주문하기 페이지에서, 주문하기를 누르게 되면,

카페1 페이지에서 즐겨찾기 추가하는 버튼을 누르게 되면, 즐겨찾기가 추가가 되는것인지.

1. 게시판에서, 한 줄이 늘어나게 되며, 매장명 : café:café\_name, 주문내용, : product: product\_name, product: product\_price (product all Price로 대체해야하는 것이 아닌지.)

2. 즐겨찾는 카페 매장 상세 페이지에서, 즐겨찾기 추가를 누르게 되면,

café : cafe\_name, café: café\_addr, 의 정보 뿌려주고, delete 관련 기능 필요.(favoriteDelete)

전체적으로, 페이징 처리 할 준비.

크게. jQuery.

function page() {

$('.paginated').each(

function() {

var pagesu = 10; // 페이지 번호 갯수

var currentPage = 0;

var numPerPage = 30; // 목록의 수

var $table = $(this);

var pagination = $("#kt\_table\_1\_paginate");

// length로 원래 리스트의 전체길이구함

var numRows = $table.find('tbody tr').length;

// Math.ceil를 이용하여 반올림

var numPages = Math.ceil(numRows / numPerPage);

// 리스트가 없으면 종료

if (numPages == 0)

return;

// pager라는 클래스의 div엘리먼트 작성

var $pager = $('<div class="pager"></div>');

var nowp = currentPage;

var endp = nowp + 10;

// 페이지를 클릭하면 다시 셋팅

$table.bind('repaginate', function() {

// 기본적으로 모두 감춘다, 현재페이지+1 곱하기 현재페이지까지 보여준다

$table.find('tbody tr').hide().slice(

currentPage \* numPerPage,

(currentPage + 1) \* numPerPage).show();

$(".pagination").empty();

if (numPages > 1) { // 한페이지 이상이면

if (currentPage < 5 && numPages - currentPage >= 5) { // 현재

// 5p

// 이하이면

nowp = 0; // 1부터

endp = pagesu; // 10까지

} else {

nowp = currentPage - 5; // 6넘어가면 2부터 찍고

endp = nowp + pagesu; // 10까지

pi = 1;

}

if (numPages < endp) { // 10페이지가 안되면

endp = numPages; // 마지막페이지를 갯수 만큼

nowp = numPages - pagesu; // 시작페이지를 갯수 -10

}

if (nowp < 1) { // 시작이 음수 or 0 이면

nowp = 0; // 1페이지부터 시작

}

} else { // 한페이지 이하이면

nowp = 0; // 한번만 페이징 생성

endp = numPages;

}

$('#kt\_table\_1\_paginate').attr('style','display:flex');

$('#kt\_table\_1\_paginate').append(function(){});

// [처음] Tag 지옥에 오신걸 환영합니다 ^^

$('<li class=" paginate\_button page-item previous" >'+

'<a href="#" pointer-events: none; class="page-link">'

+'<i class="fas fa-angle-double-left"></i></a></li>').bind(

'click',

{

newPage : page

},

function(event) {

currentPage = 0;

$table.trigger('repaginate');

$($(".pageNum")[2]).addClass('active')

.siblings().removeClass('active');

}).appendTo(pagination).addClass('clickable');

// [이전]

$('<li class=" paginate\_button page-item previous">'+

'<a href="#" pointer-events: none; class="page-link">'

+'<i class="fas fa-angle-left"></i></a></li>').bind(

'click',

{

newPage : page

},

function(event) {

if (currentPage == 0)

return;

currentPage = currentPage - 1;

$table.trigger('repaginate');

$($(".pageNum")[(currentPage - nowp) + 2])

.addClass('active').siblings()

.removeClass('active');

}).appendTo(pagination).addClass('clickable');

// [1,2,3,4,5,6,7,8]

for (var page = nowp; page < endp; page++) {

$('<li class="paginate\_button page-item">'

+'/li>')

.html('<a href="#" style="pointer-events: none;" class="page-link">' + (page + 1) +'</a>').bind(

'click',

{

newPage : page

},

function(event) {

currentPage = event.data['newPage'];

$table.trigger('repaginate');

$($(".pageNum")[(currentPage - nowp) + 2])

.addClass('active').siblings()

.removeClass('active');

}).appendTo(pagination).addClass('clickable');

}

// [다음]

$('<li class=" paginate\_button page-item previous">'+

'<a href="#" pointer-events: none; class="page-link">'

+'<i class="fas fa-angle-right"></i></a></li>').bind(

'click',

{

newPage : page

},

function(event) {

if (currentPage == numPages - 1)

return;

currentPage = currentPage + 1;

$table.trigger('repaginate');

$($(".pageNum")[(currentPage - nowp) + 2])

.addClass('active').siblings()

.removeClass('active');

}).appendTo(pagination).addClass('clickable');

// [끝]

$('<li class=" paginate\_button page-item previous">'+

'<a href="#" style= pointer-events: none; class="page-link">'

+'<i class="fas fa-angle-double-right"></i></a></li>').bind(

'click',

{

newPage : page

},

function(event) {

currentPage = numPages - 1;

$table.trigger('repaginate');

$($(".pageNum")[endp - nowp + 1]).addClass(

'active').siblings().removeClass(

'active');

}).appendTo(pagination).addClass('clickable');

$($(".pageNum")[2]).addClass('active');

});

$pager.insertAfter($table).find('span.pageNum:first').next()

.next().addClass('active');

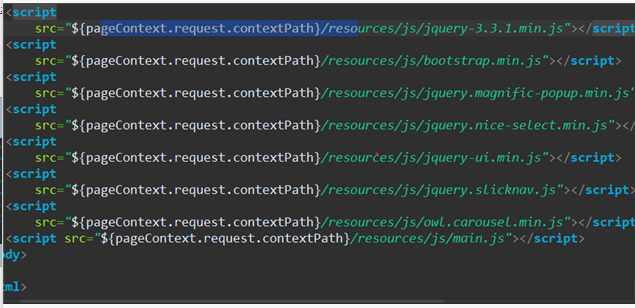
$pager.appendTo(pagination);

$table.trigger('repaginate');

});

}

Javascript,



별점 후기,

0 , 0.5 1 1.5 2 2.5 3 3.5 4 4.5 5 형식으로 잡을건지 .

별이 반개이면, 0.5를 인식하는 코드, 리퀘스트 매핑해서, 컨트롤로 가져온다. 메서드 방식 post인가.(?) dto에서는 café\_star로 빈설정

Comment service에다가. 후기 메서드를 추가해줄것인지 ,새로 만들것인지.

(User1이 준 별점,+앞으로 추가될 별점) / cnum의 개수 (총 별점 계산)

<https://pagination.js.org/>

ㄴ

<https://github.com/superRaytin/paginationjs>

<https://mvnrepository.com/artifact/com.eddmash/pagination>

com.LCY.myapp

controller

CommentController – 별점후기controller(속하게끔)

favo

model

model.dao

model.dto

CommentDto-user\_star

Service

CommentService(interface)

service.impl

CommentServiceImpl

mapper

comment