PHP #5

FORMULAIRES

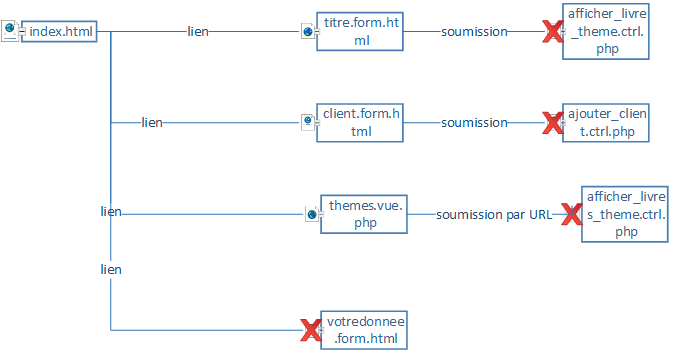
# Contexte des TP

Vous allez, sur les 4 prochains TP, compléter les fichiers constituant un site de vente de livres (et de présentation des données de votre choix), que nous avons modélisés au cours du module « Modélisation - Base de Données ».

1. Télécharger depuis *Moodle* les fichiers de départ et les décompresser dans un dossier « php » dans votre machine.

Pour cette séance, vous ferez les exercices demandés dans différents fichiers, avec l’éditeur de code de votre choix. Rappelez-vous qu’**il faut téléverser le fichier à chaque modification** pour le tester.

L’ensemble des fonctionnalités du site sont à tester depuis le fichier index.html qui s’affiche par défaut à [http://webmmi.iut-tlse3.fr/~*login*/php](http://webmmi.iut-tlse3.fr/~/php)



Une feuille de style style.css basique s’applique à ces pages web. Vous pouvez appliquer une feuille de style faite par vos soins.

# Récupération sécurisée d’un champ de formulaire

Le formulaire titre.form.html permet la saisie d’un titre pour afficher le livre correspondant (la recherche en BD sera implémentée sur le TP suivant).

1. Repérer les paramètres du formulaire nécessaires à la récupération du champ :
   * Quelle est la méthode d’envoi ?

GET

* + Vers quel script les valeurs saisies sont-elles transmises ?

Afficher\_livre\_titre.ctrl.php

* + Quel est le nom du champ ?

Titre d’un livre

1. En vous référant au cours et à la documentation de la fonction filter\_input : <https://www.php.net/manual/fr/function.filter-input.php>

compléter le script afficher\_livre\_titre.ctrl.php pour récupérer le titre saisi.

1. Tester avec un mot d’un titre de livre.
2. Tester la neutralisation de codes malveillants, comme :
   * <h2>Adobe pour altérer la page html

<script>document.location="http://i.ytimg.com/vi/0vxCFIGCqnI/maxresdefault.jpg"</script> pour une attaque XSS

* + Adobe' OR 1=1; SELECT \* FROM client; pour une injection SQL

*Remarque : observer le code source de la page pour vérifier la neutralisation de certains caractères (comme l’apostrophe en SQL) qui sont quand même affichés.*

*Remarque 2 : pour voir la différence avec une récupération insécure, tester l’appel à la fonction filter\_input sans 3ème paramètre (paramètre indiquant le filtre choisi).*

# Récupération sécurisée de plusieurs champs de formulaire

Le formulaire client.form.html doit envoyer les données d’un nouveau client au script ajouter\_client.ctrl.php

1. Compléter le formulaire client.form.html avec :
   * L’attribut method
   * L’attribut action
   * L’attribut name de chaque champ
2. Récupérer de manière sécurisée tous les champs du formulaire.
3. Tester en saisissant les informations d’un nouveau client.

# Passage de champ par la méthode GET sans formulaire !

Si la méthode POST est un peu plus sûre (car les données sont « cachées » avec l’entête http), la méthode GET a 2 avantages :

Elle permet de construire une URL comme si l’internaute était passé par le formulaire. Par exemple, la méthode GET va permettre de transmettre un thème au script afficher\_livres\_theme.ctrl.php sans passer par un formulaire.

Dans la page themes.vue.php, un lien vers chaque thème est affiché au navigateur par des echo en php.

1. Modifier chaque echo pour que chaque lien soit de la forme :

<a href=afficher\_livres\_theme.ctrl.php?theme=0>Informatique</a>

1. Compléter ensuite le script afficher\_livres\_theme.ctrl.php pour qu’il récupère le champ theme passé par la méthode GET dans l’URL.
2. Tester en cliquant sur un thème pour afficher son identifiant transmis par la méthode GET.

Pour automatiser cela, dans la page themes.vue.php la variable $tabThemes contient les thèmes.

1. Parcourir le tableau $tabThemes pour créer chaque élément a avec l’identifiant et le thème de chaque case du tableau.
2. Tester en cliquant sur un thème pour afficher son identifiant transmis par la méthode GET.

Et également, la méthode GET permet d’accéder à une page obtenue avec les données d’un formulaire.

1. Quand vous affichez la page affichant le titre saisi ou la page du thème cliqué (deux pages obtenues par la méthode GET),

Mettre en « favori » (ou « marque-page ») la page ou copier l’URL de la barre d’adresses

Puis fermer le navigateur et rouvrir la page depuis vos favoris ou en collant l’URL

La page est-elle accessible ?

1. Quand vous affichez la page récapitulant les informations du nouveau client (une page obtenue par la méthode POST),

Mettre en « favori » (ou « marque-page ») la page ou copier l’URL de la barre d’adresses

Puis fermer le navigateur et rouvrir la page depuis vos favoris ou en collant l’URL

La page est-elle accessible ?

# Partie pour l’évaluation n°2

1. Choisir un modèle conceptuel que vous avez conçu dans le module « Modélisation » (hors le modèle de la librairie).

Domaine d’étude choisi :

1. Repérer maintenant une propriété d’une entité qui vous permettra de rechercher cette entité en base de données.

Propriété choisie :

1. Dans le fichier index.html, compléter le 4ème lien de menu avec votre entité et votre propriété choisies.
2. Créer le fichier *votredonnee*.form.html composé d’un formulaire avec un seul champ pour pouvoir saisir une valeur de la propriété choisie.
3. Créer le script php qui recevra de manière sécurisée la valeur de la propriété et qui l’affichera, pour vérification.

# *Autres formulaires web (pour aller plus loin…)*

1. Créer les scripts php pour tester les formulaires html que vous avez créés au cours des modules « HTML CSS S1 » & « Intégration web S2 ».