

OBJECTIF

Réaliser les requêtes qui permettent de modifier, supprimer ou ajouter un ou plusieurs enregistrements, en fonction des besoins exprimés.

BASE UTILISEE : GESTION DES FORMATIONS

Cette base de données représente les données structurées pour la gestion des formations au sein d'un lycée. Le MCD est fourni en dernière page

Fichiers utilisés pour la génération de la base :

BaseFormation.sql pour la génération de la structure des tables

DonneesBaseFormation.sql pour l'insertion des données

Exécutez ces deux fichiers sous MySQL en utilisant l'interface PHPMyadmin

TRAVAIL A FAIRE**1. SUPPRESSION DES FORMATIONS NON DISPENSEES AU SEIN DU LYCEE**

- A. Afficher toutes les formations qui ne sont pas assurées au sein du lycée (nombre d'heures égal à 0)

```
select *
from formation
where Nbheures =0
```

id_formation	intitule_formation	Nbheures	Niveau
21	Utilisation de l'ordinateur dans l'environnement Windows	0	Debutant
22	Utilisation du réseau	0	Debutant
27	Présentation des contenus pédagogiques en ligne	0	Courant
28	Logiciels spécialisés : langues, maths ..	0	Courant

- B. Vérifiez si il existe des professeurs qui se sont inscrits aux formations précédentes :

```
SELECT *
FROM suivreformation, formation
WHERE formation.id_formation=suivreformation.id_formation
and Nbheures=0
```

- C. Supprimer toutes les formations qui ne sont pas assurées au sein du lycée (nombre d'heures égal à 0), les inscriptions à ces formations seront aussi supprimées afin de laisser la base de données dans un état cohérent

suppression des tuples qui modélisent la relation entre les formation et les professeurs

```
delete FROM suivreformation
WHERE id_formation in (21,22,27,28);
```

suppression des tuples qui modélisent les formations non assurées par le lycée

```
delete from formation
where Nbheures =0;
```

2. DEPART D'UN FORMATEUR, QUI SERA REMPLACÉ PAR UN NOUVEAU

- A. Afficher le planning des formations, on verra apparaître (en plus des informations de la table "assurerpar") le nom et prénom des formateurs ainsi que l'intitulé des formations qu'ils assurent,

```
SELECT intitulé_formation, Nom_Formateur , Prenom_Formateur, Dateform, Groupe,
NbHSeance, NumSeance, Mat_AM, NumSalle
FROM formation, formateur, assurerpar
Where formation.id_formation= assurerpar.id_formation
And assurerpar.Id_Formateur= formateur.Id_Formateur ;
```

- B. Le formateur *Cancel Christophe* ne peut plus assurer les formations qu'il a en charge, il sera remplacé par Durant Pierre (celui n'existe pas dans la table formateur)

```
# ajout du nouveau formateur
```

```
insert into formateur
values (" , 'Durant', 'Pierre');
```

```
# affiche les formateurs: noter l'identifiant de Cancel Christophe et du nouveau formateur
```

```
SELECT * FROM formateur;
```

Formateur	Nom_Formateur	Prenom_Formateur
1	Legrand	Sarah
2	Cancel	Christophe
3	Maurel	Olivier
4	Pleuveraux	François
5	Carmignac	Laurent
6	Durant	Pierre

```
# modification: changement de formateur
```

```
update assurerpar set Id_Formateur = 6
where Id_Formateur= 2 ;
```

- C. Afficher le planning des formations afin vérifier que le changement à bien été effectué

Reprendre la requête du A

3. AJOUT D'UNE FORMATION ET INSCRIPTION

- A. Ajouter une formation intitulée "Javascript" d'une durée de 12 h niveau Confirme

```
insert into formation
values ('', ' Javascript', 12, 'Confirme' )
```

- B. Tous les inscrits à la formation "Excel" niveau "Débutant" désirent participer à la formation précédemment créée. Le groupe de chaque professeur est identique. Réaliser les requêtes qui permettent de satisfaire cette demande.

Conseil : vous pouvez créer une table "tampon" qui sera détruite à la fin

création d'une table "SUIVRE" qui contient tous les professeurs qui suivent la formation Excel

```
CREATE TABLE suivre
SELECT *
FROM suivreformation
where id_formation=23
```

modification de l'identifiant de la formation de la table "SUIVRE"

il pointe maintenant vers la nouvelle formation

```
update suivre
set id_formation=43
```

#insertion des enregistrements de la table "SUIVRE" vers la table "SUIVREFORMATION"

```
insert into suivreformation
select * from suivre
```

#affichage des inscrits à la formation "Javascript"

```
select * from suivreformation
where id_formation=43
```

#suppression de la table "SUIVRE"

```
drop table suivre
```

4. MODIFICATION DE LA DATE D'UNE FORMATION

- A. Toutes les formations intitulées 'initiation linux' sont repoussées de 9 jours

#utilisation de la fonction date_add pour ajouter 9 jours à la date des formations

```
Update assurerpar
Set Dateform = DATE_ADD(Dateform , INTERVAL 9 DAY)
where id_formation=37
```

5. SUPPRESSION DE LA FORMATION "Bases de données (Access)"

- A. La formation intitulée "Bases de données (Access)" est supprimée.
Réaliser les suppressions nécessaires pour laisser la base de données dans un état cohérent.

#affichage des informations concernant la formation à supprimer, on note son identifiant

```
SELECT *
FROM formation
where intitule_formation ='Bases de données (Access)'
```

id_formation	intitule_formation	Nbheures	Niveau
35	Bases de données (Access)	12	Confirme

```
#-suppression de toutes les inscriptions à cette formation de la table suivreformation  
delete FROM suivreformation  
where id_formation =35 ;
```

```
# suppression de la formation du planning (table assurerpar)  
delete FROM assurerpar  
where id_formation =35 ;
```

```
# suppression de la formation de la liste des formations (table formation)  
delete FROM formation  
where id_formation =35 ;
```

6. CONVERSION EN MAJUSCULE

- A. Modifier les noms des formateurs pour les mettre en majuscule

```
Update formateur  
Set Nom_Formateur = UPPER(Nom_Formateur)
```

7. MODIFICATION DE SALLE

- A. Toutes les formations qui ont lieu dans les 30 jours qui suivent le jour courant (aujourd'hui donc !!!) change de salle, elles se dérouleront toutes en salle G333
(une seule requête permet de réaliser cette modification)

```
SELECT *,( to_days(dateform) - to_days(now()) ) as ecart  
from assurerpar  
where ( to_days(dateform) - to_days(now()) ) between 0 and 30 ;
```

```
Update assurerpar  
Set numsalle='G333'  
Where ( to_days(dateform) - to_days(now()) ) between 0 and 30 ;
```

8. MODIFICATION DE LA DATE DE LA FORMATION

- A. Les formations qui auront lieu le 139ème jour de l'année seront repoussées de 3 jours
(Une seule requête permet de réaliser cette modification)

```
select dateform, DAYOFYEAR(dateform)  
from assurerpar ORDER BY DAYOFYEAR(dateform);
```

```
update assurerpar  
set dateform=date_add(dateform ,INTERVAL 3 day)  
where DAYOFYEAR(dateform)=139
```

