

TP3 : Interaction avec la BDD

Accès aux données

La classe « PDO » permet, grâce à son constructeur, la connexion à une base de donnée quelconque pour peu que l'on dispose du pilote PDO associé. La méthode `query()` est utilisée pour l'interrogation. L'objet retourné est une instance de la classe `PDOStatement`. La méthode « `fetch` » permet l'obtention du résultat enregistrement par enregistrement dans un tableau numérique ou associatif.

Exercice: Affichage d'un tableau HTML reflétant le contenu d'une table de la base de donnée par interrogation PDO en PHP.

Interrogez la base de données pour construire un tableau HTML avec le contenu de la table « client ». Utilisez les fonctions de la bibliothèque « tags.lib.php » pour afficher le tableau. Ajouter la possibilité de trier par ordre alphabétique sur chaque colonne.

Modifiez le code pour afficher une table quelconque dont le nom est obtenu par une variable `http GET`.

Résultat attendu:



Contenu de la table client

id	nom	ville
1	Martin	Calais
2	Dupond	Boulogne
3	Durand	Dunkerque

Ecriture aux données

Le langage PHP est utilisé pour la vérification des données et l'insertion dans la base via un objet PDO et la méthode « `exec()` ». Par la suite la méthode des requêtes préparée sera préférée car elle protège de l'injection SQL.

Ajout d'un enregistrement dans la table client

Un formulaire HTML en code ré-entrant doit permettre l'ajout dans la table client.

L'insertion dans la base de donnée ne sera effectuée que si les données fournies sont correctes. Utilisez ou ajoutez les fonctions `check...`() de la bibliothèque déjà construite en Tp. Le champ « nom » ne doit pas être vide et ne doit pas contenir plus de caractères que le champ dans la base.

Le champ « ville » ne peut recevoir que 'Boulogne', 'Calais', 'St-Omer' ou 'Dunkerque'.

Un message de résultat (« insertion effectuée » ou un message d'erreur) est affiché après chaque soumission.

Un lien vers la page d'affichage de la table client permettra la vérification.

Résultat attendu :

Insertion dans la table client

Nom

Ville

[Voir la table client](#)

Améliorations:

Ajoutez une fonction « `select()` » à la bibliothèque « `tags.lib.php` ».

Remplacez le champ text « ville » par une liste de choix

Ajoutez la contrainte correspondante dans la table client.

Remplacez la méthode « `exec()` » par la méthode « `prepare()` » et utilisez un `PDOStatement` pour faire correspondre paramètre et valeur (méthode « `bind_param()` »). Lors de la préparation le SQL les valeurs est remplacé par le nom du champ précédé de ' : '.

Ajoutez un champ « type » dans la table client pouvant prendre les valeurs 'G' pour 'grossiste' ou 'D' pour 'détaillant'.

Ajoutez une fonction « `radio()` » à la bibliothèque « `tags.lib.php` » et modifiez le formulaire.

Ajoutez un champ « email ». et une fonction « `check_mail()` » pour la vérification (pensez à utiliser une expression régulière).

Ajoutez un champ booléen « solvable » dans la table client et une fonction `checkbox()`.

Modifications des données

La modification nécessite de préciser l'enregistrement à modifier. La sélection doit s'effectuer sur un champ sans doublon (Clef primaire ou index unique). L'information est sélectionnée par l'utilisateur. Il peut s'agir d'une liste déroulante ou d'une case à cocher ou d'un bouton 'Modifier'.

Cette information permet alors l'affichage d'un formulaire HTML pré-rempli avec les données correspondant à l'enregistrement sélectionné.

Le bouton modification provoque la mise à jour de l'enregistrement après vérification des données.

La méthode `PDO::exec('UPDATE ...')` est utilisée pour cette modification et retourne le nombre

d'enregistrements réellement affectés.

Exercice 1 : modification d'un enregistrement dans une table (sélection par bouton)

Ajouter au code de la page d'affichage d'une table, un bouton 'Modifier', qui fournira un lien vers la page 'modification.php'. Cette page réutilisera pour partie le code php écrit pour l'insertion.

Le formulaire sera pré-rempli avec les données de l'enregistrement. Après vérification des données, la commande SQL « `UPDATE` » permettra la mise à jour de l'enregistrement.

Amélioration : Remplacer la méthode « `PDO::exec` » par une requête préparée « `PDO::prepare` » et l'utilisation d'un objet `PDOStatement` pour lier les valeurs.

Amélioration : En cas de réussite une redirection permet de retourner automatiquement à la page d'affichage, en cas d'erreur un message indiquant l'erreur sera affiché au chargement de la page

Résultat attendu:

Contenu de la table client

id	nom	ville	type	email	solvable	Action	
24	Dupond	Boulogne	G	robert.dupond@foo.net	1	Modifier	Supprimer
7	Durand	Dunkerque	G		0	Modifier	Supprimer
22	Dussart	Boulogne	G	dominique.dussart@univ-littoral.fr	1	Modifier	Supprimer
20	Gérard	Boulogne	G		0	Modifier	Supprimer
15	Légrand	Boulogne	G		0	Modifier	Supprimer
18	Martin	Boulogne	G		0	Modifier	Supprimer
19	Martin	Boulogne	G		0	Modifier	Supprimer

Exercice 2 : modification d'un enregistrement dans une table (sélection par liste)

Sur le modèle de la page d'ajout d'un membre construire un formulaire HTML en code réentrant ou la sélection utilisera une liste déroulante sur le champ « nom » qui ne sera pas modifiable. De même le champ « id » déterminé automatiquement n'est pas modifiable. En retour le script affichera un message quand à la modification réelle dans la base.

Résultat attendu:

Modification dans la table client

Id

Nom

Ville

Grossiste ☒

Détaillant ☐

Email

Solvable ☒

[Retour à la table client](#)

Suppression des données

La méthode PDO ::exec('DELETE ...') est utilisée pour cette suppression, et elle retourne le nombre d'enregistrements réellement affectés et permet de vérifier que la suppression a eu lieu.

Exercice: suppression d'un enregistrement dans la table client

En réutilisant le code de la page d'affichage de la table « client » écrire une page qui permettra la sélection d'un client et sa suppression.

Une boîte de dialogue « JavaScript » demandera confirmation avant la suppression.

Résultat attendu:

Contenu de la table client

<u>id</u>	<u>nom</u>	<u>ville</u>	Action
2	Dupond	Boulogne	Supprimer
3	Durand	Dunkerque	Supprimer
6	Dussart	Boulogne	Supprimer
4	Legrand	Calais	Supprimer
5	Lepetit	Dunkerque	Supprimer
1	Martin	Calais	Supprimer

Exercice: suppression d'un enregistrement dans la table client

A partir du code de la page de suppression simple écrire une page qui permettra la sélection d'un ou plusieurs client et leur suppression.

Une boîte de dialogue « JavaScript » demandera confirmation avant la suppression.

Une boîte de dialogue « JavaScript » précisera après suppression le nombre d'enregistrements supprimés

Résultat attendu:

Suppression dans la table client

<u>id</u>	<u>nom</u>	<u>ville</u>	Selection
2	Dupond	Boulogne	<input type="checkbox"/>
3	Durand	Dunkerque	<input type="checkbox"/>
6	Dussart	Boulogne	<input type="checkbox"/>
4	Legrand	Calais	<input type="checkbox"/>
5	Lepetit	Dunkerque	<input type="checkbox"/>
1	Martin	Calais	<input type="checkbox"/>
Supprimer			