

**L2 INF245 — Examen — mai 2014,  
session 1  
des solutions**



## 1 A propos de relations

### Question 1 (4 points) :

Donner, sous forme tabulaire, le schéma et la valeur de la relation construite par chacune des requêtes ci-dessous (il peut s'agir d'un message d'erreur Oracle).

1. `select C, B, D  
from S where D <> 3;`

Erreur : D inconnu dans S.

2. `select A, B  
from R where B < 20;`

| A | B  |
|---|----|
| 1 | 11 |
| 3 | 10 |

3. `select R.A, R.B, S.C  
from R left outer join S on (R.B=S.B);`

| A | B  | C   |
|---|----|-----|
| 1 | 11 | 101 |
| 1 | 11 | 100 |
| 2 | 20 |     |
| 8 |    |     |
| 4 | 21 |     |
| 3 | 10 | 101 |
| 3 | 10 | 100 |

4. `select A, count(distinct B) as nbB,  
count(distinct C) as nbC  
from R natural join S group by A;`

| A | nbB | nbC |
|---|-----|-----|
| 1 | 1   | 2   |
| 3 | 1   | 2   |

### Question 2 (3 points) :

Reformuler en SQL chacune des requêtes listées ci-dessous. Suivre très rigoureusement les instructions fournies.

1. `select A, B from R where B in  
(select B from S where C > 100);`

```
select R.A, R.B
from R join S on (R.B = S.B)
where S.C > 100;
```

Autre version :

```
select A, B
from R natural join S
where C > 100;
```

2. `select B from S group by B having count(distinct C) = 1;`

```
select B from S
minus
select S1.B
from S S1 join S S2
on (S1.B = S2.B and S1.C <> S2.C)
```

## 2 A propos d'une bibliothèque

### 2.3 Expression de requêtes

1. Donner les adhérents en dépassement (c'est-à-dire ceux qui ont au moins un emprunt en cours depuis plus de 15 jours).

```
select distinct noAdh from LesEmprunts
where datRet is null and sysdate-datEmp >= 15
```

2. Pour chaque adhérent, donner le nombre de livres empruntés (dans le passé et au présent). On dit qu'un livre est emprunté dès lors qu'un exemplaire de ce livre est emprunté. Le résultat est 0 pour les adhérents sans emprunt.

```
select noAdh, count (distinct titre) as nbLivres
from LesAdherents left outer join LesEmprunts using (noAdh)
      left outer join LeFonds using (cote)
group by noAdh
```

3. Pour chaque livre, donner son titre et la cote de l'exemplaire de ce livre emprunté le plus souvent.

```
-- pour chaque titre, et pour chaque exemplaire de ce livre, combien d'emprunts ?
select titre, cote, count(*) as nbEmp
from LeFonds natural join LesEmprunts
group by titre, cote
-- la requête qui utilise R :
select titre, cote
from (select titre, cote, count(*) as nbEmp
      from LeFonds natural join LesEmprunts) R
where (titre, nbEmp) in
      (select titre, max(nbEmp)
       from (select titre, cote, count(*) as nbEmp
             from LeFonds natural join LesEmprunts) R
       group by titre)
```

Ou bien :

```
-- pour chaque titre, et pour chaque exemplaire de ce livre, combien d'emprunts ?
select titre, cote, count(*) as nbEmp
from LeFonds natural join LesEmprunts
group by titre, cote
-- la requête qui utilise R:
select titre, cote
from (select titre, cote, count(*) as nbEmp
      from LeFonds natural join LesEmprunts) R
natural join
(select titre, max(nbEmp) as max
 from (select titre, cote, count(*) as nbEmp
       from LeFonds natural join LesEmprunts) R
 group by titre) R1
-- <t, max> ap R1 <=> max est le nombre maximal d'emprunts pour le livre de titre t
on (R.nbEmp=R1.max)
```

## 2.4 Implantation de l'application

Nous considérons la base de données telle que spécifiée Section 2.1. Le code SQL permettant de définir les relations LesAdhérents et LeCatalogue est donné ci-dessous :

### Question 3 (2 points) :

Fournir le code SQL Oracle qui permet de définir les relations LesEmprunts et LeFonds.

```
create table LeFonds ( cote integer, titre varchar(50) constraint Fds_c0 not null,
    constraint Fds_c1 primary key (cote),
    constraint Fds_c2 foreign key (titre)
        references LeCatalogue (titre) ) ;

create table LesEmprunts ( noAdh integer, datEmp date, datRet date, cote integer,
    constraint Emp_c1 primary key (noAdh, datEmp),
    constraint Emp_c2 unique (noAdh, datRet),
    constraint Emp_c3 foreign key (cote) references LeFonds (cote),
    constraint Emp_c4 foreign key (noAdh) references LesAdherents (noAdh),
    constraint Emp_c5 check (datEmp < datRet) );
```

### Question 4 (2 points) :

Le programme AjoutAdherent-action.php écrit en PhP et dont le squelette est donné page 4 permet d'enregistrer une nouvelle adhésion et de construire un message destiné à l'utilisateur qui l'informe de l'issue de cette adhésion. Compléter le programme AjoutAdherent-action.php (utiliser la page fournie avec le sujet).

```
/* AjoutAdherent-action.php enregistre l'adhésion d'un nouvel adhérent.
 * Génère un message indiquant si l'opération s'est bien
 * déroulée ou un message d'erreur.
 * Hypothèse : la base de données est cohérente.
 */
<?php
    $noAdh = $_POST['noAdh']; $nom = $_POST['nom'];
    $prenom = $_POST['prenom']; $adresse = $_POST['adresse'];
    $anNais = $_POST['anNais'];
    // construction de la requete
    $requete = "insert into LesAdherents values(:a,:n,:p,:ad,to_date(:an,'YYYY'), sysdate);
    $curseur = oci_parse ($lien, $requete);
    oci_bind_by_name($curseur,":a",$noAdh); oci_bind_by_name($curseur,":n",$nom);
    oci_bind_by_name($curseur,":p",$prenom);
    oci_bind_by_name($curseur,":ad",$adresse);
    oci_bind_by_name ($curseur, ":an", $anNais);
    $res = oci_execute ($curseur, OCI_NO_AUTO_COMMIT);
    if ($res) {
        oci_commit($lien);
        $message = "Nouvel adhérent enregistré.";
    } else {
        oci_rollback($lien);
        $e = oci_error($curseur);
        $error = getConstraintName(e) ;
        if ($error = "ad_C0") { /* violation d'une contrainte de clef primaire */
            $message = "Erreur : le numéro d'adhérent (". $noAdh. ") existe déjà.";
        } else { /* Autre erreur SQL */
            $message = "Erreur : ". $e['message'];
        }
    }
    echo $message;
    Deconnexion ($lien);
?>
```

**Question 5 (4 points) :**

Le programme AjoutEmprunt-action.php écrit en PHP et dont le squelette est donné page 5 permet d'enregistrer un nouvel emprunt et de construire un message destiné à l'utilisateur qui l'informe de l'issue de cette adhésion. Compléter le programme AjoutEmprunt-action.php (utiliser la page fournie avec le sujet).

```

/* AddEmprunt-action.php enregistre l'emprunt de l'exemplaire cote par l'adhérent noAdh.
 * La valeur de cote et celle de noAdh ont été saisies par le biais d'un formulaire.
 * Génère un message indiquant si l'opération s'est bien déroulée ou un message
 * d'erreur.
 * Hypothèse : la base de données est cohérente.
 */
<?php
$noAdh = $_POST['noAdh'];
$cote = $_POST['cote'];
$reqTestNb = "select noAdh from LesEmprunts
              where noAdh = :n and datRet = null
              group by noAdh having count(*) >= 4";
$curseurTestNb = oci_parse ($reqTestNb);
oci_bin_by_name ($curseurTestNb, :n, $noAdh);
$resTestNb = oci_execute($curseurTestNb, OCI_NO_AUTO_COMMIT);

if (oci_fetch($curseurTestNb) {
    $message = "Nombre d'emprunts dépassé.";
} else {
    $requete = "insert into LesEmprunts values(:c, to_date(sysdate, 'DD-MM-YYYY'), null, :a)";
    $curseur = oci_parse($requete);
    oci_bind_by_name ($curseur, ":c", $cote);
    oci_bind_by_name ($curseur, ":a", $noAdh);
    $res = oci_execute ($curseur, OCI_NO_AUTO_COMMIT);
    if ($res) {
        $message = "Nouvel emprunt enregistré.";
        $oci_commit($lien);
    } else {
        $oci_rollback($lien);
        $e = oci_error($lien);
        $error = getConstraintName(e);
        if ($error = "Emp_c1") { /* violation d'une contrainte de clef primaire */
            $message = "Erreur : l'emprunt existe déjà: l'exemplaire de cote "
                . $cote . " est déjà emprunté. ";
        }
        if ($error = "Emp_c4") { /* violation d'une contrainte référentielle */
            $message = "Erreur : l'adhérent ".$noAdh." n'existe pas.";
        }
        if ($error = "Emp_c3") { /* violation de */
            $message = "Erreur : l'exemplaire ". $cote . " n'existe pas.";
        }
        if ($error<>"Emp_c1" and $error<>"Emp_c2" and $error<>"Emp_c3") {
            /* Autre erreur SQL */
            $message = "Erreur : ".$e['message'] ;
        }
    }
}
echo $message;
Deconnexion ($lien);
?>

```