PHP : Les formulaires $_GET et $_POST

Quand un utilisateur remplit un **formulaire HTML** et le soumet, les données doivent être transmises au serveur. En PHP, ces données sont automatiquement récupérées dans des **tableaux superglobaux associatifs** : `$_GET` et `$_POST`.

📌 Qu'est-ce que $_GET et $_POST ?

- Ce sont des **tableaux associatifs** : chaque champ du formulaire devient une **clé**, et la valeur saisie devient la **valeur** dans le tableau.
- Ce sont des **superglobales** : elles sont accessibles **partout** dans le code PHP, sans avoir besoin d'être déclarées comme `global`.

🧱 Fonctionnement général

	$_GET	$_POST
Méthode	Données envoyées dans l'URL	Données envoyées dans le corps de la requête
Visibilité	Visible dans la barre d'adresse	Invisible
Sécurité	Moins sécurisé (car visible)	Plus adapté aux données sensibles
Taille limite	Limitée par l'URL (~2000 caractères)	Illimitée (dépend du serveur)
Utilisation typique	Requêtes simples, liens de recherche, filtres	Formulaires, connexions, commentaires

🔍 Exemple avec $_GET

Formulaire HTML :

```html
<form action="recherche.php" method="get">
  <input type="text" name="q" placeholder="Recherche...">
  <input type="submit" value="Chercher">
</form>
```

Fichier `recherche.php` :

```php
<?php
$motCle = $_GET['q'];
echo "Vous avez recherché : $motCle";
?>
```

Résultat dans l'URL après envoi :

```
recherche.php?q=php
```

➡️ Ici, `$_GET['q']` contient la valeur saisie dans le champ de recherche.

🔐 Exemple avec `$_POST`

Formulaire HTML :

```html
<form action="contact.php" method="post">
  <input type="text" name="nom">
  <textarea name="message"></textarea>
  <input type="submit" value="Envoyer">
</form>
```

Fichier `contact.php` :

```php
<?php
$nom = $_POST['nom'];
$message = $_POST['message'];
echo "Message reçu de $nom :<br>";
echo nl2br(htmlspecialchars($message));
?>
```

➡️ Les données ne sont **pas visibles dans l'URL**, mais disponibles dans `$_POST`.

🛠️ Bonnes pratiques

1. **Toujours tester l'existence d'un champ** avant d'y accéder :

```php
if (isset($_POST['nom'])) {
    $nom = $_POST['nom'];
}
```

2. **Nettoyer les données avant affichage** :

```php
$nom = htmlspecialchars($_POST['nom']);
```

3. **Éviter les erreurs** sur les champs manquants :

```php
$nom = $_POST['nom'] ?? "Invité";
```

✅ Quand utiliser l'un ou l'autre ?

Objectif	Méthode conseillée
Rechercher une info (sans effet côté serveur)	`GET`
Lien partageable ou bookmarkable	`GET`

Objectif	Méthode conseillée
Envoyer un mot de passe, des infos personnelles	`POST`
Modifier des données (ajouter, supprimer...)	`POST`

⚠️ Une règle importante du protocole HTTP :
Une requête `GET` **ne doit jamais modifier** les données côté serveur. Elle est censée être **sûre** (sans effet secondaire).

🧠 Résumé

- `$_GET` et `$_POST` sont des **superglobaux PHP** qui permettent d'accéder aux données soumises par un formulaire.
- Ce sont des **tableaux associatifs**, où chaque **clé** correspond au nom du champ HTML.
- Choisis `GET` pour récupérer, `POST` pour modifier.
- Toujours **valider** et **nettoyer** les données avant utilisation.