* 3번 방법이 가장 호율이 좋습니다.
* 현재 방법론적으로 접근 했으며, javascript는 좀더 경량화 할 예정입니다.

방법론적으로 봐주세요.

1. **스크롤시 이미지 부분 show (jquery.lazyload.js 사용).**

<http://61.73.39.135/lonelyplanet/index_20170322_lazyload.html>

* *작동 원리*

<https://cdnjs.cloudflare.com/ajax/libs/jquery.lazyload/1.9.1/jquery.lazyload.js>

jquery.lazyload.js를 사용 하여 각각 이미지에 class=”lazy”를 추가 하여 스크롤 내릴 때 이미지 위치를 체크해서 보여 준다.

* *장점*

이미지를 처음부터 불러오지 않기 때문에 로딩시 데이터 가져올때 빠름

* *단점*

1) 가상의 dom 일때 인식을 못함. (slick.js, ajax)

2) img가 아닌 background 일때 문제가 됨

3) loading이 느리면 반응 하는데 시간이 걸림 (버그도 있음)

4) 마이크로 사이트 에서 사용 하면 좋은 효과를 기대 할수 있습니다.

Ex) https://kr.pinterest.com/

1. **스크롤시 /lonelyplanet/inc/main\_include.html 데이터 호출.**

<http://61.73.39.135/lonelyplanet/index_20170323_2_jquery.html>

* *작동 원리*

<div id="main\_include"></div> 영역을 잡는다.

$("#main\_include").load('/lonelyplanet/inc/main\_include.html', function(){

사용해서 데이터를 삽입.

* *장점*

<http://61.73.39.135/lonelyplanet/index_banner20170314.html>

위 URL 비교했을때 최대 4배정도 빠르며 최소 1초 정도 줄일수 있다.

* *단점*

As-is 론리보다 많은 수정이 발생함.(다음 리뉴얼 때 반영 예상)

1. **2번째 방법 + 메인 비쥬얼 배너(이미지 용량이 가장 큼)를 처음 1개만 노출하고 2개~마지막 배너 이미지 까지 비동기 호출.**

http://61.73.39.135/lonelyplanet/index\_20170323\_2\_jquery\_main\_fade\_bg\_lazyload.html

* 작동 원리

2번 방법 + img 태그에 data-lazy=”” 사용 하여, 이미지 호출

* 장점

**\*\* 성능 개선 전(local 기준) \*\***

* 146 requests 요청
* 총 데이터 : 46 MB
* LOAD 끝나는 시점 : 2.96s (replay시 조금씩 다름)

**\*\* 성능 개선 후(local 기준) \*\***

* 72 requests 요청
* 총 데이터 : 15.4 MB
* LOAD 끝나는 시점 : 1.26s (replay시 조금씩 다름)
* 단점

실서버 관리자에서 등록하고 있는 html까지 변경 해야 하기 때문에 먼저 반영 하는것 보다 이번 리뉴얼 작업시 반영 하는 것이 좋을 듯 싶습니다.

메인 비쥬얼 배너 2번째 노출시 비동기로 보여주기 때문에 흰색이 아주 잠깐! 보이고 있습니다.