


Documents autorisés.  
Appareils de communication mobiles interdits..

 Licence Sciences et Technologie  
Mention Informatique

Durée 2 heures.

Technologie du Web  
mai 2005

**Exercice 1 :** (Répondre sur le formulaire)

Qu'est-ce qu'une architecture 3-tiers ? Quels sont ses avantages ? Quelle(s) question(s) de cet examen relève de l'utilisation d'une architecture 3-tiers (justifier) ?

**Exercice 2 :** (Répondre sur le formulaire) Soit la table suivante, affichée par un navigateur :

A	B
D	C

Question 1 : Donnez le code HTML de l'élément `table` qui permet d'obtenir ce résultat. Vous ne vous préoccupez ici d'aucun des paramètres de style d'affichage tels que les couleurs ou le type de cadre.

Question 2 : La case centrale (contenant "X") est colorée en orange, les autres cases en jaune. Indiquez comment obtenir ce résultat. NB : votre solution doit faire en sorte que

- le fichier HTML ne contienne aucune indication de couleur.
- la coloration ne concerne que le tableau visé. Si d'autres tableaux existent dans la même page, leur style doit être le style par défaut.

**Exercice 3 :** Une base de données comporte deux tables :

- Une table **villes** qui possède les attributs **nom** (une chaîne), **pop** (un entier : le nombre d'habitants), **latitude**, **longitude** et **pays** (chaîne). La latitude est un nombre compris entre -90 et +90 (les valeurs négatives correspondent à l'hémisphère sud), la longitude est comprise entre -180 et +180. Pays est le nom de l'état auquel appartient la ville.
- Une table **états** qui comporte les attributs **nom** (une chaîne), **capitale** (une chaîne), **continent** (une chaîne).

On supposera qu'il n'y a pas d'homonymie pour les villes ni pour les états. Chaque capitale présente dans la base est supposée être présente dans la table des villes. Chaque pays référencé dans la table des villes est supposé être présent dans la table des états.

Question 1 : (Répondre sur le formulaire)

Donnez les requêtes SQL permettant d'obtenir les résultats mentionnés ci-dessous. Vous veillerez à faire les requêtes les plus simples possibles (en évitant les jointures inutiles, par exemple). Il est inutile d'utiliser des sous-requêtes.

1. Les coordonnées géographiques de la ville V.
2. Les noms de villes de l'état E.
3. Les noms et populations des villes de l'état E, par ordre décroissant de la population.
4. Les villes du continent C.
5. Le nom de l'état auquel appartient la ville V.
6. Pour chaque état, le nom et la population de sa capitale.
7. Les noms des états ayant au moins 3 villes présentes dans la base. Chaque nom doit apparaître au plus une fois.
8. La population moyenne des villes de l'état E.
9. Les états ayant au moins une ville dans l'hémisphère Sud. Chaque nom doit apparaître au plus une fois.
10. Pour chaque continent, son nom et le nombre de villes de ce continent présentes dans la table.

Question 2 : (Répondre sur le formulaire)

- 2.a Choisissez pour chaque table une clé primaire.  
2.b Indiquez la commande de création de la tables des villes.

Question 3 : (Répondre sur la copie)

Dans les questions précédentes, nous avons supposé que chaque capitale était présente dans la table des états. Nous allons maintenant construire une requête SQL permettant de vérifier si c'est exact. Concrètement, on veut obtenir la liste des capitales qui NE SONT pas dans la table des villes, sans utiliser de sous-requêtes dans une requête `select`.

- 3.a Dans cette sous-question, nous supposons connaître  $n$  le nombre de tuples de la table des villes.

Pour une capitale  $v$  présente dans la table des villes, combien y a-t-il de couples  $(v, v')$  avec  $v' \neq v$  et  $v'$  présente dans la table de villes? Même question si  $v$  n'est pas présente dans la table des villes?

En déduire une requête SQL qui fournit le résultat demandé.

- 3.b Donner une solution complète utilisant plusieurs requêtes successives et une table intermédiaire mais ne nécessitant pas de connaître au préalable le nombre de tuples de la table des villes.

Question 4 : (Répondre sur la copie) La base est supposée s'appeler `geographie` et être hébergée sur le serveur `dbserver`.

Une interface web permet à un utilisateur de rechercher le nom d'une ville dans la base pour obtenir les informations qui la concerne. Elle permet également d'ajouter une ville dans la base. Il ne vous est pas demandé d'écrire l'ensemble des codes nécessaires, mais seulement certaines fonctions :

- 4.a Construire une fonction PHP `chercherVille` à un seul argument (qui représente une chaîne). Si cette chaîne est le nom d'une ville présente dans la table des villes, alors le résultat de la fonction est un tableau dont les clés sont les noms des attributs de la table et les valeurs sont les valeurs associées à cette ville. Si l'argument n'est pas le nom d'une ville de la table, alors le résultat est `null`.

- 4.b Construire une fonction PHP `ajouterVille` dont l'argument est un tableau associatif analogue au résultat de la fonction `chercherVille` et qui représente les données d'une nouvelle ville. La fonction `ajouterVille` ajoute ces données dans la table des villes. Le résultat est `false` si l'opération a échoué, `true` sinon.

numéro de copie :



Licence Sciences et Technologie  
Mention Informatique

Technologie du Web

Durée 2 heures.

mai 2005

Exercice 1 :

Exercice 2 : Question 1 :

Question 2 :

**Exercice 3 : Question 1 :**

1	
2	
3	
4	
5	
6	
7	
8	
9	
10	

**Question 2 :****2.a**

--

**2.b**

--