

# 《AngularJS权威教程》

## 图书基本信息

书名：《AngularJS权威教程》

13位ISBN编号：9787115366470

出版时间：2014-8

作者：[美] Ari Lerner

页数：476

译者：赵望野,徐 飞,何鹏飞

版权说明：本站所提供下载的PDF图书仅提供预览和简介以及在线试读，请支持正版图书。

更多资源请访问：[www.tushu111.com](http://www.tushu111.com)

# 《AngularJS权威教程》

## 内容概要

本书是资深全栈工程师的代表性著作，由拥有丰富经验的国内AngularJS技术专家执笔翻译，通俗易懂、全面深入，是学习AngularJS不可错过的经典之作。无论是出于工作需要，还是好奇心的驱使，如果你想彻底理解AngularJS，本书都会让你感到满意。

本书将涵盖AngularJS的如下概念。

双向数据绑定

依赖注入

作用域

控制器

路由

客户端模板

服务

通过XHR实现动态内容

测试

过滤器

定制表单验证

深度测试

定制指令

专业工具

对IE的支持

# 《AngularJS权威教程》

## 作者简介

作者简介：

Ari Lerner

是一位全栈工程师，拥有多年AngularJS经验，自办并运营AngularJS电子报ng-newsletter.com，在著名硅谷工程师培训学校Hack Reactor担任AngularJS讲师。他的工作涉及软件开发的各个层次，包括基础设施开发、前端应用开发和性能优化。他目前住在旧金山一个阳光明媚的地方，还是FullStack.io创始人。

译者简介：

赵望野

前端工程师，前端基础技术组leader，曾经负责豌豆荚2.0的前端架构设计和主要开发工作，目前负责Front-end Technical Infrastructure的建设，在工作中有丰富的AngularJS使用经验。新浪微博@赵望野。

徐飞

2005年至今一直从事企业应用前端架构，对富因特网应用有较深刻的认识，致力于前端的高效开发，研究过Backbone和AngularJS的源码，翻译过讲解AngularJS基本原理的文章，对脏数据检测和基于存取器两种监听方式的差异有深刻认识。

何鹏飞

网名basecss，目前就职于腾讯CDC，任前端工程师。喜欢阅读，喜欢前端技术，崇尚开源。工作之余翻译过Grunt和Lesscss相关文档，同时也是Lesscss中文社区贡献者。

## 书籍目录

第1章 初识AngularJS	1
1.1 浏览器如何获取网页	1
1.2 浏览器是什么	2
1.3 AngularJS是什么	2
1.3.1 AngularJS有什么不同	3
1.3.2 许可	3
第2章 数据绑定和第一个AngularJS Web应用	4
2.1 AngularJS中的数据绑定	5
2.2 简单的数据绑定	6
2.3 数据绑定的最佳实践	8
第3章 模块	10
3.1 参数	11
3.1.1 name (字符串)	11
3.1.2 requires (字符串数组)	11
第4章 作用域	12
4.1 视图和\$scope的世界	12
4.2 就是HTML而已	13
4.3 作用域能做什么	14
4.4 \$scope的生命周期	14
4.4.1 创建	15
4.4.2 链接	15
4.4.3 更新	15
4.4.4 销毁	15
4.5 指令和作用域	15
第5章 控制器	16
5.1 控制器嵌套 (作用域包含作用域)	18
第6章 表达式	20
6.1 解析AngularJS表达式	20
6.2 插值字符串	21
第7章 过滤器	24
7.1 自定义过滤器	29
7.2 表单验证	29
第8章 指令简介	43
8.1 指令：自定义HTML元素和属性	44
8.2 向指令中传递数据	50
第9章 内置指令	56
9.1 基础ng属性指令	56
9.1.1 布尔属性	56
9.1.2 类布尔属性	58
9.2 在指令中使用子作用域	59
第10章 指令详解	72
10.1 指令定义	72
10.1.1 restrict (字符串)	74
10.1.2 优先级 (数值型)	75
10.1.3 terminal (布尔型)	75
10.1.4 template (字符串或函数)	76
10.1.5 templateUrl (字符串或函数)	76

10.1.6	replace (布尔型)	77
10.2	指令作用域	77
10.2.1	scope参数 (布尔型或对象)	78
10.2.2	隔离作用域	80
10.3	绑定策略	81
10.3.1	transclude	82
10.3.2	controller (字符串或函数)	84
10.3.3	controllerAs (字符串)	86
10.3.4	require (字符串或数组)	86
10.4	AngularJS的生命周期	87
10.4.1	编译阶段	87
10.4.2	compile (对象或函数)	88
10.4.3	链接	89
10.5	ngModel	90
10.5.1	自定义渲染	92
10.5.2	属性	92
10.6	自定义验证	93
第11章	AngularJS模块加载	95
11.1	配置	95
11.2	运行块	96
第12章	多重视图和路由	98
12.1	安装	98
12.2	布局模板	99
12.3	路由	99
12.4	\$location服务	103
12.5	路由模式	105
12.5.1	HTML5模式	105
12.5.2	路由事件	106
12.5.3	关于搜索引擎索引	107
12.6	更多关于路由的内容	107
12.6.1	页面重新加载	107
12.6.2	异步的地址变化	107
第13章	依赖注入	108
13.1	推断式注入声明	109
13.2	显式注入声明	110
13.3	行内注入声明	110
13.4	\$injector API	111
13.4.1	annotate()	111
13.4.2	get()	111
13.4.3	has()	111
13.4.4	instantiate()	112
13.4.5	invoke()	112
13.5	ngMin	112
13.5.1	安装	113
13.5.2	使用ngMin	113
13.5.3	工作原理	113
第14章	服务	114
14.1	注册一个服务	114
14.2	使用服务	116

14.3	创建服务时的设置项	118
14.3.1	factory()	119
14.3.2	service()	119
14.3.3	provider()	120
14.3.4	constant()	122
14.3.5	value()	122
14.3.6	何时使用value()和constant()	123
14.3.7	decorator()	123
第15章	同外界通信：XHR和服务器通信	125
15.1	使用\$http	125
15.2	设置对象	128
15.3	响应对象	130
15.4	缓存HTTP请求	131
15.5	拦截器	132
15.6	设置\$httpProvider	133
15.7	使用\$resource	134
15.8	安装	134
15.9	应用\$resource	135
15.9.1	基于HTTP GET方法	135
15.9.2	基于非HTTP GET类型的方法	136
15.9.3	\$resource实例	137
15.9.4	\$resource实例是异步的	138
15.9.5	附加属性	138
15.10	自定义\$resource方法	138
15.11	\$resource设置对象	139
15.12	\$resource服务	141
15.13	使用Restangular	142
15.14	Restangular简介	142
15.15	安装Restangular	143
15.16	Restangular对象简介	144
15.17	使用Restangular	145
15.17.1	我的HTTP方法们怎么办	146
15.17.2	自定义查询参数和头	147
15.18	设置Restangular	147
第16章	XHR实践	153
16.1	跨域和同源策略	153
16.2	JSONP	153
16.3	使用CORS	154
16.3.1	设置	154
16.3.2	服务器端CORS支持	155
16.3.3	简单请求	155
16.3.4	非简单请求	156
16.4	服务器端代理	157
16.5	使用JSON	157
16.6	使用XML	158
16.7	使用AngularJS进行身份验证	159
16.7.1	服务器端需求	159
16.7.2	客户端身份验证	160
16.8	和MongoDB通信	165

第17章	promise	168
17.1	什么是promise	168
17.2	为什么使用promise	169
17.3	Angular中的promise	170
17.4	链式请求	173
17.4.1	all(promises)	174
17.4.2	defer()	174
17.4.3	reject(reason)	174
17.4.4	when(value)	174
第18章	服务器通信	175
18.1	自定义服务器端	175
18.2	安装NodeJS	175
18.3	安装Express	176
18.4	调用API	178
18.5	使用Amazon AWS的无服务器应用	181
18.5.1	DynamoDB	181
18.5.2	简单通知服务 ( SNS )	181
18.5.3	简单队列服务 ( SQS , Simple Queue Service )	182
18.5.4	简单存储服务 ( S3 )	182
18.5.5	安全令牌服务 ( STS )	182
18.6	AWSJS + Angular	182
18.7	开始	182
18.8	介绍	184
18.9	安装	184
18.10	运行	185
18.11	用户认证/鉴权	186
18.12	UserService	190
18.13	迁移到AWS上	191
18.14	AWSService	194
18.15	在Dynamo上开始	196
18.16	\$cacheFactory	196
18.17	保存currentUser	197
18.18	上传到S3	199
18.19	处理文件上传	201
18.20	查询Dynamo	203
18.21	在HTML显示列表	204
18.22	出售我们的作品	205
18.23	使用Stripe	206
18.24	使用Firebase的无服务器应用	209
18.25	使用Firebase和Angular的三方数据绑定	210
18.26	从AngularFire开始	211
18.26.1	注册并创建一个Firebase	211
18.26.2	包含Firebase和AngularFire库	212
18.26.3	把Firebase作为依赖项添加	212
18.26.4	绑定模型到Firebase URL	212
18.26.5	数据同步	213
18.27	在AngularFire中排序	214
18.28	Firebase事件	215
18.29	显式同步	215

18.30	用AngularFire进行认证	216
18.31	认证事件	217
18.31.1	\$logout()	218
18.31.2	\$createUser()	218
18.32	使用Firebase托管部署你的Angular应用	218
18.32.1	安装Firebase工具	218
18.32.2	部署你的Web站点	219
18.33	除了AngularFire之外	219
第19章	测试	220
19.1	为什么要做测试	220
19.2	测试策略	220
19.3	开始测试	220
19.4	AngularJS测试的类型	221
19.4.1	单元测试	221
19.4.2	端到端测试	222
19.5	开始	222
19.6	初始化Karma配置文件	223
19.7	配置选项	226
19.8	使用RequireJS	231
19.9	Jasmine	233
19.9.1	细则套件	233
19.9.2	定义一个细则	233
19.10	预期	234
19.10.1	内置的匹配器	234
19.10.2	安装和卸载	237
19.11	端到端的介绍	238
19.11.1	选项输入	244
19.11.2	重复循环元素	244
19.12	模拟和测试帮助函数	245
19.13	模拟\$httpBackend	246
19.14	测试一个应用	251
19.14.1	测试路由	252
19.14.2	测试页面内容	255
19.14.3	测试控制器	257
19.14.4	测试服务和工厂	259
19.14.5	测试过滤器	263
19.14.6	测试模板	264
19.14.7	测试指令	266
19.15	测试事件	269
19.16	对Angular的持续集成	270
19.17	Protractor	270
19.18	配置	272
19.19	配置选项	273
19.20	编写测试	275
19.21	测试实践	278
19.21.1	我们的应用	278
19.21.2	测试的策略	279
19.22	建立我们的第一个测试	279
19.23	测试输入框	281



19.23.1	测试列表	282
19.23.2	测试路由	284
19.24	页面对象	285
第20章	事件	287
20.1	什么是事件	287
20.2	事件传播	287
20.2.1	使用\$emit来冒泡事件	288
20.2.2	使用\$broadcast向下传递事件	288
20.3	事件监听	289
20.4	事件对象	289
20.5	事件相关的核心服务	290
20.5.1	核心系统的\$emitted事件	290
20.5.2	核心系统的\$broadcast事件	290
第21章	架构	292
21.1	目录结构	292
21.2	模块	293
21.3	控制器	294
21.4	指令	296
21.5	测试	296
第22章	Angular动画	297
22.1	安装	297
22.2	它是如何运作的	297
22.3	使用CSS3过渡	298
22.4	使用CSS3动画	300
22.5	交错CSS过渡/动画	301
22.5.1	交错CSS过渡	301
22.5.2	交错CSS动画	302
22.5.3	什么指令支持交错动画	302
22.6	使用JavaScript动画	302
22.7	微调动画	303
22.8	DOM回调事件	304
22.9	内置指令的动画	304
22.9.1	ngRepeat动画	304
22.9.2	ngView动画	306
22.9.3	ngInclude动画	308
22.9.4	ngSwitch动画	310
22.9.5	ngIf动画	312
22.9.6	ngClass动画	314
22.9.7	ngShow/ngHide动画	316
22.10	创建自定义动画	318
22.10.1	addClass()	319
22.10.2	removeClass()	320
22.10.3	enter()	321
22.10.4	leave()	322
22.10.5	move()	323
22.11	与第三方库集成	324
22.11.1	Animate.css	324
22.11.2	TweenMax/TweenLite	324
第23章	digest循环和\$apply	326

23.1	\$watch列表	326
23.2	脏值检查	327
23.3	\$watch	328
23.4	\$watchCollection	330
23.5	页面中的\$digest循环	330
23.6	\$evalAsync列表	331
23.7	\$apply	332
23.8	何时使用\$apply	332
第24章 揭秘Angular		334
24.1	视图的工作原理	335
24.1.1	编译阶段	335
24.1.2	运行时	336
第25章 AngularJS精华扩展		337
25.1	AngularUI	337
25.2	安装	337
25.3	ui-router	337
25.3.1	安装	337
25.3.2	事件	342
25.3.3	\$stateParams	343
25.3.4	\$urlRouterProvider	344
25.3.5	创建一个导航程序	345
25.4	ui-utils	346
25.4.1	安装	347
25.4.2	mask	347
25.4.3	ui-event	347
25.4.4	ui-format	348
第26章 移动应用		350
26.1	响应式Web应用	350
26.2	交互	350
26.2.1	安装	350
26.2.2	ngTouch	351
26.2.3	\$swipe服务	352
26.2.4	angular-gestures和多点触控手势	353
26.2.5	安装angular-gestures	354
26.2.6	使用angular-gestures	354
26.3	Cordova中的原生应用程序	355
26.4	Cordova入门	356
26.4.1	Cordova开发流程	359
26.4.2	平台	359
26.4.3	插件	359
26.4.4	构建	360
26.4.5	模拟和运行	360
26.4.6	开发阶段	360
26.4.7	Angular中的Cordova服务	361
26.5	引入Angular	362
26.6	使用Yeoman构建	363
26.6.1	修改Yeoman以便使用Cordova	364
26.6.2	装配Yeoman构建	365
26.6.3	构建移动部分	365

26.6.4	处理引导程序	367
第27章	本地化	369
27.1	angular-translate	369
27.2	安装	369
27.3	教你的应用一种新语言	370
27.4	多语言支持	371
27.5	运行时切换语言	372
27.6	加载语言	373
27.7	angular-gettext	374
27.8	安装	374
27.9	用法	375
27.10	字符串提取	375
27.11	翻译字符串	377
27.12	编译新语言	378
27.13	改变语言	379
第28章	缓存	381
28.1	什么是缓存	381
28.2	Angular中的缓存	381
28.2.1	\$cacheFactory简介	381
28.2.2	缓存对象	382
28.3	\$http中的缓存	382
28.3.1	默认的\$http缓存	382
28.3.2	自定义缓存	383
28.4	为\$http设置默认缓存	384
第29章	安全性	385
29.1	严格的上下文转义：\$sce服务	385
29.2	URL白名单	387
29.3	URL黑名单	388
29.4	\$sce API	388
29.4.1	getTrusted	388
29.4.2	parse	389
29.4.3	trustAs	389
29.4.4	isEnabled	390
29.5	配置\$sce	390
29.6	可信赖的上下文类型	390
第30章	AngularJS和IE浏览器	391
30.1	Ajax缓存	393
30.2	AngularJS中的SEO	393
30.3	使Angular应用可被索引	393
30.4	服务端	393
30.4.1	hashbang语法	394
30.4.2	HTML5路由模式	394
30.5	服务端处理SEO的选项	394
30.5.1	使用Node/Express中间件	395
30.5.2	使用Apache重写URL	395
30.5.3	使用Ngnix代理URL	396
30.6	获取快照	396
30.7	使用Zombie.js获取HTML快照	397
30.8	使用grunt-html-snapshot	398

30.9	Prerender.io	399
30.10	<noscript>方法	400
第31章	构建Angular Chrome应用	401
31.1	了解Chrome应用	401
31.1.1	manifest.json	401
31.1.2	背景脚本	401
31.1.3	视图	401
31.2	构建你的Chrome应用	402
31.3	搭建框架	402
31.4	manifest.json	403
31.5	tab.html	404
31.6	在Chrome中加载应用	405
31.7	主模块	406
31.8	构建主页	406
31.9	使用Wunderground的天气API	408
31.10	设置界面	411
31.11	实现用户服务	413
31.12	城市自动填充/自动完成	415
31.13	添加时区支持	418
第32章	优化Angular应用	421
32.1	优化什么	421
32.2	优化\$digest循环	421
32.3	优化ng-repeat	423
32.4	优化\$digest调用	423
32.5	优化\$watch函数	424
32.5.1	bindonce	425
32.5.2	\$watch函数的自动优化	427
32.6	优化过滤器	427
32.6.1	不变的数据	427
32.6.2	过滤后的数据	427
32.7	页面加载优化技巧	428
32.7.1	压缩	429
32.7.2	利用\$templateCache	429
第33章	调试AngularJS	430
33.1	从DOM中调试	430
33.1.1	scope()	431
33.1.2	controller()	431
33.1.3	injector()	431
33.1.4	inheritedData()	431
33.2	调试器	431
33.3	Angular Batarang	432
33.3.1	安装Batarang	432
33.3.2	检查模型	433
33.3.3	检查性能	433
33.3.4	检查依赖图表	434
33.3.5	可视化应用	434
第34章	下一步	435
34.1	jqLite和jQuery	435
34.2	了解基本工具	436

34.3	Grunt	436	
34.4	grunt-angular-templates		439
34.4.1	安装	439	
34.4.2	用法	440	
34.4.3	可用选项	440	
34.4.4	用法	442	
34.5	Lineman	443	
34.6	Bower	445	
34.6.1	安装	445	
34.6.2	Bower简介	445	
34.6.3	配置Bower	446	
34.6.4	搜索程序包	447	
34.6.5	安装程序包	447	
34.6.6	使用程序包	447	
34.6.7	移除程序包	448	
34.7	Yeoman	448	
34.7.1	安装	448	
34.7.2	用法	449	
34.7.3	创建路由	451	
34.7.4	创建控制器	451	
34.7.5	创建自定义指令	451	
34.7.6	创建自定义过滤器	451	
34.7.7	创建视图	451	
34.7.8	创建服务	452	
34.7.9	创建装饰器	452	
34.8	配置Angular生成器	452	
34.8.1	CoffeeScript	452	
34.8.2	安全压缩	452	
34.8.3	跳过索引	452	
34.9	测试应用	452	
34.10	打包应用	453	
34.11	打包模板	453	
第35章	总结	456	

## 精彩短评

- 1、中文版文档。
- 2、很不错！
- 3、目前为止最好的一本入门书，内容很殷实，除了把文档翻译了一遍以外还有很多自己的东西。
- 4、学习中，快看完了，这本书讲得挺深
- 5、原书的设计不好，介绍内容时跳来跳去，即使重复也不应该这样。翻译的也不好，通俗易懂更谈不上。推荐看Pro AngularJS
- 6、说实话，翻译的很差劲，很多语句不通顺的地方。无可否认，译者都是Angular方面的大牛，但写作方面的能力真的不敢恭维。我甚至都怀疑这本书到底有没有校对，仿佛是为了赶时间仓促完成的毫无诚意之作。给三个星，毕竟是国内Angular的先行者。
- 7、讲得还是比较清楚的，适合angularjs初学者
- 8、名字起得不错，国内的书，讲的不够深入，不太好
- 9、已弃坑。。。
- 10、不比看angularjs官方api手册强~
- 11、一般
- 12、入门后面讲解不够详细
- 13、对于angularjs我是新手，看起来大概了解了它的基本原理，还是非常不错的。推荐一下。
- 14、内容比较全，但是不是非常的深入，很多东西都是点到为止
- 15、AngularJS的入门书，看完能快速上手Angular，但是感觉对原理性的东西讲解得不多，想要深入原理最好还要去看《Build your own Angular》
- 16、翻译太不严谨了，错误太多太多，代码根本没检查，直接复制粘贴。一会儿provider，一会儿提供者的，明明是一组选项，一会儿用中文，一会儿用英文。书的整体目录结构比较混乱，没有规律。
- 17、浅显易懂，有一个实际的例子贯穿全书，但所用版本较老
- 18、出来实习时购买的，学习angularjs必看的一本书，多谢作者
- 19、基本上可以在internet上查到相关资料
- 20、顺序有些不合理，如果没接触过angular刚开始有些内容会很费解
- 21、翻译的太水了，严重影响阅读，内容编排顺序也不好~
- 22、手册中文版
- 23、介绍基础和框架的知识挺全面的，美中不足是这本书著书时angular的版本相对现在已经有些旧了
- 24、必须力荐
- 25、之前用angular用过一段时间，最近只是将这本书粗读了一遍，还是有些收获的，所以给了四分，推荐新手读~
- 26、1.内容有些陈旧过时。2.本书是三位作者分别撰写各自章节然后拼合而成，所以三人之间的内容逻辑和结构上都存在一定问题，前后顺序以及一些概念、用法讲解的先后存在误导。
- 27、知识点是OK。但是知识结构不够系统化。
- 28、把概念讲得还算是清楚明白，但是一本400多页的技术书，不涉及原理，也是说不通的。我想买一本深入讲解angularjs实现原理的书籍，可惜这本不是。
- 29、快速看完，大致了解下有什么东西，本书就是结构散漫，看完好像有很多知识，但还是没有一个整体的印象，也就是看完可能还啥也不会。不如去看MEAN栈
- 30、很不错的一本入门书
- 31、内容蛮多的，也有深度，但是不适合初学者读。
- 32、写的比较易懂。
- 33、内容很全，不过翻译的跟屎一样
- 34、书的内容还不错，Angular的方方面面都讲到了，不过书中的错误比较多，如果英语水平够好的话，最好还是看原版，或者是在图灵社区中对照勘误来阅读。
- 35、很全面的一本书，看过之后受益匪浅。虽然很厚，干货还是不少。



# 《AngularJS权威教程》

## 精彩书评

1、书挺厚，但是读起来很快，没有需要深入思考的东西，大部分内容都是泛泛介绍angularjs，作为入门了解书挺不错，可以建立起一个对angularjs的框架，看过本书后再详细查看官方api，最后可以多阅读分析angularjs代码，这样就能掌握的差不多



## 版权说明

本站所提供下载的PDF图书仅提供预览和简介，请支持正版图书。

更多资源请访问:[www.tushu111.com](http://www.tushu111.com)