지 않은 새창열기	-	-	-
<area> 알려지지 않은 새창열기	-	-	-
5. 레이블 제공	1	0	100.0
<input> 레이블 미제공	1	0	100.0
<textarea> 레이블 미제공	-	-	-
<select> 레이블 미제공	-	-	-
6. 마크업 문법	1	0	100.0
id 중복 오류	1	0	100.0
속성이름 중복 오류	1	0	100.0
태그 열고 닫음 오류	1	0	100.0
닫는 태그 누락 오류	1	0	100.0
여는 태그 누락 오류	1	0	100.0
태그 중첩 오류	1	0	100.0

## ※ 주의사항

1. K-WAH 자동평가는 웹 접근성 표준 중 기계적으로 평가 가능한 6개 항목에 대한 점검 결과만을 제공합니다.
2. 웹 접근성의 준수여부는 K-WAH 자동평가만으로 판단할 수 없으며, 반드시 웹접근성표준의 전체 지표에 따른 수동평가를 통해 준수 여부를 확인하여야 합니다.
3. K-WAH가 수집하는 웹 페이지의 수집 순서는 네트워크의 속도에 따라 달라질 수 있습니다.
4. 평가항목 중 '기본언어정의', '마크업 문법'의 문수들은 전체 수집페이지 수 대비 문제가 없는 페이지 수의 비율로 계산됩니다.

## 웹 접근성 점검 결과

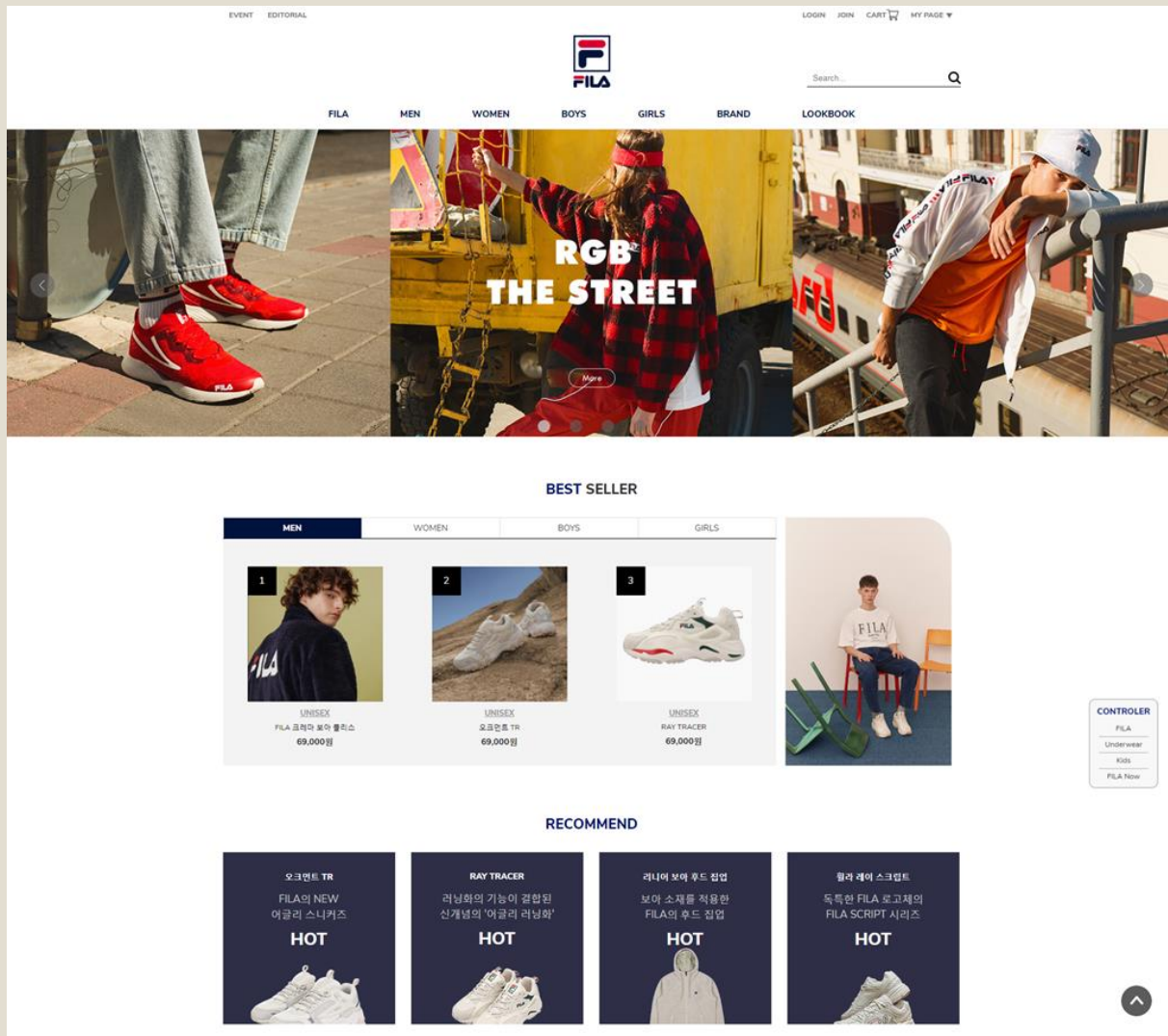
100 점

## outliner를 통해 확인한 index 페이지의 목차

1. FILA
  1. 위쪽 메뉴 영역
  2. 검색영역
  3. 메인 네비게이션 영역
    1. fila
    2. men
    3. women
    4. boys
    5. girls
    6. brand
  4. 비주얼 및 섹션 영역
  5. 비주얼 영역
  6. fila 영역
    1. best seller
    2. recommend
    3. fila
      1. clothes
      2. shoes
      3. bag
      4. acc
  7. 서브비주얼0영역
    1. disruptor 2
  8. underwear영역
    1. underwear
  9. 서브비주얼1영역
    1. only online
    2. monthly event
  10. kids영역
    1. kids
  11. 서브비주얼2영역
    1. FILA의 감성이 더해진 FILA GOLF WEAR
    2. FILA의 감성이 더해진 FILA GOLF WEAR
  12. 기획전, 인스타그램 영역
    1. fila now
    2. fila style
  13. 푸터영역
  14. fila푸터로고
    1. ABOUT FILA
    2. RECRUIT
    3. IR
    4. ETHICS
    5. ATHLETES
    6. CONTACT US

## 제 3 장

## 페이지 소개 - 연결된 페이지



### 페이지 소개

#### 총 페이지 수 : 15장

- Index 1장
- login 1장
- FILA -> about FILA 1장
- MEN -> clothes, shoes, bag, acc 총 4장
- WOMEN -> clothes 1장
- BRAND -> collabo, heritage, golf, kids, milan 총 5장
- LOOKBOOK 1장
- BEST SELLER 중 1위 제품 상세페이지 1장

### 반응형 페이지

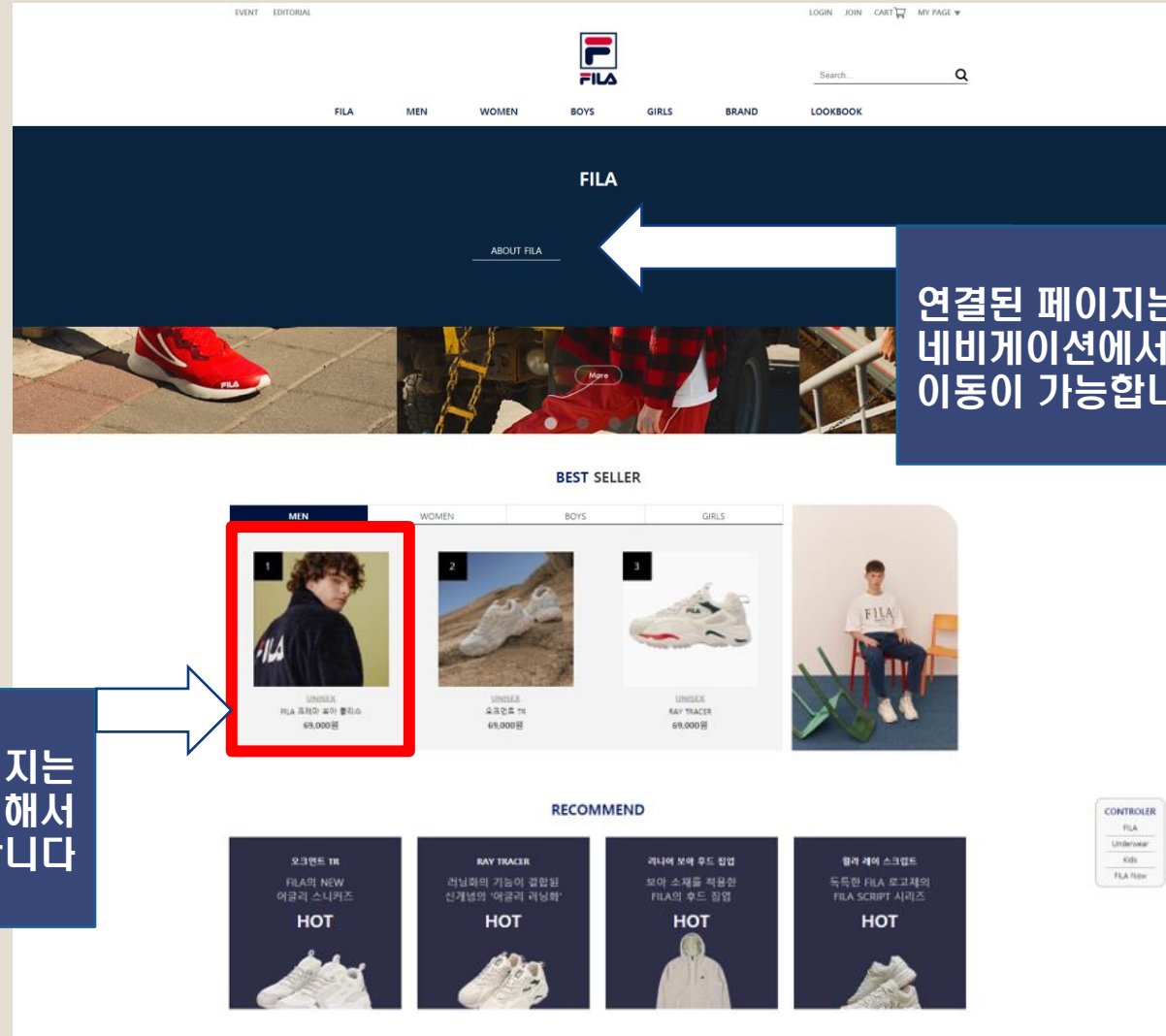
#### 반응형 가능한 페이지 수 : 8장

- Index 1장
- login 1장
- MEN -> clothes, shoes, bag, acc 총 4장
- BRAND -> heritage 1장
- LOOKBOOK 1장



## 제 3 장

## 페이지 소개 - 연결된 페이지



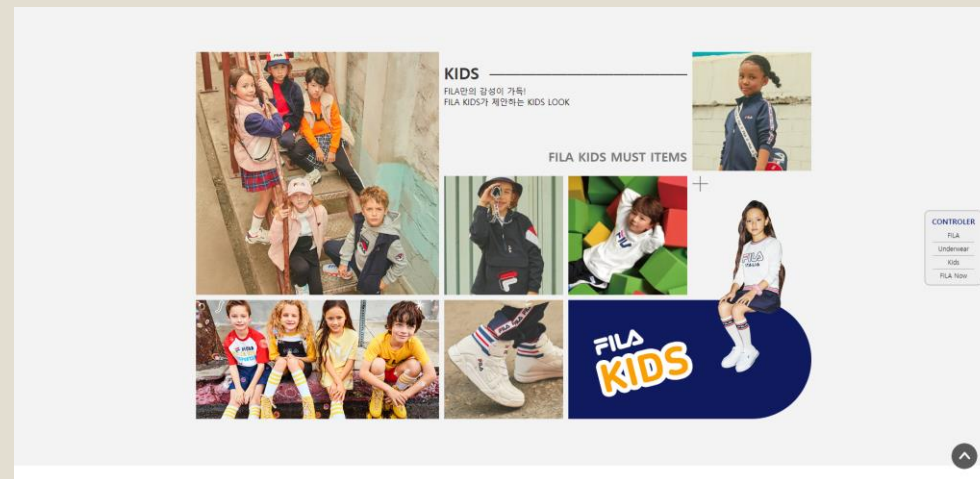
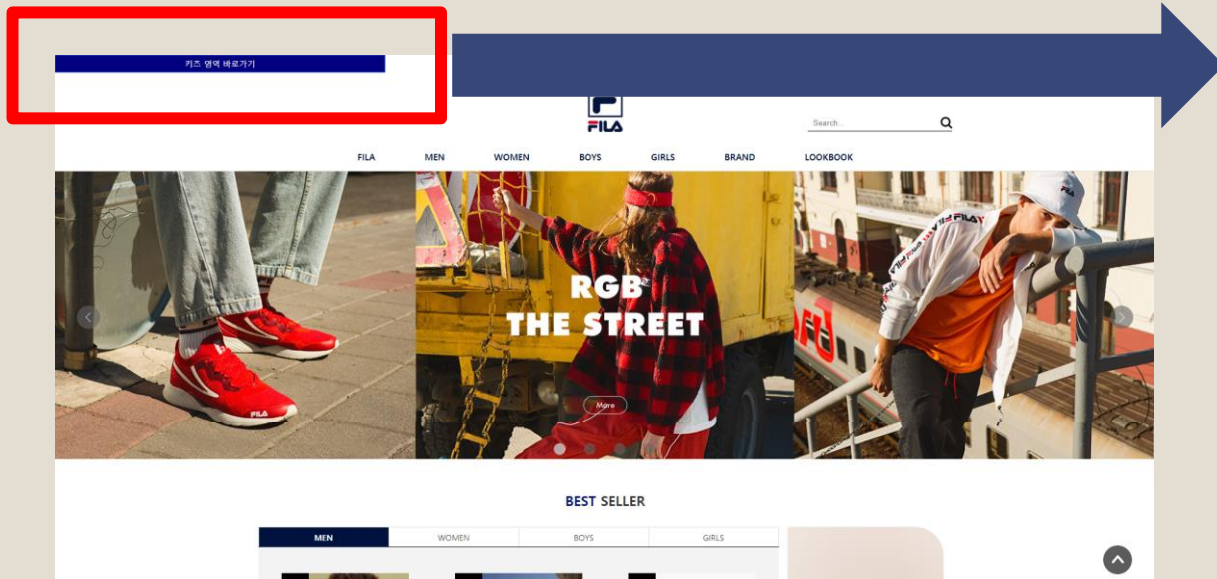
연결된 페이지는  
네비게이션에서 각각 해당하는 탭을 클릭해서  
이동이 가능합니다

제품 상세 페이지는  
BEST 1위 제품을 클릭해서  
이동이 가능합니다



## 제 4 장

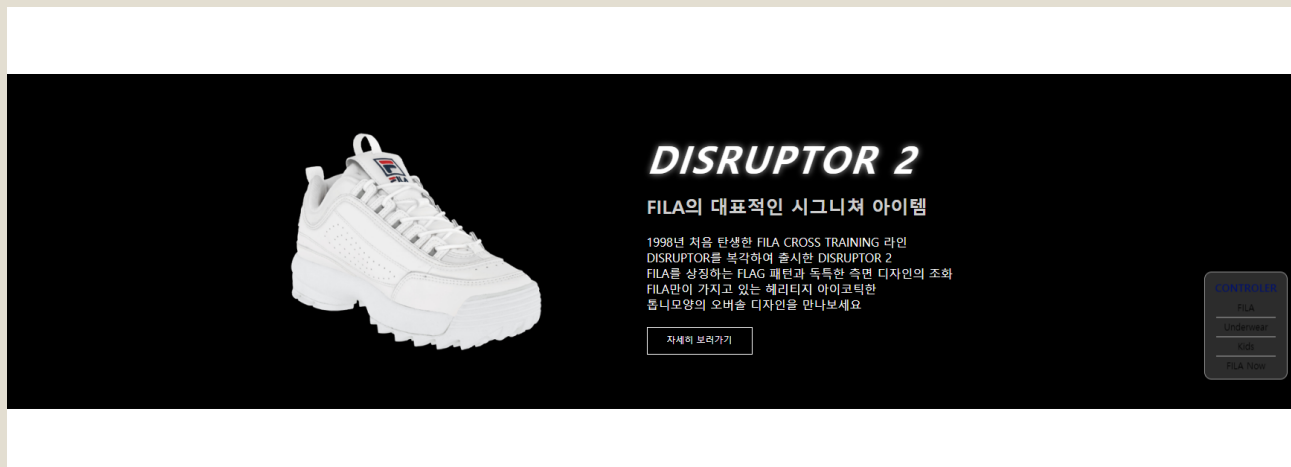
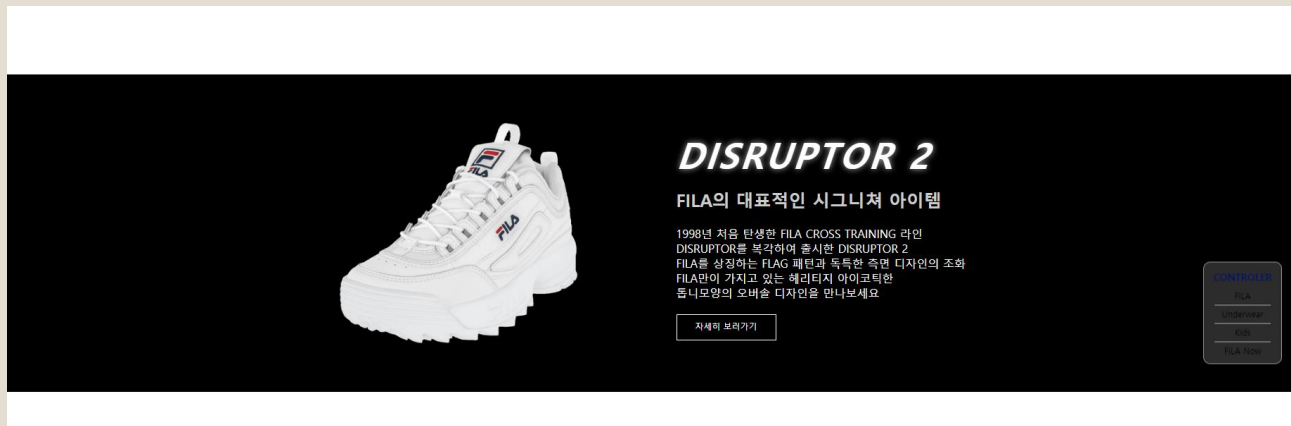
## 기능 소개 - skipNav



### skipNav

```
/*##### skipNav #####*/
#skipNav{width:30%; position:absolute; left:0; top:-30px; z-index:100;}
#skipNav p a{display:block; width:100%; height:30px; line-height:30px; text-align:center; font-size:14px;
background-color:navy; color:#fff; position:absolute;}
#skipNav p a:focus{top:30px;}
```

css의 focus기능을 이용해 tab키만을 이용해 index페이지의 모든 영역에 바로 접근할 수 있도록 skipNav를 심어두었습니다. 첫 화면에서는 아무것도 표시가 안돼지만 tab을 누르면 한 영역씩 이름이 왼쪽 상단에 표시가 되며 엔터를 누르는 순간 바로 해당 영역으로 화면이 이동하게 됩니다.



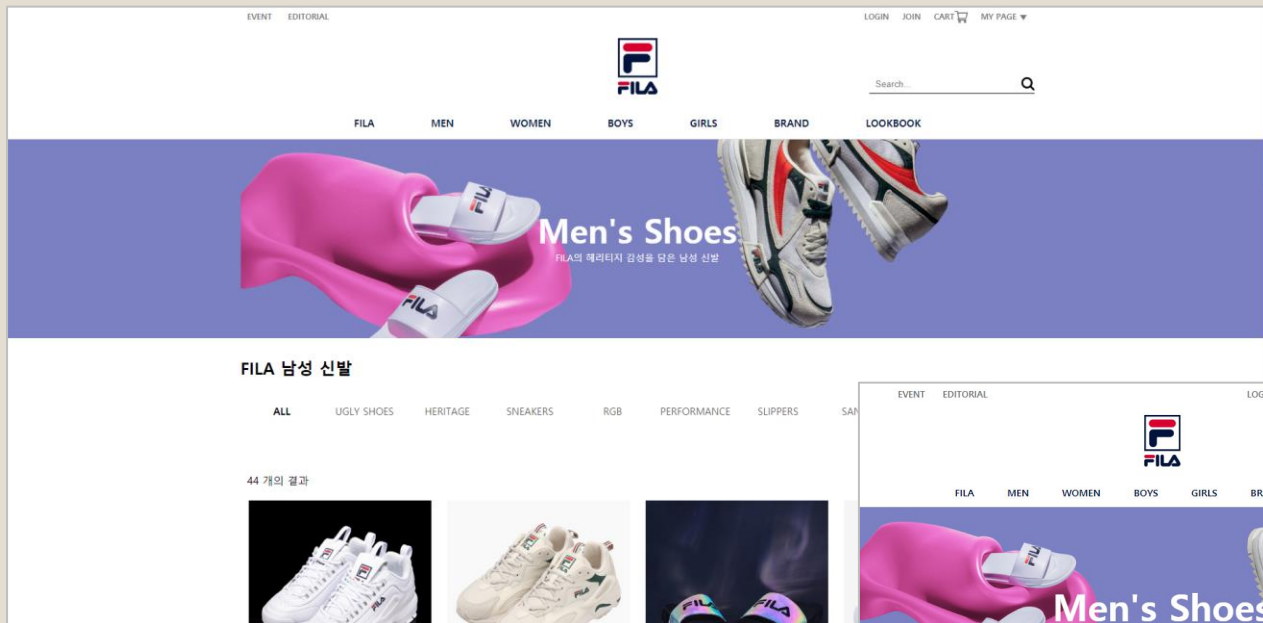
```
/*기능 적용*/
#subVisualo .subVisualo_img:hover .hmain img{display:none;}
#subVisualo .subVisualo_img p:hover{z-index:98;}
#subVisualo .subVisualo_img .ho:hover img{display:block;}
#subVisualo .subVisualo_img .h1:hover img{display:block;}
#subVisualo .subVisualo_img .h2:hover img{display:block;}
#subVisualo .subVisualo_img .h3:hover img{display:block;}
#subVisualo .subVisualo_img .h4:hover img{display:block;}
#subVisualo .subVisualo_img .h5:hover img{display:block;}
#subVisualo .subVisualo_img .h6:hover img{display:block;}
#subVisualo .subVisualo_img .hmain:hover img{display:block;}
#subVisualo .subVisualo_img .h7:hover img{display:block;}
#subVisualo .subVisualo_img .h8:hover img{display:block;}
#subVisualo .subVisualo_img .h9:hover img{display:block;}
#subVisualo .subVisualo_img .h10:hover img{display:block;}
#subVisualo .subVisualo_img .h11:hover img{display:block;}
#subVisualo .subVisualo_img .h12:hover img{display:block;}
#subVisualo .subVisualo_img .h13:hover img{display:block;}
#subVisualo .subVisualo_img .h14:hover img{display:block;}
#subVisualo .subVisualo_img .h15:hover img{display:block;}
#subVisualo .subVisualo_img .h16:hover img{display:block;}
```

## 이미지 회전

css의 hover기능을 이용해 운동화가 마우스를 따라 360도 방향을 돌리는 느낌의 기능을 구현했습니다. Hover된 구간에서 해당 이미지를 display:block; 으로 보여지게 하여 회전하는 느낌을 주었고, subVisual에 걸맞게 시각적인 효과를 극대화 하였습니다.

## 제 4 장

## 기능 소개 - 반응형 페이지



```
/*##### 반응형 맞추기 #####*/
@media all and (min-width:1025px) and (max-width:1220px){
}

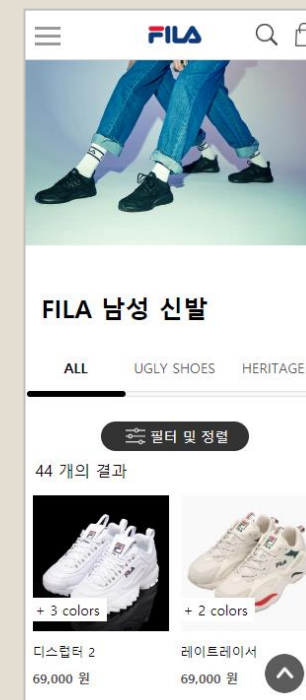
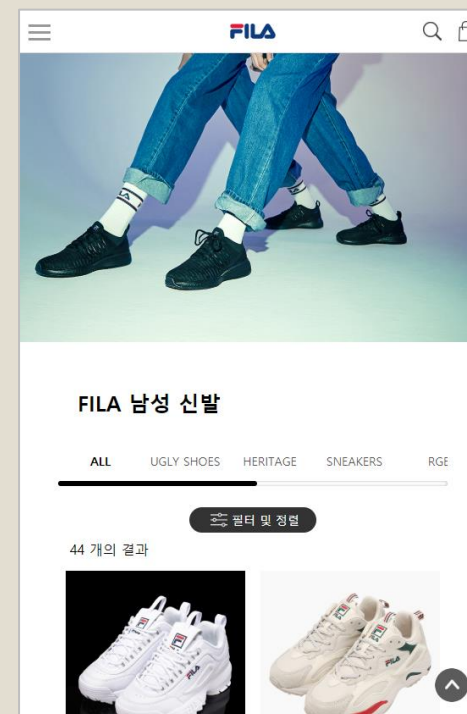
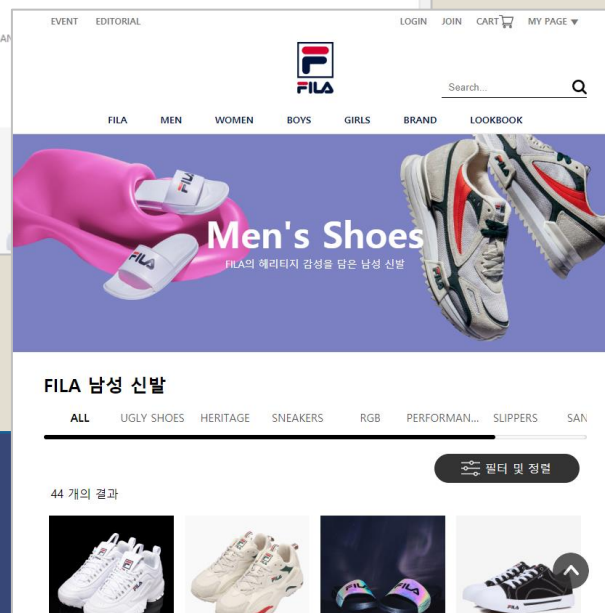
@media all and (min-width:768px) and (max-width:1024px){
}

@media all and (min-width:480px) and (max-width:767px){
}

@media all and (min-width:320px) and (max-width:479px){
}
```

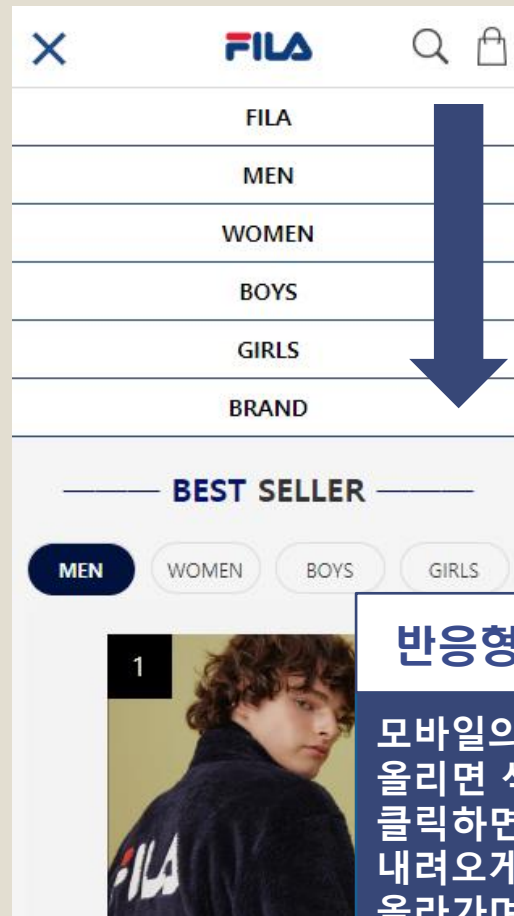
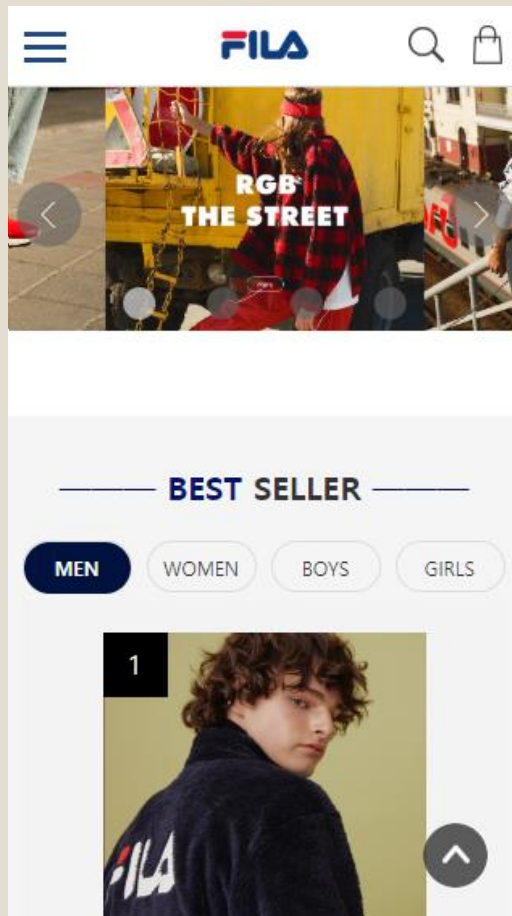
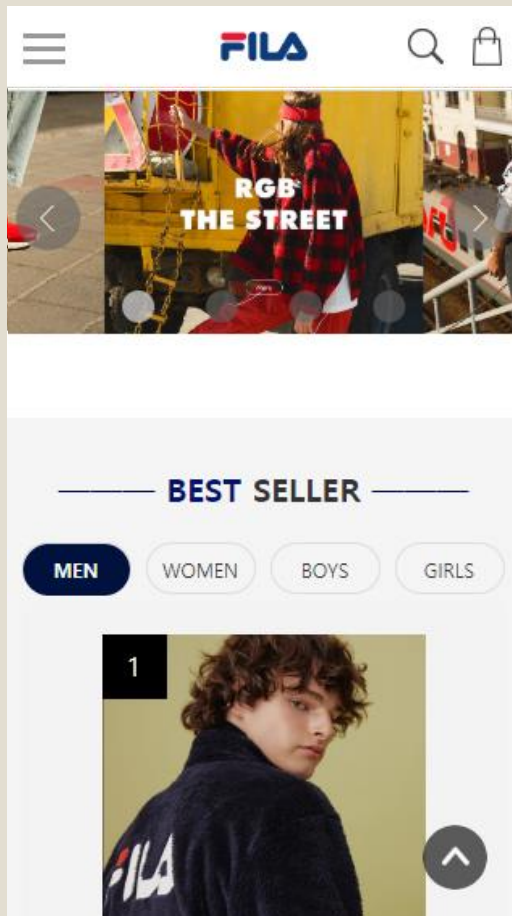
### 반응형 페이지

html5에서 미디어 쿼리를 이용해  
모바일 세로, 모바일 가로, 태블릿 pc,  
일반 pc버전의 반응형을 제작하였습니다.  
일반 pc버전은 1025px~1220px과  
공통영역인 ~1920px(max-width)로 나눠서  
제작하였습니다.



## 제 4 장

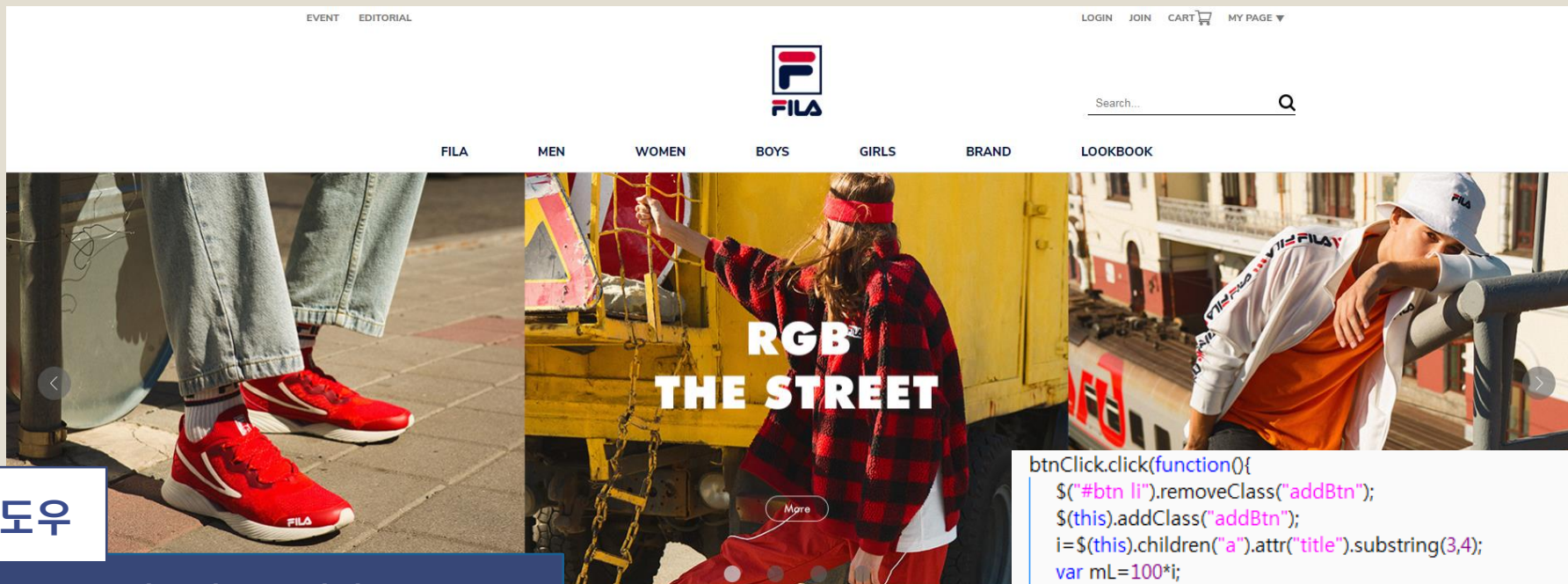
## 기능 소개 - 반응형 페이지



### 반응형 페이지

모바일의 경우 header의 category에 마우스를 올리면 색이 위에서부터 한칸씩 차례대로 채워지며 클릭하면 X자로 모양이 바뀌며 카테고리가 스르륵 내려오게 기능을 구현했으며 다시 클릭하면 스르륵 올라가며 X자는 다시 원래의 모양으로 바뀝니다. 특히 모바일의 header는 skipNav로 스크롤이 떨어지면 자동으로 fixed로 바뀌어 상단에 고정되도록 설정했습니다.



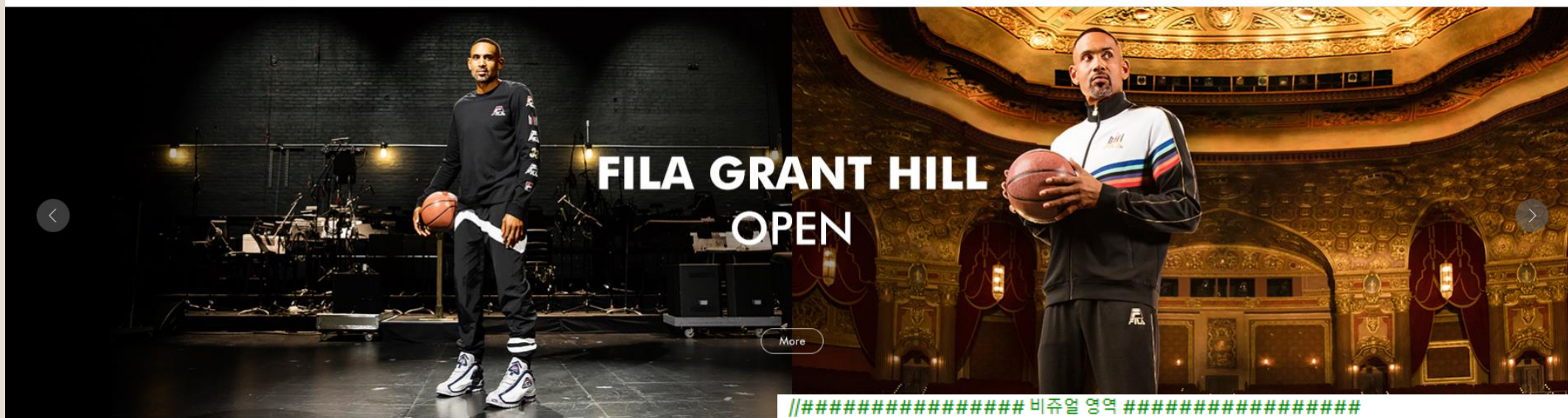


### 슬라이딩 윈도우

jQuery와 javascript를 이용해 슬라이딩 윈도우를 구현하였습니다. 버튼을 클릭했을 때, animate기능을 이용해 marginLeft값을 조정해서 슬라이딩 윈도우 기능을 제작하였습니다. index의 비주얼 영역은 버튼과 함께 작동하는 슬라이딩 윈도우로 if를 사용해서 버튼이 화살표 버튼과 함께 구동될 수 있게 만들었습니다. 이 경우에는 첫 번째 이전 혹은 네 번째 이후로는 막아놓고 반복되지 않도록 만들었습니다.

```
btnClick.click(function(){
    $("#btn li").removeClass("addBtn");
    $(this).addClass("addBtn");
    i=$(this).children("a").attr("title").substring(3,4);
    var mL=100*i;
    $("#film").animate({"marginLeft":"-" + mL + "%"},500,"swing");
});

$("#visual .next").click(function(){
    i++;
    if(i>=0&i<4){
        $("#visual #film").animate({"marginLeft":"-=100%"},500,"swing");
        $("#btn li").removeClass("addBtn");
        $("#btn li:eq("+i+")").addClass("addBtn");
    }
    else{
        i=3;
    }
});
```



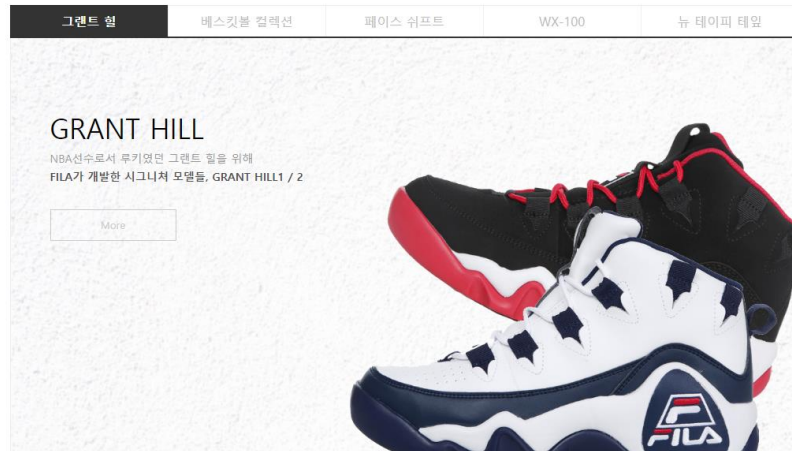
## 슬라이딩 윈도우

버튼이 없는 슬라이딩 윈도우의 경우엔 이미지 총 네 장이 계속 반복될 수 있도록 만들었습니다. append와 prepend를 이용해 첫 이미지를 맨 뒤로, 맨 마지막 이미지를 맨 앞으로 보내서 기능 구현을 했습니다.

```
##### 비주얼 영역 #####
$("#visual #film").prepend($("#film .scene:last")).css({"marginLeft":"-100%"});
$("#visual .next").click(function(){
    var film=$("#visual #film");
    $("#visual #film:not(:animated)").animate({"marginLeft":"-100%"},500,"swing",function(){
        film.append($("#visual .scene:first")).css({"marginLeft":"-100%"});
    });
});
$("#visual .prev").click(function(){
    $("#visual #film:not(:animated)").animate({"marginLeft":"+100%"},500,"swing",function(){
        var film=$("#visual #film");
        film.prepend($("#visual .scene:last")).css({"marginLeft":"-100%"});
    });
});
```

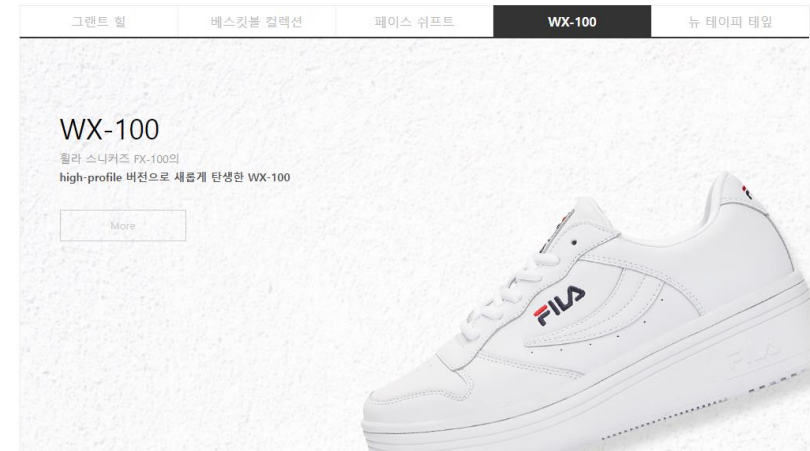
## Product

헤리티지를 기반으로 치우침 없는 조화를 추구하는 FILA의 제품을 만나보세요



## Product

헤리티지를 기반으로 치우침 없는 조화를 추구하는 FILA의 제품을 만나보세요

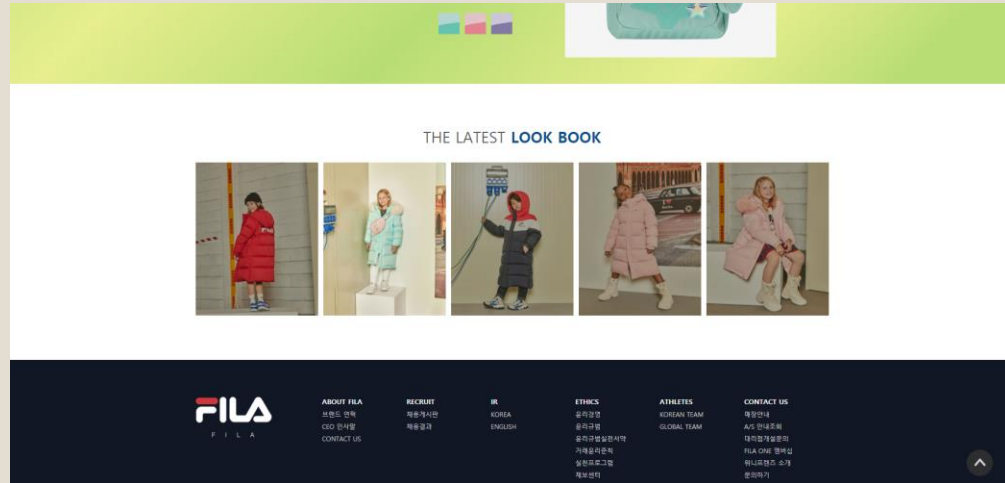


```
$("#product.tabMain li:not(:first)").hide();
$("#product.tabMenu li a").bind("click focusin",function(){
    $("#product.tabMain li").hide();
    $($this).attr("href").show();
    $("#product.tabMenu li a").removeClass("on");
    $(this).addClass("on");
    return false;
});
```

## 탭 베이직

jQuery를 이용해 해당 tab(카테고리)을 클릭하거나 탭키로 접근했을 때 해당하는 tabDesc(내용부분)만 보여지도록 show와 hide를 이용하였고, tab(카테고리)가 클릭 되어 있는걸 나타내기 위해 on이라는 class를 추가하여 나타냈습니다.

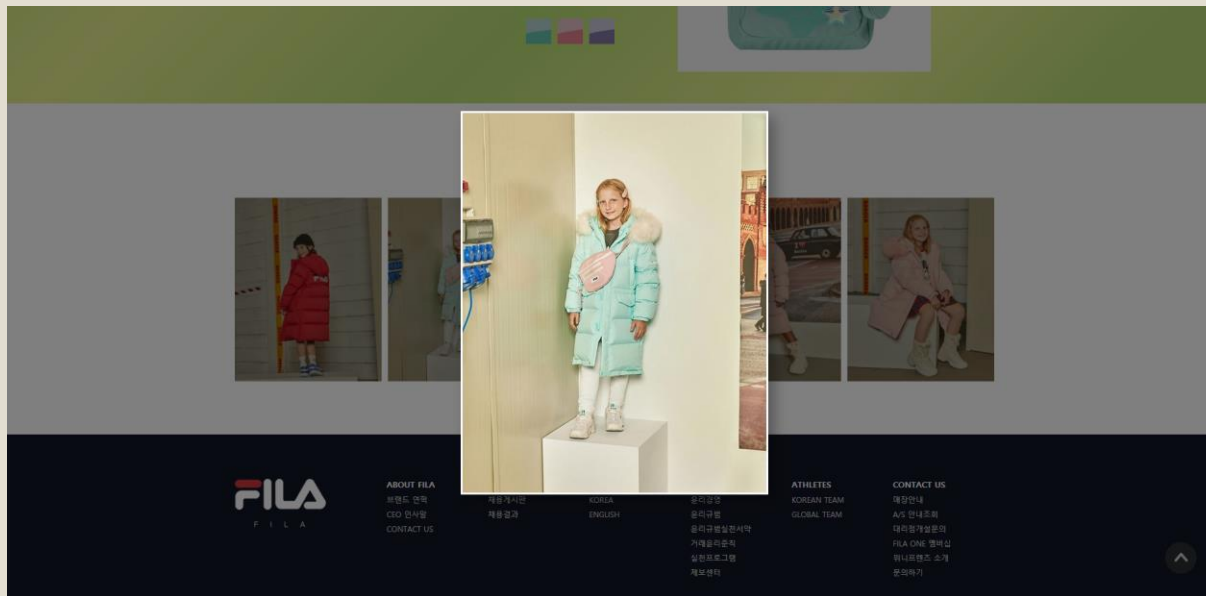




```
// modal 영역
$("#view, #view_bg").hide();
var lookbook=$("#lookbook li a");
lookbook.click(function(){
    $("body").css({"overflow-y":"hidden"});
    $("#view, #view_bg").show();
    $("#view img").attr("src",$(this).attr("href"));
    return false;
});
$("#view, #view_bg").click(function(){
    $("body").css({"overflow-y":"visible"});
    $("#view, #view_bg").hide();
});
```

## 모달 팝업

css로 해당 이미지에 마우스를 올리면 밝아지게 표현했고 jQuery와 javascript를 이용해 클릭했을 때 모달 팝업으로 해당 이미지가 확대되어 화면의 공중에 띄워지는 기능을 구현하였습니다. attr()를 이용해 이미지 소스를 불러왔고, show, hide를 이용해 모달팝업창을 나타내고 사라지게 조절하였습니다.

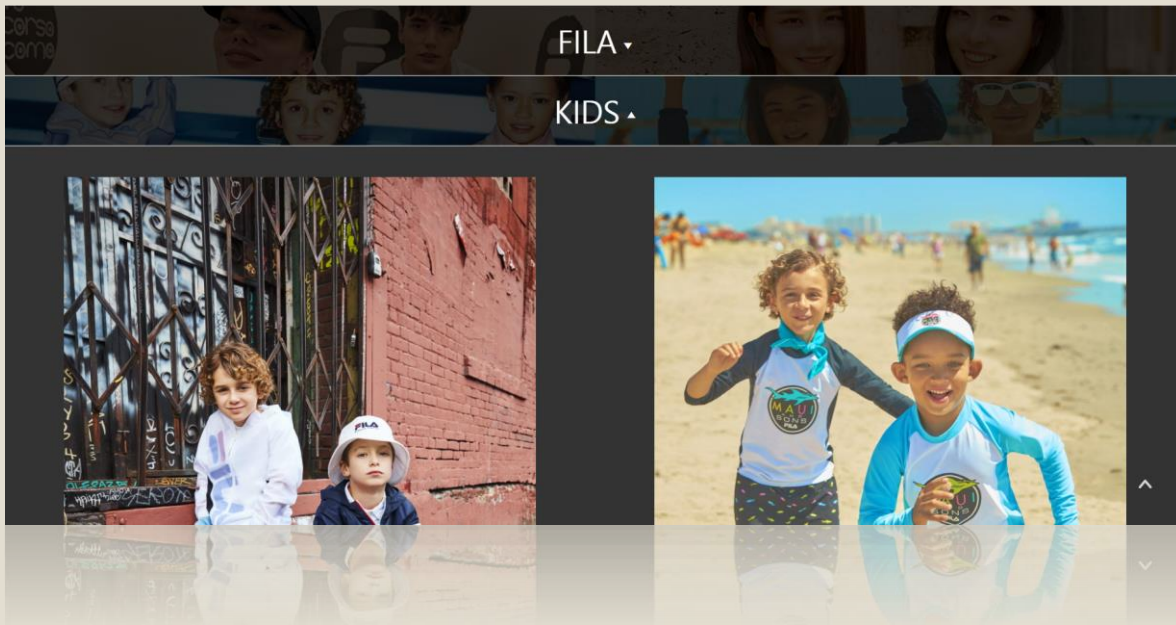




```
$("#acco .accor").hide();
$("#acco li a").bind("click focus",function(){
    $("#acco li a").removeClass("open");
    $(this).addClass("open");
    $("#acco .accor").stop().slideUp(1000,"linear");
    $(this).next(".accor").stop().slideDown(1000,"linear");
});
```

### 아코디언

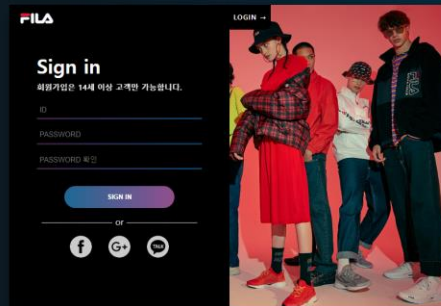
jQuery를 이용해 해당하는 영역을 클릭하거나 탭키로 접근했을 때, 영역의 이름부분의 ▼ 아이콘은 회전하도록 open이라는 이벤트class를 추가하거나 제거하였고, 아래 펼쳐지는 내용부분은 slideUp과 slideDown으로 접혔다 퍼졌다 하는 아코디언 기능을 구현했습니다.





```
$("#joinArea .loginBtn").click(function(){
    $(".black_bg").css({"left":"50%"});
    $("#joinArea").css({"left":"20%"}).fadeOut(300,"linear");
    var body=$("#body").width();
    if(body>319&&body<768){
        $("#loginArea").css({"left":"0"}).fadeIn(300,"linear");
        $("#loginArea .joinBtn").css({"left":"0"});
    }
    else{
        $("#loginArea").css({"left":"50%"}).fadeIn(300,"linear");
        $("#loginArea .joinBtn").css({"left":"-75px"});
        $("#header .join_logo").css({"opacity":"0"});
    }
    $("#joinArea .loginBtn").css({"left":"325px"});
    $(".img_bg img").css({"left":"0"});
});
```

### 로그인 페이지



jQuery와 javascript를 이용해 로그인 페이지의 기능을 적용했습니다. 첫 화면은 로그인화면인데 중앙의 join버튼을 클릭하면 검은 배경이 왼쪽으로 이동하며 sign in 영역이 나오고 다시 login버튼을 클릭하면 검은 배경이 오른쪽으로 이동하며 login영역이 나오도록 click이벤트에 left값과 fadeIn, fadeOut를 이용해 기능을 구현하였습니다.

## 제 4 장

## 기능 소개 - 다양한 마우스 이벤트들



**FILA NOW**

**FILA 기획전**  
2019

**DRIFTER CREMA**  
FILA의 스테디셀러인 DRIFTER의 19FW FUR버전.

**NEW BLACK EDITION**  
BLACK & GREY로 구성된 NEW BLACK EDITION을 만나보세요!

**FILA EXPLORE**  
필라가 출시했던 스키, 하이킹, 알파인, 트래킹 컬렉션에서 영감을 얻어 재해석한 필라 익스플로러 컬렉션

**KIDS**  
FILA만의 감성이 가득!  
FILA KIDS가 제안하는 KIDS LOOK

**FILA KIDS MUST ITEMS**

**FILA**

**CLOTHES**

MEN  
WOMEN  
BOY  
GIRL

**RECOMMEND**

**오크먼트 TR**  
FILA의 NEW 어글러 스니커즈  
**HOT**

**RAY TRACER**  
러닝화의 기능이 결합된 신개념의 '어글러 러닝화'  
**HOT**

**러니어 보아 후드 집업**  
보아 소재를 적용한 FILA의 후드 집업  
**HOT**

**월러 레어 스크립트**  
독특한 FILA 로고체의 FILA SCRIPT 시리즈  
**HOT**

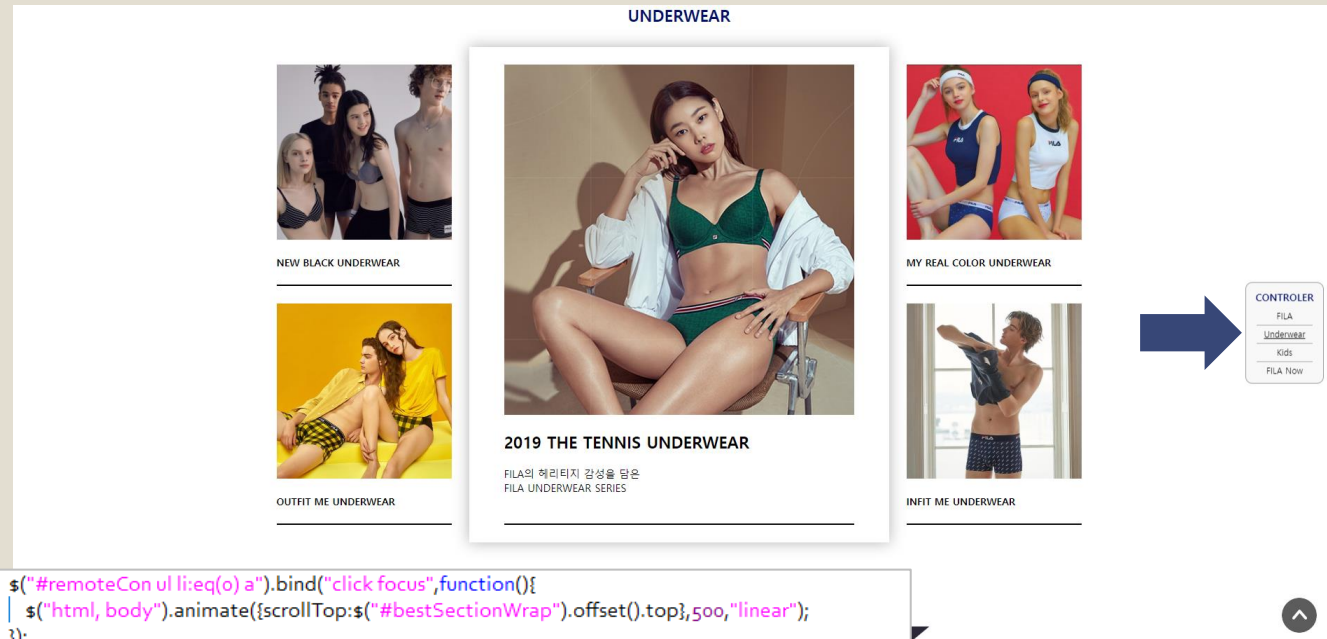
## 마우스 이벤트

css의 hover, 또는 jQuery의 mouseover 기능을 이용해 마우스가 해당 영역에 올라갔을 때, border-radius가 조절되거나 overflow: hidden이 visible로 바뀌거나 transform: scale으로 이미지가 확대되며 서브메뉴가 스르륵 올라가거나, 마우스를 따라 검은 박스가 이동하며 크기도 저절로 상대 객체에 맞게 조절되는 기능을 곳곳에 심어서 현재 마우스를 올린 부분을 눈에 띄게 표현하였습니다.



## 제 4 장

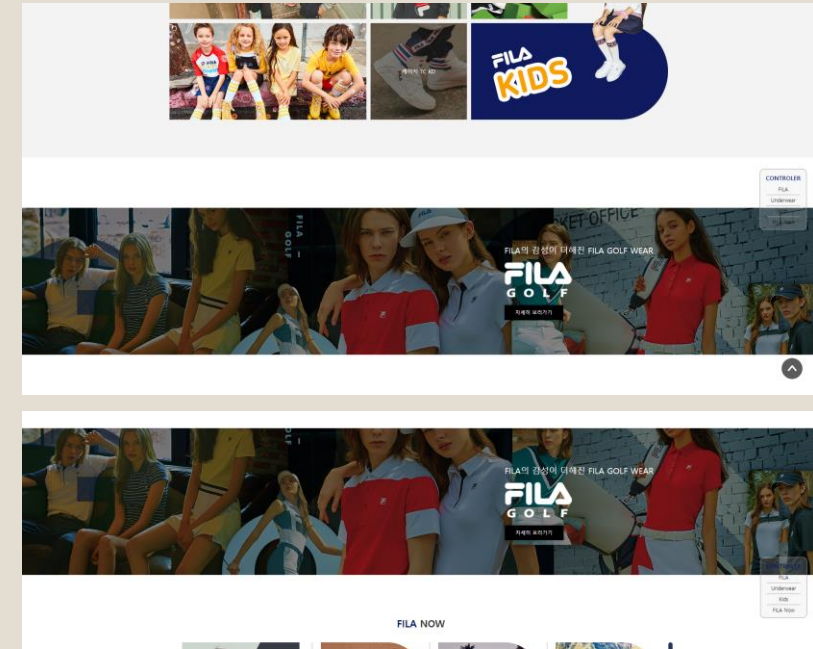
## 기능 소개 - scrollTop



```
$("#remoteControl li:eq(0) a").bind("click focus",function(){
    $("html, body").animate({scrollTop:$("#bestSectionWrap").offset().top},500,"linear");
});
$("#remoteControl li:eq(1) a").bind("click focus",function(){
    $("html, body").animate({scrollTop:$("#underwearSectionWrap").offset().top},500,"linear");
});
$("#remoteControl li:eq(2) a").bind("click focus",function(){
    $("html, body").animate({scrollTop:$("#kidsSectionWrap").offset().top},500,"linear");
});
$("#remoteControl li:eq(3) a").bind("click focus",function(){
    $("html, body").animate({scrollTop:$("#editorialWrap").offset().top},500,"linear");
});
```

### scrollTop

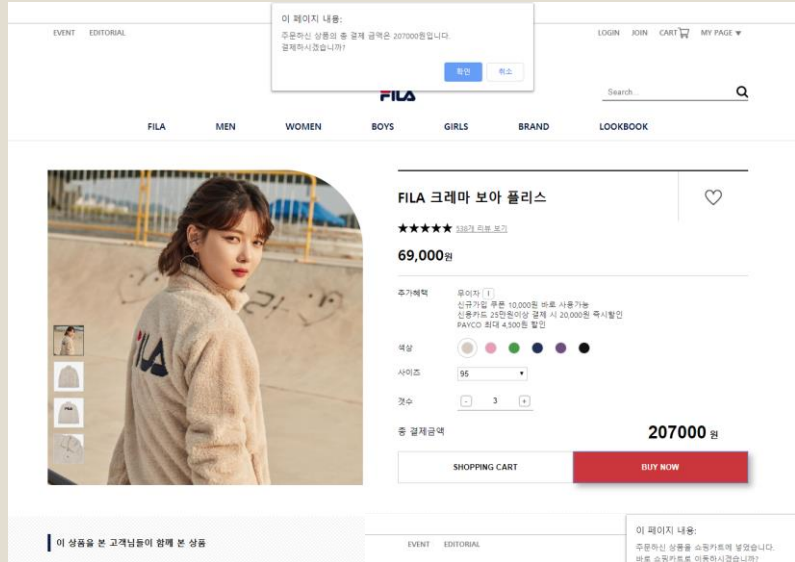
jQuery에서 scrollTop을 이용해 controller에 해당하는 이름을 클릭하면 해당 영역으로 스크롤 떨어지는 기능을 animate로 구현하였습니다. 또한 스크롤이 내려감에 따라 반응하여 배경이미지가 비슷한 속도로 올라가는 기능을 구현하여 좀 더 생동감있는 효과를 주었습니다.



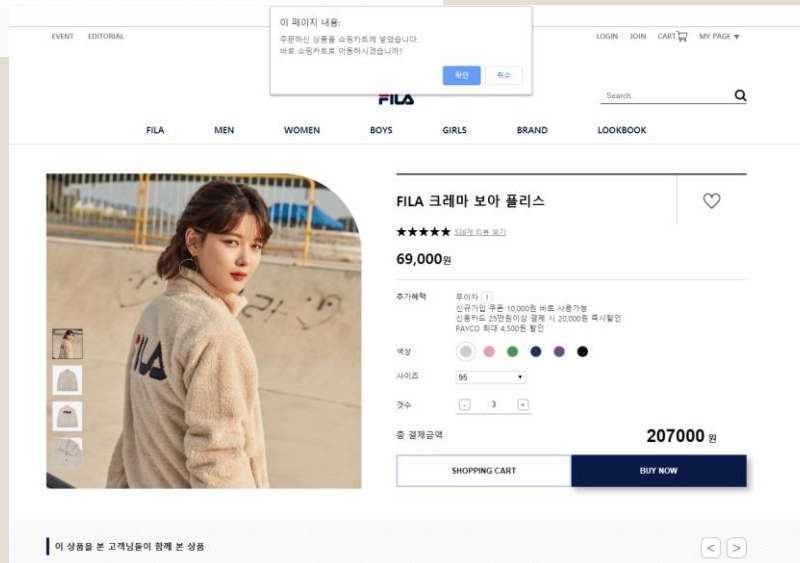
```
if(nowScroll>=4700){
    var num=(nowScroll-4700)/5;
    $("#subVisual2Wrap").css({"backgroundPosition":"center -"+num+"px"});
}
```

## 제 4 장

## 기능 소개 - confirm창을 통한 구매 확인



```
var totPrice=document.getElementById("totPrice");
var num=document.getElementById("num");
var productBest1_price=69000;
totPrice.value=productBest1_price*parseInt(num.value);
console.log(totPrice.value);
if(size0.value=="soldout"||size1.value=="soldout"||size2.value=="soldout"||
size3.value=="soldout"||size4.value=="soldout"||size5.value=="soldout"){
    alert("매진상품입니다.\n다른 Size를 골라주세요.");
    return false;
}
```



### confirm

javascript에서 다양한 조건에 따라 알림창을 띄우고, 최종적으로 결정이 되었을 때 input의 value값을 변수로 이용해서 confirm창을 통해 구매 혹은 쇼핑카트에 담을 경우 확인창을 띄워서 가격을 확인시켜주는 기능을 구현하였습니다.

**THANK YOU**  
**PORTFOLIO**