



## Директивы в AngularJS

Последнее обновление: 1.11.2015



Изучай английский по Skype!

Как вы видели ранее, для добавления элементам html дополнительной функциональности AngularJS использует директивы. Например, в предыдущих разделах мы уже познакомились с рядом директив.

Как вы уже видели, директивы объявляются в виде атрибутов, например, `<div ng-bind>`. Однако уместен еще один вариант использования: `<div class="ng-bind">`

Некоторые директивы могут образовывать самостоятельные теги: `<ng-form>`  
`</ng-form>`

В AngularJS определено более 50 встроенных директив. Они охватывают широчайший спектр функциональностей, которые востребованы во многих веб-приложениях: привязка данных, валидация форм, генерация шаблонов, обработка событий, управление html-элементами. Полный список директив вы можете посмотреть в документации. Сейчас же рассмотрим некоторые из них.

### Директива ng-repeat

Директива **ng-repeat** позволяет производить обход массивов в стиле цикла `for-each` и генерировать для них элементы html. Например, при рассмотрении контроллеров мы создали контроллер `phoneController`. Изменим его так, чтобы он передавал в представление массив:

```
var myApp=angular.module('myApp');
myApp.controller('phoneController', function($scope) {

    $scope.phones = [{
        name: 'Nokia Lumia 630',
        year: 2014,
        price: 200,
```

```

        company: {
            name: 'Nokia',
            country: 'Финляндия'
        }
    }, {
        name: 'Samsung Galaxy S 4',
        year: 2014,
        price: 400,
        company: {
            name: 'Samsung',
            country: 'Республика Корея'
        }
    }, {
        name: 'Mi 5',
        year: 2015,
        price: 300,
        company: {
            name: 'Xiaomi',
            country: 'Китай'
        }
    }
  ]
});

```

И чтобы вывести все элементы массива, применим директиву **ng-repeat**:

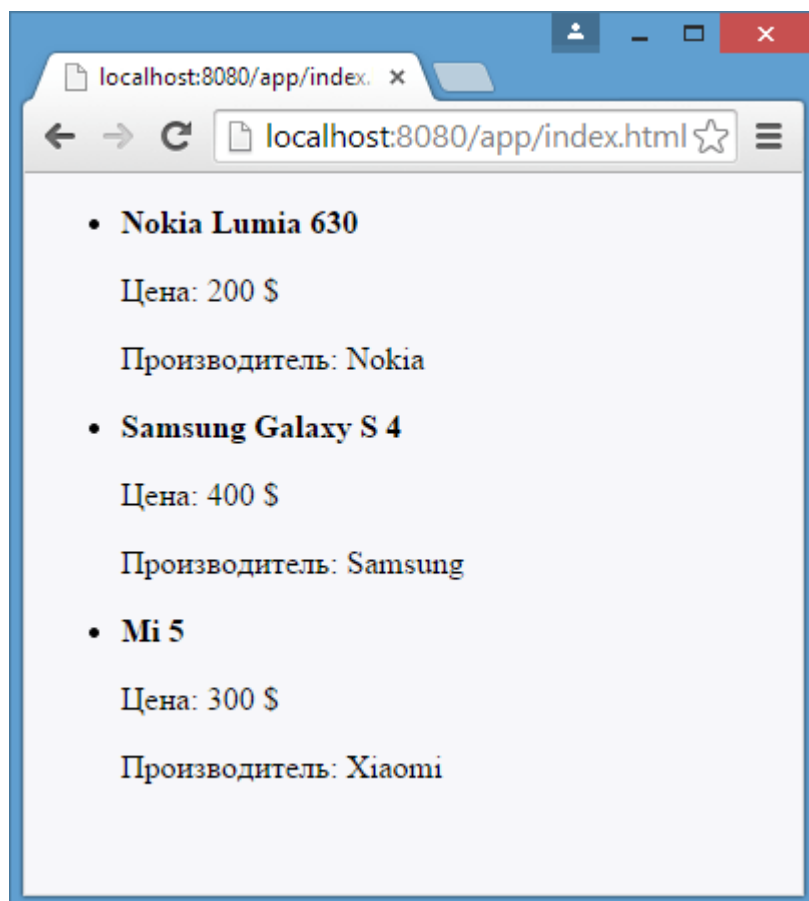
```

<!doctype html>
<html ng-app="myApp">
<head>
<meta charset="utf-8">
<link rel="stylesheet" href="css/mystyles.css" />
</head>
<body>

<div ng-controller="phoneController">
<ul>
<li ng-repeat="phone in phones">
<b>{{phone.name}}</b>
<p>Цена: {{phone.price}} $</p>
<p>Производитель: {{phone.company.name}}</p>
</li>
</ul>
</div>
<script src="js/lib/angular.min.js"></script>
<script type="text/javascript">
angular.module('myApp', []);
</script>
<script src="js/controllers/phonesController.js"></script>
</body>
</html>

```

В данном случае с помощью выражения `<li ng-repeat="phone in phones">` у нас будет создаваться для каждого объекта в наборе `phones` элемент `li`, который будет содержать определенную информацию.



## Встроенные переменные

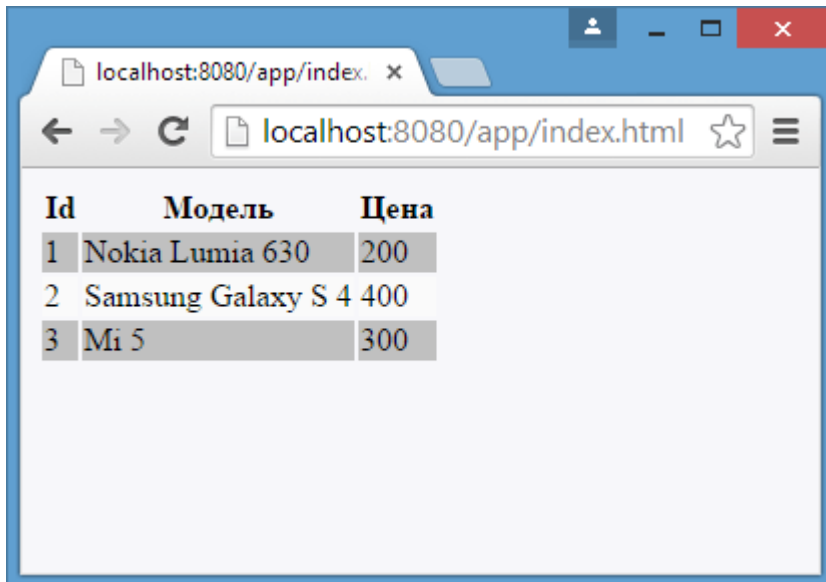
При обходе массивов мы можем использовать ряд встроенных переменных AngularJS:

- **\$index**: возвращает индекс элемента в массиве
- **\$first**: возвращает true, если элемент является первым в массиве
- **\$last**: возвращает true, если элемент является последним в массиве
- **\$middle**: возвращает true, если элемент не является ни первым, ни последним в массиве
- **\$even**: возвращает true, если элемент является четным
- **\$odd**: возвращает true, если элемент является нечетным

Используем пару переменных:

```
<style>
.odd { background-color: #fafafa;}
.even { background-color: silver}
</style>
<div ng-controller="phoneController">
<table class="table">
<thead>
<tr>
<th>Id</th>
```

```
<th>Модель</th>
<th>Цена</th>
</tr>
</thead>
<tr ng-repeat="phone in phones" ng-class="$odd ? 'odd' : 'even'">
<td>{{$index + 1}}</td>
<td>{{phone.name}}</td>
<td>{{phone.price}}</td>
</tr>
</table>
</div>
```

A screenshot of a web browser window showing a table of phones. The browser's address bar displays 'localhost:8080/app/index.html'. The table has three columns: 'Id', 'Модель', and 'Цена'. It contains three rows of data, alternating between light and dark gray backgrounds. The data is as follows:

Id	Модель	Цена
1	Nokia Lumia 630	200
2	Samsung Galaxy S 4	400
3	Mi 5	300

[Загрузить проект](#)

## Обход внутренних массивов

Возможна ситуация, когда объект в массиве, в свою очередь, сам содержит массив, и этот массив тоже надо обойти. В этом случае можно использовать вложенные директивы ng-repeat. Например, весь объект выглядит так:

```
$scope.phones = [{
  name: 'Nokia',
  models:[{
    name: "Nokia Lumia 530",
    price: 150
  },{
    name: "Nokia X2",
    price: 200
  }]
},{
  name: 'Samsung',
  models:[{
    name: "Samsung Galaxy S5",
    price: 500
  },{
    name: "Samsung Galaxy Alpha",
    price: 400
  }]
}]
```

```
});
```

Для вывода вложенного массива можно использовать следующий код:

```
<div ng-controller="phoneController">
  <ul>
    <li ng-repeat="company in phones">
      <p>{{company.name}}</p>
      <ul ng-repeat="model in company.models">
        <b>{{model.name}}</b>
        <p>Цена: {{model.price}} $</p>
      </ul>
    </li>
  </ul>
</div>
```

[Назад](#) [Содержание](#) [Вперед](#)



G+1

2 Комментариев metanit.com

1 Войти

♥ Рекомендовать    ➦ Поделиться

Лучшее в начале



Присоединиться к обсуждению...

**Борис Илларионов** • 18 дней назад

А как будет выглядеть правильная рекурсивная функция здесь? Например мы не знаем, где есть вложенные модели, а где их нет.. И чтобы не получить пустой текст? Я говорю про что-то вроде: if (value in object.childModel != undefined) -> спускаемся вниз, иначе печатаем **object.name**, object.description,.. и т.д.

^ | v • Ответить • Поделиться ›



**Михаил** • 8 месяцев назад

А в примере со списком телефонов, можно ли узнать, к какому объекту привязан элемент <li> списка, т. е., если говорить языком WPF, каков его "DataContext"?

^ | v • Ответить • Поделиться ›

✉ Подписаться    D Добавить Disqus на свой сайт    Добавить Disqus    Добавить  
🔒 Конфиденциальность



[r](#) | [Google+](#) | [Канал сайта на youtube](#) | [Помощь сайту](#)

м: metanit22@mail.ru

t.com, 2012-2016. Все права защищены.