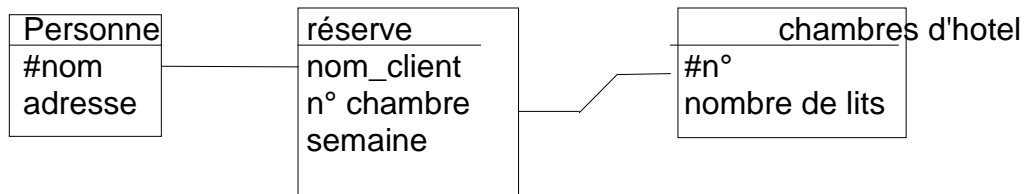


## BD40 TD 2 sur SQL,

### Base de données des réservations de chambres d'hôtel: en ACCESS



Problème 1: faire une requête sélection qui affiche les chambres disponibles une semaine donnée (infaisable en mode graphique)

- 1.1. faire une requête R1 qui affiche les chambres occupées. Observer l'instruction SQL. Faire ressortir le fait que "SELECT " définit un ensemble E, et que cet ensemble peut donc intervenir dans une autre instruction.

**E = SELECT [n°\_chambre] FROM réserve WHERE semaine= [quelle semaine?]**

- 1.2. Faire la requête R2 qui sélectionnera les chambres d'hôtel "NOT IN E".

a) faire une requête qui sélectionne toutes les chambres de l'hôtel

**SELECT [chambres\_d'hôtel].n° FROM [chambres\_d'hôtel]**

b) enlever les chambres réservées cette semaine:

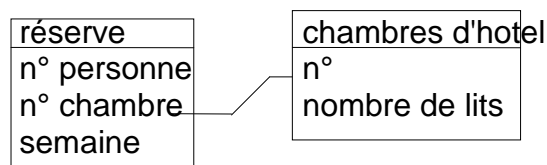
**SELECT [chambres\_d'hôtel].n° FROM [chambres\_d'hôtel]**

**WHERE chambres.n° NOT IN**

**SELECT [n°\_chambre] FROM réserve WHERE semaine= [quelle semaine?]**

- 1.2. Faire constater qu'il y a une deuxième version:

en laissant la jointure réserve.n°chambre <---> chambre.n°



qui génère une clause INNER JOIN.

**SELECT [chambres\_d'hôtel].n° FROM**

**[chambres\_d'hôtel] INNER JOIN réserve ON [n°chambre]=n°**

**WHERE chambres.n° NOT IN**

**(SELECT [n°\_chambre] FROM réserve WHERE semaine= [quelle semaine?])**

Dans cette version, un numéro de chambre n'apparaît que si la chambre a été réservée une autre semaine que celle demandée, mais pas les chambres jamais réservées.

Discuter la solution de remplacer INNER JOIN par LEFT JOIN.

Discuter la solution:

```
SELECT DISTINCTROW [chambres d'hotel].n°
FROM [chambres d'hotel], réserve
WHERE ((([chambres d'hotel].n°) NOT IN
(SELECT réserve.[n° chambre] FROM réserve
WHERE ([semaine]=[quelle semaine?]))));
```

En fait c'est idiot de faire le **produit cartésien** pour ne s'intéresser qu'à ce qui est distinct!

Problème 2: annulation des réservations: il ne faut pas enlever les réservations de la table car on veut garder un historique. faire une saisie contrôlée: créer une table des annulations de réservations: il y a une contrainte: les annulations doivent correspondre à des réservations (même chambre, même semaine, même personne)

```
INSERT INTO annulations ( personne, [date], [n° chambre] )
SELECT DISTINCTROW annulation_tmp.personne, annulation_tmp.date,
annulation_tmp.[n° chambre]
FROM annulation_tmp, annulations, réserve
WHERE (((Exists (SELECT DISTINCTROW [nom client], chambre, semaine
FROM réserve
WHERE
(
AND ( [nom client]= annulation_tmp.personne)
AND ( chambre= annulation_tmp.[n° chambre])
AND ( semaine= annulation_tmp.date )
)
))<>False)
AND ((Not Exists (SELECT DISTINCTROW [nom client], chambre, [numéro de
semaine]
FROM annulations
WHERE
(
AND ( [nom client]= annulations.personne)
AND ( chambre= annulations.[n° chambre])
AND ( semaine= annulations.date )
)
))<>False));
```