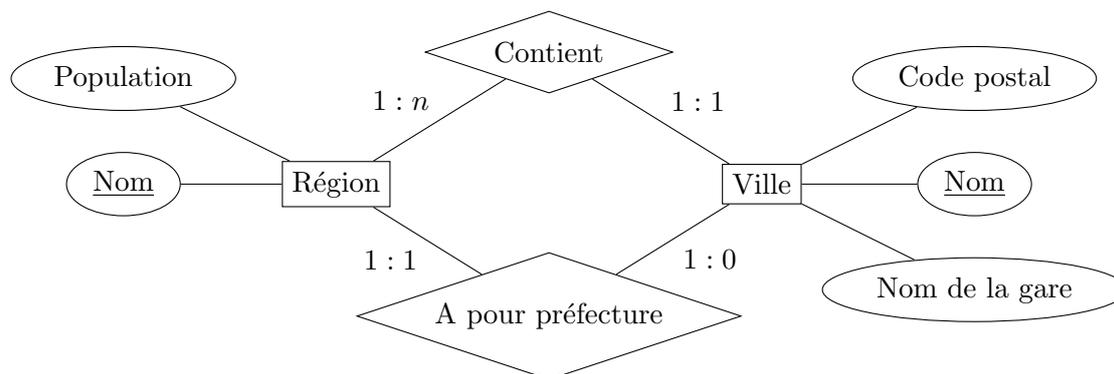


1 Exemple d'une base de donnée

Voici un exemple d'un schéma Entité-Association.



Un schéma de base de données associé peut être :

```
REGION(Nom, population)
Ville(Nom, Nom_Gare, Code_Postale)
```

Les contraintes peuvent être de plusieurs types. Cette liste n'est pas exhaustive :

- *Contraintes de propriété-attribut* : forme, liste de valeurs, fourchette de valeurs possibles, stabilité des propriétés.
- *Contrainte d'intégrité Fonctionnelle* : une des entités est totalement déterminée par la connaissance d'une ou plusieurs autres entités.
- *Contraintes inter-association* : exclusivité (même instance ne peut participer à deux associations), simultanéité (toute instance qui participe à une association participe à l'autre), totalité (toutes les instances d'une association doivent participer à au moins d'une des associations d'une collection d'association), inclusion...

Par exemple une contrainte de propriété-attribut : le Code_Postal doit être un nombre à 5 chiffres. Contrainte d'intégrité Fonctionnelle : les régions déterminent une ville (la préfecture). Contraintes inter-association : totalité : les entités de villes doivent être dans Region.

Les règles de bonne conduite :

1. Chaque entité-type doit être décrite par des propriétés qui lui sont propres.
2. Distinguer entre l'élément type et l'élément individuel représenté par l'occurrence.
3. Déterminer la ou les propriétés jouant le rôle d'identifiant. Cette propriété doit permettre de distinguer entre les occurrences de l'entité type.
4. Définir les cardinalités pour les entités types impliquées dans une association
5. Mettre en évidence les associations non porteuses d'informations
6. Une association type avec une cardinalité 1,1 ne peut pas être porteuse de propriété type.

2 TP

Des requêtes pour réviser le partiel du 23 Mars.

2.1 En vrac

1. Listez les identifiants (pid) des joueurs américains ayant participé à au moins une édition du French Open.

Solution:

```
SELECT pid FROM
  atp.player_big NATURAL JOIN
  atp.registration_big NATURAL JOIN
  atp.played_in_big
WHERE
  code='USA' AND
  tid in
  (SELECT tid FROM atp.tournament_big WHERE name='French Open');
```

2. Listez les éditions du French Open remportées par un joueur suisse.

Solution:

```
SELECT tid FROM
  atp.tournament_big NATURAL JOIN
  atp.game_big NATURAL JOIN
  atp.match_results_big
WHERE
  name='French Open' AND
  round=numrounds AND
  winner IN
  (SELECT registrnum FROM atp.registration_big NATURAL JOIN
    atp.player_big
    WHERE code='SUI');
```

3. Listez les identifiants des joueurs n'ayant jamais joué contre Nadal.

Solution:

```
WITH P1 AS (SELECT P1.PID, P1.name, M.Mid
FROM atp.player_big as P1,
  atp.registration_big AS R1,
  atp.player_big as P2,
  atp.registration_big AS R2,
  atp.game_big AS M
WHERE P1.PID = R1.PID and
  R1.registrnum in (M.registrnum1,M.registrnum2) and
  R2.registrnum in (M.registrnum1,M.registrnum2) and
  P2.PID = R2.PID and P2.last_name = 'Nadal')

SELECT P.name
FROM atp.player_big AS P
EXCEPT
SELECT P1.name
FROM P1
```

4. Listez toutes les nationalités qui ont participé à une finale de l'Australian Open.

Solution:

```

WITH AusFinal AS
(SELECT mid FROM atp.game_big NATURAL JOIN atp.tournament_big
      WHERE name='Australian Open' and numrounds=round)

SELECT DISTINCT code FROM
  atp.player_big NATURAL JOIN
  atp.registration_big JOIN
  atp.match_results_big ON winner=registrnum
WHERE mid in (SELECT * FROM AusFinale);

```

5. Créez une vue TookPart qui a deux colonnes, mid et registrnum, et qui contient tous les couples (m,r) tel que le numéro d'enregistrement r a participé au match m.

Solution:

```

CREATE VIEW TookPart AS (
(SELECT mid, registrnum1 AS registrnum FROM atp.game_big)
UNION
(SELECT mid, registrnum2 AS registrnum FROM atp.game_big)
);

```

6. Listez les identifiants des matchs (mid) opposant un joueur espagnol à un joueur américain.

Solution:

```

WITH
Am AS (SELECT registrnum FROM
      atp.registration_big NATURAL JOIN
      atp.player_big
      WHERE code='USA'),

Es AS (SELECT registrnum FROM
      atp.registration_big NATURAL JOIN
      atp.player_big
      WHERE code='USA')

SELECT mid FROM atp.game_big WHERE
  (registrnum1 in (SELECT * FROM Am) AND
  registrnum2 in (SELECT * FROM Es))
OR
  (registrnum1 in (SELECT * FROM Es) AND
  registrnum2 in (SELECT * FROM Am));

```

7. Créez une requête préparée qui étant donnés deux codes pays, liste les matchs opposant ces nationalités.

Solution:

```

PREPARE matchs_code(varchar,varchar) AS
WITH
A1 AS (SELECT registrnum FROM

```

```
atp.registration_big NATURAL JOIN
atp.player_big
WHERE code=$1),

A2 AS (SELECT registrnum FROM
atp.registration_big NATURAL JOIN
atp.player_big
WHERE code=$2)

SELECT mid FROM atp.game_big WHERE
(registrnum1 in (SELECT * FROM A1) AND
registrnum2 in (SELECT * FROM A2))
OR
(registrnum1 in (SELECT * FROM A1) AND
registrnum2 in (SELECT * FROM A2));
```

8. Listez les identifiants des matchs (mid) opposant un joueur espagnol à un joueur américain dont le joueur espagnol est sorti vainqueur.

Solution:

```
WITH
Am AS (SELECT registrnum FROM
atp.registration_big NATURAL JOIN
atp.player_big
WHERE code='USA'),

Es AS (SELECT registrnum FROM
atp.registration_big NATURAL JOIN
atp.player_big
WHERE code='USA')

SELECT mid FROM
atp.match_results_big NATURAL JOIN atp.game_big
WHERE
winner in (SELECT * FROM Es) AND
(registrnum1 in (SELECT * FROM Am) OR
registrnum2 in (SELECT * FROM Am));
```

2.2 Avec un peu d'agrégation

1. Quels tournois comptent-ils le plus d'inscrits ?

Solution:

```
WITH
Tour AS (SELECT T.Tid, count(P.registrNum) AS nbr
FROM atp.tournament_big AS T NATURAL JOIN
atp.played_in_big AS P
GROUP BY T.Tid)

SELECT tid FROM Tour
WHERE nbr = (SELECT max(nbr) FROM Tour);
```

2. Combien y a-t-il eu de matchs abandonnés à l'Australian Open de 2007?

Remarque 1 Pensez à utiliser `EXTRACT(YEAR FROM column)`.

Solution:

```
SELECT COUNT(mid) FROM
  atp.retired_match_big NATURAL JOIN
  atp.game_big
WHERE
  tid IN (SELECT tid FROM atp.tournament_big
          WHERE name='Australian Open' AND
                EXTRACT(YEAR FROM startdate)=2007);
```

3. Combien Grosjean a-t-il gagné de matchs en moyenne par tournoi?

Solution:

```
WITH
GrosNum AS (SELECT registrnum FROM atp.player_big NATURAL JOIN
            atp.registration_big
            WHERE last_name='Grosjean'),
GrosWin AS (SELECT tid, COUNT(mid) as w
            FROM atp.match_results_big NATURAL JOIN
            atp.game_big
            WHERE winner in (SELECT * FROM GrosNum)
            GROUP BY tid)

SELECT AVG(w) FROM GrosWin;
```

4. Donnez la liste des identifiants de matchs (mid) avec une colonne contenant le nombre de jeux et une colonne contenant le nombre de sets joués lors de ce match.

Solution:

```
WITH
score AS (SELECT mid, SUM(winnergames)+SUM(losergames) as ngame
          FROM atp.setscore_big GROUP BY mid)
SELECT mid, ngame, numsets FROM
  score NATURAL JOIN
  atp.match_results_big;
```

3 TD

On désire gérer les emplois du temps des différents personnels (*enseignants, enseignants-chercheurs et chercheurs*) de l'UFR, pour savoir à tout moment s'il est possible de les joindre, et où. Pour cela, on considère que, en dehors des périodes où ils peuvent être joints dans leur bureau, les personnels peuvent être en réunion, quel que soit leur statut. Une réunion est désignée par une date précise, une tranche horaire et une salle de réunion. On veut connaître les autres personnes participant à la réunion.

Chaque personne est désignée par son nom, son prénom, le bureau où on peut la joindre.

Les *enseignants* peuvent, de plus, être en cours. Un cours est identifié par la matière enseignée à laquelle est affectée toujours la même salle. Il est désigné par une période de

début et de fin (ex. de février à mai), un jour de la semaine, une tranche horaire et une salle de cours. Plusieurs enseignants peuvent enseigner la même matière dans l'année, à des jours et créneaux horaire différents. Un enseignant peut enseigner plusieurs fois la même matière dans l'année, à des périodes différentes.

Les *chercheurs* peuvent être à certaines périodes de l'année en mission en dehors de l'UFR. Une mission est désignée par une date de début et de fin, un lieu de mission avec le numéro de téléphone correspondant. Les chercheurs appartiennent à un laboratoire dont on peut joindre le secrétariat en cas d'urgence.

Les *enseignants-chercheurs* sont à la fois enseignants et chercheurs, avec un pourcentage plus ou moins grand d'enseignement (par rapport à la recherche) à effectuer. Ils peuvent donc être soit en réunion, soit en mission, soit en cours.

Établir le schéma Entité-Association de cette application. En déduire un schéma de base de données relationnelle sous forme de pattes de corbeaux. Ne pas oublier de préciser les cardinalités des associations et les identificateurs des entités. N'oubliez pas de respecter les règles de bonnes conduites.

Réfléchissez aux contraintes que vous pouvez imposer sur cette base de données.