## 练习：排序显示（jquery 与 angular）

|  |
| --- |
| 实现类似下列结构，功能为每次点击头部时，下面指标排序显示，再次点击时倒序显示。  IMG_256          分析：刚看到这个题目的时候，觉得功能挺简单，但是可能代码量（用jquery）不会很少，但没多想，还是说了下思路，然后还多嘴写了个toggle（）方法，回来寻思着试下一下吧，结果越写越郁闷，首先，toggle（）方法在1.9+的版本中已经被干掉了，不懂为什么呢，我为了实现点一次再点一次的功能，就只能自己定义个点击计数器，然后根据偶数奇数正序倒序。然后我为了每次改变每一行的不同位置，根据排序后的索引动态生成html然后插入进去，也挺郁闷的。但是还是硬着头皮写了下去，于是写完了也不想做优化了，因为想到了angularjs，实习时候学的东西不能就那样拜拜啊，于是就写了个angularjs版本，于是世界明亮了。           jQuery版：（有点笨的方法，如果你有好的方法，欢迎留言喔，教教我喔，共同学习~~ ）  <!DOCTYPE html>  <html lang="zh">  <head>  <meta charset=utf-8>  <title>demo</title>  <style type="text/css">    </style>  <script type="text/javascript" src="js/jquery-1.11.1.js"></script>  </head>  <body>  <table border="2px" cellspacing="0" cellpadding="0">  <tr class="header">  <th><span class="name">姓名</span></th>  <th><span class="friends">朋友数</span></th>  <th><span class="photos">照片数</span></th>  <th><span class="weibo">微博条数</span></th>  </tr>  <tr class="row1">  <td>大额</td>  <td>38</td>  <td>45</td>  <td>123</td>  </tr>  <tr class="row2">  <td>Pan</td>  <td>32</td>  <td>86</td>  <td>39</td>  </tr>  <tr class="row3">  <td>搞笑排行</td>  <td>63</td>  <td>76</td>  <td>98</td>  </tr>  <tr class="row4">  <td>News</td>  <td>43</td>  <td>49</td>  <td>356</td>  </tr>  </table>  <script>  var obj = {  1 : {  'pname' : '大额',  'friends': 38,  'photos': 45,  'weibo': 123  },  2 : {  'pname' : 'Pan',  'friends': 32,  'photos': 86,  'weibo' : 39  },  3: {  'pname' : '搞笑排行',  'friends': 63,  'photos': 76,  'weibo' : 98  },  4 : {  'pname' : 'News',  'friends': 43,  'photos': 49,  'weibo' : 356  }  };  var names = [],  html = [],  friend = [],  photo = [],  weibos = [];    for(var i=0;i<4;i++){  html[i+1] = '<td>'+obj[i+1].pname+'</td>'+  '<td>'+obj[i+1].friends+'</td>'+  '<td>'+obj[i+1].photos+'</td>'+  '<td>'+obj[i+1].weibo+'</td>';  }  function compare(num1,num2){  return num1-num2;  }  //jq1.8以后不支持toggle()事件了，故设一个click点击计数器  var nameCtime = 0;  $('.name').click(function(){  names = [];  for(var i=0;i<4;i++){  names.push(obj[i+1].pname);  }  if(nameCtime++ % 2 == 0){  //首字母字符编码排序  names.sort();  for(var i=0;i<names.length;i++){  var index = names.indexOf(obj[i+1].pname);  $('tr:eq('+(index+1)+')').html(html[i+1]);  }  }else{  names.sort();  names.reverse();  for(var i=0;i<names.length;i++){  var index = names.indexOf(obj[i+1].pname);  $('tr:eq('+(index+1)+')').html(html[i+1]);  }  }    });  var friendCtime = 0;  $('.friends').click(function(){  friend = [];  for(var i=0;i<4;i++){  friend.push(obj[i+1].friends);  }  if(friendCtime++ % 2 ==0){  friend.sort(compare);  for(var i=0;i<friend.length;i++){  var index = friend.indexOf(obj[i+1].friends);  $('tr:eq('+(index+1)+')').html(html[i+1]);  }  }else{  friend.sort(compare);  friend.reverse();  for(var i=0;i<friend.length;i++){  var index = friend.indexOf(obj[i+1].friends);  $('tr:eq('+(index+1)+')').html(html[i+1]);  }  }    });  var photoCtime = 0;  $('.photos').click(function(){  photo = [];  for(var i=0;i<4;i++){  photo.push(obj[i+1].photos);  }  if(photoCtime++ %2 ==0){  photo.sort(compare);  for(var i=0;i<photo.length;i++){  var index = photo.indexOf(obj[i+1].photos);  $('tr:eq('+(index+1)+')').html(html[i+1]);  }  }else{  photo.sort(compare);  photo.reverse();  for(var i=0;i<photo.length;i++){  var index = photo.indexOf(obj[i+1].photos);  $('tr:eq('+(index+1)+')').html(html[i+1]);  }  }    });  var weiboCtime = 0;  $('.weibo').click(function(){  weibos = [];  for(var i=0;i<4;i++){  weibos.push(obj[i+1].weibo);  }  if(weiboCtime++ %2 != 0){  weibos.sort(compare);  for(var i=0;i<weibos.length;i++){  var index = weibos.indexOf(obj[i+1].weibo);  $('tr:eq('+(index+1)+')').html(html[i+1]);  }  }else{  weibos.sort(compare);  weibos.reverse();  for(var i=0;i<weibos.length;i++){  var index = weibos.indexOf(obj[i+1].weibo);  $('tr:eq('+(index+1)+')').html(html[i+1]);  }  }    });    </script>  </body>  </html>   1. 其实写的过程中也发现了，自己在代码优化方面要做的事情还有很多很多，需要不断努力，后面寻思着还是有时间学着优化下上面的代码，真是感觉不太酷。        AngularJS版本：  <!DOCTYPE html>  <html ng-app>  <head>  <meta charset=utf-8>  <title>demo</title>  <style type="text/css">    </style>  <script src="js/angular.min.js"></script>  </head>  <body ng-controller="controller">  <table border="2px" cellspacing="0" cellpadding="0">  <tr class="header">  <th><span ng-click = "predicate='pname';reverse=true">姓名</span></th>  <th><span ng-click = "predicate='friends';reverse=!reverse">朋友数</span></th>  <th><span ng-click = "predicate='photos';reverse=!reverse">照片数</span></th>  <th><span ng-click = "predicate='weibo';reverse=!reverse">微博条数</span></th>  </tr>  <tr ng-repeat="o in obj|orderBy:predicate:reverse">  <td>{{o.pname}}</td>  <td>{{o.friends}}</td>  <td>{{o.photos}}</td>  <td>{{o.weibo}}</td>  </tr>  </table>  <script>  function controller($scope){  $scope.obj = [  {  'pname' : '大额',  'friends': 38,  'photos': 45,  'weibo': 123  },  {  'pname' : 'Pan',  'friends': 32,  'photos': 86,  'weibo' : 39  },  {  'pname' : '搞笑排行',  'friends': 63,  'photos': 76,  'weibo' : 98  },  {  'pname' : 'News',  'friends': 43,  'photos': 49,  'weibo' : 356  }  ];  }    </script>  </body>  </html>    通过查阅手册，文档，功能很快就实现了，而且基本没怎么写控制代码，以前实习的时候学着用的时候就觉得Angular在数据等各个方面都很强大啊，这也让我这个angular的初级菜鸟坚定了持续学习的信息哇。 |
| <table border="2px" cellspacing="0" cellpadding="0">      <tr class="header">          <th><span class="name">姓名</span></th>          <th><span class="friends">朋友数</span></th>          <th><span class="photos">照片数</span></th>          <th><span class="weibo">微博条数</span></th>      </tr>      <tr class="row">          <td>大额</td>          <td>38</td>          <td>45</td>          <td>123</td>      </tr>      <tr class="row">          <td>Pan</td>          <td>32</td>          <td>86</td>          <td>39</td>      </tr>      <tr class="row">          <td>搞笑排行</td>          <td>63</td>          <td>76</td>          <td>98</td>      </tr>      <tr class="row">          <td>News</td>          <td>43</td>          <td>49</td>          <td>356</td>      </tr>  </table>  <script>      $(function () {          var $table = $("table"),                  arr = $table.find(".row").toArray(),                  index, at, bt;            $table.find("th").click(function () {              index = $(this).index();              arr.sort(function (a, b) {                  at = $($(a).find("td").get(index)).text();                  bt = $($(b).find("td").get(index)).text();                  return isNaN(at) && isNaN(bt) ? at > bt : parseInt(at) > parseInt(bt);              });              if (arr.type && arr.type[0] == index) {                  if (arr.type[1] % 2 == 0) {                      arr.reverse();                  }                  arr.type[1]++;              } else {                  arr.type = [index, 0];              }              $.each(arr, function (i, n) {                  $table.append($(n));              });          });      });  </script> |