

## TP1

### Partie 1

Soit le schéma de la base de données relationnelle suivant :

**Employe** (numEmploye, nom, prenom, age, adresse)

**Sport** (numSport, nomSport, numLocal, salle)

**Inscription** (numSport, numEmploye, date, nbreHeures)

**Local-de-Sport** (numLocal, nomLocal)

Inscription modélise l'inscription d'un employé identifiée par num Employe à un sport identifié par numSport. Un sport est géré dans un Local-Sport identifié par numLocal.

**Exprimez, lorsque cela est possible, les requêtes suivantes en algèbre relationnelle, en calcul relationnel de tuples et en SQL.**

1. Donnez les numéros, noms et prénoms des employés inscrits dans les sports gérés par le local de nom «Victoire».
2. Donnez les numéros des employés qui ne sont inscrits à aucun sport du Local n° 1.
3. Numéros des sports assurés dans les salles 1 ou 2.
4. Numéros des sports aux quels sont inscrits tous les employés.
5. Numéros des employés inscrits à tout sport du local n° 1.

### Partie 2

1. **A partir des énoncés ci-dessus, créez les tables base de données et procédez à leurs alimentations par les données suivantes :**

**Employe** (numEmploye, nom, prenom, age, adresse)

(100, 'ABID', 'Mohamed', '35', 'SBA');  
(101, 'BEY', 'Amina', '41', 'SBA');  
(102, 'MIR', 'Ilyas', '29', 'Oran');  
(103, 'DIAFI', 'Adel', '25', 'Oran');  
(104, 'AMAR', 'Ali', '38', 'Oran');  
(105, 'BENSAID', 'Salima', '39', 'Alger');  
(106, 'NOUR', 'Khadija', '49', 'Alger');  
(107, 'NACER', 'Fatima', '29', 'Oran');

....

**Sport (numSport, nomSport, numLocal, salle)**

('S1', 'Aérobic', 'L1', '3');  
('S2', 'Athlétisme', 'L1', '1');  
('S3', 'Basket-ball', 'L2', '1');  
('S4', 'Hand-ball', 'L3', '2');

....

**Inscription (numSport, numEmploye, date, nbreHeures)**

('S1', '101', '15-01-2017', 3);  
('S1', '100', '25-02-2019', '6');  
('S2', '100', '20-03-2017', '3');  
('S1', '104', '10-05-2019', '4');  
('S3', '102', '15-01-2018', '4');  
('S1', '103', '25-10-2017', '6');  
('S4', '100', '10-02-2019', '6');  
('S1', '107', '20-12-2018', '8');  
('S2', '104', '10-07-2019', '4');  
('S3', '100', '05-06-2019', '6');  
('S4', '105', '30-04-2017', '6');  
('S1', '102', '12-02-2019', '6');  
('S1', '106', '05-10-2018', '6');  
('S1', '105', '21-09-2019', '4');  
('S1', '105', '06-09-2021', '4');

....

**Local-de-Sport (numLocal, nomLocal)**

('L1', 'Compétition');  
('L2', 'Victoire');  
('L3', 'Challenge');

....

2. Une fois les tables créées et alimentées, implémentez les requêtes SQL exprimées dans la partie 1.
3. Utilisez la commande explain plan pour calculer les coûts d'exécutions des requêtes. Comparez et commentez les résultats.

**La commande explain plan s'écrit :**

**explain plan for**

.... La requête sql.....

**select plan\_table\_output**

**from table(DBMS\_XPLAN.DISPLAY('plan\_table', null, 'typical'))**

**Corrigé du TP1**  
**(Expressivité des langages de requêtes)**

**Soit la base de données relationnelle de schéma :**

Employe (numEmploye, nom, prenom, age, adresse)

Sport (numSport, nomSport, numLocal, salle)

Inscription (numSport, numEmploye, date, nbreHeures)

LocaldeSport (numLocal, nomLocal)

**Expression des requêtes en Calcul relationnel de tuples (CRT)**

1. *Donnez les numéros, noms et prénoms des employés inscrits dans les sports gérés par le local de nom «Victoire».*

$\{e.numEmploye, e.nom, e.prenom / \exists e \text{ Employe}(e) \wedge \exists s \text{ Sport}(s) \wedge \exists l \text{ Local\_de\_Sport}(l) \\ \wedge \exists i \text{ Inscription}(i) \wedge (e.numEmploye=i.numEmploye) \wedge (i.numSport= s.numSport) \\ \wedge (s.numLocal= l.numLocal) \wedge (l.nomLocal= 'Victoire' )\}$

2. *Donnez les numéros des employés qui ne sont inscrits à aucun sport du local n° 1.*

$\{ e.numEmploye / \exists e \text{ Employe}(e) \wedge \neg (\exists i \text{ Inscription}(i) \wedge \exists s \text{ Sport}(s) \wedge (i.numSport= s.numSport) \wedge (s.numLocal='L1') \wedge (i.numEmploye = e.numEmploye) )\}$

3. *Numéros des sports assurés dans les salles 1 ou 2.*

$\{s.numSport / \exists s \text{ Sport}(S) \wedge (s.salle='1' \vee s.salle='2')\}$

4. *Numéros des sports aux quels sont inscrits tous les employés.*

$\{i.numSport / \exists i \text{ Inscription}(i) \wedge \forall e (\text{Employe}(e) \Rightarrow (\exists i' \text{ Inscription}(i') \\ \wedge (e.numEmploye= i'.numEmploye) \wedge (i'. numSport = i. numSport ) )\}$

Transformations :

$\{i.\text{numSport} / \exists i \text{Inscription}(i) \wedge \forall e \neg (\text{Employe}(e) \vee (\exists i' \text{Inscription}(i') \wedge (e.\text{numEmploye} = i'.\text{numEmploye}) \wedge (i'.\text{numSport} = i.\text{numSport})))\}$

On obtient :

$\{i.\text{numSport} / \exists i \text{Inscription}(i) \wedge \neg (\exists e (\text{Employe}(e) \wedge \neg (\exists i' \text{Inscription}(i') \wedge (e.\text{numEmploye} = i'.\text{numEmploye}) \wedge (i'.\text{numSport} = i.\text{numSport}))))\}$

5. Numéros des employés inscrits à tout sport du local n° 1.

$\{i.\text{numEmploye} / \exists i \text{Inscription}(i) \wedge \forall s (\text{Sport}(s) \wedge \text{Sport.numLocal} = 'L1') \Rightarrow (\exists i' \text{Inscription}(i') \wedge (s.\text{numSport} = i'.\text{numSport}) \wedge (i'.\text{numEmploye} = i.\text{numEmploye}))\}$

De la même manière que la requête 4, il faut appliquer les transformations pour obtenir une expression exprimable en sql.

**Expression des requêtes en algèbre relationnelle**

1. Donnez les numéros, noms et prénoms des employés inscrits dans les sports gérés par le local de nom «Victoire».

$\pi_{\text{numemploye}, \text{nom}, \text{prenom}} ((\text{employe} \bowtie_{Q1} \text{inscription}) \bowtie_{Q2} \text{sport}) \bowtie_{Q3} \sigma(\text{localdesport} ; \text{nomlocal} = "Victoire"));$

où:  $Q1 : \text{employe.numemploye} = \text{inscription.numemploye}$   
 $Q2 : \text{inscription.numsport} = \text{sport.numsport}$   
 $Q3 : \text{sport.numlocal} = \text{localdesport.numlocal}$

2. Donnez les numéros des employés qui ne sont inscrits à aucun sport du Local n° 1.

$\pi_{\text{numEmploye}} (\text{Employe}) - \pi_{\text{numEmploye}} (\text{Inscription} \bowtie_{Q2} \sigma(\text{Sport}; \text{numLocal} = 'L1')).$

3. Numéros des sports assurés dans les salles 1 ou 2.

Solution 1

$\pi_{\text{numSport}} (\sigma(\text{Sport}; (\text{salle} = '1') \vee (\text{salle} = '2')))$

Solution 2

$\pi_{\text{numSport}} (\sigma(\text{Sport}; \text{salle} = '1') \cup \sigma(\text{Sport}; \text{salle} = '2'))$

4. Numéros des sports aux quels sont inscrits tous les employés.

$\pi_{\text{numEmploye}, \text{numsport}} (\text{Inscription}) \div \pi_{\text{numEmploye}} (\text{Employe})$

5. *Numéros des employés inscrits à tout sport du local n° 1.*

$\pi_{\text{numEmploye, numSport}}(\text{Inscription}) \div \pi_{\text{numSport}}(\sigma(\text{Sport; numLocal} = \text{'L1'}))$ .

### Expression des requêtes en SQL

1. *Donnez les numéros, noms et prénoms des employés inscrits dans les sports gérés par le local de nom «Victoire».*

```
select numemploye, nom, prenom
from Employe where numemploye in
      (select numemploye from inscription where numsport in
        ( select numsport from sport where numlocal in
          ( select numlocal from localesport where nomlocal = 'victoire' )));
```

2. *Donnez les numéros des employés qui ne sont inscrits à aucun sport du Local n° 1.*

### Trois solutions possible

1.

```
select numEmploye from Employe
minus (select numEmploye from Inscription, Sport
where (Inscription.numSport=Sport.numSport and Sport.numLocal='L1'));
```

2.

```
select numEmploye from Employe
where numEmploye not in (select numEmploye from Inscription, Sport
where (Inscription.NumSport=Sport.numSport and
Sport.numLocal='L1'));
```

3.

```
select e.numEmploye from Employe e
where not exists(select i.numEmploye from Inscription i, Sport s where
(i.NumSport=s.numSport and s.numLocal='L1' and i.numEmploye=e.numEmploye));
```

3. *Numéros des sports assurés dans les salles 1 ou 2.*

1.

```
select numSport from Sport where (salle='1') OR (salle='2');
```

2.

```
select numSport from Sport where salle='1'
union select numSport from Sport where salle='2';
```

4. *Numéros des sports aux quels sont inscrits tous les employés.*

**Select distinct i.numSport from Inscription i  
where not exists(select \* from Employe e  
where not exists(select \* from Inscription j  
where i.numSport=j.numSport and e.numEmploye = j.numEmploye));**

*5. Numéros des employés inscrits à tout sport du local n° 1.*

**select distinct i.numEmploye from Inscription i  
where not exists( select \* from Sport s  
where s.numLocal='L1' and not exists(select \* from Inscription j  
where i.numEmploye=j.numEmploye and s.numSport=j.numSport));**