

Comprendre le Buffer en PHP

Le **buffer de sortie** en PHP est un mécanisme qui permet de capturer le contenu HTML ou texte généré par un script **avant de l'envoyer au navigateur**.

Pourquoi utiliser le buffer ?

- **Séparer le contenu de l'affichage** (utile pour créer des templates ou moteurs de rendu)
- **Modifier le contenu HTML** avant de l'afficher
- **Assembler dynamiquement plusieurs blocs de contenu**
- **Empêcher l'affichage prématuré** (notamment pour les redirections HTTP)

Fonctions principales

Fonction PHP	Description
<code>ob_start()</code>	Démarre le buffer
<code>ob_get_clean()</code>	Récupère le contenu du buffer et le vide
<code>ob_get_contents()</code>	Récupère le contenu sans vider
<code>ob_end_clean()</code>	Vide le buffer sans l'afficher
<code>ob_flush()</code>	Envoie manuellement le contenu du buffer au navigateur

Exemple de base

```
ob_start(); // On démarre le buffer

echo "Bonjour";
echo " le monde !";

$content = ob_get_clean(); // On récupère le texte généré

echo strtoupper($content); // Affiche : BONJOUR LE MONDE !
```

Exemple dans un système de template

Fichier : `template.php`

```
$title = "Accueil";
ob_start();
require 'page.php';
$content = ob_get_clean();
require 'layout.php';
```

Fichier : `layout.php`

```
<!DOCTYPE html>
<html>
<head><title><?= $title ?></title></head>
<body>
    <?= $content ?>
</body>
</html>
```

✓ Avantages pédagogiques

- Facilite la **compréhension du moteur de template**
- Approche **proche du fonctionnement de frameworks PHP modernes**
- Rend le code **modulaire et lisible**

⚠ À noter

- Le buffer est actif **par défaut** dans certains contextes (ex: certains CMS ou frameworks).
- Il peut interférer avec des en-têtes HTTP (`header()`) s'il n'est pas vidé correctement.