

## Exercice : Sauvegarder un formulaire dans un fichier JSON

### Exercice : Sauvegarder un formulaire en JSON

#### Objectif

Créer une page "Ajouter un message" qui :

1. Affiche un formulaire
2. Valide les données
3. Les sauvegarde dans `data/messages.json`
4. Redirige vers une liste qui affiche tous les messages

À la fin de cet exercice, tu sais faire de la persistance de données sans base de données.

## Structure du projet

```
site-modulaire/  
├── data/  
│   └── messages.json      ← créé automatiquement  
├── includes/  
│   ├── header.php  
│   ├── nav.php  
│   └── footer.php  
├── messages-liste.php    ← affiche tous les messages  
└── messages-ajouter.php ← formulaire + traitement
```

## messages-ajouter.php – Formulaire + traitement

```
<?php  
1 // =====  
2 // TRAITEMENT DU FORMULAIRE (si soumis)  
3 // =====  
4 $erreurs = []; // contiendra les messages d'erreur  
5 $success = false; // passera à true si tout est ok  
6  
7 if ($_SERVER["REQUEST_METHOD"] === "POST") {  
8  
9     // --- 1. RÉCUPÉRER et nettoyer les données ---  
10    $nom = trim($_POST["nom"] ?? "");
```

```

11     $email = trim($_POST["email"] ?? "");
12     $message = trim($_POST["message"] ?? "");
13
14     // --- 2. VALIDER ---
15     if (empty($nom)) {
16         $erreurs[] = "Le nom est obligatoire.";
17     }
18     if (empty($email) || !filter_var($email, FILTER_VALIDATE_EMAIL)) {
19         $erreurs[] = "L'email est invalide.";
20     }
21     if (empty($message)) {
22         $erreurs[] = "Le message est obligatoire.";
23     } elseif (strlen($message) < 10) {
24         $erreurs[] = "Le message doit faire au moins 10 caractères.";
25     }
26
27     // --- 3. SAUVEGARDER si pas d'erreurs ---
28     if (empty($erreurs)) {
29
30         $fichier = "data/messages.json";
31
32         // Lire les messages existants (ou tableau vide si premier message)
33         if (file_exists($fichier)) {
34             $messages = json_decode(file_get_contents($fichier), true) ?? [];
35         } else {
36             $messages = [];
37         }
38
39         // Créer le nouvel enregistrement
40         $nouveauMessage = [
41             "id" => time(), // identifiant unique (timestamp)
42             "nom" => $nom,
43             "email" => $email,
44             "message" => $message,
45             "date" => date("d/m/Y à H:i"), // ex: 14/04/2025 à 16:30
46         ];
47
48         // Ajouter à la fin du tableau
49         $messages[] = $nouveauMessage;
50
51         // Réécrire tout le fichier JSON
52         file_put_contents(
53             $fichier,
54             json_encode($messages, JSON_PRETTY_PRINT | JSON_UNESCAPED_UNICODE)
55         );
56
57         // --- 4. REDIRIGER après succès (pattern PRG) ---
58         header("Location: messages-liste.php?success=1");
59         exit;
60     }
61 }
62
63 // =====
64 // AFFICHAGE DE LA PAGE
65 // =====
66 $pageTitle = "Nouveau message";
67 require 'includes/header.php';

```

```

68 | require 'includes/nav.php';
69 |
70 | <main class="container mt-5">
71 |     <div class="row justify-content-center">
72 |         <div class="col-md-7">
73 |
74 |             <h1>Envoyer un message</h1>
75 |
76 |             <!-- Affiche les erreurs si il y en a -->
77 |             <?php if (!empty($erreurs)) : ?>
78 |                 <div class="alert alert-danger">
79 |                     <strong>Corrige les erreurs suivantes :</strong>
80 |                     <ul class="mb-0 mt-2">
81 |                         <?php foreach ($erreurs as $e) : ?>
82 |                             <li><? = htmlspecialchars($e) ?></li>
83 |                         <?php endforeach; ?>
84 |                     </ul>
85 |                 </div>
86 |             <?php endif; ?>
87 |
88 |             <form method="POST" action="" class="mt-3">
89 |
90 |                 <div class="mb-3">
91 |                     <label for="nom" class="form-label">Nom *</label>
92 |                     <input type="text"
93 |                         id="nom"
94 |                         name="nom"
95 |                         class="form-control <? = in_array("Le nom est obligatoire.", $erreur
96 |                         value="<? = htmlspecialchars($_POST['nom'] ?? '') ?>"
97 |                         required>
98 |                     </div>
99 |
100 |                     <div class="mb-3">
101 |                         <label for="email" class="form-label">Email *</label>
102 |                         <input type="email"
103 |                             id="email"
104 |                             name="email"
105 |                             class="form-control"
106 |                             value="<? = htmlspecialchars($_POST['email'] ?? '') ?>"
107 |                             required>
108 |                         </div>
109 |
110 |                     <div class="mb-3">
111 |                         <label for="message" class="form-label">Message * (min. 10 caractères)</label>
112 |                         <textarea id="message"
113 |                             name="message"
114 |                             class="form-control"
115 |                             rows="5"
116 |                             required><? = htmlspecialchars($_POST['message'] ?? '') ?></textarea>
117 |                     </div>
118 |
119 |                     <div class="d-flex gap-2">
120 |                         <button type="submit" class="btn btn-primary">Envoyer ✉</button>
121 |                         <a href="messages-liste.php" class="btn btn-outline-secondary">Annuler</a>
122 |                     </div>

```

```

118         </form>
119     </div>
120 </div>
121 </main>
122
123 <?php require 'includes/footer.php'; ?>

```

## messages-liste.php – Afficher tous les messages

```

<?php
1  $pageTitle = "Messages reçus";
2  require 'includes/header.php';
3  require 'includes/nav.php';
4
5  // Lire les messages depuis le fichier JSON
6  $fichier = "data/messages.json";
7
8  if (file_exists($fichier)) {
9      $messages = json_decode(file_get_contents($fichier), true) ?? [];
10     // Afficher les plus récents en premier (inverser le tableau)
11     $messages = array_reverse($messages);
12 } else {
13     $messages = [];
14 }
?>

15
16 <main class="container mt-5">
17
18     <div class="d-flex justify-content-between align-items-center mb-4">
19         <h1>Messages reçus</h1>
20         <a href="messages-ajouter.php" class="btn btn-primary">+ Nouveau message</a>
21     </div>
22
23     <!-- Message de succès après redirection -->
24     <?php if (isset($_GET['success'])) : ?>
25         <div class="alert alert-success alert-dismissible fade show">
26              Message envoyé avec succès !
27             <button type="button" class="btn-close" data-bs-dismiss="alert"></button>
28         </div>
29     <?php endif; ?>
30
31     <?php if (empty($messages)) : ?>
32
33         <div class="alert alert-info">
34             Aucun message pour l'instant.
35             <a href="messages-ajouter.php">Sois le premier !</a>
36         </div>
37
38     <?php else : ?>
39

```

```

40     <?php foreach ($messages as $m) : ?>
41         <div class="card mb-3 shadow-sm">
42             <div class="card-body">
43                 <div class="d-flex justify-content-between">
44                     <h5 class="card-title mb-1"><?php htmlspecialchars($m['nom']) ?></h5>
45                     <small class="text-muted"><?php htmlspecialchars($m['date']) ?></small>
46                 </div>
47                 <small class="text-muted"><?php htmlspecialchars($m['email']) ?></small>
48                 <p class="card-text mt-2"><?php nl2br(htmlspecialchars($m['message'])) ?></p>
49             </div>
50         </div>
51     <?php endforeach; ?>
52
53     <?php endif; ?>
54
55 </main>
56
57 <?php require 'includes/footer.php'; ?>

```

## Ce qu'il faut retenir

Opération	Code PHP
Lire un JSON	<code>json_decode(file_get_contents(\$f), true)</code>
Ajouter un élément	<code>\$tableau[] = \$nouvelElement</code>
Écrire un JSON	<code>file_put_contents(\$f, json_encode(\$t, JSON_PRETTY_PRINT))</code>
Vérifier si vide	<code>empty(\$tableau)</code>
Inverser ordre	<code>array_reverse(\$tableau)</code>
Date/heure	<code>date("d/m/Y à H:i")</code>
Identifiant unique	<code>time()</code> (timestamp Unix)



## Vérifie

1. Envoie 3 messages → ils apparaissent dans `messages-liste.php`
2. Ouvre `data/messages.json` dans VS Code → le JSON est bien formé
3. Essaie de soumettre le formulaire vide → les erreurs s'affichent
4. Après envoi, actualise la page liste → pas de re-soumission (grâce au redirect)



## Challenge

1. Ajoute un bouton "Supprimer" sur chaque message  
(*indice* : passe l'id en GET, filtre le tableau avec `array_filter()`, réécrits)
2. Affiche le nombre total de messages dans le titre
3. Limite l'affichage aux 10 derniers messages

