

TD2 & 3 (3,5H) + TP2 (2H) : **Traitement côté Serveur (PHP)**

Interactions avec les formulaires HTML5 + Interactions une Base de données SQLite
+ Méthodes GET / POST

1. EX1 : PASSAGE DE PARAMÈTRES AVEC LA MÉTHODE GET (TD 30')

On peut passer des paramètres à un script php au travers de l'url de la manière suivante :

http://@IPServeur/monscript.php?param1=valeur1¶m2=valeur2

1. Ecrire un script qui affiche les paramètres reçus par la méthode GET (quelque soit le nombre de paramètres reçus)
2. Modifier le script pour afficher un message «*Aucun paramètre*» s'il n'y a pas de paramètre reçu.
3. Modifier le script pour tester en plus si un des paramètres reçus s'appelle « action ». et qu'il vaut « marche ».

2. EX2 : FORMULAIRES & PARAMÈTRES POST (TD 45')

On charge la page **login_POST.php** dont voici le code source HTML

Formulaire de connexion par la méthode HTTP POST

Adresse Mail :

Mot de passe :

```
<!DOCTYPE html>
<html lang="fr" >
<head>
<meta charset="utf-8">
<title>Formulaire de connexion par la méthode HTTP POST</title>
</head>
<body>
<form id="form1" action="verif_login_POST.php" method="post">
<fieldset>
<legend>Formulaire de connexion par la méthode HTTP POST</legend>
<label for="id_mail">Adresse Mail : </label>
<input type="email" name="mail" id="id_mail" placeholder="@mail" required /><br />
<label for="id_pass">Mot de passe : </label>
<input type="password" name="pass" id="id_pass" placeholder="Mot de passe" required /><br />
<input type="submit" name="connect" value="Connection" />
</fieldset>
</form>
</body>
</html>
```

1. Relier les éléments du formulaire HTML avec le fichier **login_POST.php**
2. Quel script est appelée par ce formulaire ?
3. Quel attribut du formulaire est déterminant afin de récupérer les valeurs passées ?
4. Quelle variable PHP permet de récupérer les valeurs passées par le formulaire au script appelé?

5. Ecrire le script qui va traiter les données reçues. On commencera par tester la présence de paramètre et des clés prévues (non vides) par le formulaire. Ensuite on affichera son contenu pour vérifier les valeurs passées avec un `var_dump()` du `$_POST` puis :

Si le couple login/mot de passe est correct, on affichera un message de bienvenue, sinon on affichera un message d'erreur avec un lien pour revenir vers la page d'origine.	<pre>array(3) { ["mail"]=> string(12) "phil@free.fr" ["pass"]=> string(4) "phil" ["connect"]=> string(10) "Connection" }</pre> <p>Identification OK de l'utilisateur phil@free.fr</p>	<pre>array(3) { ["mail"]=> string(12) "phil@free.fr" ["pass"]=> string(4) "toto" ["connect"]=> string(10) "Connection" }</pre> <p>Erreur d'identification de l'utilisateur phil@free.fr</p> <p>Retour à la page d'authentification</p>
---	--	---

Utiliser la console développeur de Firefox pour voir les paramètres transmis en POST

3. EX3 : ACCÈS À BASE DE DONNÉES SQLITE PAR PHP

3.1. CRÉATION ET MISE EN PLACE D'UNE BASE DE DONNÉES SQLITE (15')

DB Browser for SQLite permet en local toutes les manipulations possibles pour une bdd SQLite et vous permettra de vérifier le bon fonctionnement de vos programmes PHP.

Attention, pour modifier votre BDD avec DB Browser for SQLite, il faudra y accéder localement puis la télécharger sur le serveur WEB par FTP.

- Ouvrir en local le fichier fourni IUT.sqlite avec DB Browser for SQLite. Vérifier que la base de données contient les deux tables suivantes.

```
CREATE TABLE "etudiants" ("mail" VARCHAR PRIMARY KEY NOT NULL DEFAULT (null) ,"mdp"
VARCHAR DEFAULT (null) ,"adresse" VARCHAR DEFAULT (null) ,"insee" INTEGER DEFAULT (null) )
```

```
CREATE TABLE "villes" ("commune" varchar(30) NOT NULL DEFAULT ('') ,"cp" varchar(5) NOT NULL
DEFAULT ('') ,"dept" varchar(30) NOT NULL DEFAULT ('') ,"insee" varchar(5) PRIMARY KEY
DEFAULT (null) )
```

- Copier cette base de données sur votre serveur web dans un dossier ex :WEB2/TD2/EX1/bdd
- Protéger l'accès à ce dossier avec un fichier .htaccess qui contient deny from all. Faites un test de lecture via HTTP depuis votre navigateur.

3.2. LISTER LES ETUDIANTS DE LA BASE DE DONNÉES (20')

Ecrire un script EX3BDD.php qui affiche tous les utilisateurs de la base de données.

Pour cela, on créera une fonction `listeEtudiants()` qui retournera un tableau PHP de tous les étudiants.

RQ: la fonction `afficheTableauHTML()` est fournie dans le fichier EX3BDD.php (déjà faite en WEB1)

WEB2 TD12 EX3 : Liste BDD ETUDIANTS

Liste tous les étudiants

mail	mdp	adresse	insee	commune	cp	dept
etu@etu.fr	etu	Rue de Lannion	22113	LANNION	22300	COTES D'ARMOR
etu1@etu.fr	etu1	Rue de Rennes	35238	RENNES	35000	ILLE ET VILAINE
etu2@etu.fr	etu2	Rue de Guingamp	22070	GUINGAMP	22200	COTES D'ARMOR
etu3@etu.fr	etu3	Rue de Vannes	56260	VANNES	56000	MORBIHAN
prof@etu.fr	prof	Rue de Taulé	29279	TAULE	29670	FINISTERE
etu4@etu.fr	etu4	Route de Trébeurden	22113	LANNION	22300	COTES D'ARMOR

// ~Fin TD2 à finir avant TD3

3.3. TESTER L'EXISTENCE D'UN COUPLE ETUDIANT / MOT DE PASSE (30')

1. Créer une fonction **utilisateurExiste(\$login,\$pass)** qui retournera un booléen vrai si un couple utilisateur/mot de passe existe dans la BDD.
2. Tester cette fonction avec au minimum deux couples utilisateur/mot de passe

Teste les étudiants	Teste les étudiants
etu@etu.fr : etu existe	etu@etu.fr : etu existe
titi@etu.fr : etu n'existe pas	titi@etu.fr : etu existe

3.4. INSÉRER UN ETUDIANT DANS LA BASE DE DONNÉES (45')

1. Créer une fonction **insereUtilisateur(\$login,\$pass,\$rue,\$codepostal,\$ville)** qui retournera un booléen vrai si l'insertion est réussie. Il faudra commencer par vérifier si la ville correspondante au code postal existe (rq : on n'insérera pas de nouvelle ville)
2. Tester cette fonction avec au moins deux comptes

Insère un étudiant	Insère un étudiant
Insertion réussie de titi@etu.fr : etu	ERREUR Insertion titi@etu.fr : etu soit la ville n'existe pas soit l'utilisateur existe déjà

4. EX4 : INTERACTION FORMULAIRE ET BASE DE DONNÉES (15')

1. Dans le fichier EX4_BDD.php copier la fonction **utilisateurExiste(\$login,\$pass)** précédemment étudiée et modifier ce script afin de rechercher si l'utilisateur qui s'authentifie via le formulaire est présent dans la base de données.
2. **Travail personnel : ranger les fonctions dans un fichier fonctions.php**

WEB2 TD12 EX4 : Formulaires & BDD ETUDIANTS	WEB2 TD12 EX4 : Formulaires & BDD ETUDIANTS
Voici la liste des paramètres reçus via HTTP par le Formulaire : <pre>array(3) { ["mail"]=> string(10) "etu@etu.fr" ["pass"]=> string(3) "etu" ["connect"]=> string(9) "Connexion" }</pre> Identification OK de l'utilisateur etu@etu.fr	Voici la liste des paramètres reçus via HTTP par le Formulaire : <pre>array(3) { ["mail"]=> string(10) "etu@etu.fr" ["pass"]=> string(4) "etu1" ["connect"]=> string(9) "Connexion" }</pre> Erreur d'identification de l'utilisateur etu@etu.fr Retour à la page d'authentification

5. EX5 : LISTE DÉROULANTE DYNAMIQUE (TD 30')

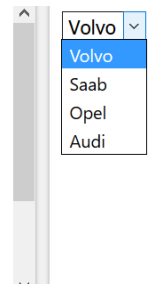
5.1. RAPPEL BALISE <SELECT>

Une balise <select> (liste déroulante en HTML) se compose de sous balises <option>.

L'attribut value sera transmis lors de l'envoi d'un formulaire qui contient une balise <select>

```
<!DOCTYPE html>
<html>
<body>

<select>
  <option value="volvo">Volvo</option>
  <option value="saab">Saab</option>
  <option value="opel">Opel</option>
  <option value="audi">Audi</option>
</select>
```

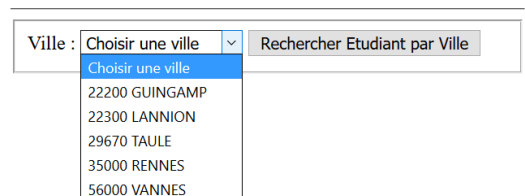


5.2. LISTE DES VILLES DES ÉTUDIANTS

On souhaite réaliser une fonction PHP afficheFormulaireEtudiantParVille() qui permettra de générer le code HTML suivant à partir des villes des étudiants existant dans la BDD :

```
<form action="" method="post">
<fieldset>
<label for="id_ville">Ville :</label>
<select id="id_ville" name="ville" size="1">
  <option value="0">Choisir une ville</option>
  <option value="22070">22200 GUINGAMP</option>
  <option value="22113">22300 LANNION</option>
  <option value="29279">29670 TAULE</option>
  <option value="35238">35000 RENNES</option>
  <option value="56260">56000 VANNES</option>
</select>
<input type="submit" value="Rechercher Etudiant par Ville"/>
</fieldset>
</form>
```

WEB2 TD12 EX5 : Formulaire dynamique & BDD ETUDIANTS



```

<?php
function afficheFormulaireEtudiantParVille(){
CNX BDD + REQUETE pour obtenir les villes correspondantes à des utilisateurs
$madb = new PDO('sqlite:bdd/IUT.sqlite'); // CNX BDD
$requete = 'SELECT DISTINCT (e.insee), commune, cp FROM etudiants e INNER JOIN villes v
ON e.insee=v.insee';
$resultat = $madb->query($requete);
$viles = $resultat->fetchAll(PDO::FETCH_ASSOC);
?>

<form action="" method="post">
<fieldset>
<label for="id_ville">Ville :</label>
<select id="id_ville" name="ville" size="1">
    <option value="0">Choisir une ville</option>
    <?php // générer la liste des options à partir de $viles

// A faire ...

?>

</select>
<input type="submit" value="Rechercher Etudiant par Ville"/>
</fieldset>
</form>
<?php
} // fin afficheFormulaireEtudiantParVille
?>

```

RQ : \$viles obtenu par la requête SQL contient par exemple :

```

array(5) {
[0]=> array(3) { ["insee"]=> string(5) "22070" ["commune"]=> string(8) "GUINGAMP" ["cp"]=> string(5) "22200" }
[1]=> array(3) { ["insee"]=> string(5) "22113" ["commune"]=> string(7) "LANNION" ["cp"]=> string(5) "22300" }
[2]=> array(3) { ["insee"]=> string(5) "29279" ["commune"]=> string(5) "TAULE" ["cp"]=> string(5) "29670" }
[3]=> array(3) { ["insee"]=> string(5) "35238" ["commune"]=> string(6) "RENNES" ["cp"]=> string(5) "35000" }
[4]=> array(3) { ["insee"]=> string(5) "56260" ["commune"]=> string(6) "VANNES" ["cp"]=> string(5) "56000" }
}

```

(1) A finir pour la première séance de TP (si ce n'est fait en TD)

6. EQUIPES DE RUGBY (TP 2H)

Une base de données respectant le schéma suivant a été implémentée à l'aide du SGBD SQLite.

```
CREATE TABLE "equipes" (  
    "numEquipe" INTEGER NOT NULL,  
    "nomEquipe" varchar2(20) NOT NULL,  
    PRIMARY KEY("numEquipe")  
);  
CREATE TABLE "joueurs" (  
    "numJoueur" INTEGER NOT NULL DEFAULT (null),  
    "nomJoueur" varchar2(20) NOT NULL DEFAULT (null),  
    "numEquipe" INTEGER NOT NULL DEFAULT (null),  
    PRIMARY KEY("numJoueur")  
);
```

6.1. CONSIGNES GÉNÉRALES

Toutes les fonctions et les formulaires devront être localisées dans un fichier unique intitulé fonctions.php qui sera à inclure dans les différents scripts à concevoir. Une feuille de style est fournie. L'encodage devra être en UTF-8. La base de données est fournie sur l'ENT (joueurs.sqlite).

6.2. AFFICHAGE DE TOUS LES JOUEURS

Ecrire un script **EX1listeJoueurs.php** pour afficher en HTML le tableau suivant.

Ce script fera appel à une fonction listeJoueur() qui retourne soit le tableau PHP soit un booléen false en cas d'erreur. L'affichage sous forme d'un tableau HTML devra être réalisé à l'aide de la fonction **afficheTableau(\$tab)** que s'il n'y a pas d'erreur

Voici la liste de tous les joueurs du TOP14

Joueur	Equipe
Penaud	Clermont
Lopez	Clermont
Parra	Clermont
Chouly	Clermont

(2) Faire valider par l'enseignant.

6.3. AFFICHAGE DES JOUEURS PAR EQUIPE (TD ET TP)

Ecrire un script **EX2listeJoueursParEquipe.php** comportant un formulaire dynamique (fonction afficheFormEquipe()) permettant la sélection des joueurs d'une équipe à l'aide d'un menu déroulant.

Ce formulaire invoquera le même script **EX2listeJoueursParEquipe.php** par méthode GET (en appelant la fonction **listerJoueursParEquipe(\$numEquipe)**) et affichera la liste des joueurs de l'équipe sélectionnée.

Sélectionner une Equipe pour afficher la liste des joueurs

Toutes les Equipes Afficher les joueurs

Voici la liste des joueurs de l'équipe : Toulouse

Joueur	Equipe
Dupont	Toulouse
Huget	Toulouse
Fickou	Toulouse

(3) Faire valider par l'enseignant.

6.4. INSERTION D'UN JOUEUR

Ecrire un script **EX3_InsererJoueurs.php** qui permet l'insertion d'un joueur dans la base de données (formulaire à compléter fonction **afficheFormulaireAjoutJoueur()**).

Votre script : (voir capture ci-dessous)

- appellera la même page avec la méthode POST
- apportera la sécurité nécessaire aux paramètres reçus avant de les insérer dans la BDD.
- vérifiera au préalable que le joueur n'est pas déjà dans la base de données avant de procéder à l'insertion.
- retournera un message indiquant la réussite ou non de l'opération
- affichera à nouveau le formulaire et le tableau des joueurs

Insertion d'un nouveau joueur

Nom joueur :

Nom equipe :

Chargement initial de la page
EX3_InsererJoueurs.php

Voici le code retour de l'opération: Le joueur existe déjà

Insertion d'un nouveau joueur

Nom joueur :

Nom equipe :

Joueur	Equipe
Penaud	Clermont
Lopez	Clermont
Parra	Clermont
Chouly	Clermont
Fofana	Clermont

Envoi du formulaire avec des données déjà présente dans la BDD

On réaffiche la liste

```
string(47) "INSERT INTO joueurs VALUES ( NULL,'Ituria','2')"
```

Voici le code retour de l'opération: Insertion réussie du joueur : 'Ituria' | 'Clermont'

Insertion d'un nouveau joueur

Nom joueur :

Nom equipe :

Joueur	Equipe
Penaud	Clermont
Lopez	Clermont
Parra	Clermont
Chouly	Clermont
Fofana	Clermont
Ituria	Clermont

Envoi du formulaire avec des données correctes

On réaffiche la liste

(4) Faire valider par l'enseignant.

6.5. MISE EN FORME AVEC BOOTSTRAP (CF WEB1)

Utiliser Bootstrap afin de pouvoir naviguer entre toutes les pages avec un menu.

(5) Faire valider par l'enseignant.