

Ing3 - Web Dev I

TP 2 – Fonction & Formulaire

Exercice 1 : Fonction & Référence

Une université veut générer automatiquement des matricules pour les nouveaux étudiants. Le format est : *PREMIERE_LETTR_PRENOM + NOM + NUMERO_INCREMENTAL*

Dans votre fichier « **exe1.php** », créer les éléments suivants :

1. Créez une variable globale **\$compteur = 1000**.
2. Ecrivez une fonction **genererMatricule(\$prenom, \$nom, &\$compteur)** :
 - Elle doit prendre le compteur **par référence**.
 - Elle génère la chaîne (ex : "SNOUFEL1000").
 - Elle incrémente le compteur de 1 à chaque appel.
 - Elle retourne le matricule.
3. Testez en appelant la fonction 3 fois pour 3 étudiant différents et affichez les résultats. Le numéro doit augmenter automatiquement (1000, 1001, 1002).

Exercice 2 : Formulaire GET

Votre objectif est de créer une page « **alert.php** », pour les administrateurs système, qui leur permet de générer des bannières d'alerte standardisées.

Partie 1 : Le Formulaire (HTML)

Créez une section (ou une page) avec un formulaire simple :

- **Méthode** : GET
- **Action** : Le nom de votre fichier PHP (ex : alert.php).
- **Champs** :
 - Un champ texte pour le **Message** (nom : msg).
 - Un menu déroulant (<select>) pour le **Type** d'alerte (nom : type) avec deux options : "Succès" (valeur success) et "Erreur" (valeur error).
 - Un bouton "Générer l'alerte".

Partie 2 : Le Traitement (PHP)

Dans le code PHP (qui reçoit les données) :

1. Créez une fonction **afficherAlerte(\$message, \$type)** qui retourne une <div> HTML avec du style CSS en ligne (style="...") :
 - Si \$type est "error" => Fond Rouge
 - Si \$type est "success" => Fond Vert

- Sinon Fond Gris.
- 2. Récupérez les données via `$_GET` (Vérifiez d'abord si elles existent avec `isset`).
- 3. Appelez la fonction et affichez le résultat.

Exercice 3 : POST & Checkboxes

Une entreprise Cloud permet à ses clients de configurer leur serveur.

Formulaire HTML :

- **Nom du Serveur** (Input Text).
- **Type d'OS** (Radio) : Ubuntu (Gratuit), Windows Server (+500 MAD).
- **Logiciels Pré-installés** (Checkbox - **Attention au tableau**) :
 - Apache (+100 MAD)
 - MySQL (+150 MAD)
 - Docker (+200 MAD)

Travail demandé (PHP) :

1. Récupérez les données via `$_POST`.
2. Calculez le prix total mensuel.
3. Affichez un récapitulatif :
 - *"Serveur [Nom] configuré sous [OS]."*
 - *"Logiciels : Apache, MySQL..."* (Utilisez une boucle `foreach` pour lire le tableau des checkboxes).
 - *"Prix Total : X MAD"*.

Exercice 4 : (Devoir Maison) AI-Recruter

Vous développez un script de pré-sélection pour les RH. Le candidat remplit un formulaire, et PHP calcule son "Score de Compatibilité".

Cahier des charges :

1. **Formulaire** :
 - Nom & Prénom.
 - Années d'expérience (Select : 0-2 ans, 3-5 ans, +5 ans).
 - Compétences (Checkboxes) : PHP, SQL, Python, HTML/CSS, Java.
2. **Logique de notation (Fonction)** :
 - Créez une fonction `calculerScore($experience, $competences)`.
 - **Expérience** : 0-2 ans (0 pts), 3-5 ans (10 pts), +5 ans (20 pts).
 - **Compétences** :

- PHP ou Python = +5 points chacun.
- SQL = +5 points.
- Java = +2 points.

3. Décision :

- Si Score > 20 : Affichez "Candidat Retenu" (En Vert).
- Si Score < 20 : Affichez "Candidat Rejeté" (En Rouge).
- Affichez le détail des points obtenus.