

JohnRo's

Portfolio.

2019-2020

WebDesigner
Developer

Contents.

Portfolio JohnRo

Index.

04

Resume

skills
experience
certification
interests
motto

06

Project 01

analysis & concept
styleguide
design
coding
validate
review

34

Project 02

analysis & concept
styleguide
design
coding
validate
review

64

Identity

personal brand design

Resyme.

“ 길이 없으면 길을 찾아라. 찾아도 없으면 길을 만들어라 ” ○

About ME

1990.12.27

heyho9012@gmail.com

+82 10 9935 5074

<https://github.com/heyho9012>

<https://heyho9012.github.io/john/>

경기도 평택시 비전동



Special Skills

Gulp

Git

Responsive Web

Interests

Watching Football & Netflix

Drawing

Coffee

Book

Professional Skills

Illustrator



Photoshop



Indesign



HTML



CSS



jQuery



Sketch



MSoffice



Work Experience

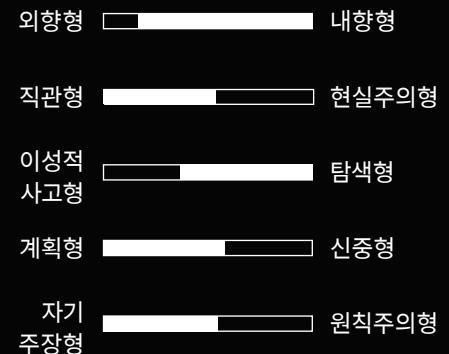
- **Contentsbridge**
한신대학교
2015.12 - 2016.04
네트워크 및 리눅스 서버 관리
CDN Service
- 2018.09 - 2019.02
청년TLO 육성사업
정보통신학과 교수 연구원

Education

- **한신대학교**
2009.03 - 2016.02
정보통신학과 졸업(3.4/4.0)
- **사회복지**
2016.06 - 2017.01
대한신학대학원대학교 사회교육원
사회복지 전공
- **그린컴퓨터아카데미**
2019.08 - 2019.12
스마트기기 UXUI 디자인 실무자
디지털앱, 웹디자인

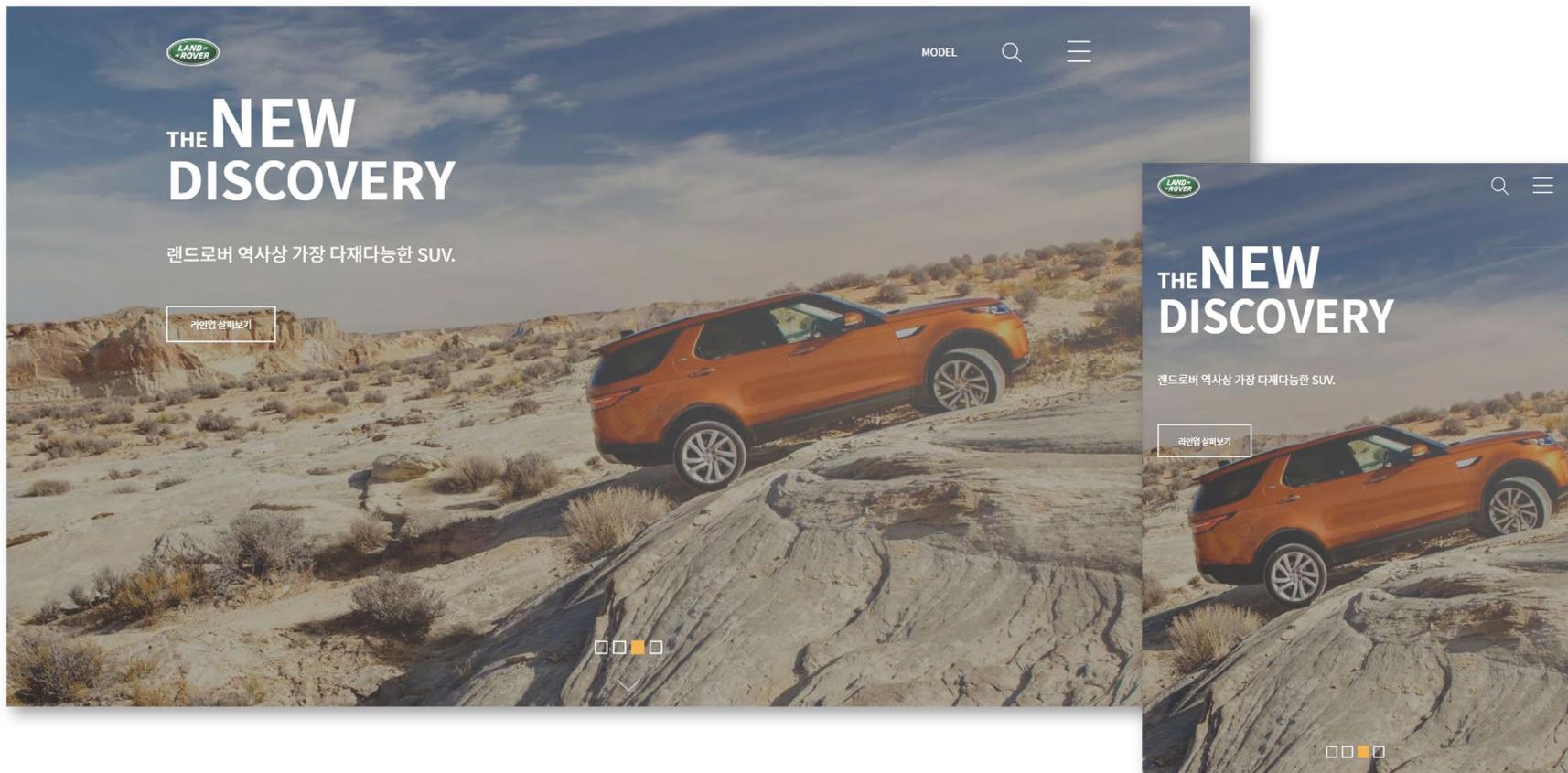


MBTI



Certification

- 웹디자인기능사 2019.12.20. 한국산업인력공단
- 컴퓨터그래픽스운용기능사 2020... 한국산업인력공단(예정)
- 정보처리기사 2015.11.30 한국산업인력공단
- 사회복지사 2급 2017.01.24 보건복지부
- TOEIC 2015.10 800



Project 01.

Analysis & Concept
Style Guide
Sketch
Proto Type
Design
Coding
Validate
Review

LAND ROVER

<https://heyho9012.github.io/landrover/>
6 weeks
HTML5, CSS3, jQuery, Photoshop, Illustrator
mobile tablet basic pc

Original site Analysis.

사이트 이름
LANDROVER

사이트 주소
<https://www.landroverkorea.co.kr>

브랜드 유형
자동차(SUV)

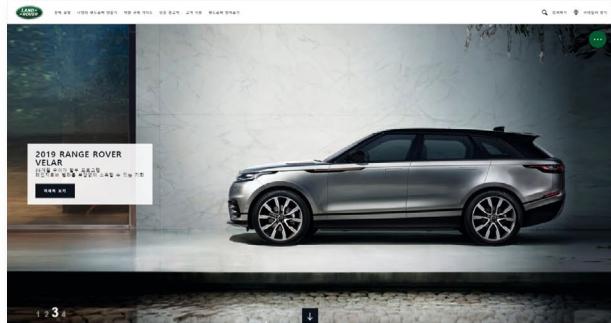
브랜드 컨셉
안정성, 전통성, 혁신

주요 고객층
30-40대

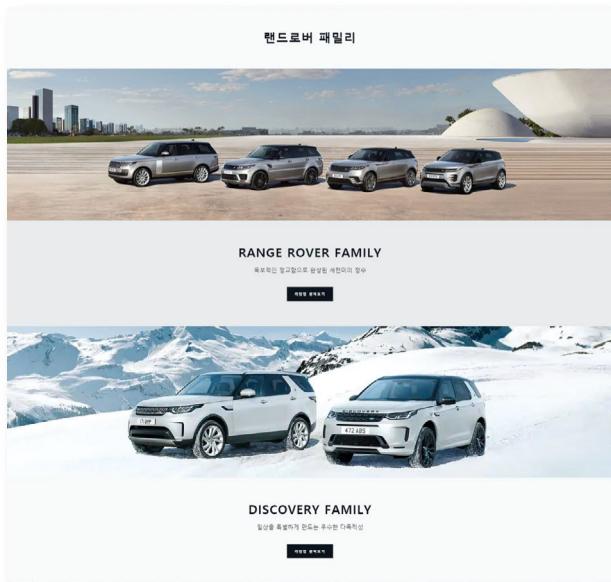
사이트 선정이유
기존 사이트보다 좀 더 흥미를
일으킬 수 있는 사이트로 만들고
브랜드 컨셉에 맞는 디자인 변경

Portfolio JohnRo

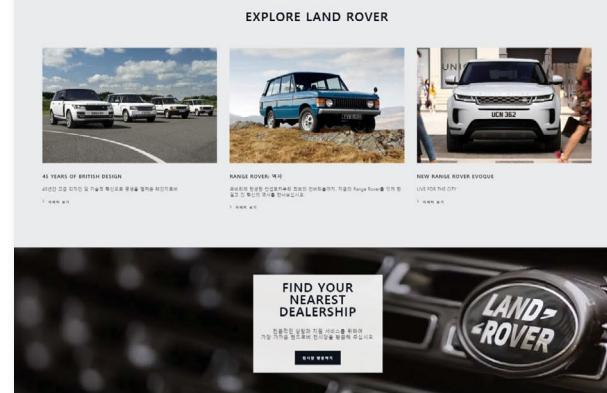
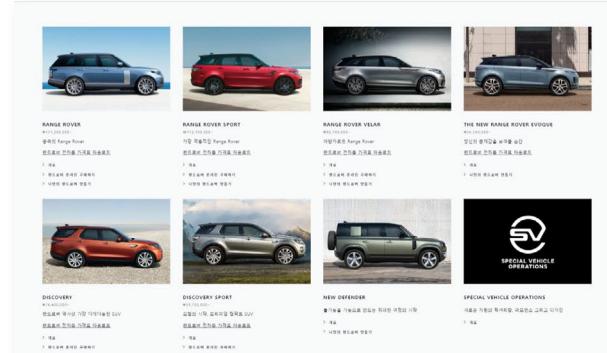
Original site.
Original site.



광고영역에 텍스트가
흰 박스 안에 있는 것이
브랜드 이미지와
어울리지 않는 디자인.



아무 기능이 없어서
사용자가 지루함을
느낄 수 있음



광고 영역을 제외하면
슬라이드 기능이 없고
나열형식으로 되어 있어
단조로움.

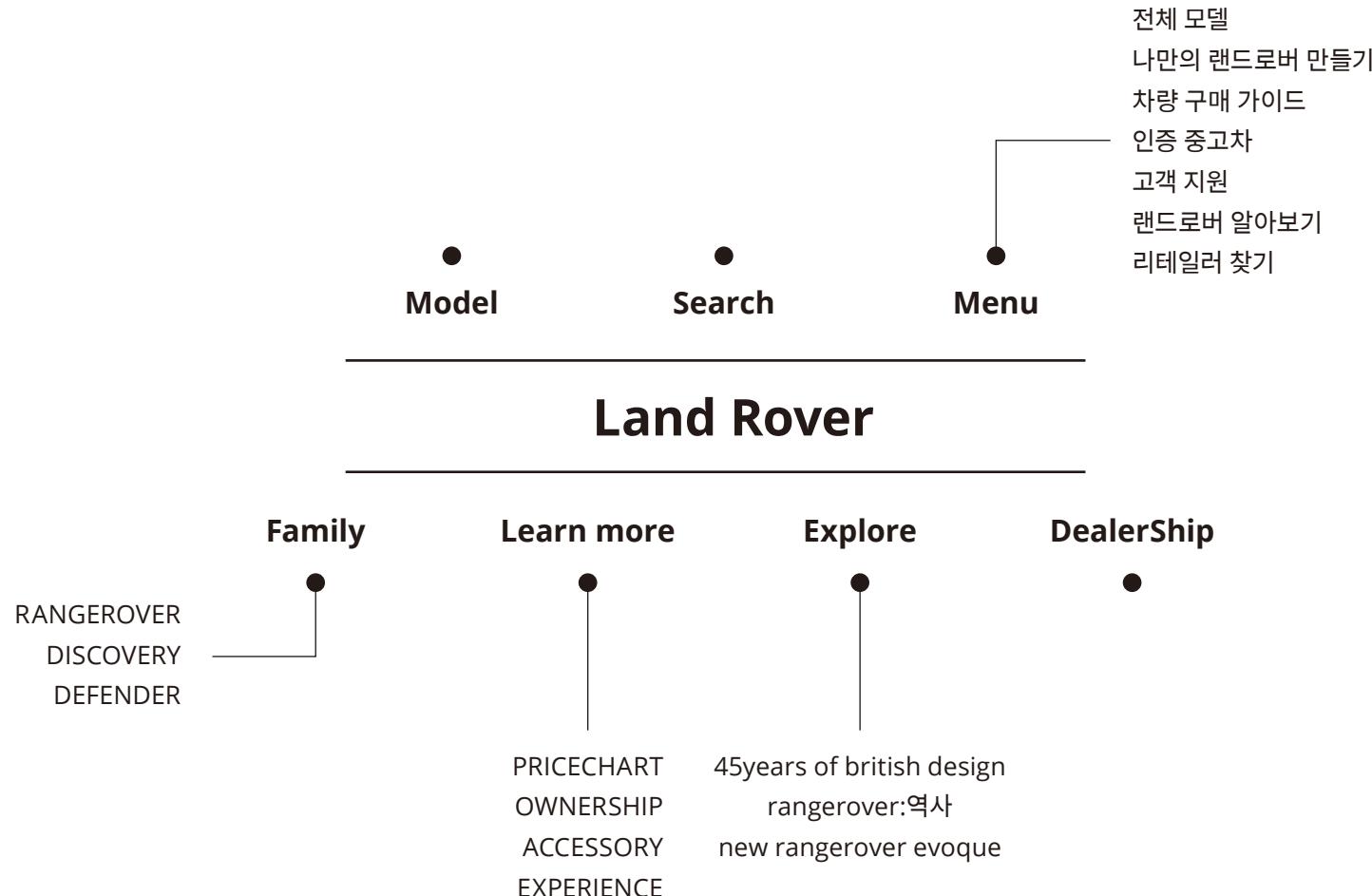
Renewal

Information Architecture

Persona

SWOT

<http://heyho9012.github.io/landrover>





윤선민

43 | IT회사원 | 서울 | 월 소득: 380

NEEDS

특징 : 차 브랜드별 특징에 관한 정보 공부

목적 : SUV 구입

요구사항 : 레이아웃이 제품의 특징을 잘 표현하지 못한다.
또, 모든 페이지가 기능도 없고 정적이어서 흥미를 떨어트린다.

Solutions

1. 동적으로 만들어 흥미를 유발시킨다.
2. 제품의 특징을 살려 주목성을 높인다.

Strengths

디자인과 편안함 보다는
성능 측면에서 우위가 있으며, 특히 안전성이 뛰어남
고급차량, 최고의 안전함이라는 요소를 적극적으로 노출

S

Opportunities

SUV라는 품목의 특성상 모든 소비자를 대상으로 한 전략에서
성공할 수 없기 때문에 시장을 세분화하여
특정 소비자를 공략할 필요가 있음

O

Weakness

마니아층만을 공략할 경우 매출면에서 부진을 면치 못할 것으로 예상

W

T

Threats

높은 가격대의 유보가격을 요구함에도 불구하고
상품 인지도는 매우 낮음 동급차종에 비해 높은 가격.
이미 시장에 진출한 Jeep Cherokee, Ford Explore의 높은 점유율

style
GuideLine

Color

Font

Button

Portfolio JohnRo

포인트 컬러로 블랙계열과 화이트로 모던하고 세련된 느낌을 의도하였습니다.
 또한, 오리지날 사이트의 컬러인 그린계열과
 어울리는 오렌지 컬러를 조합시켜 지루하지 않게 했습니다.

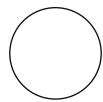
Primary



#0E0E11
RGB(14,17,17)



#060606
RGB(6,6,6)



#FFFFFF
RGB(255,255,255)

Secondary



#02512D
RGB(2,81,45)



#F89626
RGB(248,150,38)



#1D1D1D
RGB(28,29,29)



#D7E4F5
RGB(215,228,245)

Color Contrast



#0E0E11
#FFFFFF
19.3 :1



#FFFFFF
#02512D
9.5 :1



#060606
#FFFFFF
20.3 :1



#1D1D1D
#F89626
7.5 :1



#FFFFFF
#1D1D1D
16.9:1

Font

브랜드 이미지에 맞는 튼튼한 이미지의 서체로
한글과 영어를 같은 폰트로 통합해서 사용했습니다.

Font Family

Aa

Spoqa
Han Sans

가

스포카
한 산스

Aa Thin

Aa Light

Aa Regular

Aa Bold

가 Thin

가 Light

가 Regular

가 Bold

Font Size

Headers

Font-weight : Bold

Line-height : 100%

H2 header

Font-size : 58pt

H3 header

Font-size : 44pt

H4 header

Font-size : 32pt

H5 header

Font-size : 24pt

H6 header

Font-size : 16pt

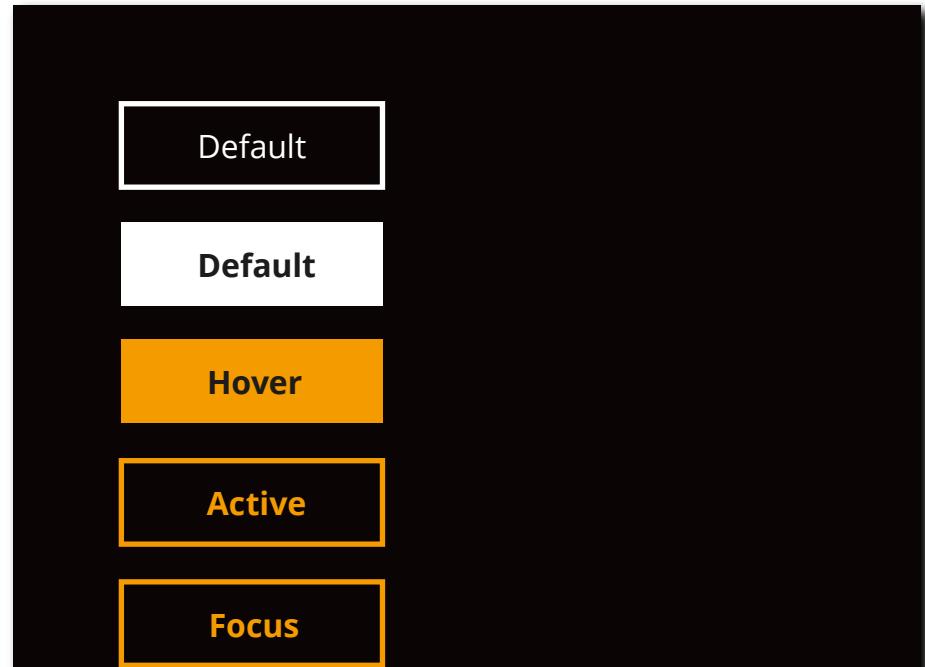
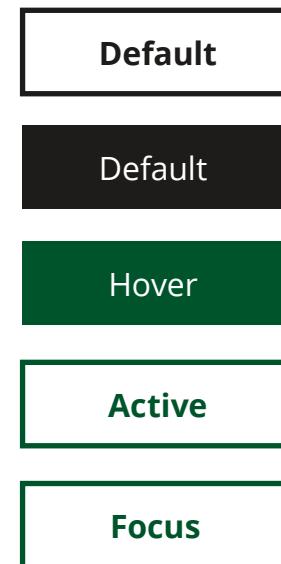
Paragraph

Font-weight : Regular

Line-height : 100%

P paragraph

Font-size : 16pt

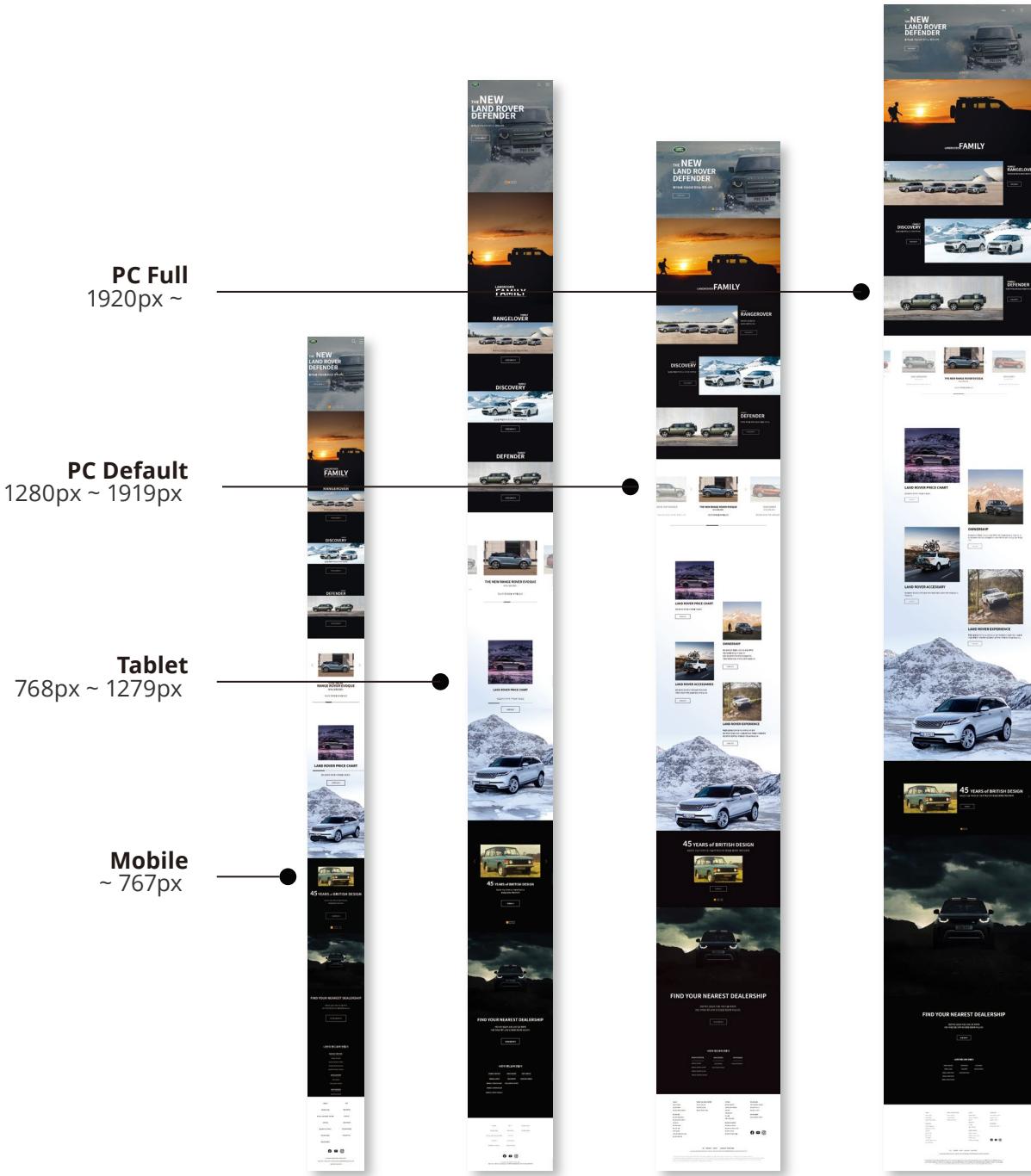


화이트 배경에는 블랙과 그린 컬러를 사용해 모던한 느낌을 주고
블랙 배경에는 화이트와 오렌지 컬러를 사용해 눈에 띄게 하였습니다.

Design

Layout
Sketch
Prototype

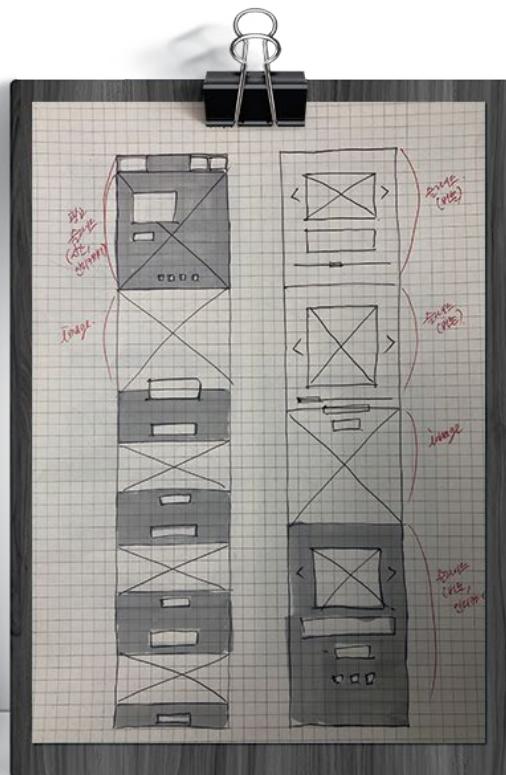
Portfolio JohnRo



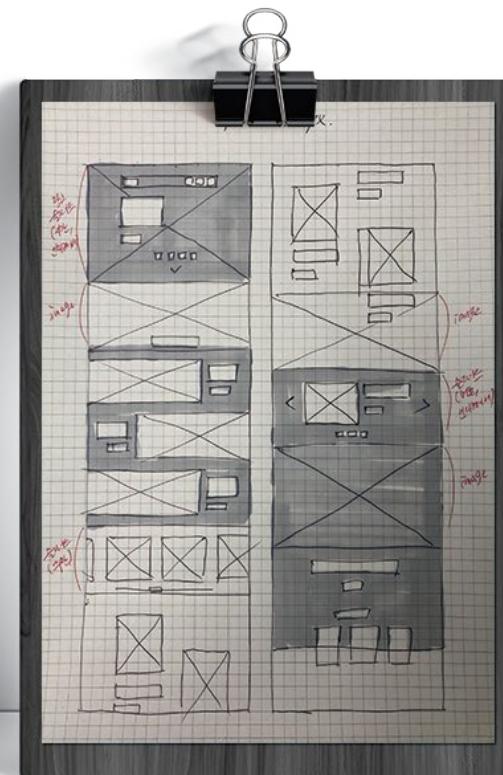
Sketch

Sketch.

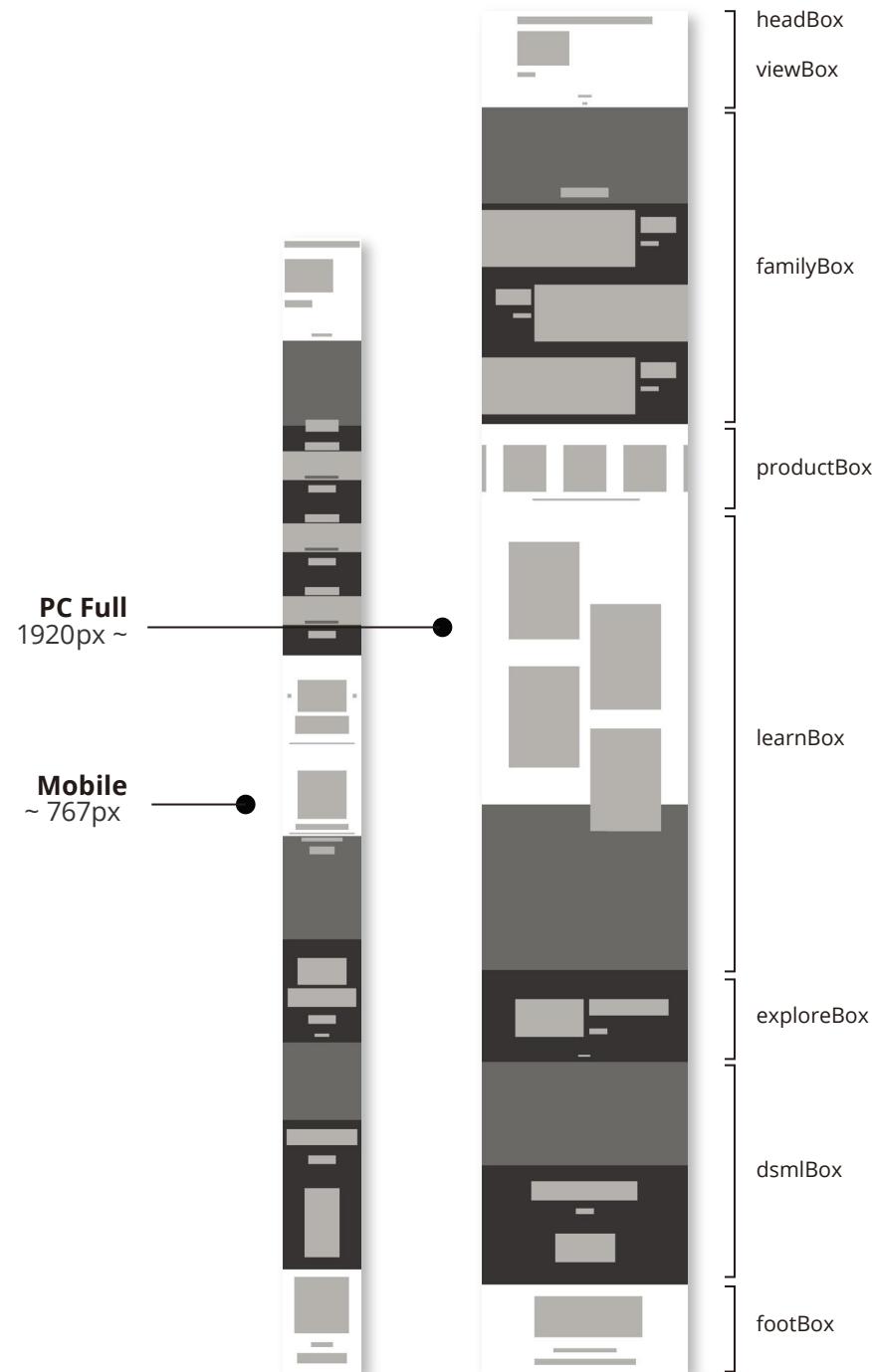
mobile



pc full(1920px)



ProtoType 제작 시 그리드 사용하여
전체 비율간격 및 레이아웃의 균일화를 하였습니다.
작업프로그램 : Illustrator



Coding.

Classification by Box.
Responsive.

Portfolio JohnRo

ViewBox

ViewBox(advertise).

jQuery

setInterval을 이용한 자동 슬라이드 구현

viewBox에 마우스 올렸을 때
setInterval 멈추고 마우스 벗어나면
setInterval 재실행

indicator 클릭 시 순서에 맞는 슬라이드로
이동 및 indicator color 변경

setInterval과 indicator 연동



SCSS

SCSS For문을 이용해 광고 이미지 삽입
가상선택자 ::before를 사용해 이미지 어둡게 만들

Mixin, Include 사용한
여러 슬라이드에 적용시킨
Indicator와 2가지 버전의 버튼

```
const viewBoxArea=viewBox.find('.view_btn_area'),
  indiUl=viewBoxArea.find('.indicator'),
  indiLi=indiUl.find('li');

viewLi.eq(-1).clone(true).prependTo(viewUl);
viewLi=viewUl.find('li');
viewUl.css({width:100 * viewLi.length + '%', marginLeft:-100+'%'});
viewLi.css({width:100 / viewLi.length + '%'});
```

```
const viewText=viewBox.find('.view_text');
let go,j=0;
const slideGo = function(){
  go=setInterval(function(){
    j++;
    if(j>=4){
      j=0;
      viewUl.css({left:100+'%'});
    }
    viewUl.animate({left:-100 * j+'%'},1200, function(){
    });
    indiLi.eq(j).addClass('action',1000);
    indiLi.eq(j).siblings().removeClass('action');
  },4500);
};
```

```
slideStop = function(){clearInterval(go)};
viewBox.on('mouseenter',function(){slideStop()});
viewBox.on('mouseleave', function(){slideGo()});
```

```
indiLi.eq(0).addClass('action');
indiLi.children('a').on('click',function(e){
  e.preventDefault();
  j=$this.parent('li').index();
  viewUl.animate({left:-100 * j + '%'});
  indiLi.eq(j).siblings().removeClass('action');
  indiLi.eq(j).addClass('action');
}); // #viewBox indicator
```

```
#viewBox{position:relative,min-height:640px;
  &::before,&::before{content:"";display:block;position:absolute;z-index:500;width:100%;height:100%;background-color:rgba(29,29,29,0.5);}
  $count:5;
  ul{position:relative;width:100%*$count,height:100%;
    li{float:left;width:100%/$count;height:100%;background-size:cover;background-repeat:no-repeat;
      background-position:70% 0;overflow:hidden;word-break:break-all;
      .view_text{position:absolute;z-index:1000;color: #fff;}
      .videoBg{position: absolute;z-index: -500;background-position: 50% 50%;}
    }
    @for $i from 1 to 4 {.view_0#{$i+1}_bg{background-image:url("../img/view/view_0+$i+.png")}}
  }
  .lineup{include lineup;
  @include indicator_square(".view_btn_area");
}
```

```
@ mixin lineup($c:#fff){
  width:168px;height:56px;text-align:center;line-height:54px;
  & a{include link;color:$c;
  p{width:100%;height:100%;font-size:0.875rem;@include border(2px,$c);box-sizing:border-box;}}
```

```
}// @ mixin Lineup
@mixin lineup_black($c:#000000){
  width:168px;height:56px;text-align:center;line-height:54px;
  & a{include link;color:$c;font-weight:bold;
  p{width:100%;height:100%;font-size:0.875rem;@include border(2px,$c);box-sizing:border-box;}}
```

```
}// @ mixin Lineup
```

```
@ mixin indicator_square($select,$pL:50,$pB:8vh){
  #{$select}{position: absolute;z-index:500;left:$pL;bottom:$pB;transform:translateX(-50%);
  width: auto; height: auto;
  >ul{width: auto; height: auto;
    >li{float: left; width: 20px; height: 20px; margin-left: 8px; border: 2px solid #fff; box-sizing: border-box;
    }
    > a{include link;
      & span{@include hidden;}}
```

HeadBox.

HeadBox(Fixed).

Click하면 메뉴창으로 화면 전환.



SCSS

Menu Icon 클릭 시
메뉴창으로 화면 전환되고
Icon이 X로 애니메이션 효과.
(rotate와 translate 사용)

```
// menu_icon
.menu_icon{position: absolute; background-color: transparent; padding: 0;
  span{display: block; position: absolute; border-radius: 1px; cursor: pointer;
    | | | background-color: #fff; transition: all .35s ease;}
  span:nth-child(1){top: 2px;}
  span:nth-child(2){top: 50%; transform: translateY(-50%);}
  span:nth-child(3){bottom: 2px;}
  &.click span:nth-child(1){top: 50%; transform: rotate(45deg) translateY(-50%);}
  &.click span:nth-child(2){opacity: 0;}
  &.click span:nth-child(3){top: 50%; transform: translateY(-50%) rotate(-45deg);}

.menu_page{display: none; position: fixed; z-index: -1; top: 0; left: 0; width: 100vw; height: 100vh;
  background-color: #1d1d1d;
  ul{@include center; height: auto;
    li{@width: auto; height: auto;
      a{@include link; color: #fff; font-size: 1rem; text-align: center;
        transition: all 0.2s linear;}}
    }
  & a:hover{color: #f89626; transform: scale(1.2);}
}
```

화면 스크롤을 내릴 시 디자인 변경.

jQuery

스크롤 내렸을 때
고정높이보다 클 시 클래스 추가

```
win.on('scroll', function(){
  let winScroll = win.scrollTop();
  // -----
  if(winScroll>headBoxH){
    headBox.addClass('action');
  }else{
    headBox.removeClass('action');
  }
})
```

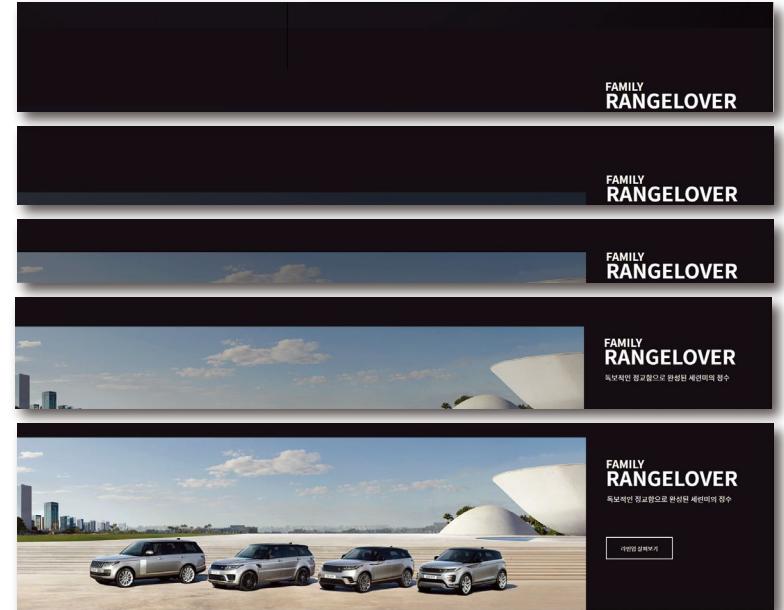
SCSS

클래스 추가 됐을 경우 HeadBox의 디자인 변경

```
#headBox.action{padding-top: 8px;
  .nav{width: 94%; height: auto;
    h1{margin-top: 2px; width: 76px; height: 40px;}
    .unb{width: auto; height: auto;
      li{margin-right: 40px;
        &:nth-child(1){font-size: 0.75rem;}
        &:nth-child(2),&:nth-child(3){margin-top: 7px; width: 30px; height: 30px;}
        &:nth-child(3){margin-right: 0; float: right;}}
      label+a{float: right;}}
    .menu_icon{width: 30px; height: 30px;}}
    .menu_icon span{width: 30px; height: 2px;}}
```



스크롤 시
아래에서 위로 이미지 올라옴



SCSS

```
.f_img{margin-top:120px;opacity:0;transition:all 1.4s ease;}.family_i .f_img.action{margin-top:0;opacity:1;}
```

클래스 추가 됐을 경우 HeadBox의 디자인 변경

jQuery

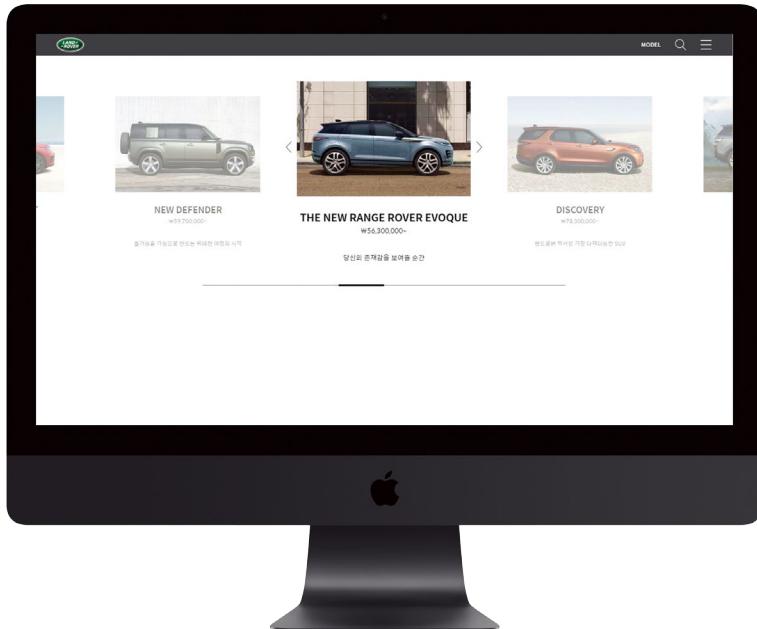
스크롤 내렸을 때
고정높이보다 클 시 클래스 추가

```
let winH=win.outerHeight();  
let headBoxH=headBox.outerHeight();  
let familyOffset1=fmainlyImg1.offset().top;  
let familyOffset2=fmainlyImg2.offset().top;  
let familyOffset3=fmainlyImg3.offset().top;  
let family_offset1=familyOffset1-(winH);  
let family_offset2=familyOffset2-(winH);  
let family_offset3=familyOffset3-(winH);
```

```
if(winScroll>family_offset1){  
    fmainlyImg1.addClass('action');  
    fmainlyImg1.addClass('action');  
}  
else{  
    fmainlyImg1.removeClass('action');  
    fmainlyImg1.removeClass('action');  
}  
if(winScroll>family_offset2){  
    fmainlyImg2.addClass('action');  
    fmainlyImg2.addClass('action');  
}  
else{  
    fmainlyImg2.removeClass('action');  
    fmainlyImg2.removeClass('action');  
}  
if(winScroll>family_offset3){  
    fmainlyImg3.addClass('action');  
    fmainlyImg3.addClass('action');  
}  
else{  
    fmainlyImg3.removeClass('action');  
    fmainlyImg3.removeClass('action');  
}  
});//#familyBox scroll
```

ProductBox.

ProductBox(models).



jQuery

model image들의 슬라이드 기능으로 버튼 클릭시
0.5초 동안 클릭 시 disabled 적용시켜 중복클릭 방지.

prependTo를 이용해 슬라이드가 반복되는거처럼 보임

버튼 클릭시 밑에 Indicator가 image 순서에 맞게 이동.

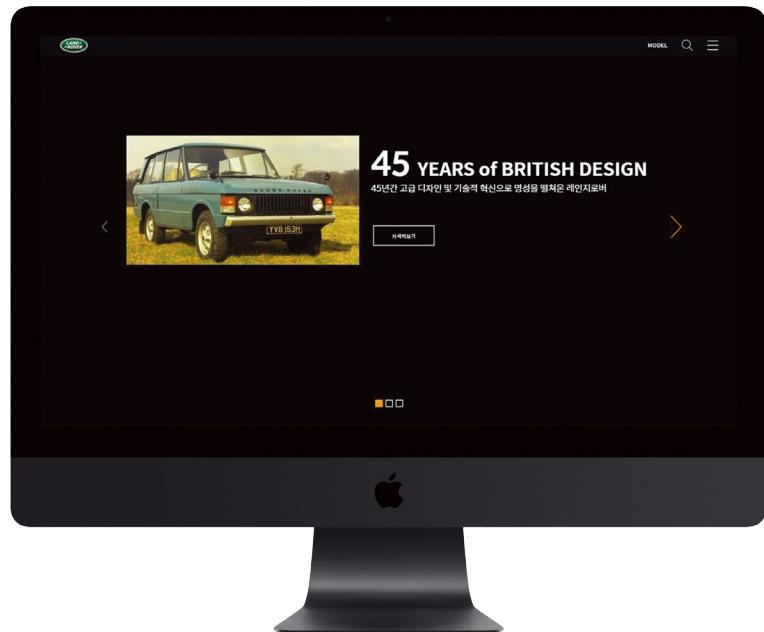
SCSS

반응형 웹을 위해 Mixin,Include 사용해서
슬라이드 디자인하고 각 반응형에 적용.

scss For문을 이용해 이미지 삽입.

```
@mixin productBox($pd,$w,$h,$bw,$bh,$c:8){  
  #productBox{position: relative; padding-top:$pd; box-sizing: border-box;  
    .product_area{position: relative; width:$w; height:$h; margin: auto;}  
    ul{position: absolute; margin-left: -300%; width: 100%*$c; height: 100%;  
      li{float: left; width: 100%/$c; height: 100%; margin: auto;  
        a{@include link($bw,$bh); margin: auto;  
          div:first-child{width: 100%; height: 100%;}}  
      }  
    }  
  }  
}
```

```
let t=0;  
productUl.find('li').eq(3).addClass('action');  
productBtn.on('click', function(e){  
  e.preventDefault();  
  e.stopPropagation();  
  let $has=$(this).hasClass('next');  
  productBtn.attr("disabled",true);  
  setTimeout(function(){productBtn.removeAttr("disabled");},500);  
  if($has){  
    t++;  
    if(t>=5){  
      productBar.css({left:-37.5+'%'})  
      t=-3;  
    }  
    productBar.stop().animate({left:12.5*t+'%'});  
    productUl.stop().animate({left:-100+'%'},function(){  
      productUl.children('li').eq(0).appendTo(productUl);  
      productUl.css({left:0});  
    });  
    productUl.find('li').eq(4).addClass('action');  
    productUl.find('li').eq(4).siblings().removeClass('action');  
  }else{  
    t--;  
    if(t<=-4){  
      productBar.css({left:50+'%'})  
      t=4;  
    }  
    productBar.stop().animate({left:12.5*t+'%'});  
    productUl.stop().animate({left:100+'%'}, function(){  
      productUl.children('li').eq(-1).prependTo(productUl);  
      productUl.css({left:0});  
    });  
    productUl.find('li').eq(2).addClass('action');  
    productUl.find('li').eq(2).siblings().removeClass('action');  
  }  
}); // #productBox button
```



jQuery

model image들의 슬라이드 기능으로 버튼 클릭시
0.5초 동안 클릭 시 disabled 적용시켜 중복클릭 방지.

prependTo를 이용해 슬라이드가 반복되는거처럼 보임
버튼 클릭시 밑에 인디케이터가 image 순서에 맞게 이동.

SCSS

scss For문을 이용해 이미지 삽입.

mixin으로 인디케이터 및 버튼이미지 적용.

```
#exploreBox{position: relative; background-color: #000000; color: #fff; text-align: center;
$count: 4;
ul{position: relative; width: 100%*$count; height: 100%;
| li{float: left; position: relative; width: 100%/$count; height: 100%;}
}
.ex_bg{background-size: cover; background-repeat: no-repeat; background-position: 50% 50%;}
@for $i from 1 to 4 { .ex_bg_#{$i}; background-image: url("../img/explore/explore_#{$i}.jpg");}
.lineup(@include lineup; @include center; @include indicator_square(".view_btn_area"));
.ex_btn(@include center; z-index: 100; width: 50%; height: 0; top: 50%);
button{float: left; position: relative; width: 30px; height: 30px; background-color: transparent;
color: #1d1d1d; padding: 0;
div{width: 100%; height: 100%}
&.next{float: right;
div(@include bg("../img/arrow/arrow_right.png", 50%, 50%));
&:hover div(@include bg("../img/arrow/arrow_right_hover.png", 50%, 50%));
}
&.prev{
div(@include bg("../img/arrow/arrow_left.png", 50%, 50%));
&:hover div(@include bg("../img/arrow/arrow_left_hover.png", 50%, 50%));
}
}
}
```

```
exploreUl.css({marginLeft: '-100%'});
exploreUl.children('li').eq(-1).clone().prependTo(exploreUl);
exploreLen++;

let i=0;
exploreBtn.on('click', function(e){
e.preventDefault();
// console.log( $(this) );
let $has=$(this).hasClass('next');
if($has){
i++;
if(i>=exploreLen-1){i=0;
exploreUl.css({left: '100%'});
}
} else{i--;
exploreUl.stop().animate({left: -100*i+'%'}, function(){
if(i<=1){
i=exploreLen-2;
exploreUl.css({left: -100*i+'%'});
}
});
indil2.eq(i).addClass('action');
indil2.eq(i).siblings().removeclass('action');
}); // #exploreBox button
```

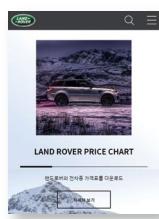
Responsive.

Responsive.

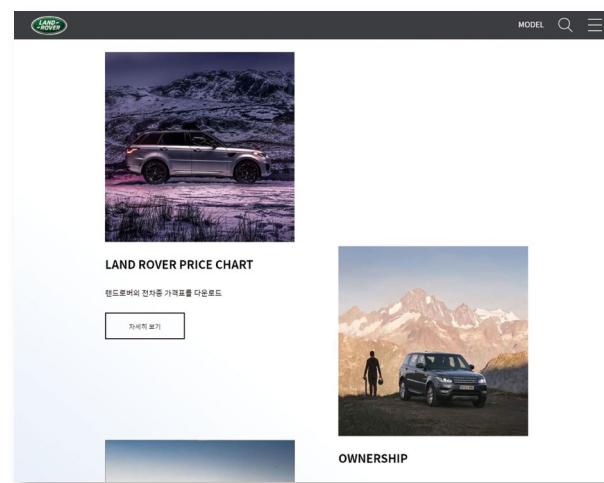
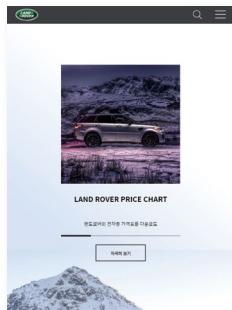
LearnBox Responsive.

mobile과 tablet은 슬라이드로 구성하고
pc default와 pc full은 아래로 나열했습니다.

mobile(~767px)

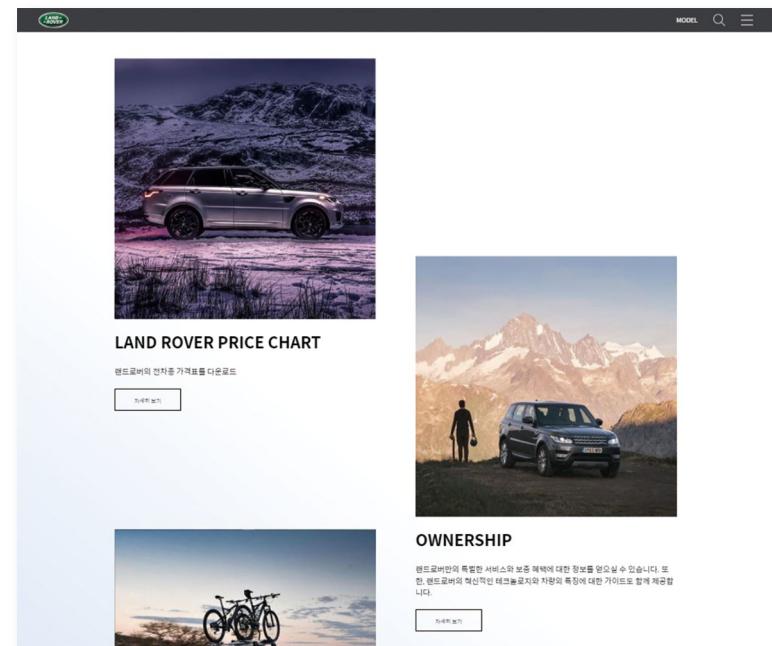


tablet(768px~1279px)



pc default(1280px~1919px)

pc full(1920px~)



Responsive.

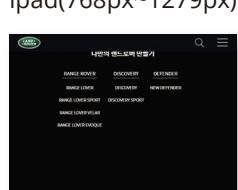
Responsive.

FootBox Responsive.

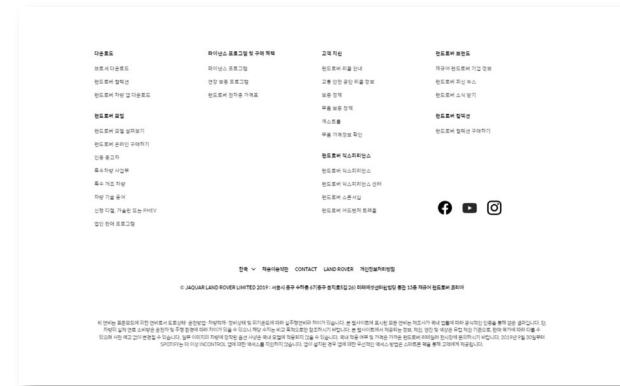
반응형으로 구성하기 위해 생각한 방법인

column 또는 float&position 중 float과 position을 사용해 구성했습니다.

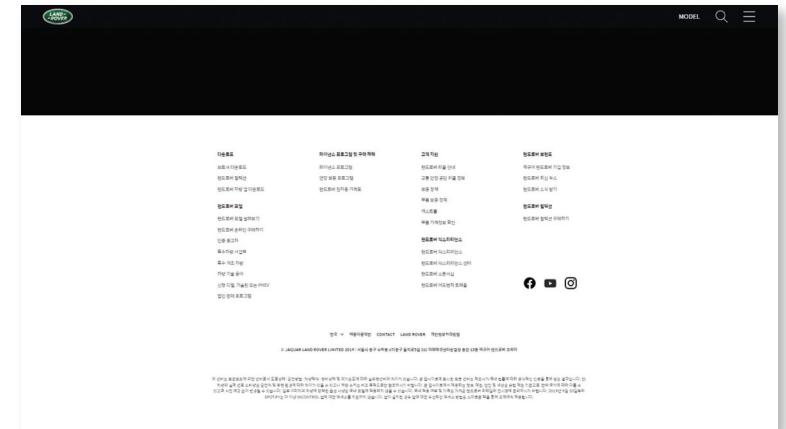
mobile(~767px)



pc default(1280px~1919px)



pc 1920px(1920px~)

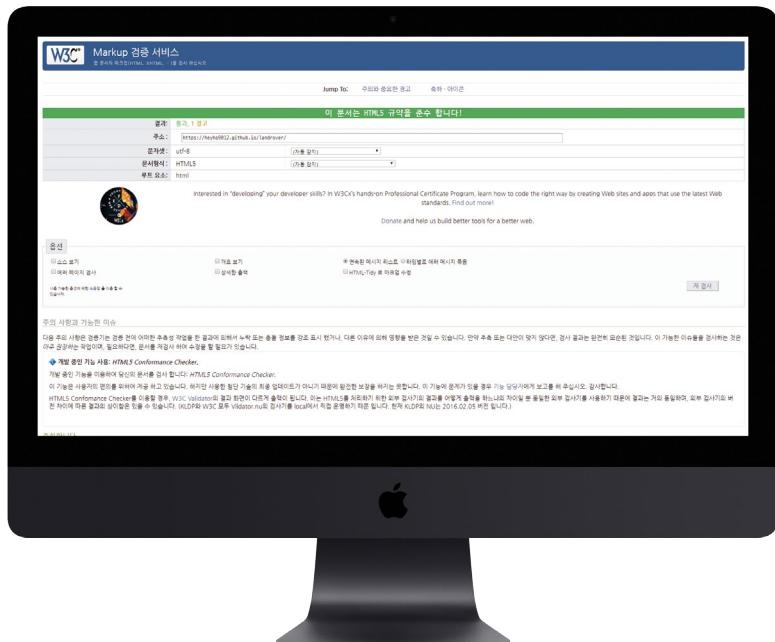


Validate.

HTML5

CSS3

Portfolio JohnRo



Validate
Validate.



Review.

What I Learned.

Improvements.

Portfolio JohnRo

What I Learned.

What I Learned

HTML5

1. 의미에 맞게 태그를 써야 함.
2. 유효성에 맞는 사용법.

CSS3

1. 반응형 웹 적용 방법.
2. SCSS를 통해 조건문, 반복문, 네스팅 변수 등 표준 CSS 보다 훨씬 많은 기능을 사용

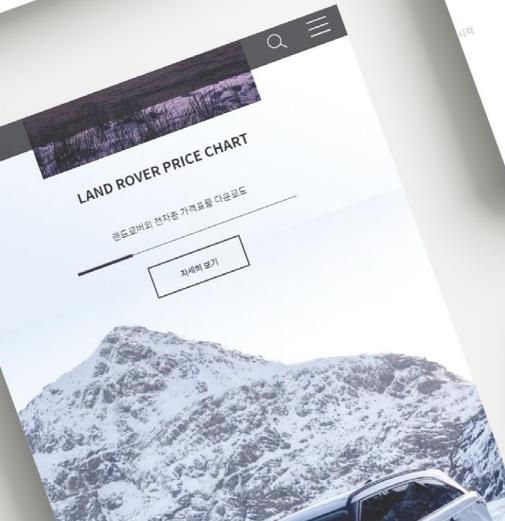
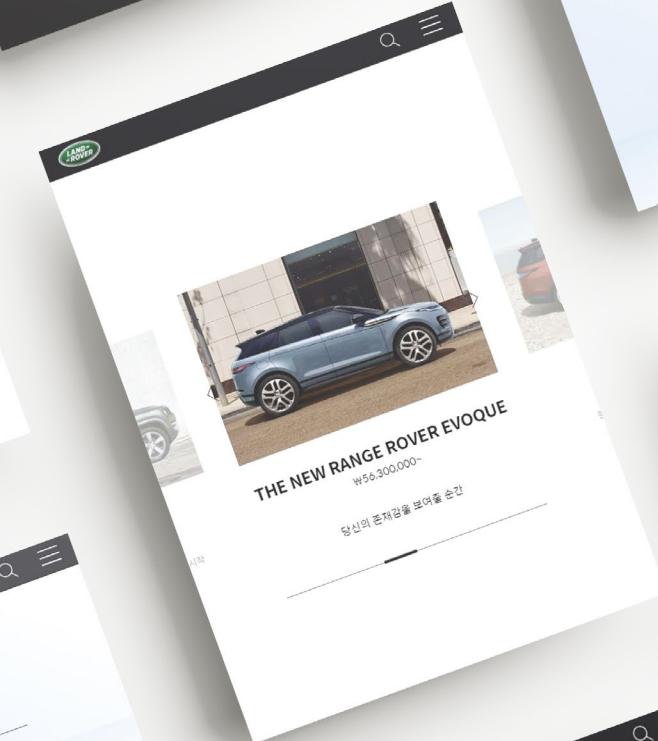
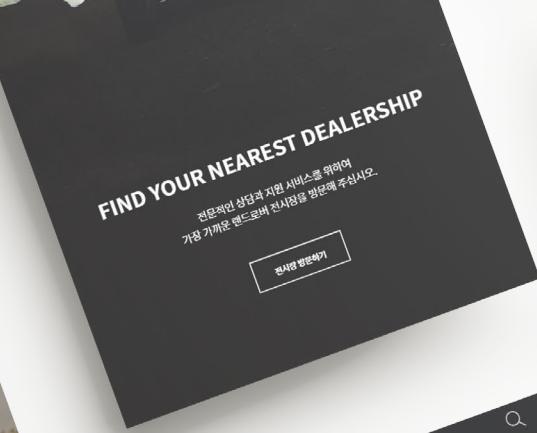
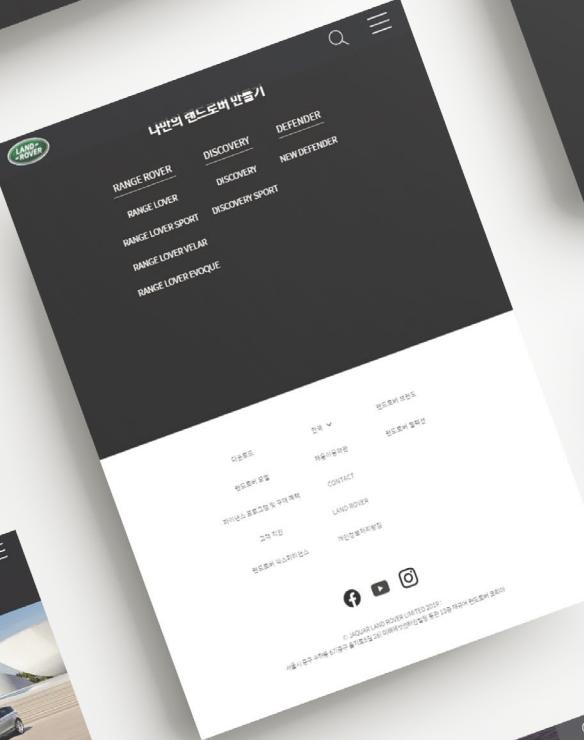
jQuery

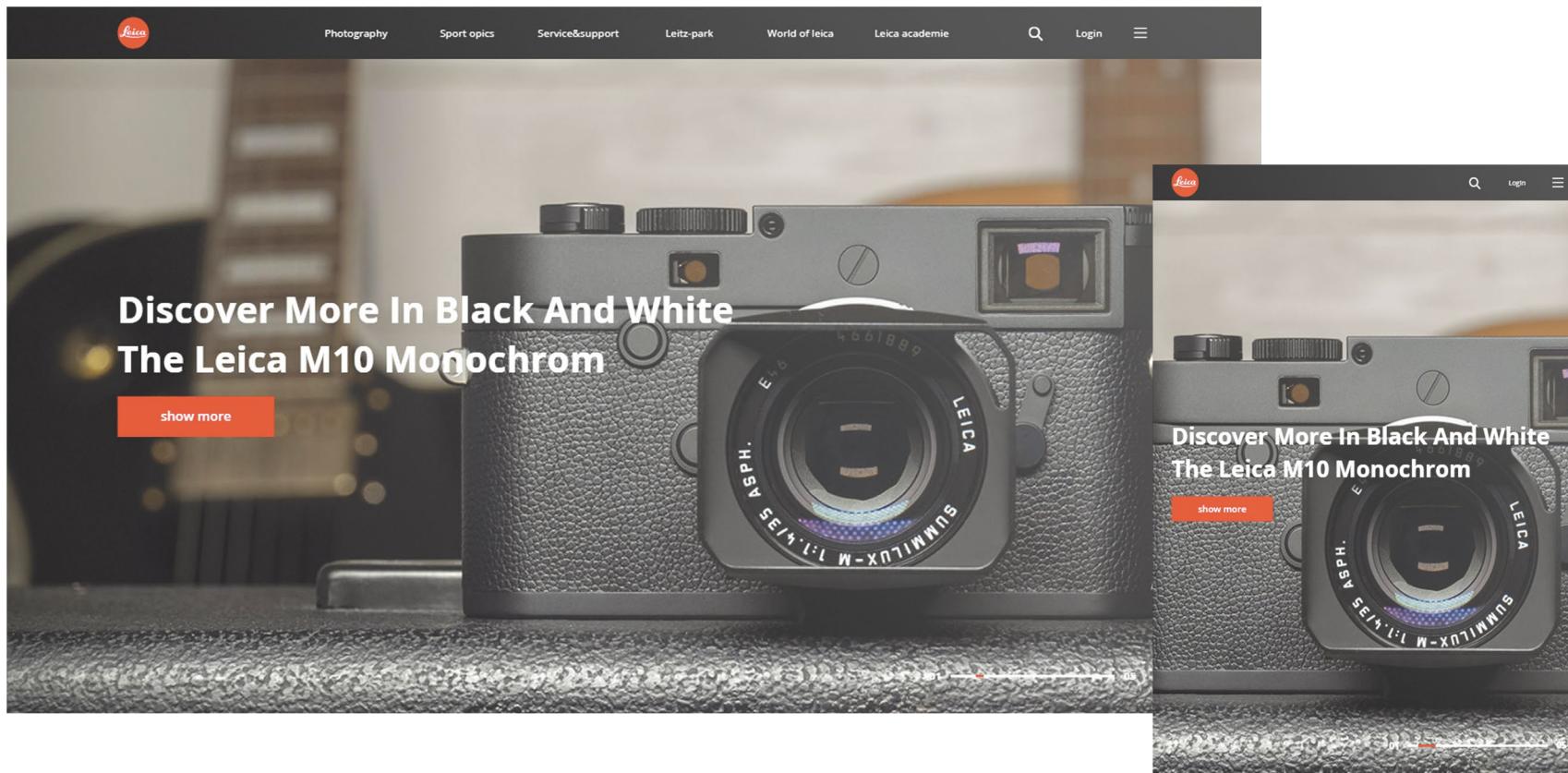
1. 여러가지 효과를 줄 수 있는 함수들. (fadeIn&Out,slideUp&Down,toggle)
2. SCSS에서 할 수 없는 적용범위가 가능함.
3. offset을 통한 scroll 기능.

Improvements.

1. 포지션의 사용을 점점 줄이고 필요한 부분에만 적용.
2. 리뉴얼 서브 페이지 추가 제작
3. 현재 explore에는 적용이 안 되어있는 display:flex에 대한 공부.







Project 02.

Analysis & Concept
Style Guide
Sketch
Proto Type
Design
Coding
Validate
Review

LEICA camera AG

<https://heyho9012.github.io/leica/>
3 weeks
HTML5, SCSS(CSS3), jQuery, Photoshop, Illustrator
mobile tablet basic pc

Original site Analysis.

사이트 이름
LEICA camera AG

사이트 주소
<https://kr.leica-camera.com/>

브랜드 유형
camera

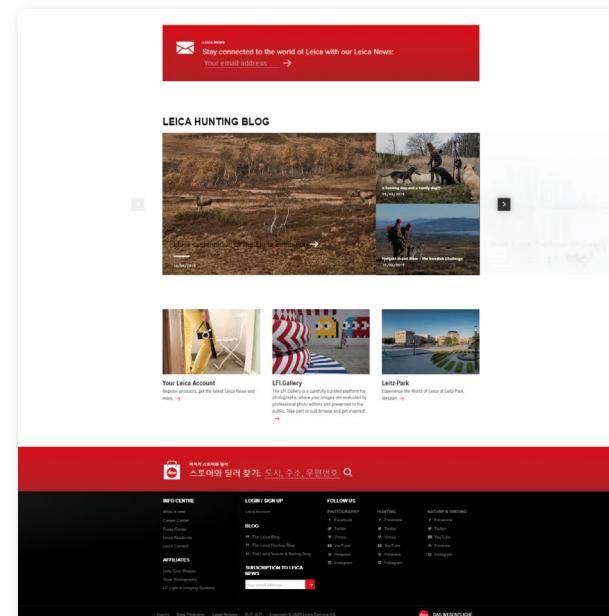
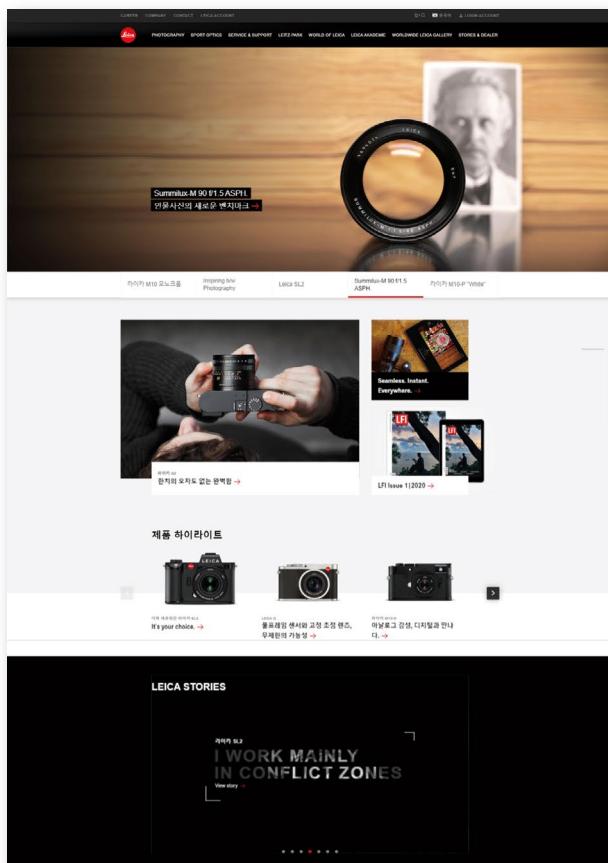
브랜드 컨셉
희귀성, 오랜역사, 명품

주요 고객층
20-50대

사이트 선정이유
기존 사이트보다 좀 더 흥미를
일으킬 수 있는 사이트로 만들기 위해

Portfolio JohnRo

Original site.



카테고리와 사이트에 있는
기능들의 대한 focus가
잡혀있지 않다.
또한, 카테고리가 복잡하고
메뉴의 sub메뉴가 너무 길다.

디자인들이 너무 심플해서
지루 할 수 있다.
한글과 영어가 의미없이
섞여 있다.

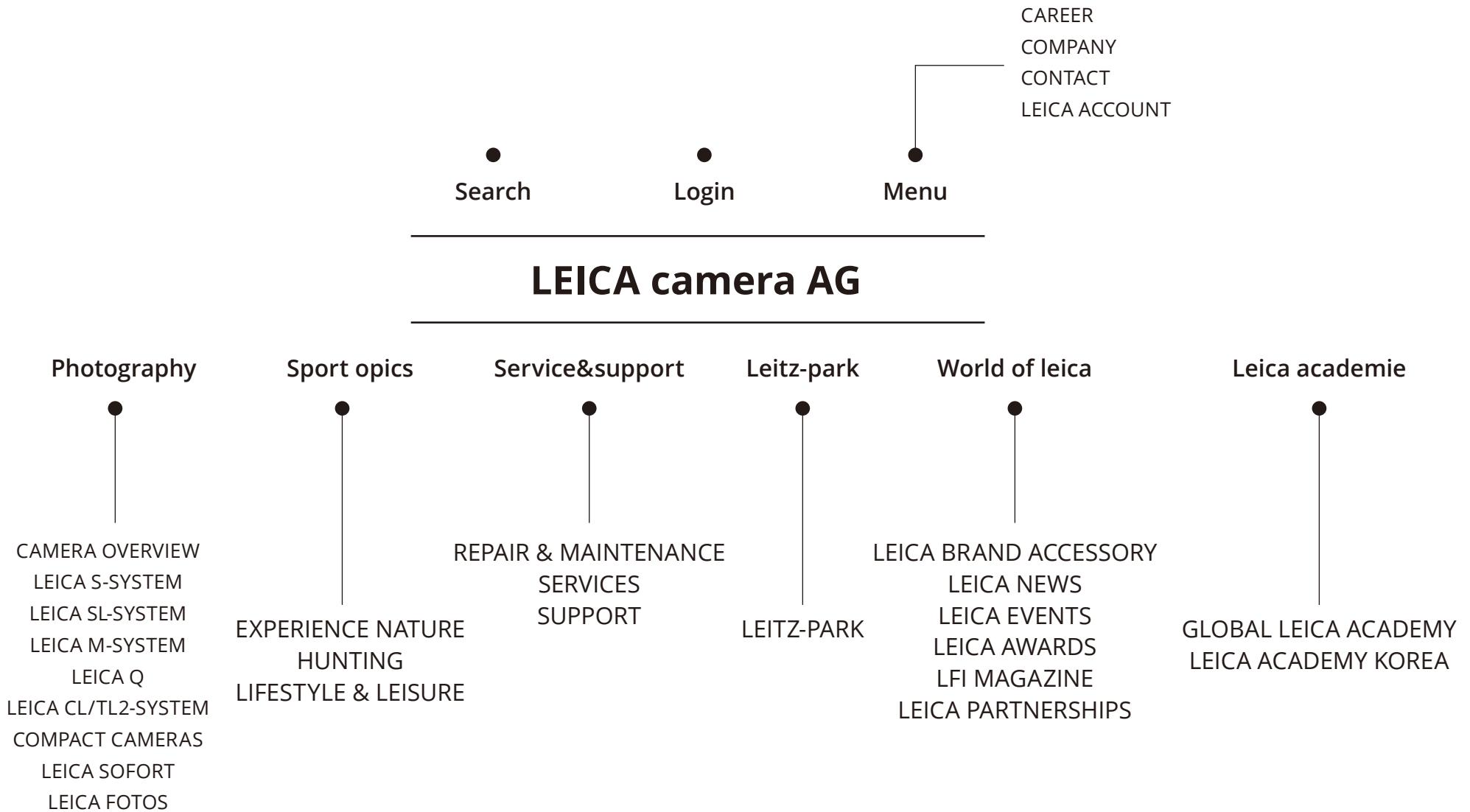
Renewal

Information Architecture

Persona

SWOT

<http://heyho9012.github.io/landrover>





이혜주

31 | 사진작가 | 수원 | 월 소득: 400

NEEDS 특징 : 한정판 카메라 수집

목적 : 카메라 구입

요구사항 : 레이아웃이 제품의 특징을 잘 표현하지 못한다.
또, 모든 페이지가 정적이어서 흥미를 떨어트리고
메뉴가 너무 많아 들어갈 곳을 찾지 못하겠다.

Solutions

1. 동적으로 만들어 흥미를 유발시키고
2. 제품의 특징을 살려 주목성을 높인다.
3. 메뉴를 사이트에 중복된 것은 제거하거나 수정해 간단하게 만든다.

Strengths

희소성, 브랜드 애호가 만족

라이카는 주로 동급 명품 브랜드, 패션 디자이너 등과 협업해 한정판 카메라를 출시한다. 전세계에서 수십~수백대만 판매되는 한정판 제품은 희소함이 가장 큰 특징이다. 소비자 사이에서 명품·프리미엄 브랜드로 알려진 제품이라면 더욱 그렇다. 라이카는 희소성을 부각하기 위해 전용 액세서리를 동봉하거나, 특별한 디자인이나 각인을 적용한다.

S

Opportunities

RF 마니아 매료, 한정판은 브랜드 강화와 수익 가져다줘

현재 디지털 RF 카메라를 제조하는 곳은 라이카 밖에 없다. 일부 크라우드펀딩 프로젝트나 스타트업이 디지털 RF 카메라를 개발하겠다고 발표했지만, 종도에 개발을 취소하거나 아직 시제품을 발표하지 못했다. RF 카메라 마니아들은 여전히 라이카 브랜드와 제품을 쓸 수밖에 없는 처지다. 가격 저항은 있지만, 라이카는 한정판 제품이 회사 수익을 늘리고 브랜드 영향력을 높이는 효과를 준다는 평가를 받는다.

O

Weakness

원래 비싸다고는 하지만, 입 벌어지는 가격

한정판 제품은 대개 일반 제품보다 비싸다. 라이카가 한정판 제품의 가격을 항상 비싸게 책정한 것은 아니다. 한국에서 한정 판매된 라이카 Q 퍼플 한정판은 일반 제품과 거의 같은 가격에 판매됐다.

T

Threats

빈약한 한정판 출시 명분, 브랜드 이미지 깨아먹는다

디지털 카메라 시대에 큰 의미 없는 한정판을 쏟아내는 라이카를 비판하는 목소리도 높다. 출시 명분이 빈약하고 역사가 너무 짧다는 것이다. 디지털 카메라 한정판은 일반 제품처럼 가격이 꾸준히 떨어지며 희소성이 덜하다. 명분이 빈약한 한정판 출시를 남발하는 것은 라이카 브랜드 이미지를 훼손할 수 있다.

style
GuideLine

Color

Font

Button

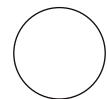
Portfolio JohnRo

포인트 컬러로 오리지날 사이트의 컬러인 레드 컬러와
블랙계열으로 깔끔하고 세련된 느낌을 주었고 베이지계열을 사용하여
화이트 컬러만 사용했을 때보다 다양한 느낌을 주려고 했습니다.

Primary



#060606
RGB(6,6,6)



#FFFFFF
RGB(255,255,255)



#E20613
RGB(226,6,19)

Secondary



#F6EBE8
RGB(246,235,242)



#909090
RGB(144,144,144)



#EOE0EO
RGB(244,244,244)

Color Contrast



#060606
#FFFFFF
20.3 : 1



#E20613
#FFFFFF
4.9 : 1



#060606
#E20613
4.1 : 1



#F6EBE8
#060606
17.3 : 1



#E20613
#060606
4.1 : 1



#060606
#EOE0EO
15.3 : 1

명도대비 3:1이상은 24px 이상 또는 굵은 18px 이상 사용

Font

Font.

브랜드 이미지에 맞는 깔끔하고 세련된 느낌의 서체로
영어는 opensans를 사용하고 한글은 기본적인 노토산스를 사용했습니다.

Font Family

Aa

Open
Sans

가

노토
산스

Aa Light

Aa Regular

Aa Semibold

Aa Bold

가 Thin

가 Light

가 Regular

가 Bold

Font Size

Headers

Font-weight : Bold Line-height : 100%

H2 header

Font-size : 32pt

H3 header

Font-size : 28pt

H4 header

Font-size : 24pt

H5 header

Font-size : 20pt

H6 header

Font-size : 16pt

Paragraph

Font-weight : Regular
Line-height : 100%

P paragraph

Font-size : 16pt



Hover : Button 180 deg rotate



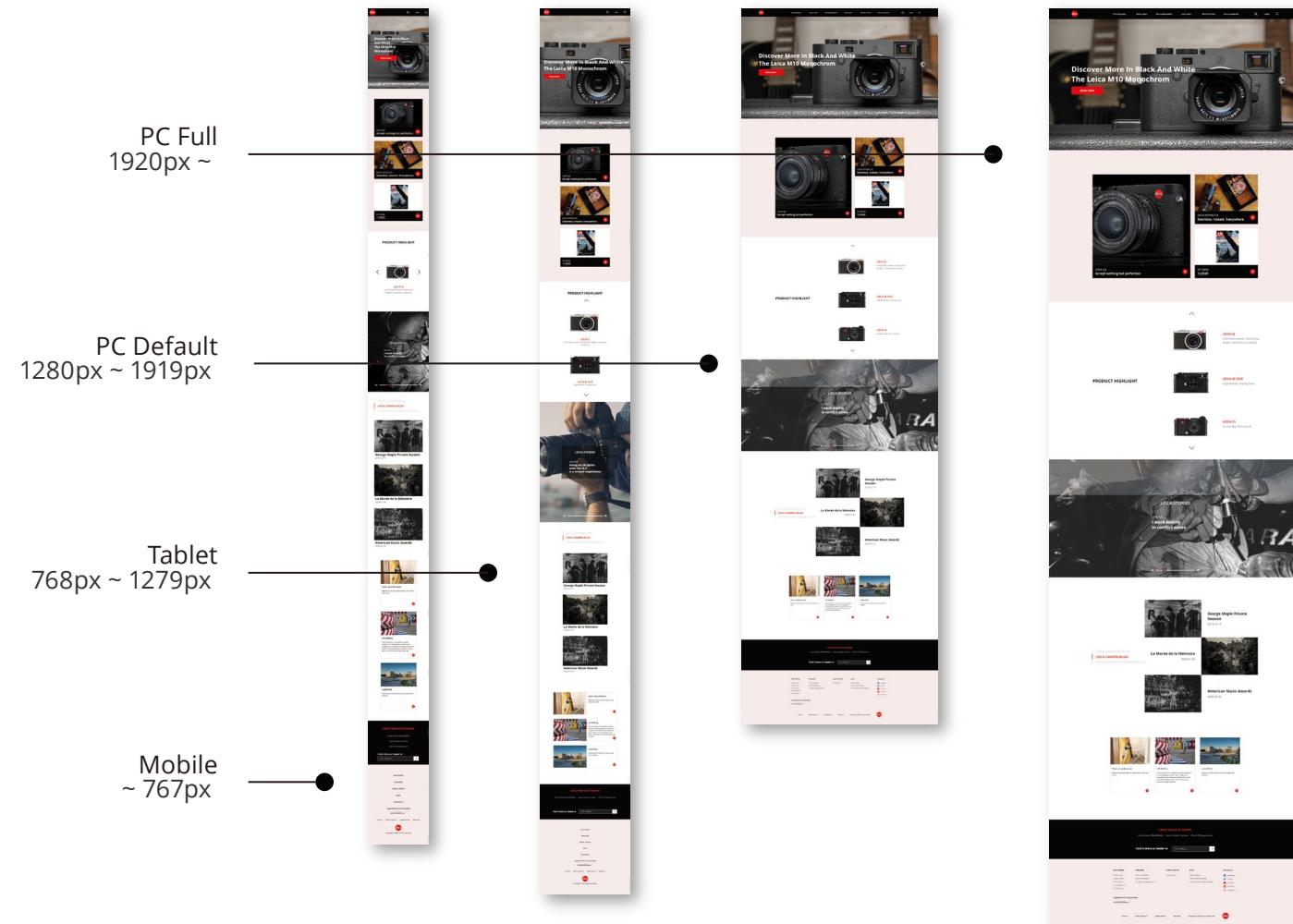
Focus : Button 180 deg rotate

버튼을 사각형과 원형의 두가지 버전으로 사용해 동적인 느낌을 주었고
컬러는 둘 다 블랙과 레드의 조합을 사용해서 브랜드 이미지의 느낌을 주었습니다.

Design

Layout
Sketch
Prototype

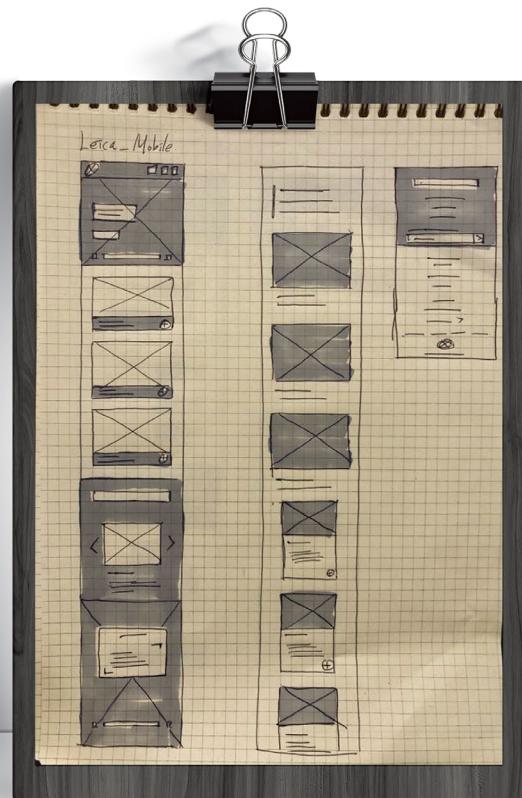
Portfolio JohnRo



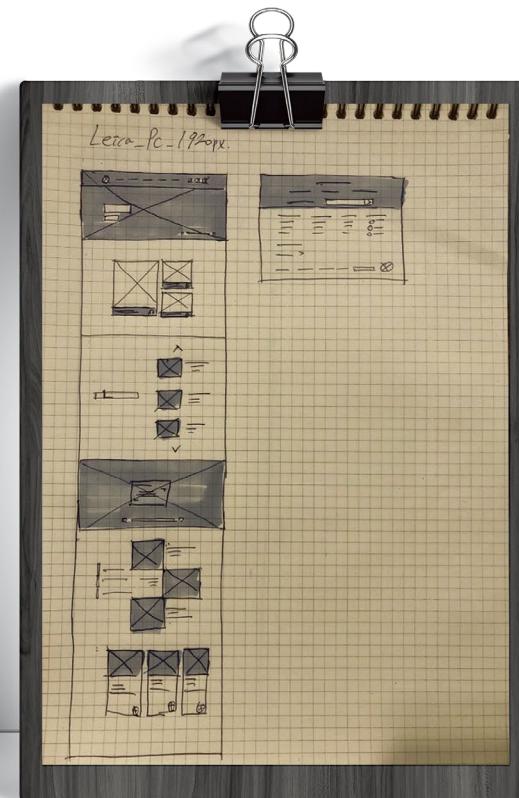
Sketch

Sketch.

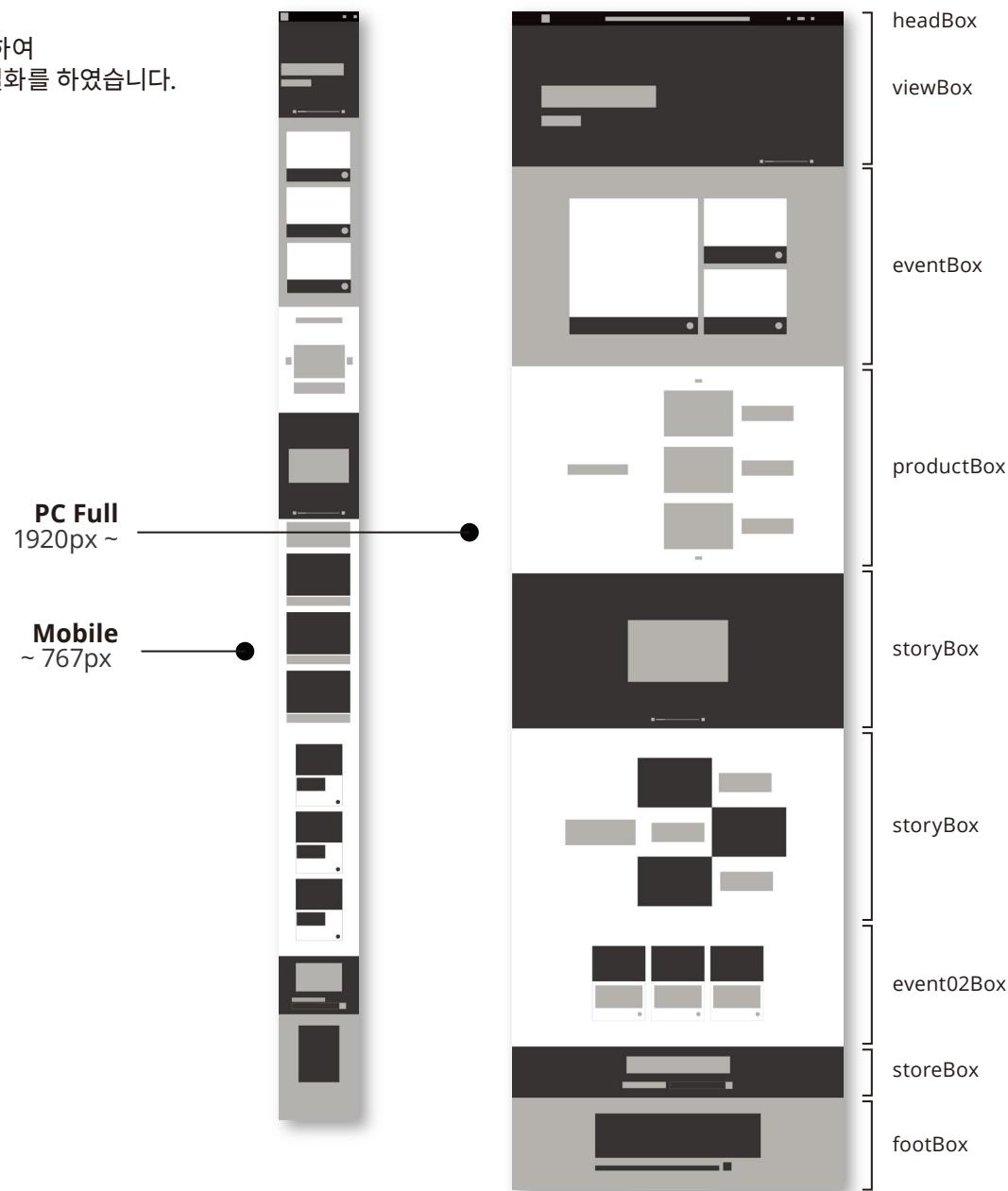
mobile



pc full(1920px)



ProtoType 제작 시 그리드 사용하여
전체 비율간격 및 레이아웃의 균일화를 하였습니다.
작업프로그램 : Illustrator



Coding.

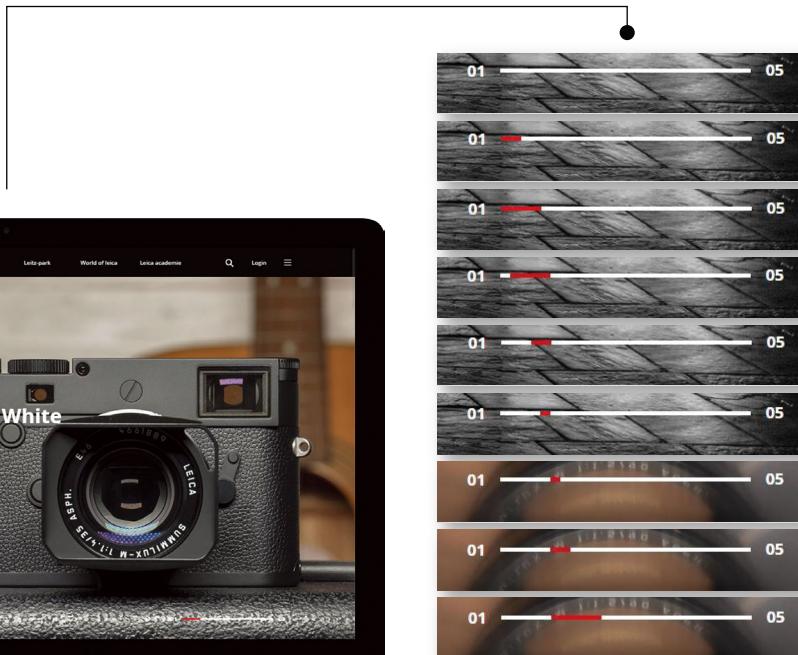
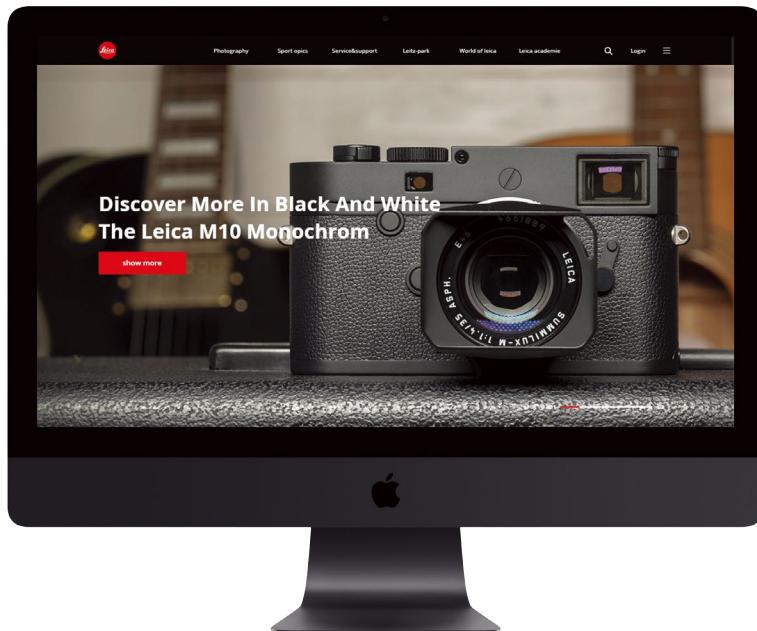
Classification by Box.
Responsive.

Portfolio JohnRo

ViewBox

ViewBox(advertise).

indicator가 왼쪽에서 오른쪽으로 이동



```
// for(let i=0;i<5;i++)indili.eq(i).css({left:'50*i}px'});
for(let i=0;i<6;i++)indilo2.eq(i).css({left:'50*i}px'});
// -----
let go,i=0,j=0,timed=2000;
indili.eq(0).find('span').animate({left:'50px'},timed*1.5);
indilo2.eq(0).find('span').animate({left:'50px'},timed*1.5);
let indifn = function(indicator,li){
  indicator.find('a').on('click',function(e){
    e.preventDefault();
    e.stopPropagation();
    let indiIndex=$(this).parent().index();
    li.eq(indiIndex).fadeIn(timed);
    li.eq(indiIndex).siblings().fadeOut(timed);
    i=li.eq(indiIndex).index()-1;
    j=li.eq(indiIndex).index()-1;
  });
};

const slideGo = function(){
  indifn(indili,viewLi);
  indifn(indilo2,storyLi);
  go=setInterval(function(){
    indili.eq(i).siblings().find('span').css({left:'-50px'});
    indilo2.eq(j).siblings().find('span').css({left:'-50px'});
    i++; if(i==5){i=0;}
    j++; if(j==6){j=0;}
    viewLi.eq(i).fadeIn(timed);
    viewLi.eq(i).siblings().fadeOut(timed);
    storyLi.eq(j).fadeIn(timed);
    storyLi.eq(j).siblings().fadeOut(timed);
    indili.eq(i).find('span').stop().animate({left:'50px'},timed*2);
    indilo2.eq(j).find('span').stop().animate({left:'50px'},timed*2);
  },timed*2);
}; slideGo(); // viewBox, storyBox SETINTERVAL fadeIn&out & indicatorBar

const slideStop = function(){clearInterval(go)};
viewDiv.on('mouseenter', function(){slideStop()}).on('mouseleave', function(){slideGo()});
storyDiv.on('mouseenter', function(){slideStop()}).on('mouseleave', function(){slideGo()});
```

jQuery

setInterval을 이용한 자동 슬라이드 구현

indicator가 순서에 맞게 위치가 왼쪽에서 오른쪽으로 이동

indicator 클릭 시 순서에 맞는 슬라이드로 이동 및 indicator 변경

setInterval과 indicator 연동

```
#viewBox{position:relative,height:100%;}
@include indicator;
@ul{position: relative; width: 100%; height: 100%;}
  >li{position: absolute; left: 0; top: 0; width: 100%; height: 100%; background-repeat: no-repeat; background-size: cover; background-position: 70% 50%; box-sizing: border-box;
    &:last-child(p{color: #060606;})
    &:not(:nth-child(1 ))(display:none);
    @for $i from 1 to 6 {&#{$i})background-image: url("../img/view/view_0"+$i+"_.jpg");}
  div{position: absolute; top: 50%; transform: translateY(-50%); width: auto; height: auto;
    p{margin-bottom: 1.25em; font-weight: bold; color: #fff;}
    a{background-color: #e20e13; color: #fff; text-align: center; font-weight: 600; padding: 0; transition: all .3s ease;
      &:hover,&:focus{background-color: #060606; color: #e20e13; font-weight: 600;}}
```

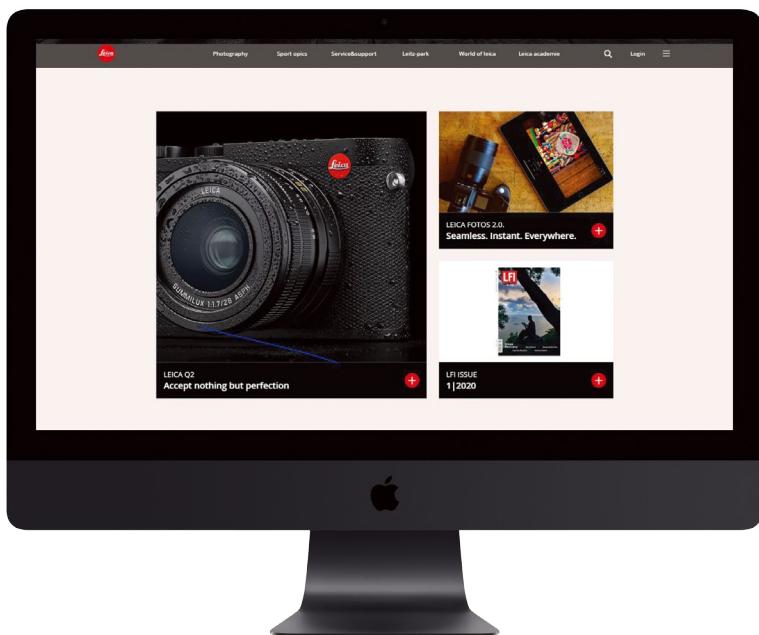
SCSS

scss For문을 이용해 광고 이미지 삽입
fadeIn,Out사용을 위해 1번쨰 슬라이드 제외 display:none

Mixin, Include 사용한
여러 슬라이드에 적용시킨 인디케이터

HeadBox.

HeadBox(Fixed).



Click하면 메뉴창으로 화면 전환.



```
button{position: absolute; top: 50%; transform: translateY(-50%); width: 20px; height: 16px;
  span{display: block; position: absolute; width: 100%; height: 2px; border-radius: 0.0625rem; background-color: #fff; transition: all .7s ease;}
  span:nth-child(1){top: 0;}
  span:nth-child(2){top: 50%; transform: translateY(-50%);}
  span:nth-child(3){bottom: 0;}
```

// -----

```
&.menu_act span{background-color: #e20613;}
```

```
&.menu_act span:nth-child(1){top: 50%; transform: translateY(-50%) rotate(45deg);}
```

```
&.menu_act span:nth-child(2){opacity: 0;}
```

```
&.menu_act span:nth-child(3){bottom: 50%; transform: translateY(50%) rotate(-45deg);}
```

```
}}
```

```
.menu_box{position: fixed; z-index: -1; top: 0; left: 0; width: 100%; height: 100%; background-color: #060606;
  div(@include img_center; width: auto; height: auto;
    ul{float: left;
      li{margin-bottom: 2rem; &:last-child{margin-bottom: 0;}}
        a{@include link; color: #fff; text-align: center; font-weight: 600;
          &:hover, &:focus{color: #e20613; font-weight: 600;}}
    }}}
```

```
#headBox.action{background-color: #rgba(6,6,6,0.7); nav>ul>li>ul{background-color: #rgba(6,6,6,0.7);}.menu_box{background-color: #rgba(6,6,6,0.8);}}
```

```
// -----
```

```
let winH=win.outerHeight();
headBoxH=headBox.outerHeight();
```



```
win.on('scroll', function(){
  let winScroll = win.scrollTop();
  // if(winScroll>headBoxH*2){
  //   headBox.addClass('action');
  // }else{
  //   headBox.removeClass('action');
  // }
}); // headBox SCROLL action
```

jQuery

스크롤 내렸을 때
고정위치보다 클 시 클래스 추가



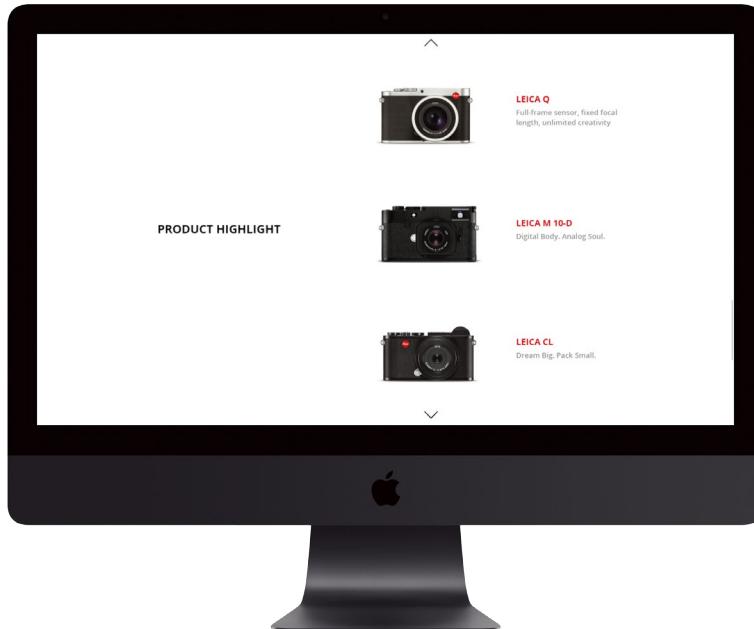
SCSS

Menu Icon 클릭 시 메뉴창으로 화면 전환되고
Icon이 X로 애니메이션 효과. (rotate와 translate 사용)

scroll 내릴 시 addClass에 의한 header와 메뉴창 opacity 80%

ProductBox.

ProductBox.(models)



Mobile에서는 가로 슬라이드로 전환

```

const ProductSlide = function(){
  productBtn.off().on('click',function(e){
    e.preventDefault();
    e.stopPropagation();

    let $hasNext=$this.hasClass('next');

    productBtn.attr("disabled",true);
    setTimeout(function(){productBtn.removeAttr("disabled")},500);

    if($hasNext){
      c++;
      productWrap.stop().animate({top:-productListH},function(){
        productWrap.children('li').eq(0).appendTo(productWrap);
        productWrap.css({top:0});
      });
    }else{
      c--;
      productWrap.stop().animate({top:productListH},function(){
        productWrap.children('li').eq(-1).prependTo(productWrap);
        productWrap.css({top:0});
      });
    }
  });

  productWrap.on('touchstart',function(event){
    startX = event.originalEvent.changedTouches[0].screenX;
  });

  productWrap.on('touchend',function(event){
    endX=event.originalEvent.changedTouches[0].screenX;
    if(startX-endX>50){
      k++;
      productWrap.stop().animate({top:-productListH},function(){
        productWrap.children('li').eq(0).appendTo(productWrap);
        productWrap.css({top:0});
      });
    }else if(endX-startX>50){
      k--;
      productWrap.stop().animate({top:productListH},function(){
        productWrap.children('li').eq(0).prependTo(productWrap);
        productWrap.css({top:0});
      });
    }
    // else if(startX-endX<50 || endX-startX<50){}
  });
}; // ProductBox list TOUCH move slide

```

```

// -----
productWrap.find('a').on('mouseenter focus',function(){
  $(this).addClass('product_action');
  $(this).parent().siblings('p').addClass('product_action');
}).on('mouseleave blur',function(){
  $(this).removeClass('product_action');
  $(this).parent().siblings('p').removeClass('product_action');
});

// ----

const DeviceSlide = function(winW){
  if(winW <= mobile){
    ProductSlideM();
  }else{
    ProductSlide();
  }
}; DeviceSlide(beforeW);

// -----

```

jQuery

productListH=productWrap.find('li').outerHeight(true)로 반응형마다 슬라이드 적용 높이 다르게 만듦.

model image들의 슬라이드 기능으로 버튼 클릭시 0.5초 동안 클릭 시 disabled 적용시켜 중복클릭 방지.

prependTo, appendTo를 이용해 슬라이드가 반복되는거처럼 보임

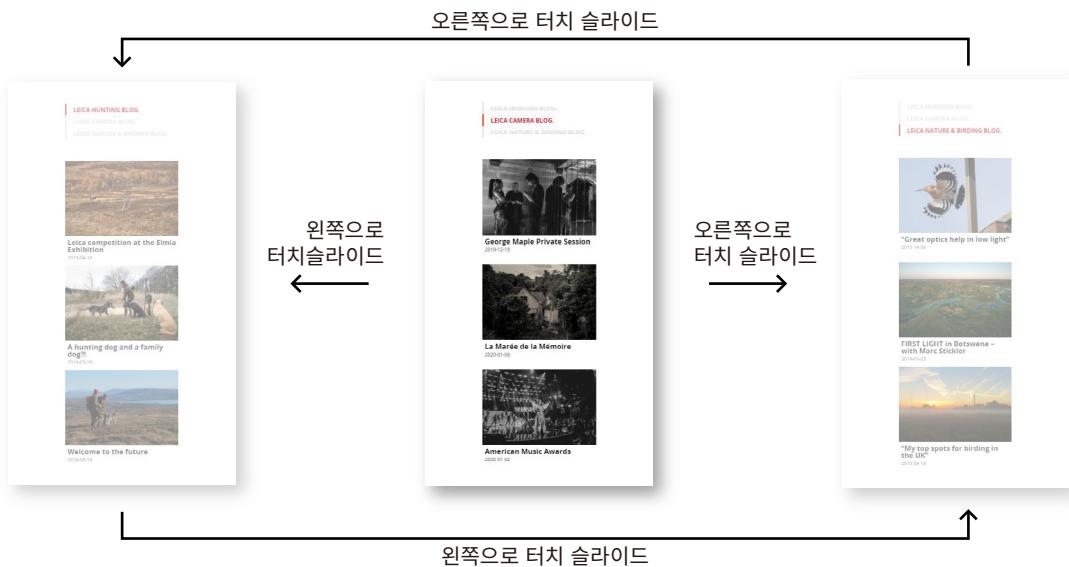
슬라이드를 함수화하여 모바일사이즈에서는 ProductSlideM() 적용. (Mobile은 가로 슬라이드 tablet,PC는 세로 슬라이드)



BlogBox.



mobile & tablet일 때



jQuery

왼쪽 타이틀 클릭 시 fadeIn & Out으로 변경됨.

mobile, tablet의 경우 터치 슬라이드 적용해
왼쪽, 오른쪽으로 터치이동 시 타이틀 클릭 시와
마찬가지로 fadeIn & Out 적용.

```

let k=1,timed=3000;

blogTitle.find('li').find('a').on('click focus',function(e){
  e.preventDefault();
  let blogIndex=$('#this').parent().index();
  blogCon.children('li').eq(blogIndex).stop().fadeIn(timed/6).siblings().stop().fadeOut(timed/6);
  k-blogCon.children('li').eq(blogIndex).index();
  $('#this').parent().addClass('blog_act').siblings().removeClass('blog_act');
}); // blogBox list CLICK fadeIn&Out
//-----
let startX,endX;

blogBox.on('touchstart',function(event){
  startX = event.originalEvent.changedTouches[0].screenX;
});

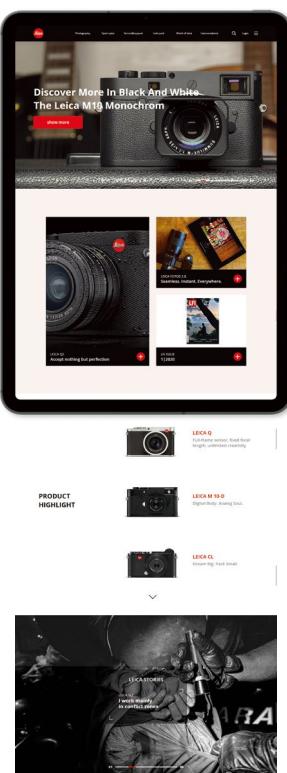
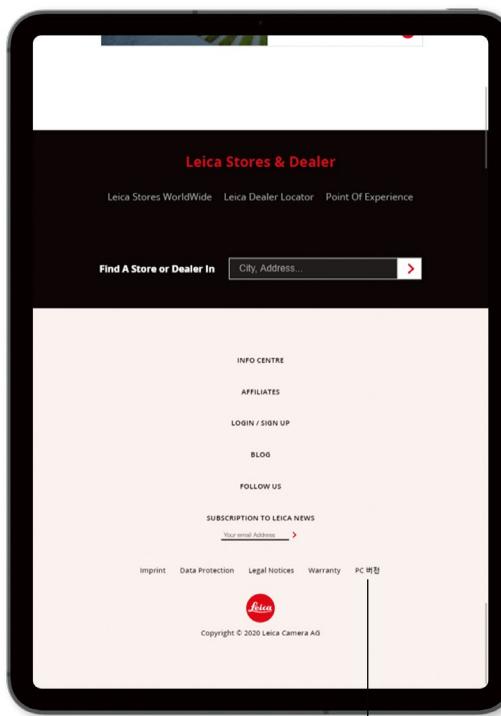
blogBox.on('touchend',function(event){
  endX=event.originalEvent.changedTouches[0].screenX;
  if(startX-endX>50){
    k--;
    if(k<=-2){k=1;}
    blogCon.children('li').eq(k).stop().fadeIn(timed/6).siblings().stop().fadeOut(timed/6);
    blogTitle.children('li').eq(k).addClass('blog_act').siblings().removeClass('blog_act');
  }else if(endX-startX>50){
    k++;
    if(k>=3){k=0;}
    blogCon.children('li').eq(k).stop().fadeIn(timed/6).siblings().stop().fadeOut(timed/6);
    blogTitle.children('li').eq(k).addClass('blog_act').siblings().removeClass('blog_act');
  }
  // else if(startX-endX<50 || endX-startX<50 ){}

}); // blogBox list TOUCH move fadeIn&Out

```

ExploreBox.

ExploreBox.



mobile & tablet일 경우 PC버전 링크 만듦.



```
let count = 0;
const PcVer = function(){
  footBox.find('.pc_ver').on('click',function(e){
    e.preventDefault();
    $('html,body').stop().animate({scrollTop:0},0);
    $('meta').eq(1).attr({"content":"width=1280"});
    $('#viewBox').css('height: 800px');
    ProductSlide();
    footList01.find('dt').find('a').off('click');
    count++;
  });
}

// -----
const DeviceFoot = function(winW){
  if(winW <= tablet){
    FootFn();
    PcVer();
  }
}; DeviceFoot(beforeW);

// -----
```

jQuery

PcVer 함수를 만들어 tablet 이하 사이즈 일 경우만 적용 가능하게 만들고 PC버전 링크를 클릭 했을 때 meta태그의 content의 width를 pc size로 변경.

사이즈를 변경하면 새로고침되도록 만들었으므로 PC버전 클릭 시 새로고침은 되지 않고 productBox 슬라이드 함수 pc용으로 변경.

footBox의 아코디언 적용 안되게 off('click') 사용.

Responsive.

Responsive.

Responsive.

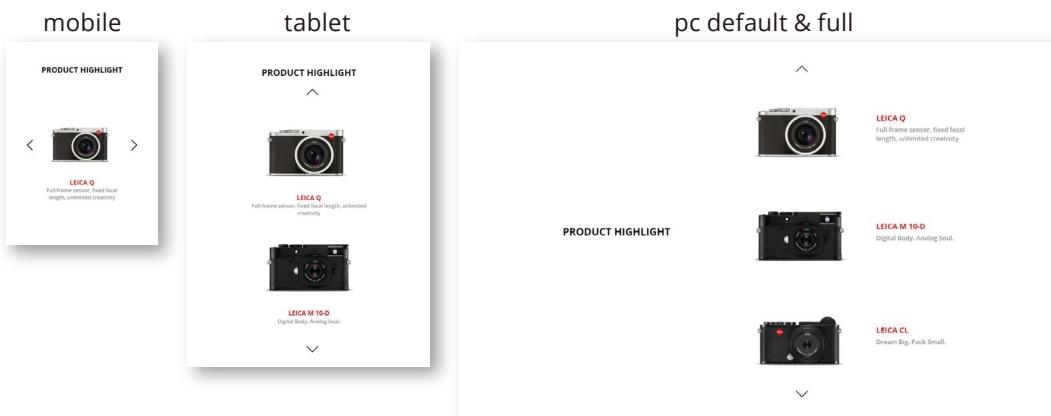
mobile, tablet, pcdefault(1280px), pcfull(1920px)로 정해진 사이즈로 변경되면
새로고침 적용.

jQuery

jQuery로 rwd파일을 만들어 반응형별로 사이즈가(가로값) 변경되면
새로고침 적용(firefox에서도 적용되게 window.location과 location.reload() 사용).

PcVer클릭시 count가 10이 되게 하고 1일 때 새로고침안되게 적용.

디바이스 별 사이즈 정의하고 고정 사이즈가 되면 새로고침.



ProductBox Responsive.

let productListH=productWrap.find('li').outerHeight(true)
반응형 사이즈 마다 li의 height값이 다르기 때문에 li의 적용값을 다르게 하기 위해 새로고침 후 height값 적용

```
const mobile=767, tablet=1279, laptop=1919, pc=1920;
const device = ['mobile', 'tablet', 'laptop', 'pcfull'];

// browser userAgentLowerCase();
let nowb = null;
if (browser.indexOf('firefox') !== -1) {
  nowb = 'firefox';
} else {
  nowb = 'ohter';
}

// 사이즈 변경 체크
$(window).on('resize', function () {
  let afterW = $(window).outerWidth(true);
  let afterDevice = DeviceSet(afterW);
  if (beforeDevice !== afterDevice) { // 가로값이 변경되었을 때만 새로고침(세로값 변화 상관없음)
    if (nowb === 'firefox') {
      window.location = window.location;
    } else if (count==1){
      count=0;
    } else {
      location.reload();
    }
  }
});

// 디바이스 크기 체크
const DeviceSet = function(winW){
  if(winW <= mobile){
    nowSize = device[0];
  }else if(winW > mobile && winW <= tablet){
    nowSize = device[1];
  }else if(winW > tablet && winW <= laptop){
    nowSize = device[2];
  }else{
    nowSize = device[3];
  }
  return nowSize;
};

const DeviceData = function(wid){
  switch(wid){
    case device[0]:
      break;
    case device[1]:
      break;
    case device[2]:
      break;
    case device[3]:
      break;
  }
}; // DeviceData 함수

// 디바이스 크기 체크
const DeviceSet = function(winW){
  if(winW <= mobile){
    nowSize = device[0];
  }else if(winW > mobile && winW <= tablet){
    nowSize = device[1];
  }else if(winW > tablet && winW <= laptop){
    nowSize = device[2];
  }else{
    nowSize = device[3];
  }
  return nowSize;
}; // DeviceSet 조건문 할수화 처리
let beforeDevice = DeviceSet(beforeW);
DeviceSet(beforeW); // DeviceSet함수실행

setTimeout(function(){
  DeviceData(beforeDevice);
}, 200);
```

Responsive.

Responsive.

FootBox Responsive.

반응형으로 구성 및 mobile과 tablet에서만 footBox의 메뉴들을 accordion toggle 적용.

mobile(~767px)



tablet(768px~1279px)



pc default(1280px~1919px)



pc full(1920px~)



jQuery

tablet 사이즈 이하에서만 accordion 함수 적용.

```
// -----
const FootFn = function(){
  footList01.find('dt').find('a').on('click',function(e){
    e.preventDefault();
    $(this).parent().siblings('dd').stop().slideToggle();
    for(let i=0;i<4;i++){
      $(this).parent().parent().parent().siblings().find('dd').eq(i).stop().slideUp();
    }
  });
};

// -----
const DeviceFoot = function(winW){
  if(winW <= tablet){
    FootFn();
    PcVer();
  }
}; DeviceFoot(beforeW);

// -----
```

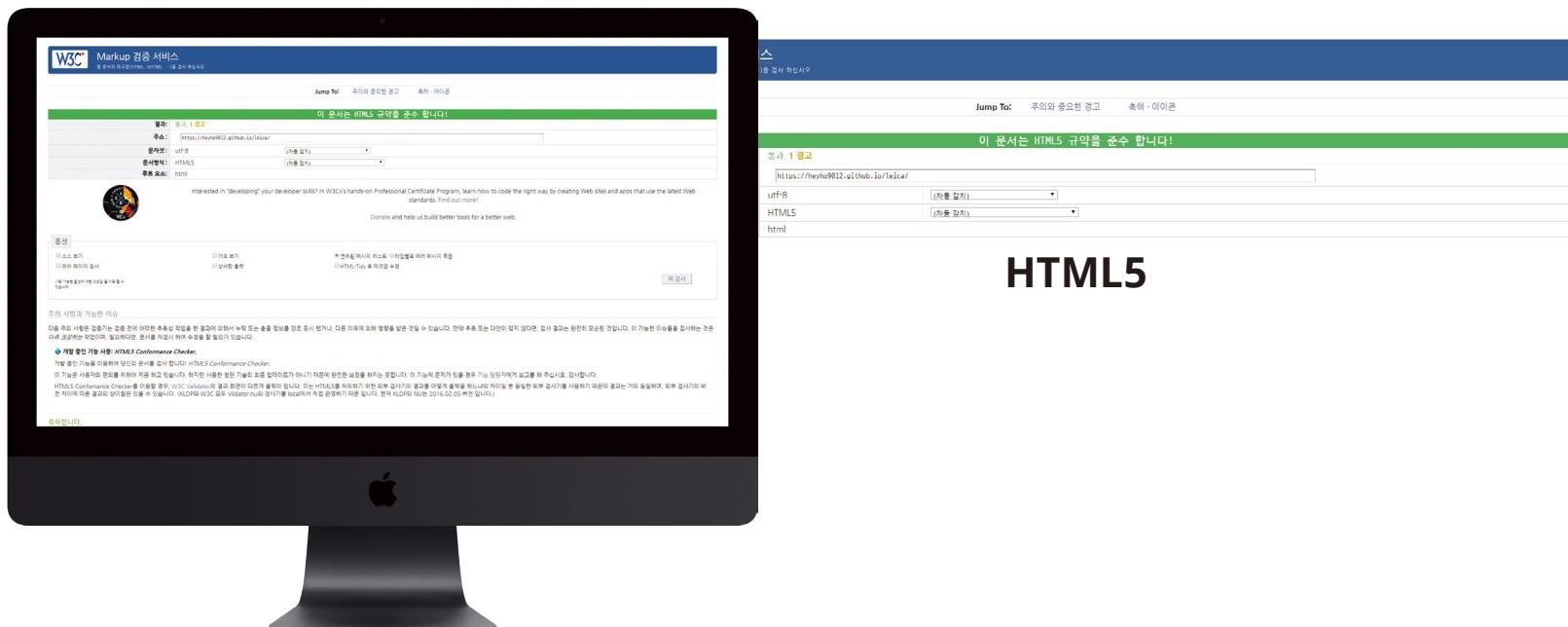
Validate.

HTML5

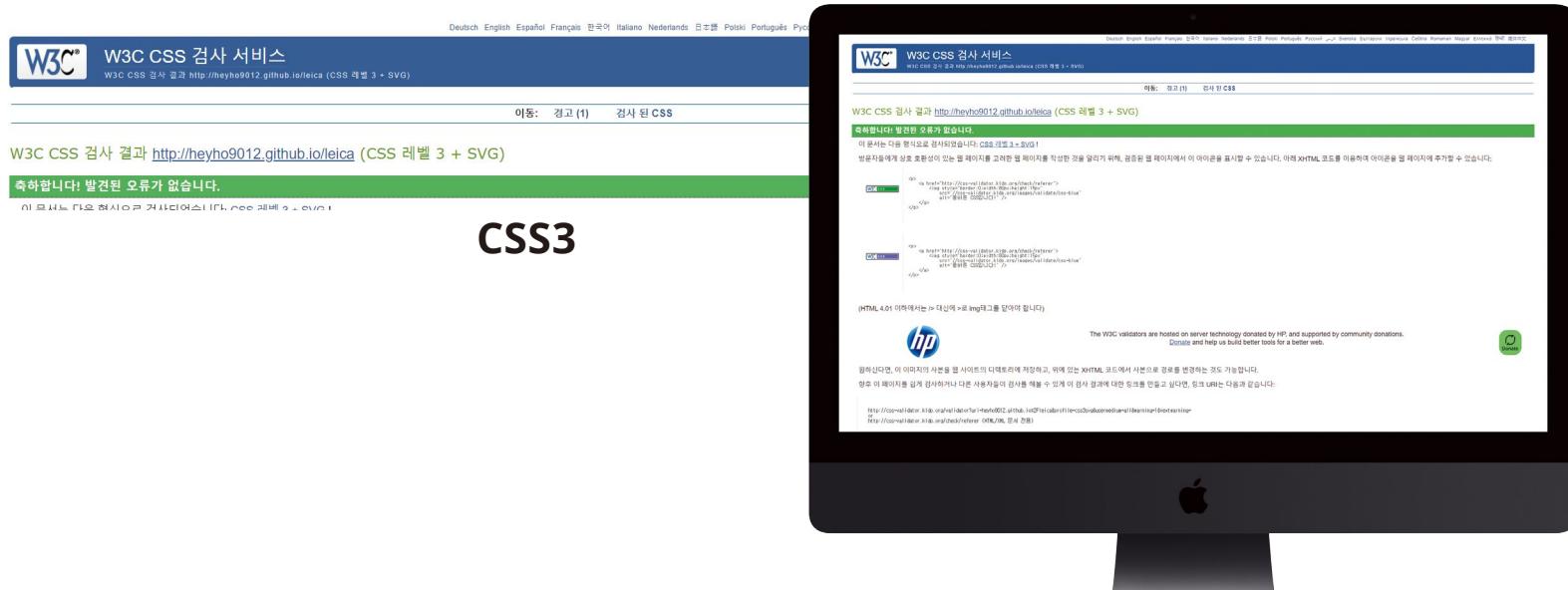
CSS3

Portfolio JohnRo

Validate
Validate.



HTML5



CSS3

Review.

What I Learned.

Improvements.

Portfolio JohnRo

What I Learned.

What I Learned

HTML5

- 1. 첫번째 사이트보다 더 깔끔한 코딩
- 2. 박스(article)별로 html을 나눠서 작업
- 3. form, label, input태그에 대한 이해도

SCSS(CSS3)

- 1. height:100% 적용
- 2. header와 footer를 나눠서
작업해 sub페이지에 적용

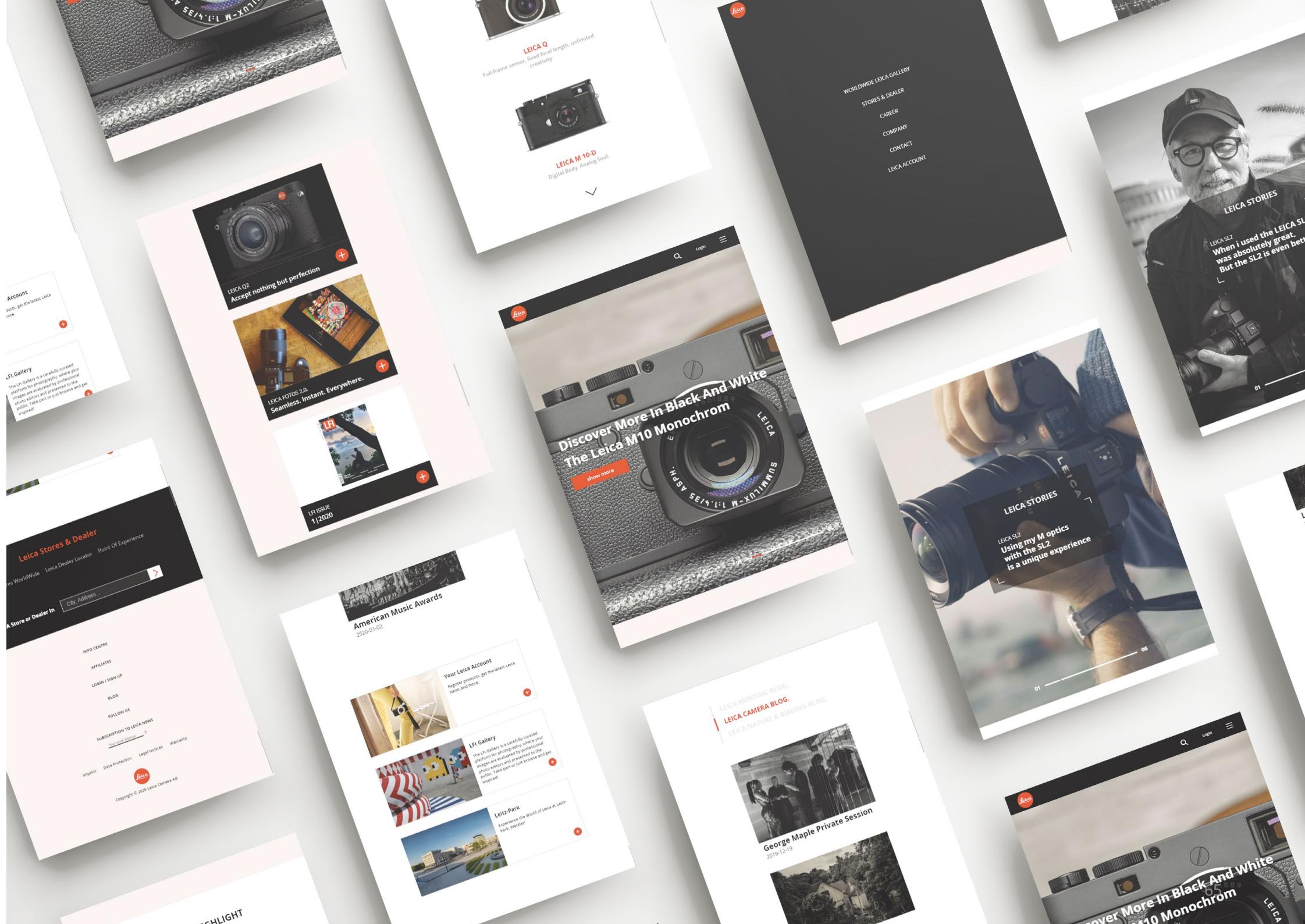
jQuery

- 1. touch이벤트를 통해 모바일에서 터치 슬라이드 기능을 적용
- 2. 반응형 웹으로써 사이즈가 변경되면
새로고침 및 사이즈별 기능이 달라지게 함
- 3. 함수를 사용해 좀 더 간단하게 적용

Improvements.

- 1. jQuery에 대한 공부를 해서 이해도가 좀 더 높아짐.
- 2. 반응형별 기능 및 구조 적용.





Identity.

Concept

Design

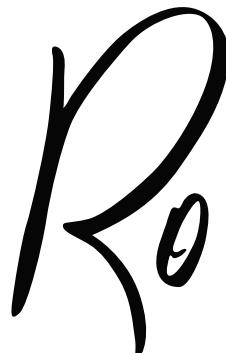
Size

Color

Concept

성을 영어로 바꿔 Ro를 필기체같이 형상화한 로고

Design



Size

Default Size

20mm x 20mm
(57px x 57px)

Maximum Size

56mm x 56mm
(160px x 160px)

Color

Black

#060606
R 6 G 6 B 6
C 0 M 0 Y 0 K 100

White

#FFFFFF
R 255 G 255 B 255
C 0 M 0 Y 0 K 0