

Epreuve : Technologie Web et Technologie Objet
Durée : 2 heures

Exercice 1 (4 points)

On dispose d'un fichier XML nommé concours.xml dont le texte est le suivant :

```
<xml>
<USINE poste="1">
<Atelier numero=1>
<ouvrier nom="Talhi" prénom="Said" />
<ouvrier nom="Sedrati" prénom="Maamer" />
<technicien nom="Nezzar" prénom="Rafik" />
</Atelier><Atelier numero=3>
<ouvrier nom="Guezouli" prénom="Lyamine" />
<ouvrier nom="Guezouli" prénom="Larbi" />
<technicien nom="Hamouid" prénom="Khaled" />
<technicien nom="Malekem" prénom="Lamine" />
</Atelier>
<Atelier numero=2>
<ouvrier nom="Zidani" prénom="Madjid" />
</Atelier>
</USINE>
</xml>
```

Ce document est-il bien formé ? Est-il valide ? Expliquer pourquoi.

En admettant que ce document soit correct, écrire une transformation XSL qui compte le nombre d'ateliers et de personnes (pour notre exemple, une personne est soit un ouvrier soit un technicien).

Exercice 2 (2 points)

Créer un formulaire comprenant un groupe de champs. Le formulaire doit permettre la saisie du nom, prénom, adresse, ville et code postal.

Les données sont ensuite traitées par un fichier PHP séparé récupérant les données et les affichant avec une boucle foreach dans un tableau HTML.

Exercice 3 (4 points)

On suppose qu'on dispose de la table de données nommée Etudiant dont voici un extrait :

nom	module	contrôle	note
Nezzar	TechObjet	1	15
Nezzar	TechObjet	2	14
Abida	TechWeb	1	16
Abida	TechWeb	2	16
Frah	TechWeb	2	12
...

Donner les requêtes MySQL qui permettent de répondre aux questions suivantes:

1. Combien y a-t-il d'étudiants si on admet que dans la table Etudiant chaque valeur de "nom" différente correspond à un étudiant?
2. Quel est le "nom" du premier étudiant par ordre alphabétique ?
3. Quel étudiant a eu la meilleure de toutes les notes, quelle que soit le module ou le type de contrôle (contrôle continu (1), contrôle final (2))?
4. Quel étudiant a eu la plus mauvaise moyenne en Technologie Web (TechWeb), quelque soit le type de contrôle?

Exercice 4 (6 points):

Un transpondeur diffuse la position courante d'un avion. Nous allons le considérer dans cet exercice comme étant un objet. La position de l'avion est considérée comme un attribut de l'objet transpondeur. Une telle position est modifiée en utilisant un système de positionnement satellite. L'objet transpondeur doit être capable de modifier périodiquement la position en utilisant des données fournies par deux satellites satA et satB. Soit les attributs suivants de l'objet transpondeur :

- a. satA et satB : deux objets (de classe « satellite ») représentant deux satellites.
- b. currentPosition : la position courante de l'avion
- c. cA et cB : les coordonnées de l'avion fournies par les deux satellites satA et satB.
- d. theNavigator : un objet de classe Navigator.

La classe satellite fournit une méthode position() retournant la position de l'avion détectée par le satellite.

La classe Navigator fournit la méthode "compute (coordA, coordB)". Cette dernière permet de calculer la position de l'avion en se basant sur les coordonnées, coordA et coordB, qui sont fournies par les deux satellites satA et satB.

On vous demande de :

1. expliquer la différence entre serveurs et objets actifs,
2. justifier clairement que l'objet transpondeur doit être un objet actif,
3. écrire une classe thread en Java pour des objets actifs transpondeurs (Mettez les commentaires nécessaires à la compréhension de votre solution).

Exercice 5 (4 points):

Soient des objets de classes quelconques tels que nombres réels, chaînes de caractères, et graphes. Nous voulons définir une seule opération « add » qui permettra d'additionner deux nombres réels, de concaténer deux chaînes de caractères ou de composer deux graphes. On vous demande de :

Proposer une conception de l'opération add dans une modélisation orientée objet. Votre solution doit être facilement extensible pour prendre en compte d'autres classes d'objets, autres que celles explicitées ci-dessus. Les notions fondamentales que vous utiliseriez dans votre conception doivent être expliquées clairement,

Fournir un squelette (ébauche ou esquisse) d'une implémentation (en Java) de votre conception.

L'implémentation doit mettre en évidence la prise en compte des classes des objets à additionner ainsi que l'extensibilité à d'autres classes.