



AngularJS 정렬(OrderBy) 기능

Table of Contents

- AngularJS 정렬(OrderBy) 기능
 - Filter 및 Data 바인딩 전체 완성 소스
 - Controller.js 소스
 - 정렬 기능 설명 및 적용
 - 정렬 순서 및 바인딩 하는 모델 설정
 - 실제 소스 동작 화면
 - plunker를 위한 실시간 테스트 방법
 - 최종 요약 정리

Filter 및 Data 바인딩 전체 완성 소스

우선 아래와 같은 간단한 예제를 작성해봅니다.

```
1 <!DOCTYPE html>
2 <html lang="ko" ng-app>
3 <head>
4   <meta charset="utf-8">
5   <title>AngularJS orderBy(정렬)</title>
6   <script src="https://ajax.googleapis.com/ajax/libs/angularjs/1.2.10/angular.min.js"></script>
7   <script src="controller.js"></script>
8 </head>
9 <body ng-controller="StudyMemberListCtrl">
10
11   <input type="text" ng-model="searchContent"/>
12
13   <!-- orderBy의 정렬 순서를 지정하는 AngularJS 모델을 설정함. -->
14
15   <select ng-model="orderProperty">
16     <option value="no" selected>번호</option>
17     <option value="name" selected>이름</option>
18     <option value="tpcode" selected>사원코드</option>
19   </select>
20
21   <!-- 당연한 이야기지만, 위의 ng-model 과 아래 filter 의 Model 이름은 같아야 필터링 기능이 정상적으로 동작된다. -->
22
23   <ul>
24
25     <li ng-repeat="member in orders | filter:searchContent | orderBy:orderProperty">
26
27       <!-- orderBy항목이 추가되었다!
28         orderBy는 value 값에 할당된 값을 모델에 반영하여 해당 요소로 정렬된다.
29       -->
30
31       <!-- 필터 기능 + 정렬 기능 동시에 모델에 반영되어 바인딩 된다! -->
32
33       번호:{{member.no}}<br/>
34       이름:{{member.name}}<br/><!-- font 태그 추가 실습 -->
35       사원코드:{{member.tpcode}}<br/>
36       <p>한 마디:{{member.message}}</p>
37
38     </li>
39   </ul>
40 </body>
41 </html>
```



지난 번 예제와 정렬 기능을 수행하기 위한 기능이 어떤 곳인지 한 번 생각해 보자.

자, 다음 Controller.js 파일을 아래와 같이 작성해보자. 서로 다른 키워드로 넉넉하게 입력해 두면 더욱 좋다.

```
1 function StudyMemberListCtrl($scope){
2     $scope.orders = [
3         {"no":0,"name":"이남희","message":"I love Groovy","tpcode":"a"},
4         {"no":1,"name":"한용희","message":"I love Apple","tpcode":"b"},
5         {"no":2,"name":"이준호","message":"I love Ipad","tpcode":"c"},
6         {"no":3,"name":"윤정분","message":"I love Jobs","tpcode":"d"},
7         {"no":4,"name":"김선준","message":"I love AirPlane","tpcode":"e"},
8         {"no":5,"name":"김현하","message":"I love Computer","tpcode":"f"},
9         {"no":6,"name":"손지성","message":"I love Money","tpcode":"g"},
10        {"no":7,"name":"백성진","message":"I love Car","tpcode":"h"},
11        {"no":8,"name":"김은기","message":"I love AngularJs","tpcode":"i"}
12    ];
13
14    //정렬 기본값 지정! (지정하지 않으면 선언된 순서대로 표시!)
15    $scope.orderProperty = "name";
16 }
```



\$scope.orderProperty <- 이 부분을 주목해서 보기 바란다. 오늘 예제의 핵심이다.

정렬 기능 설명 및 적용

```
//정렬 기본값 지정! (지정하지 않으면 선언된 순서대로 표시!)
$scope.orderProperty = "name";
```

컨트롤러에서 정렬 **기준값**을 지정해 주는 부분이다. 지정하지 않으면 정의된 **순서**대로 표시된다.

정렬 순서 및 바인딩 하는 모델 설정

orderProperty를 설정함으로써, 정렬시 **새로운 모델**이 생성되며, li 태그의 orderBy 부분과 매핑되며, 실시간으로 바인딩 되는 것을 확인 할 수 있다.(아래 사진 참조)

```
<ul>

  <li ng-repeat="member in orders | filter:searchContent | orderBy:orderProperty">

    <!-- orderBy항목이 추가되었다!
         orderBy는 value 값에 할당된 값을 모델에 반영하여 해당 요소로 정렬된다.
    -->

    번호:{{member.no}}<br/>
    이름:{{member.name}}<br/><!-- font 태그 추가 실습 -->
    사원 코드:{{member.tpcode}}<br/>
    <p>한 마디:{{member.message}}</p>

  </li>
</ul>
```



선택트 박스의 orderProperty 부분과 li 태그의 orderBy:orderProperty 부분은 **일치**가 되어야 한다는 점을 잊지말자!

실제 소스 동작 화면

아래는 **실행 기본 화면**이다. 정렬은 기본값(번호)으로 되는 것을 알 수 있다.

사원코드 ▼

- 번호:0
이름:이남희
사원코드:a

한 마디:I love Groovy
- 번호:1
이름:한용희
사원코드:b

한 마디:I love Apple
- 번호:2
이름:이준호
사원코드:c

한 마디:I love Ipad
- 번호:3
이름:윤정부
사원코드:d

한 마디:I love Jobs
- 번호:4
이름:김선준
사원코드:e

한 마디:I love AirPlane

name으로 정렬했을 경우

이름

- 번호:4
이름:김선준
사원코드:e

한 마디:I love AirPlane
- 번호:8
이름:김은기
사원코드:i

한 마디:I love AngularJs
- 번호:5
이름:김현하
사원코드:f

한 마디:I love Computer
- 번호:7
이름:백성진
사원코드:h

한 마디:I love Car
- 번호:6
이름:손지성
사원코드:g

한 마디:I love Money

사원코드 및 성이 김씨인 사람으로 검색했을 경우

김

사원코드

- 번호:4
이름:김선준
사원코드:e

한 마디:I love AirPlane
- 번호:5
이름:김현하
사원코드:f

한 마디:I love Computer
- 번호:8
이름:김은기
사원코드:i

한 마디:I love AngularJs

위 결과화면에서 확인할 수 있듯이, 실시간으로 **필터 + 정렬 + 바인딩** 통해 모델을 생성한 후 컨트롤러에 의하여 리스트가 동기화 되어 화면에 적용되는 되는 것을 확인할 수 있다.

plunker를 위한 실시간 테스트 방법

<http://plnkr.co/edit/dVYmPFuQM0PdQvZ5g7nK>

마찬가지로 위 주소를 클릭하면 오늘 예제 소스를 작성한 후 실시간으로 **변경, 테스트** 할 수 있다. **가능하면 한 번 해보도록 하자.**



HTML 화면 또는 js파일 수정 후 위의 **RUN** 버튼을 클릭하면 된다.



Plunker는 자동 저장 및 실시간 실행, jquery, angularjs 및 기타 js **자동완성** 도구들을 제공한다.

최종 요약 정리

자, 다시 한번 오늘 예제 정렬 동작방식을 정리해 보도록 하자.

1. 셀렉트 정렬 기준을 선택하면 설정된 값으로 **새로운 모델을 만들어** 낸다.
2. 변화된 모델은 즉시, **컨트롤러에 의해 감지되어 연결된 뷰에 즉시 반영된다.**
3. 반영됨과 동시에 DOM을 **자동으로 업데이트** 시키므로, 결과 값을 바로 확인 할 수 있다.

링크 목록

- <http://plnkr.co/edit/dVYmPFuQM0PdQvZ5g7nK> - <http://plnkr.co/edit/dVYmPFuQM0PdQvZ5g7nK>