

Les éléments à télécharger sont disponibles à l'adresse suivante :  
**<http://www.editions-eni.fr>**  
 Saisissez la référence ENI de l'ouvrage **RI5JASAP** dans la zone de recherche  
 et validez. Cliquez sur le titre du livre puis sur le bouton de téléchargement.

## Partie 1 : Présentation de JavaScript et programmation à partir d'algorithmes

### Chapitre 1-1

#### Présentation du langage JavaScript

1. Définition et rapide historique . . . . . 15
2. Prérequis pour un apprentissage aisé du langage. . . . . 17
3. Outillage nécessaire . . . . . 17
4. Positionnement du JavaScript face à d'autres technologies  
de développement web (HTML, CSS, PHP...) . . . . . 18

### Chapitre 1-2

#### Développement à partir d'algorithmes

1. Présentation de la notion d'algorithme . . . . . 19
2. Notion de variable . . . . . 20
  - 2.1 Présentation des notions de variable et de type . . . . . 20
  - 2.2 Types de base et opérations associées . . . . . 21
  - 2.3 Intérêt des types . . . . . 22
  - 2.4 Utilisation des variables dans des expressions . . . . . 24
  - 2.5 Tableau récapitulatif des opérateurs . . . . . 24
3. Manipulation des variables . . . . . 25
  - 3.1 Nommage des variables . . . . . 25
  - 3.2 Affectation . . . . . 26
  - 3.3 Exercice n°1 : Inversion du contenu de deux variables mémoire . . . . . 27
  - 3.4 Affichage des résultats . . . . . 28
  - 3.5 Exercice n°2 : Surfaces de cercles . . . . . 30
  - 3.6 Saisie au clavier . . . . . 31
  - 3.7 Exercice n°3 : Surface et volume d'une sphère . . . . . 32

## 2 \_\_\_\_\_ Apprendre à développer

avec JavaScript

4. Fonctions prédéfinies. . . . .	32
4.1 Exercice n°4 : Affichage de la longueur d'un nom. . . . .	33
4.2 Exercice n°5 : Détermination des initiales. . . . .	34
5. Traitements conditionnés. . . . .	36
5.1 Exercice n°6 : Polynôme du second degré. . . . .	38
5.2 Exercice n°7 : Libellé du mois en clair. . . . .	39
5.3 Exercice n°8 : Libellé du mois en clair (Suivant ... Finsuivant). . . . .	41
6. Structures itératives. . . . .	42
6.1 Principe des itérations. . . . .	42
6.2 Structures itératives de base. . . . .	42
6.3 Exercice n°9 : Moyenne de 10 nombres. . . . .	45
6.4 Exercice n°10 : Moyenne d'une série de n nombres. . . . .	45
6.5 Exercice n°11 : Plus Grand Commun Diviseur par la méthode des divisions successives. . . . .	48
6.6 Structure itérative Pour. . . . .	50
6.7 Exercice n°12 : Calcul de la moyenne de 10 nombres. . . . .	51
6.8 Exercice n°13 : Décompte du nombre de voyelles dans un mot. . . . .	52
7. Tableaux à dimension unique. . . . .	54
7.1 Exercice n°14 : Décompte des nombres pairs dans un tableau. . . . .	55
8. Tableaux à dimensions multiples. . . . .	56
8.1 Exercice n°15 : Mini-tableur. . . . .	56
9. Procédures, fonctions et passage de paramètres. . . . .	58
9.1 Les objectifs. . . . .	58
9.2 Les procédures. . . . .	58
9.3 Exercice n°16 : Appel d'une procédure avec passage de paramètres. . . . .	59
9.4 Les fonctions. . . . .	60
9.5 Exercice n°17 : Appel d'une fonction avec passage de paramètres. . . . .	61

## Partie 2 : Bien débiter avec JavaScript

### Chapitre 2-1

#### Bases du langage JavaScript

1. Méthodologie d'apprentissage . . . . .	63
2. Variables (déclaration et typage) . . . . .	64
2.1 Exercice n°2 : Surfaces de cercles . . . . .	64
2.2 Exercice n°3 : Surface et volume d'une sphère . . . . .	70
2.3 Exercice n°4 : Nombre de lettres d'un mot . . . . .	72
2.4 Exercice n°5 : Détermination des initiales . . . . .	73

### Chapitre 2-2

#### Conditionnement des traitements

1. Présentation de la syntaxe . . . . .	75
2. Exemples . . . . .	77
2.1 Exercice n°6 : Polynôme du second degré . . . . .	77
2.2 Exercice n°8 : Impression du libellé d'un mois . . . . .	78

### Chapitre 2-3

#### Traitements itératifs (boucles)

1. Présentation de la syntaxe des boucles . . . . .	81
2. Boucle while . . . . .	82
2.1 Syntaxe . . . . .	82
2.2 Exercice n°9 : Moyenne de 10 nombres saisis au clavier . . . . .	82
2.3 Exercice n°10 : Moyenne d'une série de n nombres saisis au clavier . . . . .	83
3. Boucle do while . . . . .	85
3.1 Syntaxe . . . . .	85
3.2 Exercice n°11 : Moyenne d'une série de n nombres saisis au clavier . . . . .	85
4. Boucle for . . . . .	86
4.1 Syntaxe . . . . .	86
4.2 Exercice n°12 : Moyenne d'une série de 10 nombres saisis au clavier . . . . .	87
4.3 Exercice n°13 : Décompte du nombre de voyelles dans un mot . . . . .	88

## Chapitre 2-4

### Tableaux

1. Tableaux à dimension unique . . . . .	91
1.1 Syntaxe . . . . .	91
1.2 Exercice n°14 : Décompte des nombres pairs dans un tableau . . . . .	92
2. Tableaux à dimensions multiples . . . . .	94
2.1 Syntaxe . . . . .	94
2.2 Exercice n°15 : Mini-tableur . . . . .	95

## Chapitre 2-5

### Procédures et fonctions

1. Les procédures . . . . .	97
1.1 Syntaxe . . . . .	98
1.2 Exercice n°16 : Appel d'une procédure avec passage de paramètres . . .	98
2. Les fonctions . . . . .	100
2.1 Syntaxe . . . . .	100
2.2 Exercice n°17 : Appel d'une fonction avec passage de paramètres . . .	100

## Partie 3 : Approche POO sous JavaScript

### Chapitre 3-1

#### Approche "objet" en JavaScript

1. Introduction . . . . .	103
2. Programmation orientée objet au travers d'exemples . . . . .	103
2.1 Séquence 1 : Déclaration des objets JavaScript en méthode "Inline".	104
2.2 Séquence 2 : Création des objets JavaScript par constructeur . . . . .	105
2.3 Séquence 3 : Variables privées dans une instance d'objet . . . . .	106
2.4 Séquence 4 : Passage de paramètre(s) à un constructeur . . . . .	107
2.5 Séquence 5 : Non-partage des méthodes par les instances d'objets . .	107
2.6 Séquence 6 : Notion de prototype . . . . .	109
2.7 Séquence 7 : Surcharge d'une méthode . . . . .	110
2.8 Séquence 8 : Extension d'un prototype . . . . .	111
2.9 Séquence 9 : Mécanisme de l'héritage . . . . .	112

2.10 Séquence 10 : Limite de l'héritage de la séquence n°9	114
2.11 Séquence 11 : Une seconde limite à notre héritage	115

## Chapitre 3-2 Objets de base de JavaScript

1. Présentation	117
2. Les objets de base	117
2.1 Objet Array	118
2.2 Objet Date	118
2.3 Objet Math	125
2.4 Objet window	128
2.5 Objet navigator	137
2.6 Objet String	139

## Chapitre 3-3 Les nouveautés d'EcmaScript 6

1. Présentation générale	145
2. Apports au niveau de la Programmation Orientée Objet	146
2.1 Notion de prototype	146
2.2 Surcharge d'une méthode	149
2.3 Extension de prototype	152
2.4 Héritage	154
2.5 Premier exemple POO en EcmaScript 6	158
2.6 Héritage en EcmaScript	161
2.7 Méthodes getter, setter et static en EcmaScript 6	164
3. Fonctions fléchées (arrow functions)	169
3.1 Avantages des fonctions fléchées	169
3.2 Exemple	169
4. Structures Map, Set et boucle for of	174
4.1 Présentation générale	174
4.2 Exemple	175
5. Portée des variables (var ou let)	183
5.1 Présentation générale	183
5.2 Exemple	183

# 6 \_\_\_\_\_ Apprendre à développer

avec JavaScript

6. Promesses (promise) .....	187
6.1 Présentation générale .....	187
6.2 Exemple .....	187
7. Déstructuration .....	191
7.1 Présentation générale .....	191
7.2 Exemple .....	191

## Partie 4 : Gestion de formulaire et modèle DOM

### Chapitre 4-1

#### Saisie de données via des formulaires

1. Pilotage des contrôles de saisie via JavaScript .....	195
1.1 Contrôle de saisie sur un champ texte .....	195
1.2 Contrôle de numéricité d'une saisie dans un champ texte .....	201
1.3 Contrôle de caractères alphabétiques d'une saisie dans un champ texte .....	204
1.4 Contrôle de caractères alphabétiques et numériques d'une saisie dans un champ texte .....	204
1.5 Contrôle de longueur d'une saisie dans un champ texte .....	205
1.6 Contrôle de saisie sur une adresse e-mail .....	206
1.7 Contrôle d'un choix dans une liste déroulante (version simplifiée) ..	206
1.8 Contrôle d'un choix dans une liste déroulante (version étendue) ..	209
1.9 Contrôle d'un choix par bouton radio .....	212
1.10 Contrôle d'un choix par case à cocher .....	216

### Chapitre 4-2

#### Modèle DOM

1. Introduction .....	221
1.1 Définition de DOM .....	221
1.2 Définition de l'arborescence .....	222
2. Apprentissage du modèle DOM .....	224
2.1 Script "Hello World!" .....	224
2.2 Différence entre write et writeln .....	225
2.3 Gestion des liens hypertextes .....	226

2.4	Gestion des images . . . . .	228
2.5	Gestion des formulaires et de leurs balises . . . . .	230
2.6	Gestion des ancres . . . . .	232
2.7	Gestion de la navigation entre pages web . . . . .	234
2.8	Affichage de caractéristiques générales du document . . . . .	239
2.9	Gestion des boutons dans les formulaires . . . . .	241
2.10	Gestion des tableaux (balise HTML table) . . . . .	250

## Chapitre 4-3

### Exploration de flux XML via DOM

1.	Notion de flux XML . . . . .	279
2.	Exemples . . . . .	280
2.1	Exemple 1 : Affichage d'un contenu d'e-mail codé en XML . . . . .	280
2.2	Exemple 2 : Liste des marques des voitures (fichier voitures.xml) . . . . .	283
2.3	Exemple 3 : Liste des marques des voitures avec une boucle . . . . .	285
2.4	Exemple 4 : Liste des nœuds rattachés à la racine . . . . .	287
2.5	Exemple 5 : Liste des champs (nœuds) de chaque voiture . . . . .	289
2.6	Exemple 6 : Remplacement d'une valeur de nœud . . . . .	291
2.7	Exemple 7 : Accès aux attributs . . . . .	292
2.8	Exemple 8 : Accès à un nœud parent . . . . .	293
2.9	Exemple 9 : Parcours arrière des nœuds . . . . .	294
2.10	Exemple 10 : Remplacement systématique d'une valeur d'attribut . . . . .	295
2.11	Exemple 11 : Conversion XML en HTML . . . . .	296
2.12	Exemple 12 : Suppression d'un nœud dans un flux XML . . . . .	298

## Partie 5 : Cookies et mécaniques de persistance

### Chapitre 5-1

#### Gestion des cookies en JavaScript

1.	Notion de cookie . . . . .	303
2.	Écriture d'un cookie . . . . .	304
3.	Lecture d'un cookie . . . . .	306
4.	Suppression d'un cookie . . . . .	308

## Chapitre 5-2

### Stockage local de données

1. Présentation générale des solutions ..... 309
  - 1.1 Stockage par sessionStorage ..... 310
  - 1.2 Stockage par localStorage ..... 310
2. Mise en œuvre du Web Storage au travers d'exemples ..... 311
  - 2.1 Exemple 1 : Stockage par localStorage de chaînes de caractères .... 311
  - 2.2 Exemple 2 : Stockage dans le localStorage d'un objet JavaScript. ... 319

## Chapitre 5-3

### Stockage distant (Ajax - PHP - MySQL - XML)

1. Présentation générale de la solution ..... 329
2. Mise en œuvre du stockage distant au travers d'exemples ..... 330
  - 2.1 Exemple 1 : Accès Ajax sur BDD MySQL  
(liste de l'ensemble des voitures) ..... 331
  - 2.2 Exemple 2 : Accès MySQL via Ajax ..... 352

## Chapitre 5-4

### Stockage distant (Ajax - PHP - MySQL - JSON)

1. Présentation générale de la solution ..... 359
2. Mise en œuvre du stockage distant au travers d'exemples ..... 361
  - 2.1 Exemple 1 : Présentation du système de notation JSON ..... 361
  - 2.2 Exemple 2 : Lecture d'un fichier JSON via XMLHttpRequest ..... 365
  - 2.3 Exemple 3 : Lecture d'un fichier JSON  
via XMLHttpRequest et un script serveur en PHP ..... 371
  - 2.4 Exemple 4 : Lecture d'une table MySQL  
via XMLHttpRequest (serveur PHP et flux JSON) ..... 373
  - 2.5 Exemple 5 : Recodage de l'exemple 4 avec une liste déroulante. .... 378



## Partie 6 : Géolocalisation, dessin et graphiques de gestion

### Chapitre 6-1

#### Géolocalisation

1. Principe de la géolocalisation .....	383
2. Exemples d'applications de géolocalisation .....	384
2.1 Exemple 1 : Affichage de la carte de l'Ouest de la France. ....	384
2.2 Exemple 2 : Affichage de la carte de l'Ouest de la France (marqueur) .....	390
2.3 Exemple 3 : Affichage de la carte de l'Ouest de la France (marqueur et cercles de population) .....	393
2.4 Exemple 4 : Affichage de la carte de l'Ouest de la France (informations météorologiques) .....	396
2.5 Exemple 5 : Affichage de la carte de Rennes Centre-Sud (avec photo des sites importants) .....	403
2.6 Exemple 6 : Affichage de la carte de Rennes (Street View) .....	404

### Chapitre 6-2

#### Dessin (HTML5 CANVAS)

1. Présentation de l'API HTML5 CANVAS .....	409
2. Exemples d'applications de l'élément <canvas> .....	410
2.1 Exemple 1 : Tracé d'un simple carré .....	410
2.2 Exemple 2 : Tracé d'une grille de TicTacToe .....	413
2.3 Améliorations possibles sur le jeu du TicTacToe .....	420

### Chapitre 6-3

#### Graphiques de gestion

1. Différentes solutions de conception de graphiques de gestion. ....	421
2. Exemples d'utilisation des API Google Charts. ....	421
2.1 Exemple 1 : Tracé d'un histogramme .....	422
2.2 Exemple 2 : Tracé d'un graphique en secteurs .....	427
2.3 Exemple 3 : Tracé d'une carte .....	429

## Partie 7 : Frameworks JavaScript

### Chapitre 7-1

#### Positionnement des frameworks JavaScript

1. Présentation générale des frameworks JavaScript ..... 433
  - 1.1 Frameworks « front-end » ..... 433
  - 1.2 Frameworks « back-end » ..... 434
  - 1.3 Solutions de développement « hybride » ..... 434
2. Les frameworks Node.js, Svelte, React et React Native ..... 435

### Chapitre 7-2

#### Installation de Node.js

1. Présentation du framework Node.js ..... 437
2. Installation du framework Node.js ..... 438

### Chapitre 7-3

#### Framework Svelte

1. Présentation du framework Svelte ..... 447
2. Site svelte.dev ..... 448
3. Création locale d'un projet Svelte ..... 451
4. Installation de Microsoft Visual Studio Code ..... 452
5. Projets Svelte ..... 453
  - 5.1 « eni\_svelte\_01 » - Première application ..... 453
  - 5.2 « eni\_svelte\_02 » - Importance des commentaires ..... 459
  - 5.3 « eni\_svelte\_03 » - Mise en place d'une image ..... 461
  - 5.4 « eni\_svelte\_04 » - Intégration de balises HTML ..... 463
  - 5.5 « eni\_svelte\_05 » - Un premier bouton ..... 465
  - 5.6 « eni\_svelte\_06 » - Le rôle du \$ dans les formules ..... 467
  - 5.7 « eni\_svelte\_07 » - Champs de saisie ..... 469
  - 5.8 « eni\_svelte\_08 » - Saisie avec curseur ..... 472
  - 5.9 « eni\_svelte\_09 » - Cases à cocher ..... 474
  - 5.10 « eni\_svelte\_10 » - Boutons « radio » ..... 477

5.11 « eni_svelte_11 » - Liste . . . . .	480
5.12 « eni_svelte_12 » - Liste et boutons « radio » . . . . .	482
5.13 « eni_svelte_13 » - Boutons Svelte Materialify . . . . .	484
5.14 « eni_svelte_14 » - Liste déroulante Svelte Materialify . . . . .	486
5.15 « eni_svelte_15 » - Switches Svelte Materialify . . . . .	488

## Chapitre 7-4 Framework React

1. Présentation de React . . . . .	491
2. Création locale d'un projet React . . . . .	492
3. Projets React basiques . . . . .	493
3.1 « eni_react_bases_01 » . . . . .	493
3.2 « eni_react_bases_02 » . . . . .	499
3.3 « eni_react_bases_03 » . . . . .	500
3.4 « eni_react_bases_04 » . . . . .	503
3.5 « eni_react_bases_05 » . . . . .	506
3.6 « eni_react_bases_06 » . . . . .	510
4. Les props React . . . . .	513
4.1 « eni_react_props_01 » . . . . .	513
4.2 « eni_react_props_02 » . . . . .	515
4.3 « eni_react_props_03 » . . . . .	517
5. Les librairies tierces pour React . . . . .	519
5.1 « eni_react_material-ui_table_01 » . . . . .	520
5.2 « eni_react_material-ui_table_02 » . . . . .	523
5.3 « eni_react_material-ui_table_03 » . . . . .	526
5.4 « eni_react_recharts_01 » . . . . .	529
5.5 « eni_react_recharts_02 » . . . . .	533
5.6 « eni_react_recharts_03 » . . . . .	536
6. Lecture de fichiers JSON sous React . . . . .	539
6.1 Différents types de flux JSON . . . . .	539
6.2 « eni_react_lecture_json_local » . . . . .	540

7.	Interactions avec un serveur PHP-MySQL. ....	548
7.1	« eni_react_php_mysql_01 » .....	548
7.2	« eni_react_php_mysql_02 » .....	555
7.3	« eni_react_php_mysql_03 » .....	570
7.4	« eni_react_php_mysql_04 » .....	574
7.5	« eni_react_php_mysql_05 » .....	584
8.	La navigation sous React (routage) .....	589

## Chapitre 7-5 Framework React Native

1.	Approches de développement pour périphériques mobiles. ....	595
1.1	Développements web, natif et hybride .....	595
1.1.1	Applications web .....	596
1.1.2	Applications natives. ....	596
1.1.3	Applications hybrides .....	597
1.2	Les trois principales plateformes .....	598
1.2.1	Apple iOS .....	598
1.2.2	Android. ....	599
1.2.3	Windows Phone, Windows 10 Mobile .....	599
2.	Présentation du framework React Native .....	599
3.	Projets React Native .....	601
3.1	Application « eni_react_native_helloworld » .....	601
3.2	Application « eni_react_native_list_view » .....	618
3.3	Application « eni_react_native_input_text » .....	623
3.4	Application « eni_react_native_picker_basique » .....	629
3.5	Application « eni_react_native_hook » .....	633
3.6	Application « eni_react_native_php_mysql » .....	637
3.7	Présentation d'une application CRUD PHP-MySQL complète. ....	643
3.8	Application « eni_react_native_villes_ajout » .....	644
3.9	Application « eni_react_native_villes_liste » .....	653
3.10	Application « eni_react_native_villes_modification » .....	658
3.11	Application « eni_react_native_villes_suppression » .....	669

## Chapitre 7-6 Framework Vue3.js

1. Présentation de Vue3.js . . . . .	675
1.1 Points communs entre React.js et Vue3.js . . . . .	675
1.2 Les points forts de Vue3.js . . . . .	676
1.3 Les points faibles de Vue3.js. . . . .	677
2. Applications basiques Vue3.js . . . . .	677
2.1 Application « eni_vue3js_liste_voitures_01.html » . . . . .	677
2.2 Application « eni_vue3js_liste_voitures_02.html » . . . . .	680
2.3 Application « eni_vue3js_liste_voitures_03.html » . . . . .	681
2.4 Application « eni_vue3js_liste_voitures_04.html » . . . . .	682
2.5 Application « eni_vue3js_liste_voitures_05.html » . . . . .	685
2.6 Application « eni_vue3js_liste_voitures_06.html » . . . . .	687
2.7 Application « eni_vue3js_liste_voitures_07.html » . . . . .	691
2.8 Application « eni_vue3js_liste_voitures_08.html » . . . . .	696
2.9 Application « eni_vue3js_liste_voitures_09.html » . . . . .	698
3. Applications Vue3.js couplées avec MySQL . . . . .	707
3.1 Content Delivery Network (CDN) . . . . .	708
3.2 Node.js Package Manager (NPM) . . . . .	708
3.3 Interface en ligne de commandes (CLI). . . . .	710
3.4 Application « eni_vue3js_voitures_mysql_1 » . . . . .	710
3.4.1 Création de l'application dans un sous-répertoire . . . . .	710
3.4.2 Lancement de l'application . . . . .	712
3.4.3 Modifier les scripts générés (squelette) par le template . . . . .	712
3.5 Application « eni_vue3js_voitures_mysql_2 » . . . . .	726
3.6 Application « eni_vue3js_voitures_mysql_3 » . . . . .	732
3.7 Application « eni_vue3js_voitures_mysql_4 » . . . . .	739

## Partie 8 : JavaScript et l'Intelligence Artificielle

### Chapitre 8-1

#### Mise en œuvre de la librairie Math.js

1. Introduction . . . . .	747
2. Application « eni_mathjs_01.html » - Statistiques basiques sur une matrice 3*3 . . . . .	747
3. Application « eni_mathjs_02.html » - Statistiques Iris (échantillon partiel) . . . . .	750
4. Application « eni_mathjs_03.html » - Statistiques Iris (échantillon complet) . . . . .	759
5. Application « eni_mathjs_04.html » - Régression linéaire Iris (échantillon partiel) . . . . .	766

### Chapitre 8-2

#### Mise en œuvre de la librairie TensorFlow.js

1. Algorithme de classification automatique . . . . .	777
2. Application « eni_classification_iris.html » . . . . .	778

Index . . . . .	791
-----------------	-----

Les éléments à télécharger sont disponibles à l'adresse suivante :

**<http://www.editions-eni.fr>**

Saisissez la référence ENI de l'ouvrage **RI2TYP** dans la zone de recherche et validez. Cliquez sur le titre du livre puis sur le bouton de téléchargement.

## Avant-propos

### Chapitre 1

#### Introduction

1. Un peu d'histoire. ....	9
2. ECMAScript .....	13
3. Pourquoi TypeScript ? .....	17
3.1 Transpilation .....	17
3.2 Typage statique .....	20
4. Les dessous du typage. ....	22
4.1 Les fichiers de définition .....	22
4.2 Typage structurel .....	24
5. L'architecture de TypeScript .....	27
5.1 Core TypeScript Compiler. ....	27
5.2 Standalone TS Compiler .....	28
5.3 Language Service. ....	28
5.4 Options de compilation .....	28
6. Environnement de développement .....	30
6.1 Node.js. ....	30
6.2 Visual Studio Code. ....	32
6.3 Installer TypeScript .....	38

## Chapitre 2

### Types et instructions basiques

1. Variable et portée . . . . .	41
2. Types basiques . . . . .	45
2.1 Introduction . . . . .	45
2.2 Types primitifs basiques . . . . .	45
2.3 Types primitifs : undefined et null . . . . .	49
2.4 Types par référence : array et tuples . . . . .	52
2.5 Types par référence : object . . . . .	55
2.6 Types spéciaux : any et void . . . . .	58
2.7 Affirmation de type . . . . .	60
2.8 Type spécial : unknown . . . . .	61
3. Décomposition . . . . .	64
4. Énumération . . . . .	68
5. Conditionnelle If...Else . . . . .	75
6. Optional chaining . . . . .	77
7. Opérateurs . . . . .	78
7.1 Opérateurs arithmétiques . . . . .	79
7.2 Opérateurs unaires . . . . .	79
7.3 Opérateurs de comparaison . . . . .	80
7.4 Opérateurs logiques combinatoires . . . . .	81
8. Zoom sur les opérateurs . . . . .	81
8.1 Opérateurs unaires + et - . . . . .	82
8.2 Opérateur unaire ! . . . . .	82
8.3 Opérateurs == et === . . . . .	83
8.4 Opérateurs    et && . . . . .	84
8.5 Opérateur de coalescence null (??) . . . . .	86
8.6 Opérateur de décomposition . . . . .	86
9. Boucles . . . . .	88
9.1 Boucles for...in . . . . .	88
9.2 Boucle for...of . . . . .	90



10. Symbol . . . . .	91
11. Itération et collections . . . . .	95
11.1 Iterator . . . . .	95
11.2 Générateurs . . . . .	97
11.3 Map et Set . . . . .	98
11.3.1 Map . . . . .	98
11.3.2 Set . . . . .	99
11.3.3 WeakMap et WeakSet . . . . .	100
12. Fonctions . . . . .	100
12.1 Les bases . . . . .	100
12.2 Paramètre rest . . . . .	102
12.3 Les closures . . . . .	105
12.4 This et les fonctions fléchées . . . . .	106
12.5 Type fonction . . . . .	111
13. Prototype . . . . .	112
14. Gestion des exceptions . . . . .	113

## Chapitre 3

### La programmation orientée objet

1. Introduction . . . . .	117
2. Les classes . . . . .	118
3. Les propriétés . . . . .	121
4. Les méthodes . . . . .	124
5. Les constructeurs . . . . .	130
6. Le statisme . . . . .	132
7. L'accessibilité des membres . . . . .	135
8. L'encapsulation . . . . .	143
9. L'héritage . . . . .	146
10. L'abstraction . . . . .	153
11. Les interfaces . . . . .	157

12. Le polymorphisme .....	168
13. Les principes SOLID .....	170
13.1 Introduction aux principes SOLID .....	170
13.2 Le principe de responsabilité unique .....	170
13.3 Le principe d'ouverture à l'extension, fermeture à la modification .....	173
13.4 Le principe de substitution de Liskov .....	176
13.5 Le principe de ségrégation des interfaces .....	181
13.6 Le principe d'inversion des dépendances .....	183

## Chapitre 4

### Les modules

1. Introduction .....	187
2. Historique .....	188
2.1 Pattern module .....	188
2.2 AMD et CommonJs .....	190
2.3 Standardisation .....	195
3. La norme ECMAScript 2015 .....	195
3.1 Export .....	199
3.2 Import .....	203
3.3 Import assert .....	206
4. Gestion des types .....	206
5. Chargement et résolutions .....	208
5.1 Résolutions .....	208
5.2 Chargement .....	211

## Chapitre 5

### La généricité

1. Introduction .....	213
2. Déclaration de base .....	213
3. Classes et interfaces .....	216

4. Contraintes .....	219
5. Générique et constructeur .....	225

## Chapitre 6 Les décorateurs

1. Introduction .....	227
2. Décorateurs expérimentaux. ....	229
2.1 Décorateurs expérimentaux de classe .....	230
2.2 Décorateurs expérimentaux de méthode .....	232
2.3 Décorateurs expérimentaux de propriété .....	235
2.4 Décorateurs expérimentaux de paramètre .....	239
2.5 Fabrique de décorateurs .....	242
2.6 Métadonnées. ....	244
3. Décorateurs ECMAScript. ....	253
3.1 Décorateurs ECMAScript de classe .....	255
3.2 Décorateurs ECMAScript de méthode .....	257
3.3 Décorateurs ECMAScript de propriété .....	259
3.4 Décorateurs ECMAScript d'accesseurs .....	262
3.4.1 Décorateur "getter" .....	262
3.4.2 Décorateur "setter". ....	264
3.4.3 Décorateur d'accesseurs automatiques .....	265

## Chapitre 7 Asynchronisme

1. Introduction .....	271
2. Callback hell .....	272
3. Promesse .....	274
3.1 Historique .....	274
3.2 Promesses ECMAScript 2015. ....	275
3.2.1 Résolution des promesses .....	275
3.2.2 Then, catch et finally .....	277
3.2.3 Méthodes statiques .....	282

3.2.4 Conclusion .....	283
4. Async/await .....	284

## Chapitre 8

### Système de types avancés

1. Introduction .....	289
2. Alias de type .....	289
3. Type never .....	293
4. Type union et intersection .....	295
4.1 Union .....	295
4.2 Intersection .....	297
5. Type guards .....	300
5.1 Introduction .....	300
5.2 Opérateur typeof .....	301
5.3 Opérateur instanceof .....	302
5.4 Opérateur in .....	304
5.5 Union discriminante .....	307
5.6 Type guards définis par l'utilisateur .....	309
5.7 Gestion de null et undefined .....	312
5.8 Cas impossible .....	315
5.9 Fonction d'affirmation .....	316
6. Types littéraux .....	318
7. Modèle de type littéral .....	320
8. Type index .....	324
9. Mapped type .....	329
9.1 Avant TypeScript 2.1 .....	329
9.2 Les opérateurs .....	332
9.3 Remappage des clés .....	333
9.4 Mapped type natif .....	334
10. Affirmation constante .....	337
11. Tuples variadiques .....	339

12. Type conditionnel .....	341
12.1 Les bases .....	341
12.2 Distributivité .....	343
12.3 Opérateur infer .....	346
12.4 Type conditionnel natif .....	347
13. Opérateur satisfies .....	351

## Chapitre 9

### TypeScript et la programmation fonctionnelle

1. Introduction .....	355
2. Fonction pure et impure .....	357
3. Immutabilité .....	361
4. Itération .....	368
5. Conditions .....	372
6. Fonction partielle .....	373
7. Currying .....	374
8. Pattern Matching .....	377
9. Composition de fonctions .....	380
10. Pour aller plus loin... ..	384

## Chapitre 10

### Un premier projet avec Node.js

1. Introduction .....	385
2. Mise en place du projet .....	386
2.1 Création du fichier package.json .....	387
2.2 Mise en place de TypeScript .....	388
2.2.1 Installation .....	388
2.2.2 Configuration de TypeScript .....	389
2.3 Une première application .....	396

3. Création du Framework MVC . . . . .	399
3.1 Registre des routes . . . . .	400
3.2 Fabrique de contrôleurs . . . . .	403
3.3 Le décorateur Controller . . . . .	405
3.4 La fabrique de décorateurs Action . . . . .	406
3.5 Création du contrôleur EmployeeController . . . . .	408
3.6 Initialisation de l'application . . . . .	410
4. Validation et OpenApi . . . . .	415
4.1 Typage des schémas . . . . .	416
4.2 Le décorateur de propriété SchemaProperty . . . . .	421
4.3 Liaison entre une action et un modèle . . . . .	422
4.4 Le décorateur de méthode Model . . . . .	425
4.5 Gestion des schémas dans la classe Server . . . . .	428
4.6 L'explorateur d'API . . . . .	429
4.7 L'action post du contrôleur EmployeeController . . . . .	430
5. Accès et persistance des données . . . . .	435
5.1 Typage des entités métier . . . . .	436
5.2 Persistance des données . . . . .	437
5.3 Repository . . . . .	439
6. Inversion des dépendances . . . . .	444
6.1 Le conteneur de dépendances . . . . .	445
6.2 Le décorateur Inject . . . . .	448
6.3 Injection des dépendances . . . . .	449
6.4 Stockage des données dans un fichier . . . . .	451
7. Allez plus loin . . . . .	453
Index . . . . .	455

## Avant-propos

## Introduction

1. Préambule .....	11
2. La stack MEAN.....	12
3. L'importance d'Angular .....	13
4. La différence entre Angular et AngularJS .....	14

## Chapitre 1 JavaScript

1. Pourquoi JavaScript ? .....	15
2. ECMAScript 6.....	16
3. Comment programmer en JavaScript ?.....	17
4. Rappel des bases .....	19
4.1 Les variables .....	19
4.2 Les boucles.....	20
4.3 Les tableaux.....	21
4.4 Les objets.....	22
4.5 Les types internes .....	24
4.6 Le transtypage.....	27
4.7 Les expressions régulières.....	29
4.8 Les blocs d'instructions .....	30
4.9 Les structures .....	32
4.9.1 Les structures de données usuelles .....	32
4.9.2 La structure conditionnelle .....	33
4.9.3 La structure switch .....	34
4.9.4 Les structures itératives.....	35

# 2 \_\_\_\_\_ Angular et Node.js

Développement web full stack avec MEAN

5. Les méthodes avancées . . . . .	36
5.1 Les promesses . . . . .	36
5.2 async/await . . . . .	38
5.3 .map() . . . . .	39
5.4 .filter() . . . . .	40
5.5 .reduce() . . . . .	41
5.6 .includes() . . . . .	43
5.7 .startsWith() et .endsWith() . . . . .	44
6. La programmation objet en JavaScript . . . . .	46
6.1 Le principe de la programmation objet . . . . .	46
6.2 Les factory functions . . . . .	46
6.3 La propriété prototype et son chaînage . . . . .	47
7. Conclusion . . . . .	49

## Chapitre 2 TypeScript

1. Le langage TypeScript . . . . .	51
2. Les différences avec JavaScript . . . . .	52
3. Le transpiler et son utilité . . . . .	54
4. Les types . . . . .	55
4.1 Les types de base . . . . .	55
4.2 Le typage de variables non scalaires . . . . .	55
4.3 Le type enum . . . . .	56
4.4 Le type générique any . . . . .	57
4.5 Le typage des fonctions . . . . .	57
5. Les fonctions . . . . .	58
5.1 Définir une fonction . . . . .	58
5.2 Les paramètres et la valeur de retour . . . . .	59
5.3 La fonction anonyme . . . . .	60
5.4 La fonction Constructor . . . . .	60



6. Les classes	62
6.1 La création d'une classe	62
6.2 La création d'une instance d'objets	63
6.3 L'accessibilité	64
6.4 L'héritage et son extension	65
6.5 La surcharge « override »	67
7. Les interfaces	68
7.1 Définir une interface	68
7.2 Le comportement d'une interface	69
8. Les décorateurs	71
9. Conclusion	72

## Chapitre 3 Node.js

1. Présentation	75
2. L'installation de Node.js 18	76
2.1 Sous Ubuntu	76
2.2 Sous Windows	77
2.3 Sous Mac	78
3. Les modules	80
3.1 La notion de module	80
3.2 Le gestionnaire de module npm	81
3.3 npmjs.com	82
3.4 package.json	83
4. La création du premier serveur Node.js	85
4.1 Fastify	85
4.2 Express	86
4.3 Choisir entre Fastify et Express	87

# 4 \_\_\_\_\_ Angular et Node.js

Développement web full stack avec MEAN

4.4	Les protocoles REST	88
4.4.1	La création des routes REST	89
4.4.2	La consultation d'une ressource via la méthode GET	92
4.5	Le module FS (FileSystem)	94
5.	Les middlewares	96
5.1	La notion de middleware	96
5.2	La création d'un middleware de journalisation d'événements (log)	99
6.	La mise en place HTTPS	102
7.	Conclusion	103

## Chapitre 4 MongoDB

1.	Qu'est-ce que MongoDB ?	105
2.	Pourquoi utiliser un SGBD NoSQL et MongoDB ?	106
3.	Les fonctionnalités de MongoDB	108
4.	Les collections et les documents	110
4.1	Les collections	110
4.2	Les documents	111
5.	Les index	112
6.	L'installation de MongoDB	113
6.1	Sous Ubuntu	113
6.2	Sous Windows	114
6.3	Sous Mac	117
7.	L'utilisation de MongoDB en ligne de commande	119
8.	La création et l'insertion de documents dans une collection	121
9.	L'importation et l'exportation de documents	122

10. L'interrogation d'une collection . . . . .	123
10.1 Les différentes méthodes d'interrogation . . . . .	123
10.2 La recherche des documents selon certains critères . . . . .	124
10.3 Les opérateurs de comparaison, logiques et ensemblistes . . . . .	125
11. La recherche des documents grâce aux expressions régulières . . . . .	128
12. Le référencement des documents et jointures . . . . .	129
12.1 Les différentes méthodes de référencement des documents . . . . .	129
12.2 Les objets imbriqués . . . . .	131
12.3 Les jointures . . . . .	133
13. La mise à jour et la suppression d'un document . . . . .	134
14. Conclusion . . . . .	136

## Chapitre 5

### Connexion entre MongoDB et Node.js

1. Implémenter le CRUD avec Node.js et MongoDB . . . . .	137
2. Mongoose . . . . .	139
3. La création d'un projet de test . . . . .	141
3.1 La création d'une application avec Node.js et MongoDB . . . . .	141
3.2 La connexion au serveur MongoDB depuis Node.js . . . . .	143
3.3 L'insertion et l'interrogation de données à partir d'un serveur Node.js . . . . .	145
3.4 Les méthodes find() et toArray() . . . . .	146
3.5 La synchronisation des requêtes . . . . .	147
3.6 Les fonctions de callback et le module async . . . . .	148
4. L'interrogation de MongoDB via les routes gérées par Express . . . . .	149
4.1 La structure d'un serveur Node.js interrogeant MongoDB . . . . .	149
4.2 Le CORS . . . . .	150
4.3 Les exemples de gestion de routes . . . . .	150
4.4 La création d'une collection de produits et la recherche sur les produits . . . . .	152
5. Conclusion . . . . .	154

# 6 Angular et Node.js

Développement web full stack avec MEAN

## Chapitre 6 Angular

1. Angular, un framework applicatif moderne . . . . .	155
1.1 Angular, une évolution radicale par rapport à AngularJS . . . . .	156
1.2 La modularité des applications Angular . . . . .	157
1.2.1 Des modules pour organiser le code . . . . .	158
1.2.2 Des composants et des services pour créer des fonctionnalités . . . . .	160
1.3 Manipuler les composants comme des balises HTML . . . . .	161
1.4 Développement d'applications Angular en TypeScript . . . . .	162
2. La mise en place d'une application Angular . . . . .	163
2.1 L'installation d'Angular CLI . . . . .	163
2.2 La création du projet Angular avec Angular CLI . . . . .	164
2.3 La structure des dossiers du projet Angular . . . . .	165
2.4 La mise à jour du projet Angular via Angular CLI . . . . .	166
2.5 Générer du code JavaScript dans une application Angular . . . . .	168
3. Les décorateurs en Angular : une fonctionnalité essentielle pour améliorer les composants . . . . .	169
4. La création d'un nouveau composant . . . . .	169
4.1 Créer un premier composant avec Angular . . . . .	169
4.2 Le composant racine de l'application Angular . . . . .	171
4.3 Intégrer un nouveau composant dans le composant racine de l'application Angular . . . . .	171
5. Le cycle de vie d'un composant dans Angular . . . . .	172
5.1 Les principaux hooks . . . . .	172
5.1.1 ngOnChanges() . . . . .	172
5.1.2 ngOnInit() . . . . .	174
5.1.3 ngDoCheck() . . . . .	176
5.1.4 ngOnDestroy() . . . . .	177

5.2	Encore plus de hooks	179
5.2.1	Les hooks personnalisés	179
5.2.2	Les hooks de test	179
5.2.3	Les hooks de validation	179
5.2.4	Les hooks de sécurité	179
5.2.5	Les hooks de performance	180
6.	Les templates	180
6.1	L'imbrication des templates	181
6.2	Les templates externalisés	182
7.	Data bindings entre le composant et le template	184
7.1	Accès aux éléments du DOM	184
7.2	Interpolation d'une variable dans un template	186
7.3	Property binding	187
7.4	Event binding	188
7.5	Two-way data binding	189
8.	Les directives	190
8.1	Les directives structurelles	190
8.1.1	La directive ngFor	191
8.1.2	La directive ngIf	193
8.2	Les directives attributs	194
8.2.1	La directive ngClass	195
8.2.2	La directive ngStyle	197
9.	Conclusion	198

# 8 \_\_\_\_\_ Angular et Node.js

Développement web full stack avec MEAN

## Chapitre 7

### Intégration d'un serveur Node.js avec Angular

1. La notion de client-serveur entre Angular et Node.js. . . . . 201
2. Le module HttpClientModule. . . . . 202
3. La mise en place d'un environnement de test  
entre Angular et Node.js . . . . . 204
  - 3.1 La création d'un projet Node.js simple . . . . . 204
  - 3.2 La création de la base de l'application Angular . . . . . 208
    - 3.2.1 L'intégration du module HttpClientModule. . . . . 209
    - 3.2.2 Le requêtage du serveur Node.js. . . . . 210
4. Conclusion . . . . . 212

## Chapitre 8

### Angular : Routage

1. À quoi sert le routage ? . . . . . 213
2. La différence entre routage Node.js et routage Angular . . . . . 214
3. La mise en place d'un routage . . . . . 215
  - 3.1 Le tag <base> . . . . . 216
  - 3.2 Le module @angular/router . . . . . 217
  - 3.3 La création d'une route de navigation. . . . . 218
  - 3.4 La directive <router-outlet>. . . . . 220
4. Le lazy loading . . . . . 221
  - 4.1 L'intérêt d'un lazy loading. . . . . 221
  - 4.2 La mise en place du lazy loading . . . . . 223
  - 4.3 La stratégie de préchargement (preloading strategy) . . . . . 224
  - 4.4 La mise en place de la stratégie de préchargement . . . . . 224
5. Les route guards : la police du routage . . . . . 225
  - 5.1 Mise en garde . . . . . 226
  - 5.2 La méthode canActivate. . . . . 227
  - 5.3 La méthode canDeactivate. . . . . 228

6. Conclusion .....	229
---------------------	-----

## Chapitre 9

### Tests et mise en production

1. Tests Angular .....	231
1.1 Le framework de test Jasmine .....	232
1.2 Les différents types de tests .....	233
1.2.1 La couverture de code (code coverage) .....	234
1.2.2 Tester un service Angular .....	236
1.2.3 Tester un composant Angular .....	237
1.3 La création d'un test avec Jasmine .....	238
2. Tests Node.js. ....	239
2.1 Le framework de test Jest. ....	240
2.2 La création d'un test avec Jest .....	241
3. L'introduction à l'approche TDD .....	243
3.1 Explication et exemple .....	244
3.2 Quelques recommandations .....	247
4. Le déploiement .....	248
4.1 La commande CLI ng build .....	249
4.2 La mise en production d'une application Node.js .....	250
4.3 L'hébergement sur Apache .....	252
5. Conclusion .....	254

Conclusion .....	255
------------------	-----

Index .....	257
-------------	-----