

TP Bases de Données n°1 : création d'une base de données

PTSI Lycée Eiffel

31 janvier 2014

1 Installation d'easyphp

Comme je doute très fortement que les logiciels nécessaires à nos TP de bases de données aient été installés dans la salle B23, nous allons commencer par là : allez à l'adresse suivante

<http://www.easyphp.org/easyphp-devserver.php>

et téléchargez la version la plus récente d'easyphp. Cliquez sur tout ce qu'il faut pour que l'installation se lance effectivement, surveillez de temps en temps que ça a l'air de fonctionner, et faites les exercices qui suivent avant de lancer réellement ce que vous avez téléchargé.

2 Manipulation de bases de données « sur papier »

On souhaite faire le schéma entités/association d'une base de données modélisant une médiathèque. On considère que chaque disque de la médiathèque contient plusieurs plages, que chaque plage contient une oeuvre unique, mais qu'une oeuvre peut s'étendre sur plusieurs plages. Pour chaque plage, on associera les interprètes. On donne les attributs suivants pour chacune des entités (en gras la clé primaire) : **titre** et année pour le disque ; **numéro** et durée pour chaque plage ; **titre** et année pour les oeuvres ; **numéro d'identification**, nom, prénom pour les interprètes.

1. Effectuer le schéma entités /associations correspondant, en précisant les cardinalités.
2. On souhaite ajouter les instruments joués, en admettant que chaque artiste joue d'un seul instrument sur chaque plage où il joue. Où vaut-il mieux ajouter l'attribut instrument ?
3. Ajouter dans le schéma une entité auteur, en lui ajoutant les attributs que vous souhaitez, et en tenant compte du fait qu'une oeuvre peut avoir plusieurs auteurs.

Vous pouvez maintenant créer une base de données personnelle, dans le domaine de votre choix, mais qui satisfait aux conditions suivantes : au moins trois tables distinctes, avec dans chaque au moins deux attributs, et des relations entre tables. Remplissez chaque table avec au moins cinq instances de chaque entité.

3 Création d'un BDD sous MySQL

Il est temps de faire un peu de manipulation avec le beau joujou que nous avons installé tout à l'heure. Lancez donc ce cher easyphp. Normalement, il doit lancer une espèce de page d'aide complètement inutile au démarrage, mais s'il ne se passe rien d'autre (ou même rien du tout), c'est essentiellement normal. Vous devriez tout de même avoir désormais une belle icône easyphp dans votre barre d'outils pas loin de l'horloge. Faites un petit clic droit sur l'icône, puis sélectionnez l'onglet **Administration**. Il doit vous lancer un onglet Internet avec plein de choses bizarres dedans, dont une possibilité d'ouvrir un onglet « Administration MySQL ». Ouvrez-le donc. Voilà, ça y est, l'interface de gestion de bases de données est en route. Après vous être demandés comment on peut encore faire des choses aussi moches en 2014, tentez d'effectuer les opérations suivantes :

- Créez une base de données, à laquelle vous pouvez même donner le nom de votre choix (reprenez par exemple ce que vous avez mis au point dans le deuxième exercice de la section précédente). Oubliez la colonne *Interclassement* à côté, il n'y a que des choses incompréhensibles et inutiles pour vous dedans. Normalement, votre base de données, une fois créée, est visible dans la partie gauche de l'interface. Cliquez dessus.
- Créez dans votre base une (ou plusieurs) tables en spécifiant à chaque fois le nombre de colonnes de la table. Remplissez pour chacune le tableau de structure qui vous est proposé. Si vous ne comprenez rien à ce que vous avez sous les yeux, vous pouvez tenter, à vos risques et périls, un clic sur les petites cases contenant des points d'interrogation, qui vous renvoient en général vers un manuel MySQL relativement abscons. Sinon, vous pouvez aussi me poser des questions. Précisons quand même qu'il est indispensable de préciser la clé primaire de votre table (dans la colonne **INDEX**) et que la colonne intitulée **A_I** permet une auto-incrémentation de la valeur pour les variables de type **INT** (ainsi, si vous créez une table dont la clé primaire est un numéro, en cochant cette case, ce numéro sera automatiquement augmenté d'une unité à chaque fois que vous ajoutez une nouvelle ligne à votre table). Essayez de modifier certains des paramètres après création de votre table.
- Remplissez chacune de vos tables avec quelques données via l'onglet **Insérer**. On ne va trop se préoccuper des problèmes techniques de relations entre les différentes tables pour ce premier TP, mais essayez quand même de créer des tables ayant des liens entre elles (par exemple, comme on avait vu en cours, une table de tournois de tennis ayant une colonne avec les vainqueurs, et une autre qui contient une liste de joueurs où ces vainqueurs vont sûrement apparaître).
- Essayez d'effectuer dans votre base de données une recherche élémentaire via l'onglet **Rechercher**. Comme c'est quand même bien pourri, passez plutôt par l'onglet **Requête** en essayant un peu de comprendre par vous-même comment ça fonctionne (ce n'est pas si compliqué, et ça vous poussera à faire des tests).
- Pour les plus motivés, passez maintenant à l'onglet **SQL** et tentez d'effectuer une recherche via une requête SQL, c'est-à-dire une commande dans le langage SQL directement interprétable par votre gestionnaire de bases de données. Une telle commande contiendra les mots-clés **SELECT** (qui introduira la recherche), **FROM** qui indiquera dans quelle table il faut chercher, et **WHERE** (suivi d'une condition) qui indiquera les critères de recherche. Par exemple, `SELECT * FROM 'ptsi' WHERE 'prénom'='Lucas'` devrait vous afficher tous les éléments de la table nommée `ptsi` dont le champ `prénom` contient la chaîne de caractère `Lucas` (donc, si la table n'est pas remplie avec les pieds, la liste des élèves dont le prénom est `Lucas`). Si vous voulez sélectionner, par exemple, tous les élèves dont le prénom commence par un `L`, vous pouvez utiliser d'autres mots-clés, en remplaçant l'égalité dans la sélection précédente par une commande du style `WHERE 'prénom' LIKE 'L%'` (ce qui signifie en gros que le champ `prénom` ressemble à un `L` suivi de n'importe quoi). Nous apprendrons bien évidemment les bases du langage SQL petit à petit pour mieux comprendre le fonctionnement de ces requêtes. Pour information, on peut tout faire à l'aide de requêtes, y compris les créations et le remplissage des tables.