

Les éléments à télécharger sont disponibles à l'adresse suivante :  
**<http://www.editions-eni.fr>**  
Saisissez la référence de l'ouvrage **EP3JASP** dans la zone de recherche  
et validez. Cliquez sur le titre du livre puis sur le bouton de téléchargement.

## Avant-propos

1. Préambule .....	23
2. Introduction .....	25
3. Organisation du livre .....	27
4. Public visé .....	28
5. Pourquoi Spring .....	29
6. Prérequis pour aborder Spring et Java EE .....	29
7. Objectifs du livre .....	31

## Chapitre 1

### Éléments constitutifs du framework

1. Introduction .....	33
2. Historique .....	33
3. Utilité de Spring dans un projet .....	37
4. Vue d'ensemble et sujets abordés dans l'ouvrage .....	41
4.1 Les modules Spring .....	41
4.2 Sujets abordés .....	43
4.3 Version des briques Spring utilisées dans l'ouvrage .....	43
4.4 Version des ressources externes .....	44
4.5 Gestion des dépendances Maven .....	44
4.6 Utilisation d'un BOM (Bill Of Materials) Maven .....	46
5. Compléments .....	49
6. Points clés .....	49

## Chapitre 2 Spring et les design patterns

1. Introduction . . . . .	51
2. Le singleton en Java . . . . .	52
2.1 Préoccupation . . . . .	53
2.2 Solution Spring . . . . .	54
2.3 Exemple . . . . .	55
3. L'inversion de contrôle . . . . .	57
4. Façade . . . . .	59
5. Fabrique . . . . .	60
6. Active record . . . . .	61
7. Décorateur . . . . .	62
8. Proxy . . . . .	62
9. Modèle Vue Contrôleur (MVC) . . . . .	64
10. Le contrôleur de partie frontale (Front Controller) . . . . .	65
11. Les assistants de vue (View Helper) . . . . .	66
12. Les templates . . . . .	67
13. Stratégie . . . . .	68
14. Points clés . . . . .	69

## Chapitre 3 Rappels sur les éléments externes à Spring

1. Codage equals et hashCode . . . . .	71
1.1 Description de la problématique . . . . .	74
1.2 Mise en œuvre . . . . .	75
2. Projet Lombok . . . . .	77
3. Les systèmes de log . . . . .	79

4. Bases de données H2 . . . . .	81
4.1 Description de la problématique . . . . .	81
4.2 Mise en œuvre . . . . .	82
4.2.1 Installation . . . . .	82
4.2.2 Configuration du POM. . . . .	82
4.2.3 Connexion à la base de données en utilisant JDBC . . . . .	82
4.2.4 Utiliser un listener de servlet pour démarrer et arrêter la base . . . . .	83
5. Projets Maven . . . . .	84
5.1 Description de la problématique . . . . .	85
5.2 Mise en œuvre . . . . .	85
5.3 Installation de Maven sous Windows . . . . .	86
5.4 Utilisation d'un archétype Maven . . . . .	87
5.5 Contenu du fichier pom.xml dans les cas simples utilisés dans ce livre . . . . .	88
6. Rappels Java 8 et 9 . . . . .	89
6.1 Nouveauté Java 8+ . . . . .	89
6.2 Le langage de programmation . . . . .	91
6.3 Collections . . . . .	91
6.4 Sécurité . . . . .	92
6.5 Les dates . . . . .	92
6.6 Dates locales . . . . .	92
6.7 Concurrency . . . . .	93
6.8 Optional . . . . .	93
6.9 Parallel arrays . . . . .	94
6.10 Lambda . . . . .	95
6.11 L'API Stream . . . . .	96
7. Points clés . . . . .	97

## Chapitre 4 Le conteneur Spring

1. Introduction . . . . .	99
2. Les origines . . . . .	99
3. Les modules fondamentaux . . . . .	101
3.1 Composition d'un bean . . . . .	101
3.2 Le singleton et le prototype . . . . .	102
3.3 Les objets fondamentaux du package core . . . . .	102
3.3.1 Le PropertyEditor . . . . .	102
3.4 Les PropertyValue . . . . .	105
3.5 Le BranWrapper . . . . .	105
3.6 La BeanFactory . . . . .	105
3.7 L'interface BeanDefinition . . . . .	106
3.8 Le PropertyPlaceholderConfigurer . . . . .	107
3.9 Les objets fondamentaux du package context . . . . .	109
3.10 Relations entre le Bean, sa BeanFactory ou son Contexte . . . . .	110
4. Configuration des beans . . . . .	111
4.1 Configuration par un fichier XML . . . . .	112
4.2 Configuration avec des annotations . . . . .	113
4.3 Configurer les beans applicatifs implicitement . . . . .	115
4.4 Configuration par lambdas . . . . .	116
5. Utilisation des beans : injection par setters et constructeurs . . . . .	117
5.1 Mapping via le constructeur en XML . . . . .	118
5.2 Comparaison des méthodes d'instanciation . . . . .	119
5.3 Autres annotations de configuration . . . . .	120
5.3.1 L'annotation @Primary . . . . .	120
5.3.2 L'annotation @Profile et @Conditionnal . . . . .	121
6. Contrôle du cycle de vie : construction et destruction . . . . .	131
7. Exemple illustrant les mappings standards . . . . .	132
7.1 Le projet Maven . . . . .	132
7.2 Fichier de configuration de Spring . . . . .	132

7.3	Dépendance Spring Core et système de log . . . . .	133
7.4	Dépendance vers les bibliothèques de tests unitaires. . . . .	134
7.5	Exemple illustrant l'utilisation des logs. . . . .	135
7.6	Fichier de configuration spécifique pour les tests. . . . .	135
8.	Le Spring Expression Language . . . . .	137
8.1	Utilisation de l'ExpressionParser . . . . .	137
8.2	L'EvaluationContext . . . . .	138
8.3	Utilisation avec @Value . . . . .	139
9.	Serveur J2EE, Java EE et Jakarta EE . . . . .	140
9.1	Généralités . . . . .	140
9.2	Web services . . . . .	146
9.2.1	Via une servlet. . . . .	146
9.2.2	Serveur intégré . . . . .	146
9.3	Les EJB . . . . .	149
9.3.1	Fonctionnement . . . . .	149
9.3.2	Création de l'EJB distant. . . . .	152
9.3.3	Configuration de Maven côté client . . . . .	153
9.3.4	Création du client . . . . .	154
9.3.5	Intégration plus poussée. . . . .	155
9.3.6	Le design pattern Business Delegate . . . . .	157
10.	Points clés . . . . .	161

## Chapitre 5

### Configuration avancée

1.	Introduction . . . . .	163
2.	Les fichiers de ressources . . . . .	163
2.1	Les fichiers de ressources standards. . . . .	164
2.2	Les fichiers de ressources chargés via le contexte Spring . . . . .	165

# 6 --- Java Spring

Le socle technique des applications Java EE

2.3	Les fichiers de ressources chargés	
	via un service ResourceLoaderAware .....	166
2.3.1	Le programme principal .....	166
2.3.2	Le service ResourceLoaderAware .....	167
3.	Convertisseurs et formateurs .....	168
3.1	Built-in converters .....	168
3.1.1	Pour les types standards .....	168
3.1.2	Pour les tableaux et les listes .....	169
3.1.3	Pour les énumérations .....	170
3.1.4	Pour les objets en général .....	170
3.2	Convertir un Array en Collection et String .....	172
3.3	Convert Factory .....	173
3.3.1	Classe StringToUtilisateurConverterFactory .....	173
3.3.2	Classe UtilisateurToStringConverterFactory .....	173
3.3.3	Classe LesConversions5 .....	174
3.4	Les formateurs par défaut .....	174
3.4.1	Classe LesConversions6 .....	174
3.5	Les formateurs personnalisés .....	175
3.5.1	Classe CarteDeCredit .....	175
3.5.2	Classe CarteDeCreditParser .....	176
3.5.3	Classe CarteDeCreditPrinter .....	176
3.5.4	Classe CarteDeCreditFormatter .....	177
3.5.5	Classe LesConversions7 .....	177
4.	Les BeanWrappers, le binding et les validateurs .....	178
4.1	Classe LesBeanWrappers .....	178
4.2	Classe UtilisateurValidator .....	179
5.	Points clés .....	180

**Chapitre 6****Programmation orientée aspect avec Spring**

1. Introduction . . . . .	181
2. Pourquoi l'AOP . . . . .	187
3. Les concepts d'AOP . . . . .	188
4. Limites de Spring AOP et utilisation d'AspectJ . . . . .	190
5. Le support @AspectJ dans Spring . . . . .	190
5.1 Activation du support . . . . .	191
5.2 Activation de @AspectJ avec configuration XML . . . . .	191
5.3 Activation de @AspectJ avec configuration Java . . . . .	192
5.4 Déclaration d'un aspect . . . . .	192
5.5 Déclaration d'un pointcut . . . . .	193
5.6 Déclaration des greffons simples . . . . .	195
5.7 Les types génériques . . . . .	197
5.8 Détermination de noms d'arguments . . . . .	198
5.9 Ordre d'appel des greffons interceptant le point de jonction . . . . .	198
5.10 Injection . . . . .	199
5.11 Le mécanisme des proxies . . . . .	199
5.12 La face cachée des proxies en AOP . . . . .	200
6. Les API bas niveau pour Spring AOP . . . . .	203
6.1 L'interface PointCut . . . . .	203
6.2 L'interface ClassFilter . . . . .	203
6.3 L'interface MethodMatcher . . . . .	204
7. Points clés . . . . .	204

## Chapitre 7 Les tests et Spring

1. Introduction .....	205
2. Les mock objects .....	206
2.1 Mocks spécialisés pour "environnement" .....	207
2.2 Support .....	207
2.2.1 Utilités générales .....	207
2.2.2 Spring MVC .....	208
2.3 Tests d'intégration .....	209
2.3.1 Vue d'ensemble .....	209
2.3.2 Mise en cache du contexte de test .....	209
2.3.3 Tests back et front .....	210
2.4 Annotations .....	211
2.4.1 @ContextConfiguration .....	211
2.4.2 @WebAppConfiguration .....	212
2.4.3 @ContextHierarchy .....	212
2.4.4 @ActiveProfiles .....	213
2.4.5 @TestPropertySource .....	213
2.4.6 @DirtiesContext .....	214
2.4.7 Interface TestExecutionListener .....	215
2.4.8 @TransactionConfiguration .....	216
2.4.9 @Transactional .....	216
2.4.10 @Rollback .....	217
2.4.11 @BeforeTransaction .....	217
2.4.12 @AfterTransaction .....	217
2.4.13 @Sql, @SqlConfig et @SqlGroup .....	217
2.5 Annotations standards .....	218
3. Les frameworks de test JUnit et testNG .....	219
3.1 Utilisation avec JUnit .....	219
3.1.1 Spring JUnit Runner .....	219
3.1.2 @IfProfileValue .....	219
3.1.3 @ProfileValueSourceConfiguration .....	220

3.1.4	@Timed . . . . .	220
3.1.5	@Repeat . . . . .	221
3.1.6	Méta-annotations de support pour les tests . . . . .	221
3.2	Les arcanes du framework Spring TestContext . . . . .	225
3.2.1	Classes et interfaces du framework de tests . . . . .	225
3.2.2	Configuration du TestExecutionListener par les annotations . . . . .	227
3.2.3	Les TestExecutionListeners . . . . .	228
3.2.4	Mélange XML et classes annotées . . . . .	232
3.3	Configuration des contextes de tests . . . . .	232
3.3.1	Configuration de contexte avec des initialiseurs de contexte . . . . .	232
3.3.2	L'héritage dans la configuration des contextes . . . . .	233
3.3.3	Prise en compte des profils d'environnement . . . . .	234
3.3.4	Configuration de contexte avec des fichiers de propriétés de test . . . . .	239
3.3.5	Déclarer un fichier de propriétés pour les tests . . . . .	239
3.3.6	Détection du fichier de propriétés par défaut . . . . .	240
3.4	Hiérarchie des contextes de tests. . . . .	241
3.4.1	Héritage et surcharge des propriétés de test . . . . .	241
3.4.2	Chargement d'un WebApplicationContext . . . . .	242
3.4.3	Conventions . . . . .	242
3.4.4	La sémantique de ressources explicites . . . . .	243
3.4.5	Injecter des objets mockés . . . . .	244
3.4.6	Cachabilité du contexte de test . . . . .	244
3.4.7	Les hiérarchies des contextes . . . . .	246
3.4.8	L'injection de dépendance dans les tests . . . . .	249
3.5	Le scope session lors d'un test de requête . . . . .	251
3.5.1	Le scope session lors d'un test de requête . . . . .	251
3.5.2	Test de beans de scope request. . . . .	252
3.5.3	Configuration d'un bean de scope session. . . . .	252

3.6	Les transactions . . . . .	253
3.6.1	La gestion des transactions. . . . .	253
3.6.2	Les transactions managées par le test . . . . .	254
3.6.3	Activation et désactivation des transactions . . . . .	254
3.6.4	Comportement du commit et du rollback d'une transaction . . . . .	256
3.6.5	Exécution de code en dehors d'une transaction . . . . .	257
3.6.6	Configuration d'un gestionnaire de transactions . . . . .	257
3.6.7	Démonstration de toutes les annotations liées à la transaction . . . . .	258
3.7	Les scripts SQL . . . . .	259
3.7.1	Exécuter des scripts SQL . . . . .	259
3.7.2	La sémantique du path des ressources. . . . .	262
3.7.3	La détection de script par défaut . . . . .	262
3.7.4	Déclaration de plusieurs @Sql . . . . .	263
3.7.5	Phases d'exécution pour les scripts . . . . .	264
3.7.6	Script de configuration avec SqlConfig . . . . .	264
3.7.7	La gestion des transactions pour @Sql . . . . .	265
3.8	Classes de support TestContext . . . . .	266
3.8.1	Classes de support de junit . . . . .	266
3.8.2	Spring JUnit Runner . . . . .	268
3.8.3	Classes de support TestNG . . . . .	268
3.9	Framework Spring MVC Test . . . . .	270
3.9.1	Projet autonome . . . . .	270
3.9.2	Tests côté serveur . . . . .	271
3.9.3	Définir les attendus. . . . .	276
3.9.4	Ajouts de filtres. . . . .	278
3.9.5	Tests REST côté client . . . . .	278
3.10	Autres ressources . . . . .	279
4.	Points clés . . . . .	280

**Chapitre 8**  
**Partie back de l'application**

- 1. Description de la problématique ..... 281
- 2. Mise en œuvre. .... 284
  - 2.1 Configuration d'une entité de la couche domaine ..... 284
  - 2.2 Accès à l'objet du domaine ..... 286
  - 2.3 Requêtes typées ..... 287
  - 2.4 Cache de niveaux 1 et 2 ..... 287
  - 2.5 Bean Validation (JSR-303) ..... 288
  - 2.6 L'API Criteria. .... 291
  - 2.7 Accès à la base de données ..... 291
  - 2.8 Le fichier persistence.xml ..... 292
  - 2.9 Tests JPA. .... 293
- 3. Points clés ..... 295

**Chapitre 9**  
**Spring dans un contexte web JSP**

- 1. Spring MVC ..... 297
  - 1.1 Fonctionnement global ..... 298
    - 1.1.1 Configuration simple XML ..... 298
    - 1.1.2 Configuration par annotation ..... 302
  - 1.2 Éléments complexes du contrôleur ..... 303
    - 1.2.1 Fonctionnement global du contrôleur ..... 303
    - 1.2.2 Exemple de la classe Simple ..... 304
    - 1.2.3 Simple revisité ..... 305
    - 1.2.4 Par le path ..... 305
    - 1.2.5 Par un motif sur le path ..... 306
    - 1.2.6 Par le path et une méthode. .... 307
    - 1.2.7 Mappé par path + méthode + présence de paramètres de query ..... 307
    - 1.2.8 Mappé par path + méthode + présence d'un header. . 308

1.2.9	Par l'absence d'un header	308
1.2.10	Par consommation	309
1.2.11	Par production via <code>Accept=application/json</code>	310
1.2.12	Par production via <code>Accept=application/xml</code>	310
1.2.13	Paramètres de query	311
1.2.14	Groupes de paramètres de query	312
1.2.15	Variable du path	313
1.2.16	Variable Matrix	313
1.2.17	Variables Matrix multiples	314
1.2.18	Corps de la requête	315
1.2.19	En-tête et corps de la requête	315
1.2.20	Arguments dans la requête	316
1.2.21	Reader sur la requête	317
1.2.22	InputStream de la requête	318
1.2.23	Arguments de la réponse	318
1.2.24	Writer de la réponse	319
1.2.25	OutputStream de la réponse	320
1.2.26	Session	320
1.2.27	Handler personnalisé	321
1.2.28	Utiliser l'annotation <code>@ResponseBody</code> seule	322
1.2.29	Utiliser l'annotation <code>@ResponseBody</code> avec UTF-8 requis	323
1.2.30	Utiliser l'annotation <code>@ResponseBody</code> avec UTF-8 produit	323
1.2.31	Réponse avec un statut personnalisé	324
1.2.32	Réponse avec un en-tête personnalisé	325
1.2.33	Obtention d'un cookie	325
1.2.34	Lire une chaîne de caractères	326
1.2.35	Écrire une chaîne de caractères	326
1.2.36	Lire les données encodées dans l'URL	327
1.2.37	Retourner une <code>MultiValueMap</code>	327
1.2.38	Lire une structure XML	328

- 1.2.39 Écrire dans une structure XML  
via Accept=application/xml ..... 329
- 1.2.40 Lire une structure JSON ..... 329
- 1.2.41 Écrire dans une structure JSON  
via Accept=application/json ..... 330
- 1.2.42 HTML généré par un template JSP ..... 331
- 1.2.43 Mapping à partir d'un modèle ..... 332
- 1.2.44 Utiliser des variables dans un template de vue ..... 332
- 1.2.45 Data binding avec des variables d'URI ..... 333
- 1.2.46 Types primitifs ..... 333
- 1.2.47 Date ..... 336
- 1.2.48 Conversion des collections ..... 337
- 1.2.49 Utiliser des collections formatées ..... 338
- 1.2.50 Utiliser des objets personnalisés ..... 338
- 1.2.51 Utiliser un convertisseur personnalisé ..... 339
- 1.2.52 Validation ..... 343
- 1.2.53 @ExceptionHandler dans un contrôleur ..... 344
- 1.2.54 @ExceptionHandler global ..... 344
- 1.2.55 Templates de String pour les URI ..... 344
- 1.2.56 UriComponentsBuilder ..... 345
- 2. Client REST ..... 346
  - 2.1 Utilisation du RestTemplate ..... 346
  - 2.2 Le bean de domaine avec l'annotation REST  
pour les champs manquants ..... 347
- 3. Spring Security ..... 347
  - 3.1 Introduction à Spring Security ..... 347
  - 3.2 Spring Security dans un environnement web ..... 350
    - 3.2.1 Authentification par Spring ..... 350
    - 3.2.2 Authentification par page de login personnalisée ..... 351
    - 3.2.3 Authentification par base de données ..... 351
- 4. Points clés ..... 353

## Chapitre 10 Intégration JSF2

1. Introduction .....	355
2. Mojarra .....	357
3. Architecture .....	358
4. Cycle de vie .....	360
4.1 Requête .....	360
4.2 Restore View ou Reconstruct Component Tree .....	360
4.3 Apply Request Value .....	361
4.4 Perform Validation .....	361
4.5 Synchronize Model ou Update Model Values .....	361
4.6 Invoke Application Logic .....	362
4.7 Render Response .....	362
4.8 Réponse .....	362
4.9 Fichier web.xml .....	362
4.10 Dépendances .....	363
4.11 Fichier faces-config.xml .....	363
4.12 Bean managé sans Spring .....	365
4.13 Exemple de vue JSF .....	365
4.14 Aperçu d'un composant JSF .....	366
5. Intégration Spring .....	367
5.1 Architecture .....	368
5.2 Dépendances .....	368
5.3 Fichier web.xml .....	368
5.4 Fichier faces-config.xml .....	369
5.5 Couches basses (back) .....	370
6. Points clés .....	370

**Chapitre 11**  
**Application Spring Angular**

- 1. Introduction ..... 371
- 2. La partie backend ..... 372
  - 2.1 Génération d'un backend ..... 372
- 3. La partie frontend ..... 375
  - 3.1 Angular CLI ..... 375
  - 3.2 Création du projet initial ..... 376
  - 3.3 Démarrage de l'application ..... 376
  - 3.4 Créer un service Compte ..... 377
- 4. Points clés ..... 379

**Chapitre 12**  
**Spring-HATEOAS**

- 1. Introduction ..... 381
  - 1.1 Exemple de liens hypermédias codés manuellement ..... 384
  - 1.2 Aller plus loin avec l'hypermédia ..... 386
  - 1.3 Autoconfiguration par annotation ..... 387
  - 1.4 Fournisseurs de relations ..... 387
  - 1.5 Fournisseur d'URI compacte ..... 388
  - 1.6 Support du côté client ..... 389
  - 1.7 Découverte de liens coté client ..... 390
  - 1.8 Utilisation du @RepositoryRestResource ..... 390
- 2. Points clés ..... 400

## Chapitre 13 Documentation Spring REST Docs

1. Introduction . . . . .	401
1.1 Dépendance sur la bibliothèque de tests . . . . .	404
1.2 Dépendance sur les plug-ins Maven . . . . .	404
1.3 Les extraits (snippets) . . . . .	406
2. Exemple JUnit 5 . . . . .	406
3. Requête et réponse . . . . .	408
4. Réponse avec un JSON imbriqué . . . . .	409
4.1 Les paramètres de requête . . . . .	410
4.2 Les paramètres inclus dans le path . . . . .	410
4.3 Les Request parts . . . . .	410
4.4 Les Request parts payloads . . . . .	411
4.5 Les champs . . . . .	411
4.6 Liens hypermédias dans la réponse . . . . .	411
4.7 Les en-têtes . . . . .	412
5. Personnalisation de la documentation . . . . .	413
5.1 Utilisation de <code>@AutoConfigureRestDocs</code> . . . . .	413
6. Couplage Swagger 2 . . . . .	413
6.1 Utilisation Springfox . . . . .	414
6.2 En dehors de Spring Boot . . . . .	415
7. Utilisation avec Spring Data Rest . . . . .	416
8. Récapitulatif sur la documentation générée . . . . .	417
9. Points clés . . . . .	417

**Chapitre 14**  
**Spring Boot**

- 1. Introduction ..... 419
- 2. Configuration des exemples..... 420
  - 2.1 Configuration Maven pour la version 1.5 de Spring Boot ... 420
  - 2.2 Configuration Maven pour la version 2 de Spring Boot .... 421
  - 2.3 Utilisation du hot swapping ..... 421
  - 2.4 Packaging et lancement de l'application..... 422
  - 2.5 Application Spring MVC minimum ..... 422
- 3. L'auto-configuration Spring Boot ..... 423
  - 3.1 La configuration automatique des Beans Spring ..... 423
- 4. Les starters ..... 425
  - 4.1 Les starters courants..... 425
  - 4.2 Les starters orientés messages ..... 428
  - 4.3 Les bases de données..... 428
  - 4.4 Les web services ..... 429
  - 4.5 Les moteurs de rendus ..... 429
  - 4.6 Les starters moins courants ..... 430
- 5. Spring MVC ..... 431
- 6. Personnalisation de la bannière ..... 432
- 7. Événements d'applications..... 432
- 8. La récupération des arguments de la ligne de commande ..... 433
  - 8.1 CommandLineRunner ..... 433
  - 8.2 ApplicationRunner ..... 434
  - 8.3 La configuration yaml et profiles..... 434
- 9. La configuration des logs ..... 435
  - 9.1 Les logs en couleur ..... 436
  - 9.2 Choix du type de logs..... 440
- 10. L'auto-configuration pour Spring MVC ..... 440
- 11. La gestion des sessions ..... 441

12. Le guide de migration de la version 1.5 à la version 2.0 . . . . .	443
12.1 Fichiers de configuration . . . . .	443
12.2 Comportements différents . . . . .	443
12.3 Démarrage. . . . .	443
12.4 Utilisation de ApplicationRunner ou CommandLineRunner .	444
12.5 Configuration externalisée. . . . .	444
12.6 Développement d'applications web . . . . .	445
12.6.1 Web Starter en tant que dépendance transitive . . . . .	445
12.7 Sécurité . . . . .	445
13. Points clés . . . . .	446

## Chapitre 15 Spring et le noSQL

1. Introduction . . . . .	447
2. Les modèles de données . . . . .	448
2.1 Modèle Clé-valeur. . . . .	449
2.2 Modèle Documents . . . . .	449
2.3 Modèle Orienté colonnes. . . . .	449
2.4 Les bases orientées Graph . . . . .	450
3. Principes des bases de données . . . . .	450
3.1 Consistance des données . . . . .	450
4. Pourquoi et quand utiliser une base noSQL . . . . .	453
5. Problèmes liés à l'utilisation des bases NoSQL. . . . .	455
6. Limitations des bases de données noSQL . . . . .	456
7. Spring et le noSQL . . . . .	456
8. Cache de données . . . . .	457
8.1 Cache simple. . . . .	458
9. Cacher des données avec GemFire. . . . .	460
10. GemFire en tant que base de données noSQL . . . . .	462

11. Redis en autonome . . . . .	464
11.1 Utilisation de Redis pour le cache de données . . . . .	464
11.2 Utilisation de Redis pour gérer des messages . . . . .	466
12. MongoDB . . . . .	469
12.1 MongoDB avec Spring Boot . . . . .	469
12.2 MongoDB avec une API REST . . . . .	471
13. Points clés . . . . .	476

## Chapitre 16 Spring Batch

1. Introduction . . . . .	477
2. Architecture d'un batch . . . . .	481
3. Exemple Spring Batch version 4 . . . . .	482
4. Chargeur H2 depuis un CSV . . . . .	483
5. Dépendances Spring Batch 3 et 4 . . . . .	487
6. Autres nouveautés de la version 4 . . . . .	488
7. Points clés . . . . .	489

## Chapitre 17 Les intergiciels à messages (MOM)

1. Introduction . . . . .	491
2. Implémentations open source . . . . .	493
3. Implémentations propriétaires . . . . .	494
4. Les cas d'utilisation . . . . .	494
5. JMS et ActiveMQ . . . . .	497
5.1 Utilisation simplifiée . . . . .	498

6. RabbitMQ.....	501
6.1 Spring AMQP et RabbitMQ .....	501
6.2 Exemple RabbitMQ .....	503
7. Points clés .....	505

## Chapitre 18

### Spring et Kotlin

1. Introduction .....	507
2. Caractéristiques principales du langage Kotlin .....	511
2.1 Les méthodes et les fonctions .....	511
2.2 L’immuabilité des objets.....	511
2.3 Les types .....	512
2.4 La gestion des valeurs nulles .....	515
2.5 Les appels chaînés sécurisés .....	516
2.6 Les lambdas.....	517
3. Contrôleur Spring MVC, Spring Boot en Kotlin .....	517
3.1 Fonction principale.....	518
3.2 Test associé à la fonction principale .....	518
4. Les plug-ins.....	524
5. Points clés .....	525

## Chapitre 19

### Introduction à Spring Reactor et Spring Webflux

1. Introduction .....	527
2. Spring Reactor .....	528
2.1 Présentation .....	528
2.2 Utilisation de Reactor Core .....	532
2.2.1 Les threads.....	532
2.2.2 Les FutureTask .....	532
2.2.3 Le CompletableFuture .....	533

2.2.4	Flux et Mono	534
2.3	Les tests	538
2.3.1	Vérifications avec StepVerifier	538
2.3.2	Emissions manuelles avec TestPublisher	541
3.	WebFlux	543
3.1	Définition du terme réactif	544
3.2	Les couches réactives	545
3.2.1	la couche contrôleur	545
3.2.2	La couche des services	549
3.2.3	La couche repository	550
3.2.4	Repository réactif R2DBC	550
4.	Client réactif	555
5.	Les tests avec WebFlux	556
5.1	Tests unitaires	556
5.1.1	Tests unitaires avec des applications réactives	556
5.1.2	Utilisation de MockWebServer	557
5.2	Tests d'intégration	559
5.2.1	Utilisation de @WebFluxTest avec WebTestClient	559
6.	Pour aller plus loin	562
7.	Points clés	562

## Chapitre 20

### Spring et JHipster

1.	Introduction	563
2.	Généralités	564
2.1	JHipster web	565
2.2	Personnalisation de l'outil JHipster	570
2.3	Niveaux de personnalisations	571
3.	Créer un blueprint	576
3.1	Blueprint pour utiliser lombok dans le domain	577
3.2	Exécution du blueprint local	581

4.	JHipster comme outil multitechnologie . . . . .	582
4.1	Côté client . . . . .	582
4.2	Côté serveur . . . . .	584
4.3	Côté déploiement . . . . .	586
4.4	Côté Spring . . . . .	587
5.	Structure du projet . . . . .	588
5.1	La partie front . . . . .	588
5.2	Les grandes lignes de la partie back . . . . .	588
5.2.1	Spring Boot . . . . .	589
5.2.2	La classe serveur HTTP/servlet . . . . .	595
6.	JHipster et WebFlux . . . . .	607
6.1	Configuration DatabaseConfiguration . . . . .	608
6.2	Configuration DateTimeFormatConfiguration . . . . .	609
6.3	Configuration LocaleConfiguration . . . . .	609
6.4	Configuration ReactorConfiguration . . . . .	609
6.5	Configuration SecurityConfiguration . . . . .	610
6.6	Configuration WebConfigurer . . . . .	610
6.7	Les tests . . . . .	610
7.	Points clés . . . . .	610
	Index . . . . .	611