



tr
e
o
x
e

Angular

Développez vos applications **web**
avec le framework **JavaScript**
de **Google**

Version numérique
OFFERTE !

www.editions-eni.fr

Daniel DJORDJEVIC
William KLEIN



Sébastien
OLLIVIER

Avant-propos

- 1. Pourquoi cet ouvrage ? 13
- 2. À qui s'adresse cet ouvrage ? 14
- 3. Structure de l'ouvrage. 14

Chapitre 1 Introduction

- 1. État des lieux du développement web. 17
- 2. Angular 18
 - 2.1 Organisation par composants 18
 - 2.2 TypeScript. 19
 - 2.3 Les spécifications ES6. 19
 - 2.4 DOM Virtuel 19
 - 2.5 Rendu côté serveur possible. 20
- 3. AngularJS vs Angular 21

Chapitre 2 Ma première application

- 1. Introduction 23
- 2. Le setup 23
 - 2.1 Installation de npm 23
 - 2.2 Installation de TypeScript et initialisation du projet. 25
 - 2.3 Installation des dépendances Angular. 26
- 3. Mon premier composant 28
- 4. Mon premier module 31
- 5. Lancement de l'application 32
- 6. Angular CLI à la rescousse 34

Chapitre 3

Fondamentaux d'Angular

1. Introduction	37
2. Les composants.....	37
3. Les templates	39
3.1 Binding	40
3.2 Événements JavaScript.....	42
3.3 Listes	43
3.4 Afficher/cacher un élément	45
3.5 Liens.....	48
3.6 Styles.....	49
3.7 Directives	53
4. Les décorations	55
5. Les pipes	56
5.1 Utiliser un pipe	56
5.2 Les pipes du framework	57
5.3 Créer un pipe	58
5.3.1 Les pipes de transformation.....	58
5.3.2 Les pipes filtres	60
5.4 Les pipes purs et impurs.....	61
6. Les modules.....	63
7. Observable et promise	64
8. Préparer l'application pour la production	66

Chapitre 4 TypeScript

1. JavaScript	67
2. TypeScript.....	69
2.1 Syntaxe	70
2.1.1 Variables	70
2.1.2 Fonctions.....	71
2.1.3 Classes.....	71
2.1.4 Arrow Function	73
2.1.5 Union type	74
2.1.6 Interface	75
2.1.7 Générique	76
2.1.8 Les décorateurs	76
2.2 Typings	77
2.3 Compilation	78
2.4 EcmaScript 6 et 7 dès maintenant.....	79
2.5 Angular et TypeScript	80

Chapitre 5 Angular CLI

1. Introduction	81
2. Qu'est-ce qu'Angular CLI ?	81
2.1 La définition	81
2.2 Les commandes principales	83
2.2.1 Créer un nouveau projet Angular CLI : ng new.....	83
2.2.2 Compiler l'application : ng build	83
2.2.3 Servir l'application avec ng serve.....	87
2.2.4 Exécuter les tests avec Angular CLI.....	88
2.3 Les commandes de scaffolding.....	91
2.4 La génération d'un composant.....	92

2.5	De angular-cli à @angular/cli	99
2.5.1	Migrer	99
2.5.2	Les changements à effectuer	99
3.	Créer une application avec Angular CLI	102
3.1	Installer Angular CLI	102
3.2	Créer le socle initial	102
4.	Configurer un projet Angular CLI	106
4.1	Schéma du fichier de configuration d'Angular CLI	106
4.2	Gérer les assets lors de la compilation	110
4.3	Rajouter des environnements	113
4.4	Intégrer une librairie externe	114
4.5	Intégrer un préprocesseur CSS	115
4.6	Ajouter des entités de manière globale	116
4.6.1	Ajouter un script globalement	116
4.6.2	Ajouter une feuille de style globalement	117
4.7	Éjecter la configuration Webpack	119
5.	Compilation Ahead-Of-Time (AOT)	119

Chapitre 6

Les composants

1.	Introduction	123
2.	Qu'est-ce qu'un composant ?	123
2.1	Une première définition	123
2.2	Créer un premier composant	125
2.2.1	Syntaxe inline	125
2.2.2	Une réelle syntaxe, découpée en plusieurs fichiers	126
2.2.3	Utiliser un composant dans son application	127
2.3	Afficher les propriétés d'un composant	129

3. Les inputs et outputs	131
3.1 Les inputs d'un composant	131
3.1.1 Déclarer une variable en tant qu'Input	131
3.1.2 Un exemple concret avec une liste	132
3.1.3 Donner un nom personnalisé à son input	134
3.2 Les outputs d'un composant	135
3.3 Donner un nom personnalisé à son output	138
4. Interaction entre composants	138
4.1 Passer une donnée du parent vers l'enfant à l'aide d'un input ..	139
4.2 Intercepter un changement de valeur à l'aide d'un setter	141
4.3 Notifier le parent à l'aide d'un EventEmitter en output	143
4.4 Observer les changements d'input avec ngOnChanges	145
4.5 Utiliser une variable locale	148
5. Les décorateurs @ViewChild et @ViewChildren	150
5.1 Récupérer une référence avec @ViewChild	150
5.2 Récupérer plusieurs références avec @ViewChildren	152
6. Les composants Angular et la View Encapsulation	154
6.1 Le Shadow DOM	154
6.2 Spécifier une View Encapsulation	155
6.3 Les types de View Encapsulation	156
6.3.1 La View Encapsulation émulée	156
6.3.2 La View Encapsulation native	159
6.3.3 Aucune View Encapsulation	160

Chapitre 7

Les services

1. Introduction	163
2. Qu'est-ce qu'un service ?	163
3. Déclarer son service	164
4. Utiliser son service	164

5. Rendre son service asynchrone	166
5.1 Les promises	166
5.2 Les observables	169
5.3 Que choisir ?	174
6. Notifier lorsque les données changent	175

Chapitre 8

L'injection de dépendances

1. Principe de base	181
2. Injection de dépendances dans Angular	182
2.1 Enregistrement global	183
2.2 Enregistrement dans un composant	184
2.3 Injecter une dépendance	185
3. Provider	186
3.1 useClass	186
3.2 UseExisting	187
3.3 UseFactory	189
3.4 UseValue	192
3.5 OpaqueToken	192
3.6 Dépendance optionnelle	196
3.7 Injection restreinte	197
3.8 Restriction de l'enregistrement d'une dépendance	199

Chapitre 9

Le requêtage HTTP

1. Introduction	201
2. Obtenir et envoyer des données	201
3. Transformer des données	204

4. Communiquer de manière sécurisée	205
4.1 Authentification basique	205
4.2 Authentification OAuth	208
5. Simuler le requêtage HTTP	212

Chapitre 10

Les interactions utilisateur

1. Qu'est-ce que l'évent binding ?	219
2. S'abonner à un événement	220
3. Récupérer une entrée utilisateur	221
3.1 Comment manipuler l'objet \$event ?	221
3.2 Utiliser un typage fort pour \$event	222
3.3 Une alternative grâce à un template reference variable	223
3.4 Utiliser un template reference variable avec un événement ..	224
3.5 Filtrer les entrées utilisateur	225

Chapitre 11

Les formulaires

1. Les formulaires basés sur un template	229
2. Créer un composant formulaire	230
2.1 Le composant	230
2.2 La vue et le data binding	231
2.2.1 La syntaxe ngModel	232
2.2.2 ngModel en détail	232
2.2.3 L'utilisation de ngModel dans un cas concret	232
2.3 Intégrer le formulaire dans l'application	233
3. Les états et la validité d'un champ	234
3.1 Les états d'un input	234
3.2 Styliser selon la validité	236
4. Soumettre le formulaire	237

5. Les formulaires et les FormControls	238
5.1 Les contrôles et les groupes de contrôles	238
5.2 Les validateurs intégrés	240
5.3 Créer un validateur personnalisé	241
5.4 Les validateurs asynchrones	242

Chapitre 12

Le routage

1. Introduction	245
2. Définir les routes d'une application	245
3. Le rendu de composant	247
4. Naviguer dans son application	249
5. Récupération des données de routage	250
6. Outlet nommé	254
6.1 Définir des outlets nommés	254
6.2 Naviguer avec des outlets nommés	256
7. Resolver	257

Chapitre 13

Les directives

1. Introduction	263
2. Qu'est-ce qu'une directive?	263
2.1 Introduction	263
2.2 Directives communes	264
2.2.1 NgIf	264
2.2.2 NgFor	264
2.2.3 NgStyle	265
2.2.4 NgClass	265

3. Les directives d'attribut	266
3.1 Créer une directive d'attribut	266
3.2 Interagir avec les événements du DOM	268
3.3 Passer des valeurs aux directives d'attribut.....	269
4. Les directives structurelles	272
4.1 La balise <template> et l'astérisque.....	272
4.2 Créer une directive structurelle	273

Chapitre 14

Tester son application

1. Introduction	279
2. Les tests unitaires	279
2.1 Introduction aux tests avec Jasmine	280
2.2 Exécuter du code avant ou après chaque test	283
2.3 Les matchers mis à disposition	284
2.3.1 Comment utiliser un matcher ?	285
2.3.2 Des exemples de matchers	285
2.3.3 Négation d'un matcher	288
2.4 Créer un matcher personnalisé	289
2.4.1 Créer une librairie de matchers personnalisés.....	289
2.4.2 Utiliser un matcher personnalisé.....	291
2.5 Les composants.....	292
2.5.1 Le composant à tester.....	292
2.5.2 Le TestBed.....	293
2.5.3 Vérifier que le composant est bien instancié	294
2.5.4 Contrôler les propriétés du composant.....	295
2.5.5 S'assurer que le rendu du composant soit cohérent	295
2.6 Les services	297
2.6.1 Le service à tester	297
2.6.2 Tester le service.....	297
2.6.3 S'assurer que le service est bien injecté.....	298

2.7	Les directives.....	299
2.7.1	La directive à tester.....	299
2.7.2	Tester la directive.....	300
2.8	Injecter un faux service.....	302
3.	Les tests e2e.....	303
3.1	Lancer les tests e2e.....	304
3.2	Écrire un test e2e.....	305
3.2.1	Interagir avec le navigateur.....	306
3.2.2	Récupérer un élément du DOM et interagir avec.....	307
3.2.3	Interagir avec les éléments DOM.....	308
3.2.4	Tester le composant.....	310
3.3	Écrire un test e2e, en mieux.....	311

Chapitre 15

Le cross-platform avec Angular

1.	Apache Cordova.....	315
1.1	Créer un projet Apache Cordova.....	316
1.2	Angular dans Apache Cordova.....	318
1.3	Utiliser un plug-in Apache Cordova.....	324
2.	Ionic 2.....	329
2.1	Créer un projet Ionic.....	329
2.2	Utiliser les composants Ionic.....	330
2.3	Amorcer une application Ionic.....	334
2.4	Naviguer dans une application Ionic.....	337
2.5	Utiliser un plug-in avec Ionic.....	339
3.	NativeScript.....	342
3.1	Créer une application avec NativeScript.....	343
3.2	Utiliser les composants NativeScript.....	344
3.3	Amorcer une application NativeScript.....	348
3.4	Naviguer dans une application NativeScript.....	349
3.5	NativeScript et Web.....	352

Chapitre 16**Pour aller plus loin**

1. Introduction	355
2. Rendu côté serveur	355
2.1 Principe de la mise en place	356
2.2 Création du module serveur	357
2.3 Compilation AOT	359
2.4 Serveur Node.js	361
2.5 Compilation et exécution	369
3. La détection de changement	371
3.1 Pourquoi la détection de changement ?	371
3.2 Les Zones, ou comment notifier Angular ?	374
3.2.1 Le comportement de la détection de changement	376
3.2.2 L'immuabilité et la stratégie de détection de changement OnPush	377
3.3 Encore plus de contrôle sur la détection de changement	380
4. Le cycle de vie d'un composant	381
4.1 La présentation des lifecycle hooks	382
4.1.1 Les différents hooks	382
4.1.2 Utiliser un lifecycle hook	383
4.2 Le cycle de vie d'un composant	383
4.2.1 Le constructeur	386
4.2.2 ngOnInit	387
4.2.3 ngOnChanges	387
4.2.4 ngDoCheck	388
4.2.5 ngAfterContentInit	389
4.2.6 ngAfterContentChecked	390
4.2.7 ngAfterViewInit	390
4.2.8 ngAfterViewChecked	391
4.2.9 ngOnDestroy	391
Index	393