



Collection

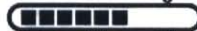
EXPERT IT

Zend Framework et PHP

Programmation par composants

Christophe HARO

Fichiers à télécharger



www.editions-eni.fr

eni
Editions

INFORMATIQUE TECHNIQUE

Les exemples cités tout au long de cet ouvrage sont téléchargeables
à l'adresse suivante : <http://www.editions-eni.fr>
Saisissez la référence ENI de l'ouvrage **EIZEN** dans la zone de recherche
et validez. Cliquez sur le titre du livre puis sur le lien de téléchargement.

Chapitre 1

Introduction

| | |
|---|----|
| 1. Introduction | 9 |
| 2. Pré-requis | 11 |
| 3. Structure du livre | 13 |
| 4. Conventions adoptées | 15 |
| 5. Bibliographie et webographie | 17 |

Chapitre 2

Installations

| | |
|--|----|
| 1. Introduction | 19 |
| 2. Installation du Zend Framework | 19 |
| 2.1 Téléchargement | 20 |
| 2.2 Ajustements de la configuration | 20 |
| 3. Installations et configurations complémentaires | 21 |
| 4. Installer PHP Data Object (PDO) et le pilote du SGBDR | 24 |
| 4.1 Présentation du problème | 24 |
| 4.2 Installation de PHP Data Object | 25 |
| 5. Installer PEAR::HTML_Table et PEAR::HTML_Common | 26 |
| 6. Bibliographie et webographie | 33 |
| 7. Conclusion | 34 |

Chapitre 3

Architecture d'une application et le Zend Framework

| | |
|---|----|
| 1. Introduction | 35 |
| 2. Qualité logicielle et architecture | 36 |
| 2.1 Modèle en couches | 37 |
| 2.2 Internet et l'architecture client/serveur | 39 |

| | | |
|-----|---|----|
| 2.3 | Architecture en couches et application Web : le patron de conception Modèle-Vue-Contrôleur (MVC) | 40 |
| 3. | Cadre d'application (framework) et architectures | 42 |
| 3.1 | Introduction au framework de Zend | 42 |
| 3.2 | Le Zend Framework et ses composants | 42 |
| 4. | Bibliographie et webographie | 45 |
| 5. | Conclusion | 45 |

Chapitre 4

Configuration d'une application objet

| | | |
|-------|---|----|
| 1. | Introduction | 47 |
| 2. | Préparer l'environnement applicatif | 48 |
| 2.1 | Comprendre le chargement automatique des classes | 49 |
| 2.2 | Distribution des fichiers et conventions de désignation | 52 |
| 2.3 | Chargement automatique des classes : Zend_Loader | 53 |
| 2.4 | Variables globales et configuration des applications : principes | 54 |
| 2.4.1 | Gestion des données globales : Zend_Registry | 55 |
| 2.4.2 | Configuration des applications : Zend_Config | 57 |
| 3. | Configuration des applications | 66 |
| 3.1 | Les fichiers de configuration | 66 |
| 3.2 | Le chargeur de configuration | 68 |
| 4. | Distribution des fichiers dans l'arborescence du site | 76 |
| 4.1 | Architecture | 76 |
| 4.2 | Protection des répertoires du site | 79 |
| 5. | Webographie | 81 |
| 6. | Conclusion | 81 |

Chapitre 5

Écrire une application console

| | | |
|-----|--|----|
| 1. | Introduction | 83 |
| 2. | Présentation de l'application exemple | 84 |
| 2.1 | Analyse syntaxique de la ligne de commande | 85 |
| 2.2 | L'automate d'analyse de la ligne de commande | 87 |
| 2.3 | Le composant Zend_Console_Getopt | 88 |

- 2.4 Analyse de l'application. 91
- 3. Le script pilote palindrome.php et les fichiers de configuration 92
- 4. La classe d'analyse Console_Palindromes_Commande. 97
- 5. La classe Console_Palindromes_Palindrome. 100
 - 5.1 Les méthodes principales. 100
- 6. La classe Configurations_Palindrome. 103
 - 6.1 Les initialisations 105
 - 6.2 Filtrage des données et Zend_Filter 108
 - 6.3 Filtrage des données
et Configurations_Palindrome::initFiltres() 110
 - 6.4 Définir un filtre. 113
- 7. Utiliser 117
 - 7.1 Lancer l'exécution du script principal. 117
 - 7.2 L'arborescence des fichiers 119
 - 7.3 Extensions. 120
- 8. Conclusion 120

Chapitre 6

Accès aux données

- 1. Introduction 121
- 2. Connexion à la base de données 122
 - 2.1 Réaliser la connexion au SCBDR. 123
 - 2.2 La base de données exemple 127
- 3. Présentation de Zend_Db 130
 - 3.1 Requêtes simples et résultats. 130
 - 3.2 Lecture des données de la base 133
 - 3.3 Insert, update, delete. 140
 - 3.3.1 Insérer de nouvelles données 140
 - 3.3.2 Mettre à jour des enregistrements. 143
 - 3.3.3 Supprimer des tuples. 144
 - 3.4 Transactions 145
 - 3.5 Requêtes élaborées : Zend_Db_Select 146
 - 3.6 Analyser le comportement : Zend_Db_Profiler 156
- 4. Correspondance Objet/Relationnel (ORM) 162
 - 4.1 Introduction en forme d'avertissement 163

| | | |
|-----|--|-----|
| 4.2 | La correspondance Objet/Relationnel : présentation | 164 |
| 4.3 | Requêtes de sélection simples | 170 |
| 4.4 | Écriture dans la base | 173 |
| 4.5 | Naviguer entre les classes de correspondance. | 175 |
| 4.6 | Persistance | 181 |
| 5. | Webographie | 186 |
| 6. | Conclusion | 187 |

Chapitre 7**Le patron de conception Modèle-Vue-Contrôleur**

| | | |
|-----|--|-----|
| 1. | Introduction | 189 |
| 2. | Flux de contrôle d'une requête et Zend_Controller | 191 |
| 2.1 | Présentation des classes de Zend_Controller : théorie | 191 |
| 2.2 | L'objet requête. | 193 |
| 2.3 | Chemin (<i>workflow</i>) d'une requête HTTP | 193 |
| 3. | Démarrage (<i>bootstrap</i>) de l'application | 198 |
| 3.1 | Le script principal index.php | 199 |
| 3.2 | La classe principale Routeur | 200 |
| 4. | Présentation de l'application | 202 |
| 4.1 | Le contrôleur d'actions des cas d'utilisation | 203 |
| 4.2 | Le modèle | 208 |
| 4.3 | Liste des établissements et des examens : les méthodes d'actions. | 210 |
| 4.4 | Les scripts de vues : etablissements.phtml et examens.phtml | 211 |
| 4.5 | La méthode d'action candidatsparetablissementAction() | 214 |
| 4.6 | Le script de la vue candidats-par-etablissement.phtml | 219 |
| 4.7 | La méthode du modèle | 220 |
| 4.8 | La classe responsable de la configuration. | 221 |
| 5. | La mise en cache des informations | 223 |
| 5.1 | Présentation de Zend_Cache | 224 |
| 5.2 | Mise en œuvre dans l'application EXAMENS | 225 |
| 6. | Webographie | 228 |
| 7. | Conclusion | 228 |

Chapitre 8**Confidentialité**

| | |
|--|-----|
| 1. Introduction | 229 |
| 2. Confidentialité des applications Web | 231 |
| 2.1 Identifier | 231 |
| 2.2 Authentifier | 233 |
| 2.3 Autoriser | 234 |
| 3. Zend_Auth et l'authentification | 234 |
| 3.1 Présentation de Zend_Auth | 234 |
| 3.2 Authentification à l'aide d'une table de base de données | 238 |
| 3.3 Authentification HTTP | 242 |
| 3.3.1 Configuration de l'adaptateur | 242 |
| 3.3.2 Définition du résolveur | 245 |
| 3.3.3 Définition d'un résolveur interrogeant une table de base de données : principes | 249 |
| 3.3.4 Implémentation du résolveur Resolveur_DbTable | 252 |
| 4. La validation des données et Zend_Validate | 255 |
| 4.1 Présentation de Zend_Validate | 256 |
| 4.2 Les classes de Zend_Validate | 261 |
| 4.3 Écrire une classe de validation | 262 |
| 4.4 Contrôler la robustesse des mots de passe | 266 |
| 5. Zend_Acl et les autorisations | 272 |
| 5.1 Les ressources | 272 |
| 5.2 Les rôles | 275 |
| 5.3 Autoriser | 280 |
| 6. Extension de l'application exemple | 282 |
| 6.1 Les cas d'utilisation | 282 |
| 6.2 Les rôles et leur implémentation | 284 |
| 7. Contrôle d'accès dans l'application EXAMENS | 285 |
| 7.1 Les ressources et les contrôleurs d'actions | 286 |
| 7.2 Implémentation du contrôle d'accès | 291 |
| 8. Zend_Auth et l'authentification : implémentation | 302 |
| 9. Description de l'application exemple : implémentation | 311 |
| 10. Conclusion | 314 |

Chapitre 9**Internationalisation**

| | | |
|-------|---|-----|
| 1. | Introduction | 315 |
| 2. | Zend_Locale | 317 |
| 2.1 | Sélection automatique de la locale | 317 |
| 2.2 | Utilisation des locales automatiques | 318 |
| 2.3 | Locale initialisée pour toute une application | 320 |
| 3. | Expression localisée des messages | 320 |
| 3.1 | Choisir un adaptateur de traduction | 321 |
| 3.2 | Générer un fichier de traduction pour Zend_Translate | 322 |
| 3.2.1 | Instanciation de Zend_Translate | 323 |
| 3.2.2 | Génération du fichier des messages à traduire | 324 |
| 3.2.3 | Traduire les messages | 326 |
| 3.2.4 | Distribution des fichiers de traduction dans les dossiers du site | 328 |
| 3.3 | Compléments à l'étude de Zend_Translate | 329 |
| 3.3.1 | Utiliser plusieurs langues différentes | 329 |
| 3.3.2 | Accélérer l'utilisation des traductions | 330 |
| 3.3.3 | Intégrer des données calculées dans une traduction | 331 |
| 4. | Localisation de l'application EXAMENS | 332 |
| 4.1 | Traduction des messages | 333 |
| 4.2 | Adaptation des vues | 336 |
| 4.3 | Adaptation de index.php ou de Routeur.php | 337 |
| 5. | Zend_Date | 338 |
| 6. | Zend_Currency | 341 |
| 7. | Journalisation | 343 |
| 7.1 | Présentation de Zend_Log | 344 |
| 7.2 | Journaliser | 345 |
| 7.3 | Adapter la journalisation | 346 |
| 7.3.1 | Utiliser plusieurs rédacteurs | 346 |
| 7.3.2 | Définir de nouveaux niveaux de priorité | 347 |
| 7.3.3 | Définir un nouveau type d'événement | 347 |
| 7.4 | Journalisation des exceptions | 347 |
| 7.5 | Formateurs | 348 |
| 7.6 | Les filtres | 348 |
| 7.7 | Journalisation dans l'application EXAMENS | 349 |

8. Webographie 353
9. Conclusion 353

Chapitre 10
Services Web

1. Introduction 355
2. Exemples d'accès à un service Web par programme 359
 2.1 Un client pour interroger le service de recherche Yahoo! 359
 2.2 Écrire un service interrogé par un navigateur 361
 2.3 Un client avec Zend_Http_Client 364
3. Zend Framework et les services Web 366
4. Utiliser un service Web avec le Zend Framework :
 Zend_Service_Yahoo 367
5. Representational State Transfer et Zend_Rest 370
 5.1 Définition du serveur 372
 5.2 Définition du client 377
6. Webographie 378
7. Conclusion 378

Annexe 379

Index 381