

# PHP : Site modulaire avec include

## PHP : Site modulaire avec include

### Le problème

---

Imagine un site de 8 pages. Chaque page a le même header et le même footer.  
Tu changes le logo → tu ouvres **8 fichiers**. Tu oublies une page → bug en prod. 🤔

**Solution** : découper le site en morceaux réutilisables.

---

### include et require

---

```
1 | <?php include 'fichier.php'; ?>
2 | <?php require 'fichier.php'; ?>
```

Les deux insèrent le contenu d'un autre fichier à cet endroit.

Quelle différence ?

**include**

**require**

Fichier manquant Warning, le script **continue** Fatal error, le script **s'arrête**

À utiliser pour    Blocs optionnels                    Blocs essentiels (header, footer)

✅ **Règle** : pour header et footer, utilise toujours **require**.

Si le header manque, la page est cassée — autant arrêter tout de suite.

---

### Structure de projet recommandée

---

```
php-pratique/
├── includes/
│   ├── header.php      ← ouverture HTML + <head> + Bootstrap
│   ├── nav.php         ← barre de navigation
│   └── footer.php      ← fermeture HTML + scripts
├── index.php
├── contact.php
└── produits.php
```

Le dossier **includes/** regroupe tous les blocs partagés.

Les pages sont à la racine du projet.

---

### Créer les blocs partagés

---

# includes/header.php

```
1 <!DOCTYPE html>
2 <html lang="fr">
3 <head>
4   <meta charset="UTF-8">
5   <meta name="viewport" content="width=device-width, initial-scale=1">
6   <title><?= $pageTitle ?? "Mon site" ?></title>
7   <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0/dist/css/bootstrap.min.css" rel="
8 </head>
9 <body>
```

<?= \$pageTitle ?? "Mon site" ?> :

- <?= ... ?> est un raccourci pour <?php echo ... ?>
- ?? signifie : si \$pageTitle n'existe pas, utilise "Mon site"

# includes/nav.php

```
1 <nav class="navbar navbar-expand-lg navbar-dark bg-dark">
2   <div class="container">
3     <a class="navbar-brand" href="index.php">MonSite</a>
4     <div class="navbar-nav">
5       <a class="nav-link" href="index.php">Accueil</a>
6       <a class="nav-link" href="produits.php">Produits</a>
7       <a class="nav-link" href="contact.php">Contact</a>
8     </div>
9   </div>
10 </nav>
```

# includes/footer.php

```
1 <footer class="bg-dark text-white text-center py-3 mt-5">
2   <p class="mb-0">&copy; <?= date('Y') ?> Mon Site – Tous droits réservés</p>
3 </footer>
4
5 <script src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0/dist/js/bootstrap.bundle.min.js"></s
6 </body>
7 </html>
```

date('Y') affiche l'année courante automatiquement.

# Créer une page qui utilise les blocs

## index.php

```
<?php
1 // On définit le titre AVANT d'inclure le header
2 $pageTitle = "Accueil";
```

```

3 | require 'includes/header.php';
4 | require 'includes/nav.php';
?>
5 |
6 | <main class="container mt-4">
7 |     <h1>Bienvenue</h1>
8 |     <p>Ceci est la page d'accueil.</p>
9 | </main>
10 |
11 | <?php require 'includes/footer.php'; ?>

```

## contact.php

```

<?php
1 | $pageTitle = "Contact";
2 | require 'includes/header.php';
3 | require 'includes/nav.php';
?>
4 |
5 | <main class="container mt-4">
6 |     <h1>Nous contacter</h1>
7 |     <p>Email : contact@monsite.be</p>
8 | </main>
9 |
10 | <?php require 'includes/footer.php'; ?>

```

**Résultat** : les deux pages ont exactement le même header et footer.  
 Tu changes `nav.php` → **toutes les pages** sont mises à jour. ✓

## Les chemins de fichiers

Le chemin est **relatif au fichier qui fait l'include**.

```

1 | // Depuis index.php (à la racine)
2 | require 'includes/header.php'; // ✓ cherche includes/ dans le même dossier
3 |
4 | // Depuis un fichier dans un sous-dossier (ex: admin/dashboard.php)
5 | require '../includes/header.php'; // ✓ .. remonte d'un niveau

```

💡 Astuce : utilise `__DIR__` pour un chemin absolu, sans ambiguïté :

```

1 | require __DIR__ . '/includes/header.php';

```

## Ce qu'on a maintenant

- Un seul `header.php` → modifier Bootstrap ? Une seule ligne à changer.

- Un seul `nav.php` → ajouter un lien ? Une seule ligne à changer.
- Chaque page a son propre `$pageTitle`.

C'est la base de tout site PHP professionnel.

---

## Prochaine étape

---

Exercice : [construire son premier site modulaire](#) →