

Examen de Bases de Données Avancées. Mercredi 06 Février 2013 - Durée: 2h.

Aucune erreur de syntaxe ne sera tolérée.

Questions de cours 10 points

1. On veut modéliser l'aspect statique d'une application. Justifier l'utilisation de l'un des deux formalismes graphiques existants (comparaison).
2. Dans le cas de la modélisation avec l'approche objet, comment vous justifiez le passage au modèle relationnel ou au modèle relationnel objet pour implémenter la base de données ?
3. Donner la définition d'un type incomplet. Dans quels cas son utilisation devient impérative ?
4. Donner un exemple (en SQL3) concret et complet explicitant l'utilité des types incomplets.
5. Expliquer avec un exemple concret le principe de la persistance (le principe du mapping) ainsi que ses types.
6. Comment créer le mécanisme faisant la persistance des objets sauvegardés dans une table (Relationnel - non 1FN, Relationnel-Objet) ?
7. Citer les restrictions d'utilisation des concepts de déclaration de l'héritage et de l'instanciation.
8. Donner un exemple concret pour expliciter ces restrictions.

Exercice 10 points

On souhaite implémenter sous Oracle une base de données permettant de faire le suivi des sportifs lors des compétitions internationales multisports (type jeux olympiques) du point de vue des contrôles antidopage. Le schéma SQL3 de cette base de données est le suivant :

```
CREATE TYPE T_Control AS OBJECT ( NumCtrl NUMBER(6), Type VARCHAR(15), DateCtrl DATE, Resultat CHAR(1))
```

/

```
CREATE TYPE TAB_Control AS TABLE OF T_Control
```

/

```
CREATE TYPE T_Athlete AS OBJECT ( NumDossard NUMBER(5), Nom VARCHAR(20), Prenom VARCHAR(20), Controles TAB_Control)
```

/

```
CREATE TYPE TAB_Athlete AS TABLE OF T_Athlete
```

/

```
CREATE TYPE T_Equipe AS OBJECT( NumEq NUMBER(4), CodePays VARCHAR(3), Sport VARCHAR(15), Athletes TAB_Athlete)
```

/

```
CREATE TABLE Equipe OF T_Equipe (CONSTRAINT pk_equipe PRIMARY KEY (NumEq)) NESTED TABLE Athletes STORE AS Stockage_athletes (NESTED TABLE Controles STORE AS Stockage_controles);
```

On a peuplé la table Equipe avec les données ci-après.

NumEq	CodePays	Sport	NumDossard	Nom	Prénom	NumCtrl	Type	DateCtrl	Result
1	USA	100 mètres	1	Green	Maurice	1	Sanguin	03/11/05	N
			2	Lewis	Carl	2	Sanguin	03/11/05	N
2	USA	Basket	23	Jordan	Michael	3	Urinaire	03/11/05	N
			8	Bryant	Kobe	4	Urinaire	03/11/05	N
			34	O'Neal	Shaquille	5	Urinaire	03/11/05	N
3	UK	100 mètres	4	Sphinx	Le	6	Sanguin	03/11/05	P
						7	Sanguin	04/11/05	N
4	UK	Aviron	5	Smith	John	8	Urinaire	03/11/05	N
			6	Smith	Jack	9	Urinaire	03/11/05	P
						10	Urinaire	04/11/05	P
5	FRA	Aviron	9	Dupond	Albert	11	Urinaire	03/11/05	N
			7	Martin	Maurice	12	Urinaire	03/11/05	N
6	FRA	Basket	10	Bilba	Jim	13	Urinaire	03/11/05	N
			11	Parker	Tony	14	Urinaire	03/11/05	N
			12	Diaw	Boris				

1) Insertion et interrogation

Ecrire les requêtes permettant de :

- Montrer comment les équipes 1, 4 et 6 ont été insérées (codes).
- Lister les équipes (CodePays, Sport) avec les athlètes (Nom, Prénom) les constituants.
- Dresser la liste des contrôles effectués par l'athlète n° 4 de l'équipe n° 3.
- Trouver le nombre de contrôles positifs par athlète.
- Afficher la date du dernier contrôle pour chaque athlète.
- Lister l'(es) équipe(s) la(es) plus contrôlée(s).

2) Mise à jour des tables imbriquées

- Insérer l'athlète <30, 'Pietrus', 'Michael', pas de contrôle> dans l'équipe n° 6.
- Insérer les contrôles <15, 'Urinaire', '23/01/2007', 'N'> et <16, 'Sanguin', '23/01/2007', 'N'> pour l'athlète n° 30 de l'équipe n° 6.
- Modifier à « Négatif » le résultat des contrôles de l'athlète n° 6 de l'équipe n° 4 passés avant le 04/11/2005.

Bon Courage.

Corrigé de l'examen de Bases de Données Avancées.

Mercredi 06 Février 2013 - Durée: 2h.

Questions de cours (10 Points)

1. On veut modéliser l'aspect statique d'une application. Justifier l'utilisation de l'un des deux formalismes graphiques existants (comparaison).

Le modèle Entité/Association

Simple et pratique : il n'y a que 3 concepts : entités, associations et attributs.

Il permet de modéliser rapidement des structures pas trop complexes.

Passage au relationnel et (programmation non orienté objet)

1.00 Pt

.....

Modèle de classes

Il permet de modéliser des structures plus complexes.

Passage au relationnel objet

Programmation orienté objet

1.00 Pt

.....

2. Dans le cas de la modélisation avec l'approche objet, comment vous justifiez le passage au modèle relationnel ou au modèle relationnel objet pour implémenter la base de données ?

Passage au relationnel :

1. *Avantage du relationnel*

Simple, modèle formel

Première forme normale

0.75 Pt

Passage au Relationnel-objet

1. *Inconvénients du relationnel*

Jointure complexe (reconstruire un objet complexe)

Première forme normale

0.75 Pt

2. *Avantage du relationnel objet*

Hybridation des concepts de l'orienté objet (héritage, méthodes, pointeurs, ...) avec ceux du relationnel (structure tabulaire, colonne, tuple, ...)

3. Donner la définition d'un type incomplet. Dans quels cas son utilisation devient impérative ?

Un type incomplet est un ADT déclaré sans corps, on fait appel à ce genre de déclaration lors de l'existence de références croisées

1.00 Pt

4. Donner un exemple (en SQL3) concret et complet explicitant l'utilité des types incomplets.

On ne peut pas créer un type qui utilise un type qui n'a pas encore été créé. Que faire quand on a des références croisées (par exemple, un type ville qui pointe vers un type pays et un

type pays qui pointe sur un type ville) ? On déclare alors un type incomplet puis on le mettra à jour plus tard.

Exemple

```
CREATE TYPE t_ville  
/ -- création du type incomplet
```

1.00 Pt

```
CREATE TYPE t_pays AS OBJECT  
( ... ,  
... ,  
ville REF t_ville,  
... )  
/
```

```
CREATE OR REPLACE t_ville AS OBJECT -- modification du type incomplet  
( ... ,  
... ,  
... ,  
pays REF t_pays,  
... ,  
... )  
/
```

5. Expliquer avec un exemple concret le principe de la persistance (le principe du mapping) ainsi que ses types.

La persistance est le fait de pouvoir sauvegarder (persister) les objets créés après la fin du programme.

Type : 1. Colonne, 2. ligne, 3. transitoire

1.00 Pt

6. Comment créer le mécanisme faisant la persistance des objets sauvegardés dans une table (Relationnel - non IFN, Relationnel-Objet) ?

En déclarant le type : INSTANTIABLE: création du constructeur du type.

1.00 Pt

7. Citer les restrictions d'utilisation des concepts de déclaration de l'héritage et de l'instanciation.

- 1^{ère} Rest : Un type NOT INSTANTIABLE ne peut pas être défini FINAL ;
- 2^{ème} Rest : Un sous-type NOT INSTANTIABLE doit hériter d'un type INSTANTIABLE.

1.00 Pt

8. Donner un exemple concret pour expliciter ces restrictions.

Exemple : avec erreurs :

```
CREATE TYPE Pilote_type UNDER Employe_type  
(nbHvol NUMBER, compagnie VARCHAR(6))  
NOT INSTANTIABLE FINAL (Restriction 1)
```

0.50 Pts

```
CREATE OR REPLACE TYPE Employe_type AS OBJECT  
(codemp CHAR(6), nom VARCHAR(30), datenais DATE, paye NUMBER(6,2))  
NOT INSTANTIABLE NOT FINAL
```

1.00 Pts

```
CREATE TYPE Pilote_type UNDER Employe_type  
(nbHvol NUMBER, compagnie VARCHAR(6))  
NOT INSTANTIABLE FINAL
```


Corrigé de l'examen de Bases de Données Avancées.

Mercredi 06 Février 2013 - Durée: 2h.

Exercice (10 Points)**1) Insertion et interrogation****1. Montrer comment les équipes 1, 4 et 6 ont été insérées**

05
 INSERT INTO Equipe VALUES(1, 'USA', '100 metres', TAB_Athlete(T_Athlete(1, 'Green', 'Maurice', TAB_Control(T_Control(1, 'Sanguin', '3/11/2005', 'N'))), T_Athlete(2, 'Lewis', 'Karl', TAB_Control(T_Control(2, 'Sanguin', '3/11/2005', 'N')))));

25
 INSERT INTO Equipe VALUES(4, 'UK', 'Aviron', TAB_Athlete(T_Athlete(5, 'Smith', 'John', TAB_Control(T_Control(8, 'Urinaire', '3/11/2005', 'N'))), T_Athlete(6, 'Smith', 'Jack', TAB_Control(T_Control(9, 'Urinaire', '3/11/2005', 'P'), T_Control(10, 'Urinaire', '4/11/2005', 'P'))));

15
 INSERT INTO Equipe VALUES(6, 'FRA', 'Basket', TAB_Athlete(T_Athlete(10, 'Bilba', 'Jim', TAB_Control(T_Control(13, 'Urinaire', '3/11/2005', 'N'))), T_Athlete(11, 'Parker', 'Tony', TAB_Control(T_Control(14, 'Urinaire', '3/11/2005', 'N'))), T_Athlete(12, 'Diaw', 'Boris', TAB_Control()));

2. Liste des équipes (CodePays, Sport) avec les athlètes (Nom, Prénom) les constituants

1
 SELECT e.CodePays, e.Sport, a.Nom, a.Prenom FROM Equipe e, TABLE(e.Athletes) a;

Ou bien

SELECT e.CodePays, e.Sport, CURSOR (SELECT a.Nom, a.Prenom FROM TABLE(e.Athletes) a) FROM Equipe e;

3. Liste des contrôles effectués par l'athlète n° 4 de l'équipe n° 3.

1
 SELECT * FROM TABLE (SELECT a.Controles FROM TABLE (SELECT e.Athletes FROM Equipe e WHERE e.NumEq = 3) a WHERE a.NumDossard = 4);

Ou bien

SELECT c.NumCtrl, c.Type, c.DateCtrl, c.Resultat FROM Equipe e, TABLE (e.Athletes) a, TABLE (a.Controles) c WHERE e.NumEq = 3 AND a.NumDossard = 4;

4. Nombre de contrôles positifs par athlète

1
 SELECT a.Nom, a.Prenom, COUNT(*) FROM Equipe e, TABLE(e.Athletes) a, TABLE(a.Controles) c WHERE c.Resultat = 'P' GROUP BY a.Nom, a.Prenom;

5. Date du dernier contrôle pour chaque athlète

SELECT a.Nom, a.Prenom, MAX(c.DateCtrl) FROM Equipe e, TABLE(e.Athletes) a, TABLE(a.Controles) c GROUP BY a.Nom, a.Prenom;

6. Équipe(s) la(es) plus contrôlé(es)

SELECT e.NumEq, e.CodePays, e.Sport FROM Equipe e, TABLE(e.Athletes) a, TABLE(a.Controles) GROUP BY e.NumEq, e.CodePays, e.Sport HAVING COUNT(*) = (SELECT MAX(COUNT(*)) FROM Equipe e1, TABLE(e1.Athletes) a1, TABLE(a1.Controles) GROUP BY e1.NumEq);

2) Mises à jour des tables imbriquées

a) Insérer l'athlète <30, 'Pietrus', 'Michael', pas de contrôle> dans l'équipe n° 6

INSERT INTO TABLE (SELECT Athletes FROM Equipe e WHERE e.NumEq = 6) VALUES (T_Athlete (30, 'Pietrus', 'Michael', TAB_Controlé()));

b) Insérer les contrôles <15, 'Urinaire', '23/01/2007', 'N'> et <16, 'Sanguin', '23/01/2007', 'N'> pour l'athlète n° 30 de l'équipe n° 6

INSERT INTO TABLE (SELECT a.Controles FROM Equipe e, TABLE (e.Athletes) a WHERE e.NumEq = 6 AND a.NumDossard = 30) VALUES (T_Controlé(15, 'Urinaire', '23/01/2007', 'N'));

INSERT INTO TABLE (SELECT a.Controles FROM TABLE (SELECT Athletes FROM Equipe WHERE NumEq = 6) a WHERE a.NumDossard = 30) VALUES (T_Controlé(16, 'Sanguin', '23/01/2007', 'N'));

c) Modifier à « Négatif » le résultat des contrôles de l'athlète n° 6 de l'équipe n° 4 passés avant le 04/11/2005

UPDATE TABLE (SELECT a.Controles FROM Equipe e, TABLE(e.Athletes) a WHERE e.NumEq = 4 AND a.NumDossard = 6) c SET c.Resultat = 'N' WHERE c.DateCtrl < '04/11/2005';