



# Cours 4

## Relations M à N, sub- SELECT, GROUP BY ... HAVING

BINV1030-1

Gestion de données : bases



# Agenda du jour

- Relations de M à N
- Sous-requêtes
- GROUP BY ... HAVING ...
- bd3.albums



# Relations de M à N



# Table commandes

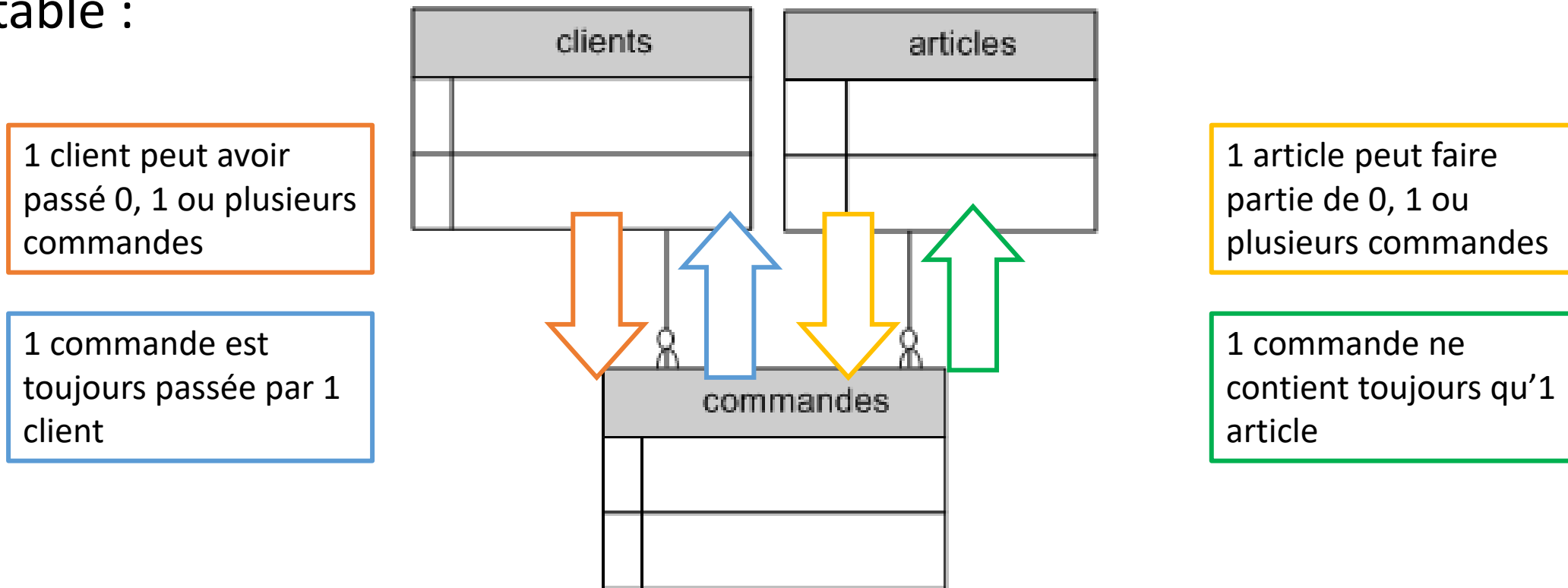
numcommande	jour	mois	année	numclient	article	quantite	prixunitaire
1	5	9	2000	1	stylo M3	20	150
2	6	9	2000	18	styloM3	80	150
3	6	9	2000	18	cartouche	160	20
5	8	9	2000	3	classeurL	10	80
8	1	10	2000	1	stylo M3	30	150

Redondance → prévoir un CATALOGUE D'ARTICLES



# L'idée : créer une relation de M à N (p. 19)

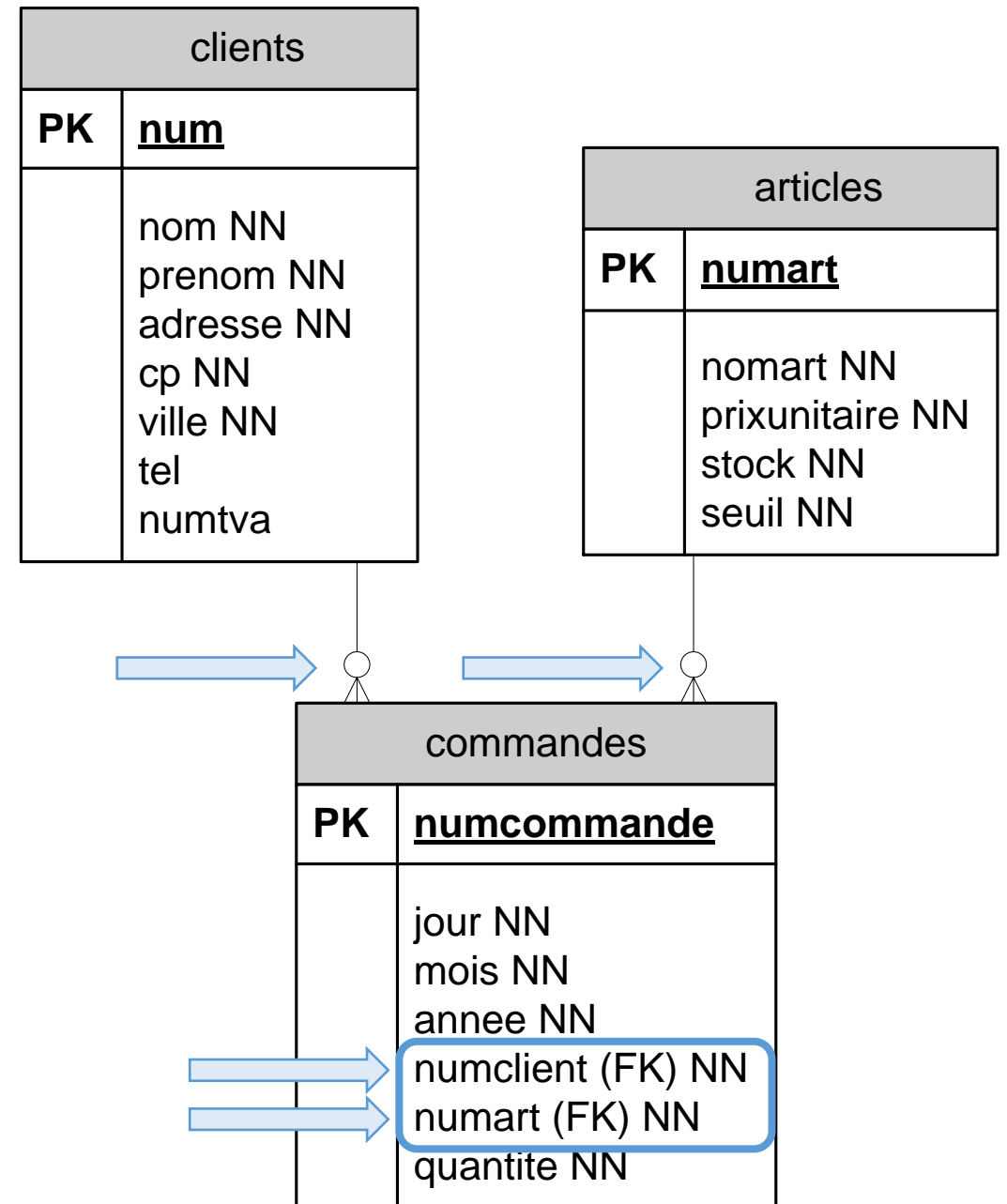
On pourrait extraire cette information et la stocker dans une autre table :





# Relation de M à N

2 relations de 1 à N  
→ 2 FK

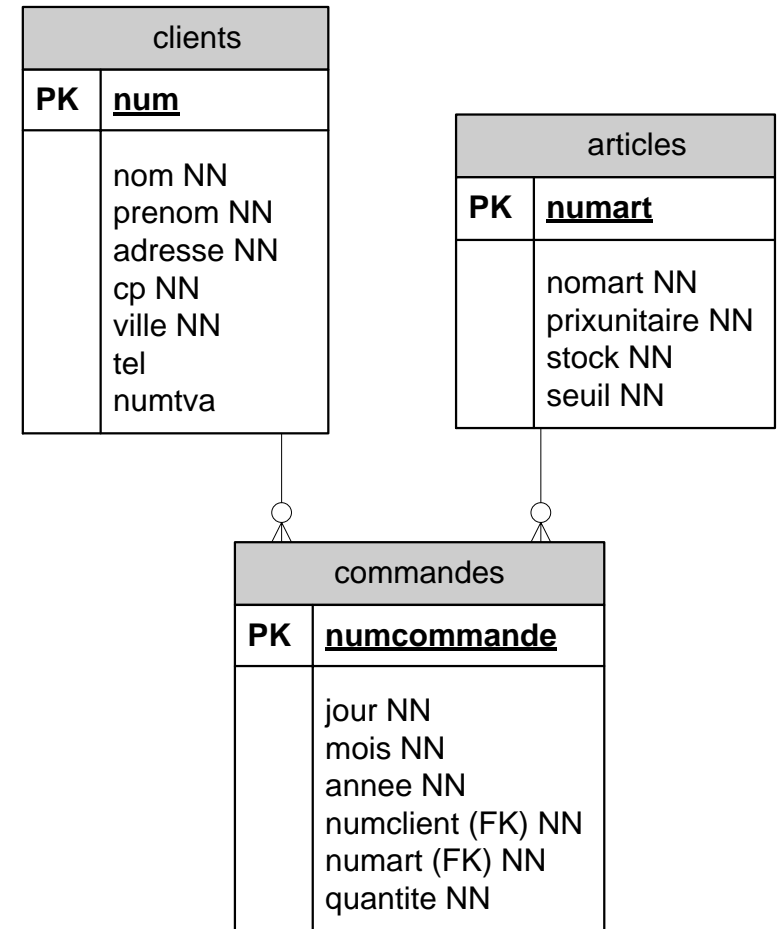




# CREATE TABLE (p. 20)

```
CREATE TABLE articles
(numart      int          PRIMARY KEY,
nomart       varchar(30)  NOT NULL,
prixunitaire int          NOT NULL,
stock        int          NOT NULL,
seuil        int          NOT NULL);
```

```
CREATE TABLE commandes
(numcommande int          PRIMARY KEY,
jour         int          NOT NULL,
mois         int          NOT NULL,
annee        int          NOT NULL,
numclient    int          NOT NULL REFERENCES clients(num),
numart       int          NOT NULL REFERENCES articles(numart),
quantite     int          NOT NULL);
```





# Contenu des tables

commandes :

numcommande	jour	mois	année	numclient	numart	quantite
1	5	9	2000	1	3	20
2	6	9	2000	18	3	80
3	6	9	2000	18	4	160
5	8	9	2000	3	5	10
8	1	10	2000	1	3	30

articles :

numart	nomart	prixunitaire	stock	seuil
1	stylo M1	100	200	100
2	stylo M2	130	125	100
3	stylo M3	150	150	100
4	cartouche	20	950	200
5	classeur L	80	23	20
6	classeur XL	150	12	20

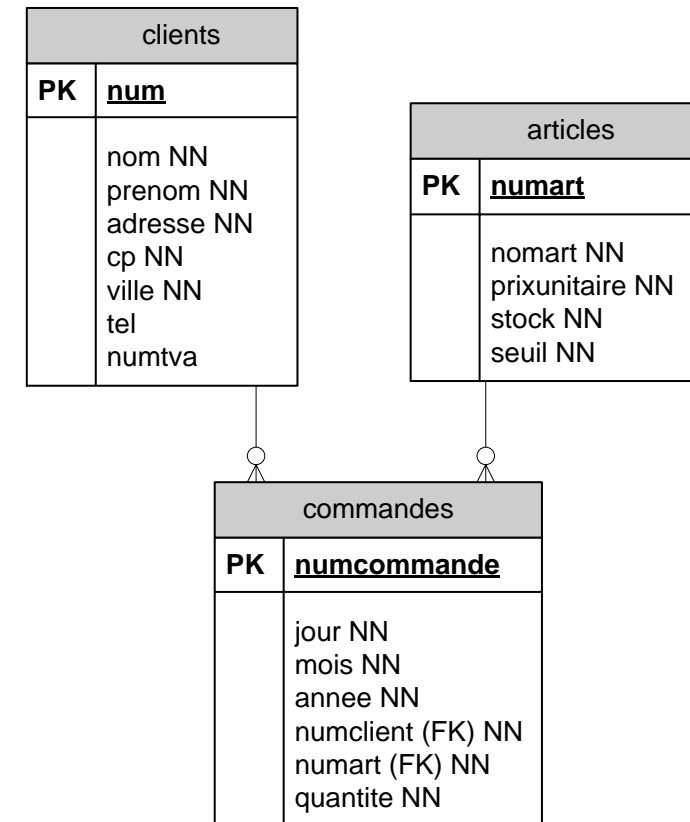




# Exemple

- Quels sont les clients de Wavre qui ont commandé en l'an 2000 des "stylo M3", et en quelle quantité? Donnez le numéro de commande et la quantité commandée ainsi que les nom, prénom et numéro de client

```
SELECT co.numcommande, co.quantite,  
       cl.nom, cl.prenom, cl.num  
FROM clients cl, commandes co,  
      articles ar  
WHERE cl.num = co.numclient  
      AND co.numart = ar.numart  
      AND cl.ville = 'Wavre'  
      AND co.annee = 2000  
      AND ar.nomart = 'stylo M3'
```





# Jusqu'à présent dans la table commandes :

commandes :

numcommande	jour	mois	année	numclient	numart	quantite
1	5	9	2000	1	3	20
2	6	9	2000	18	3	80
3	6	9	2000	18	4	160
5	8	9	2000	3	5	10
8	1	10	2000	1	3	30

clients	
PK	<u>num</u>
	nom NN prenom NN adresse NN cp NN ville NN tel numtva

articles	
PK	<u>numart</u>
	nomart NN prixunitaire NN stock NN seuil NN

commandes	
PK	<u>numcommande</u>
	jour NN mois NN annee NN numclient (FK) NN <b>numart (FK) NN</b> quantite NN

**MAIS** toujours que  
1 seul article par commande



# Papeterie Kes'Kiako

Rue de la Page, 58  
8954 Clairefontaine

## BON DE COMMANDE

Num : 2016256398

**CLIENT :** Jean BON

Adresse : Rue du Saucy Son, 48

8596 CERVELAS

0489/256398

DATE du bon de comm

Lignes de  
commande

Article	Prix unitaire	Quantité	Total
cartouche	0,2 €	15	3€
classeurL	0,8€	20	16€
stylo M3	1,50€	15	22,5€
TOTAL :			41,5€

Signature du client pour accord :

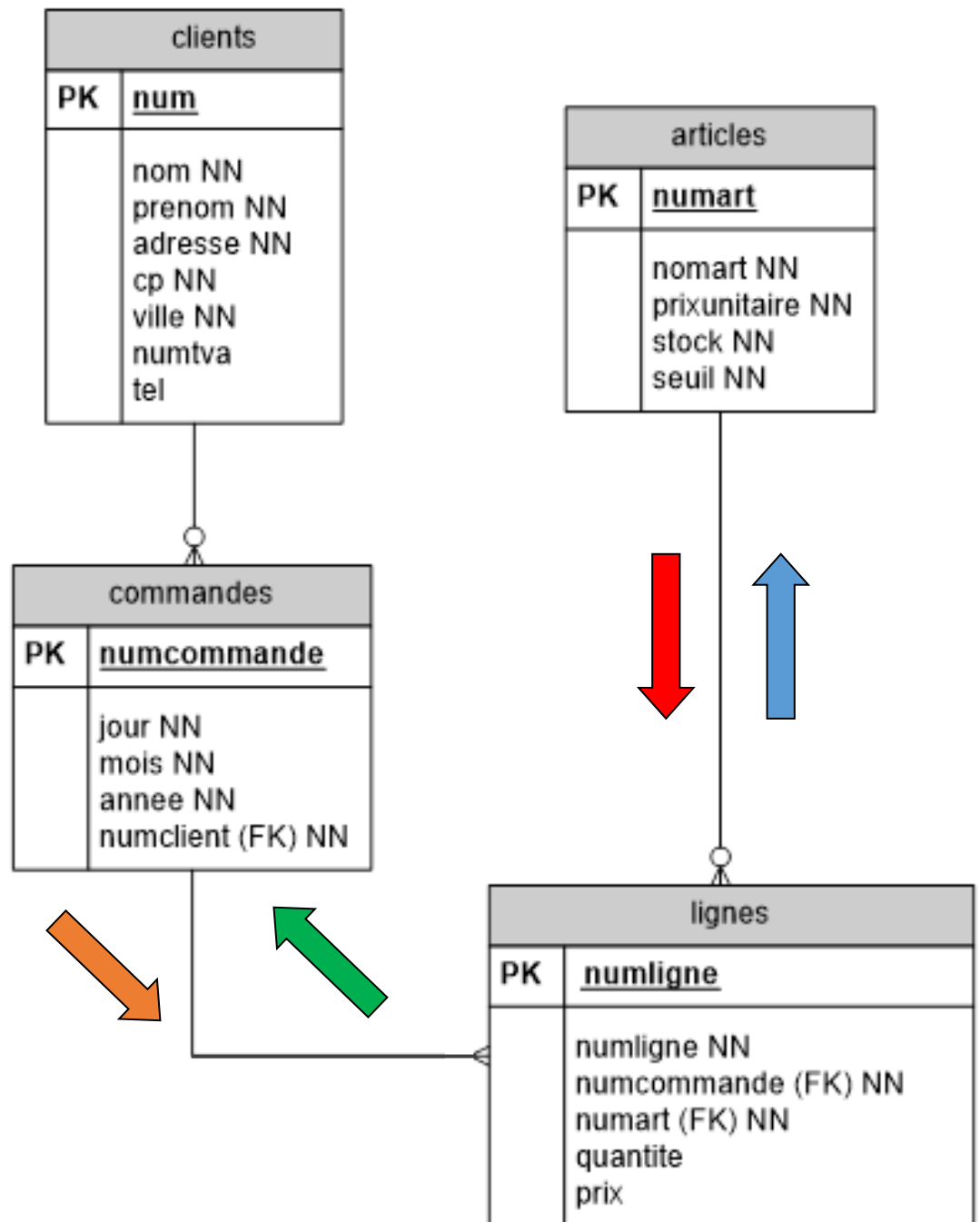
Signature de l'employé :



# Solution

Table lignes de commandes :

- Une commande a 1 ou plusieurs lignes
- Une ligne fait partie d'une seule commande
- Un article peut être commandé 0, 1 ou plusieurs fois
- Une ligne de commande porte sur un seul article

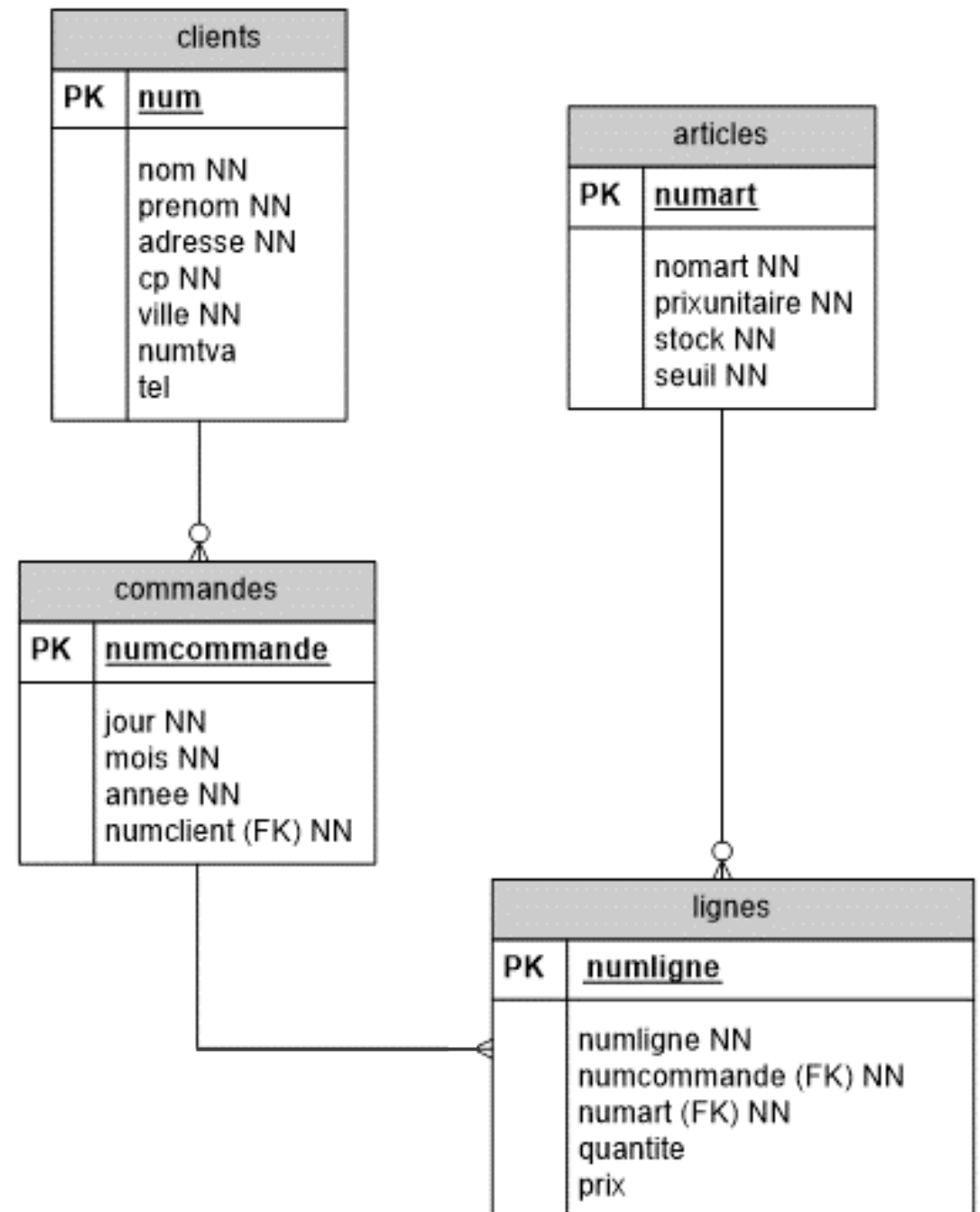




# Choix de la PK (1) (p. 22)

num ligne	num commande	numart	quantite	prix
1	1	3	4	5
2	1	4	1	4
3	2	3	2	5
...				

1 seule numérotation des lignes

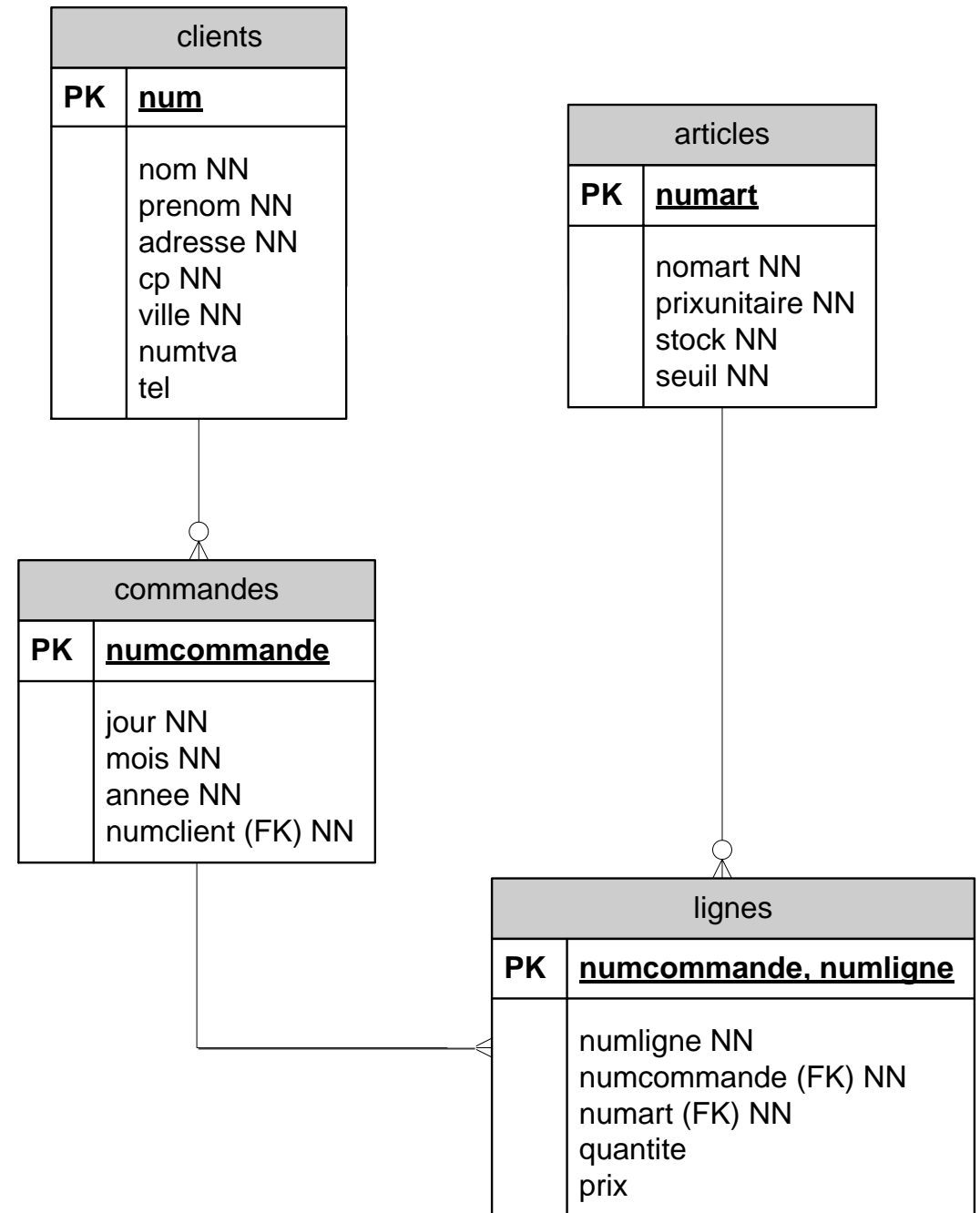




# Choix de la PK (2)

num ligne	num commande	numart	quantite	prix
1	1	3	4	5
2	1	4	1	4
1	2	3	2	5
...				

1 numérotation des lignes  
par commande

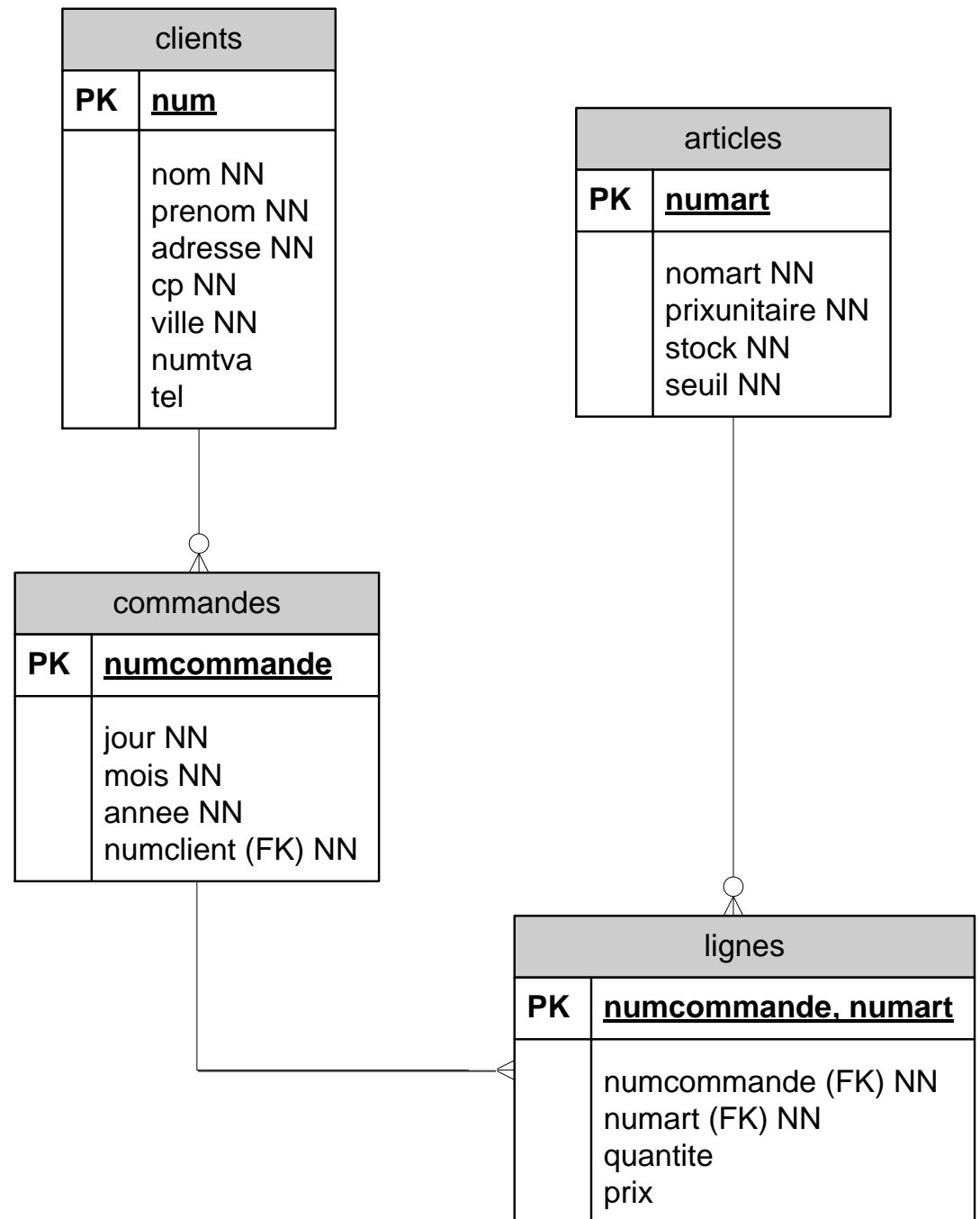




# Choix de la PK (3)

num commande	numart	quantite	prix
1	3	4	5
1	4	1	4
1	5	2	5
1	3	1	5

1 article ne peut être commandé  
qu'une seule fois par commande  
**Contraignant !!**

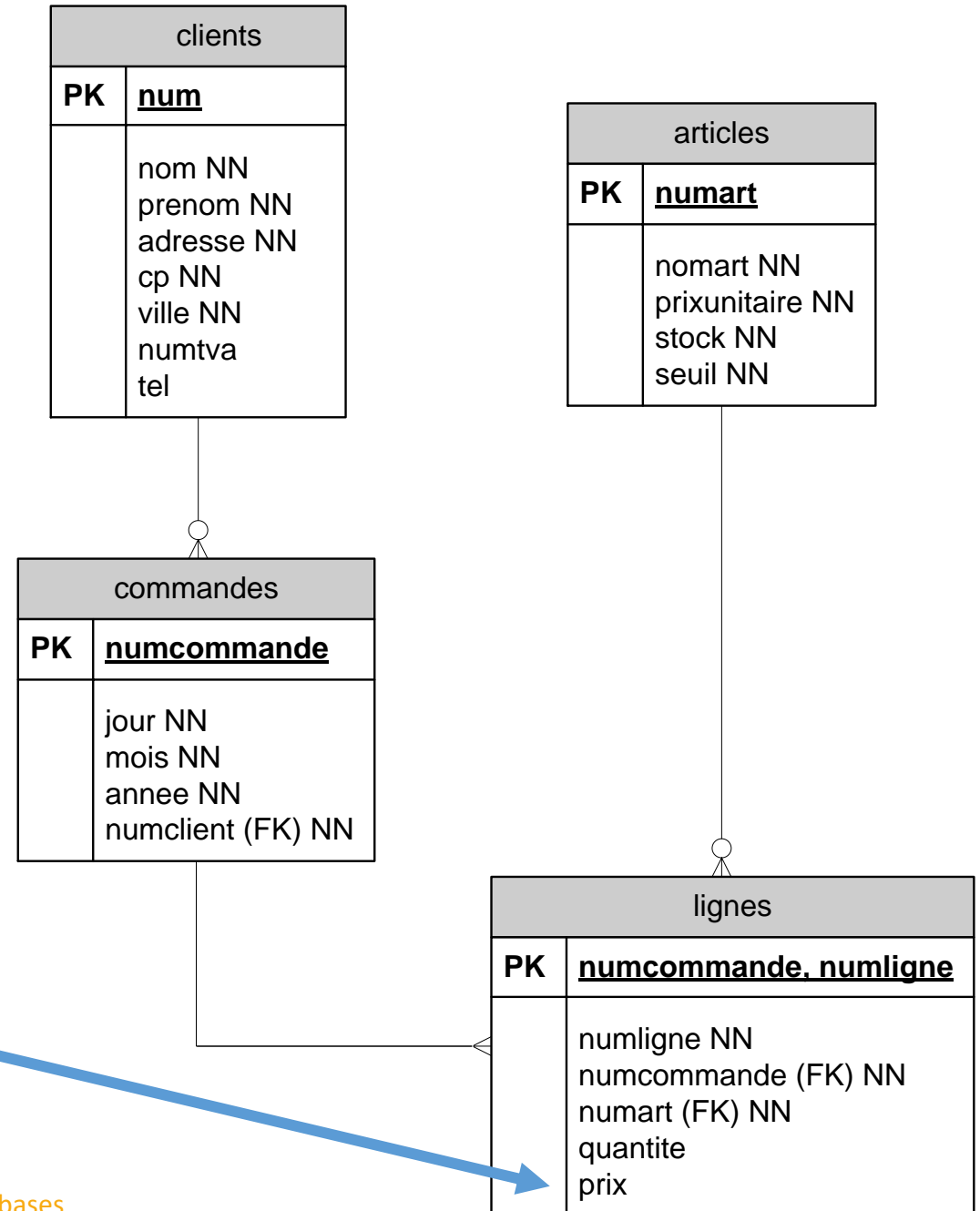




# Choix de la PK : conclusion

- Si aucune contrainte n'est exprimée dans l'énoncé, la solution 2 sera la solution préférée.

A quoi sert ce prix ?







# CREATE TABLE lignes (p. 23)

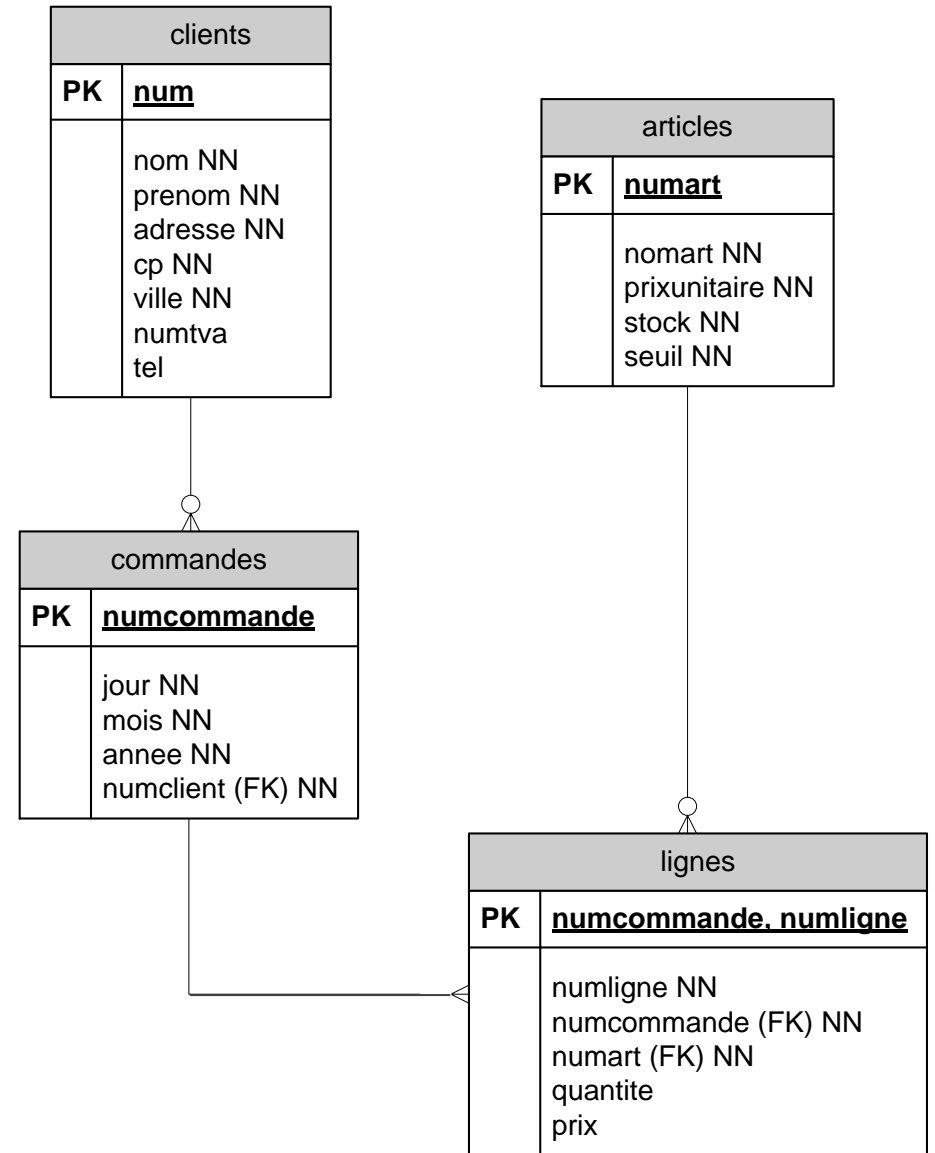
```
CREATE TABLE lignes
(numligne      int      NOT NULL,
numcommande    int      NOT NULL      REFERENCES commandes (numcommande)
numart         int      NOT NULL      REFERENCES articles (numart) ,
quantite       int,
prix           int,
PRIMARY KEY (numcommande, numligne))
```

lignes	
PK	<u>numcommande, numligne</u>
	numligne NN numcommande (FK) NN numart (FK) NN quantite prix



# Exercice 1

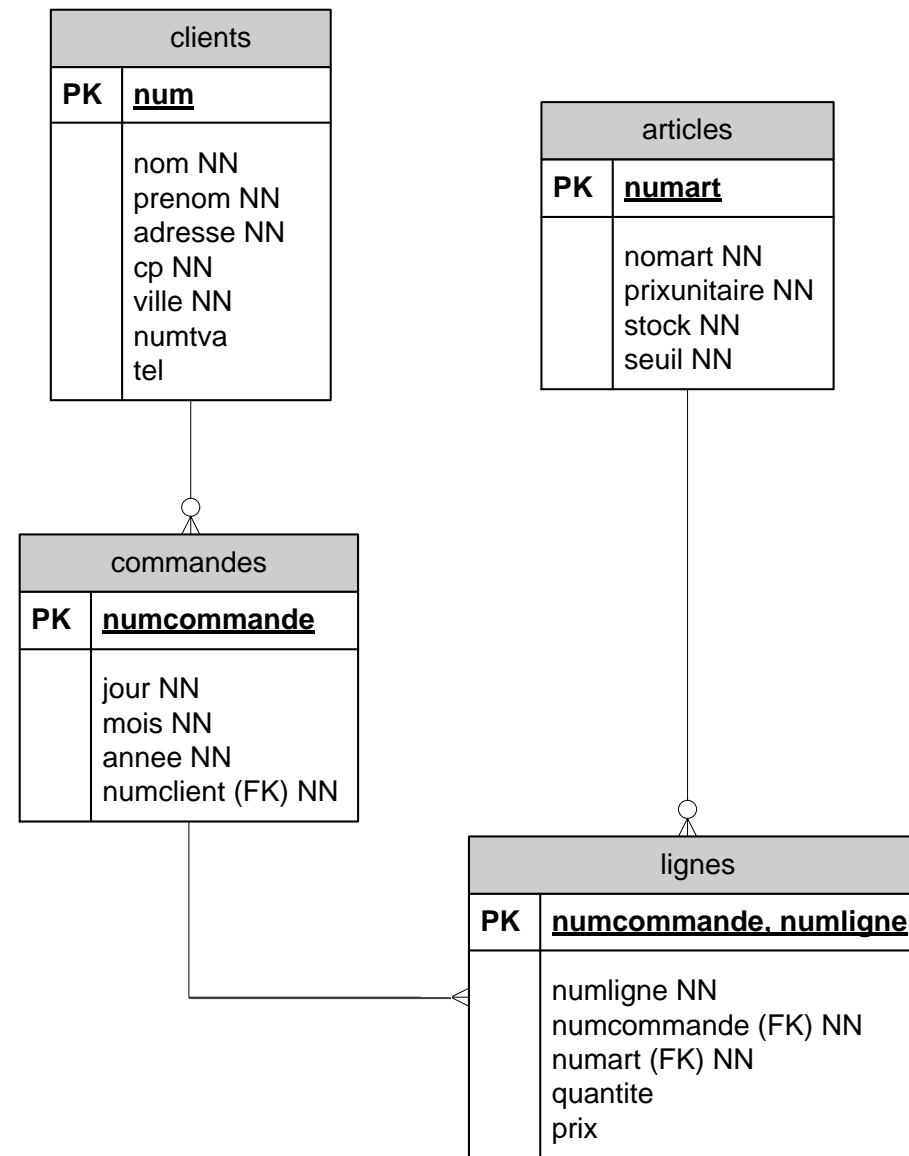
- Quelle est l'année de la commande la plus récente contenant l'article 'Stylo M3' ?





## Exercice 2

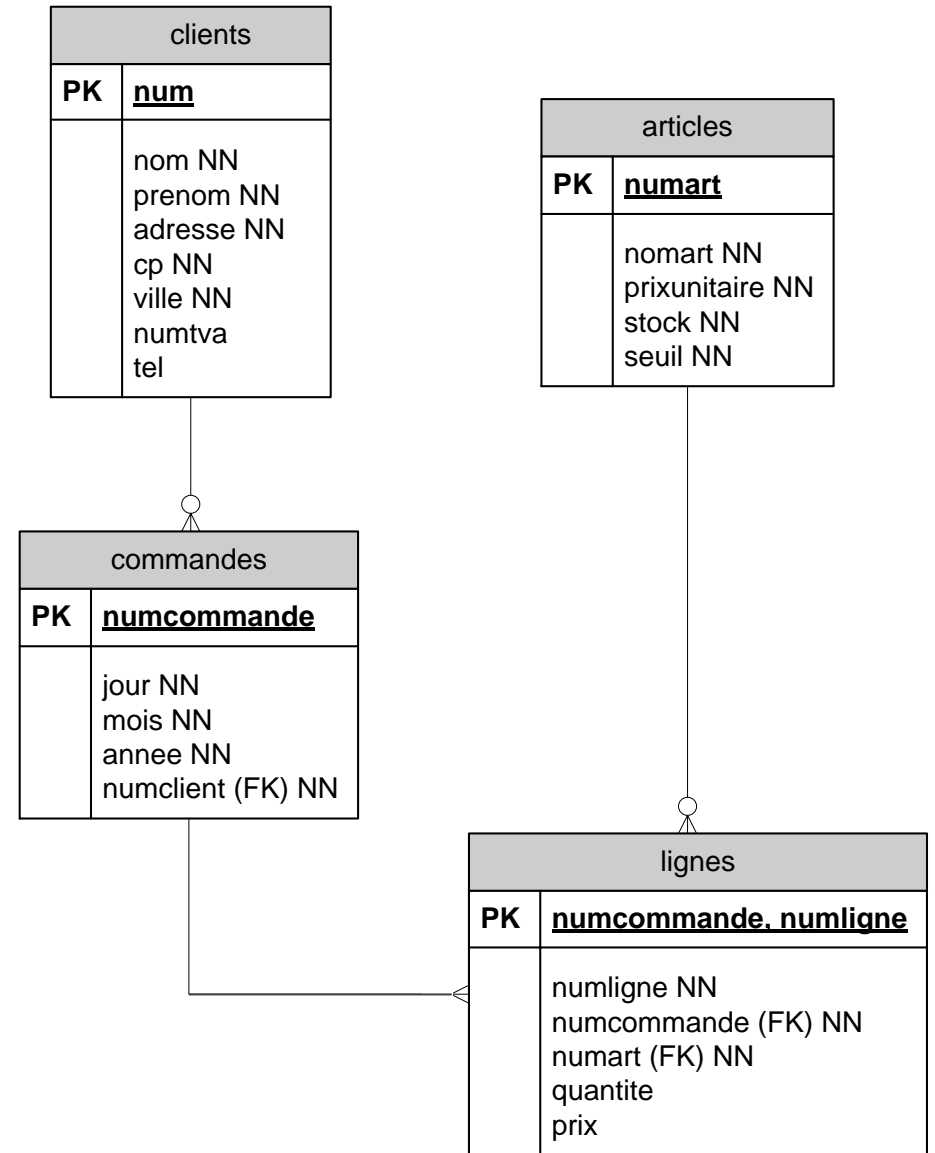
- Quel est le prix total de la commande numéro 100 ?





# Exercice 3

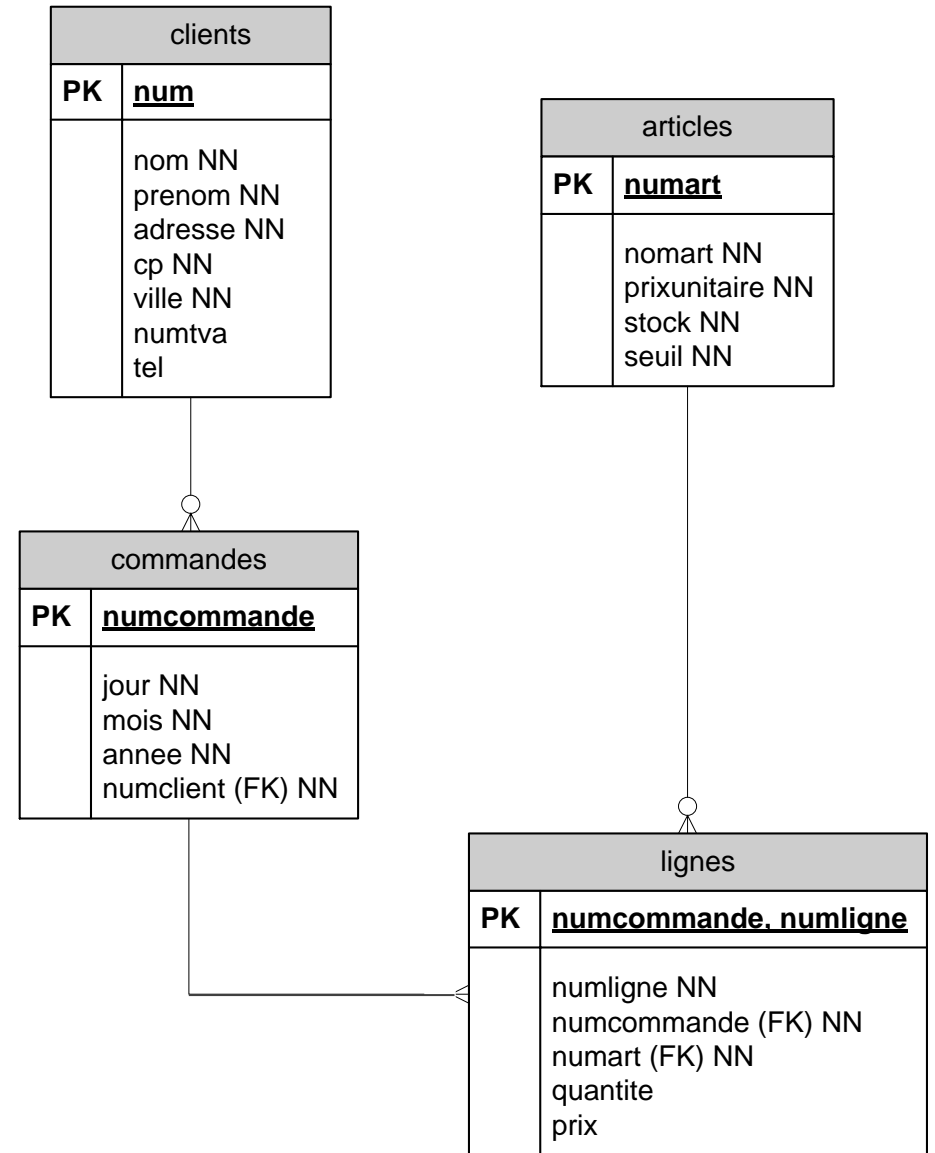
- Quelles sont les numéros des commandes qui portent sur l'article 'Stylo M3' ? Attention à la casse.





# Exercice 4

- Quels sont les numéros des commandes qui portent sur l'article 'Stylo M3' et au moins un autre article?





# Sub-SELECT ou sous-requêtes



# Sous-requêtes (p. 26)

- Une sous-requête consiste à exécuter une requête à l'intérieur d'une autre requête.
- Une sous-requête est aussi appelée une requête imbriquée, une requête en cascade ou un sub-SELECT
- Une sous-requête peut elle-même contenir une autre sous-requête



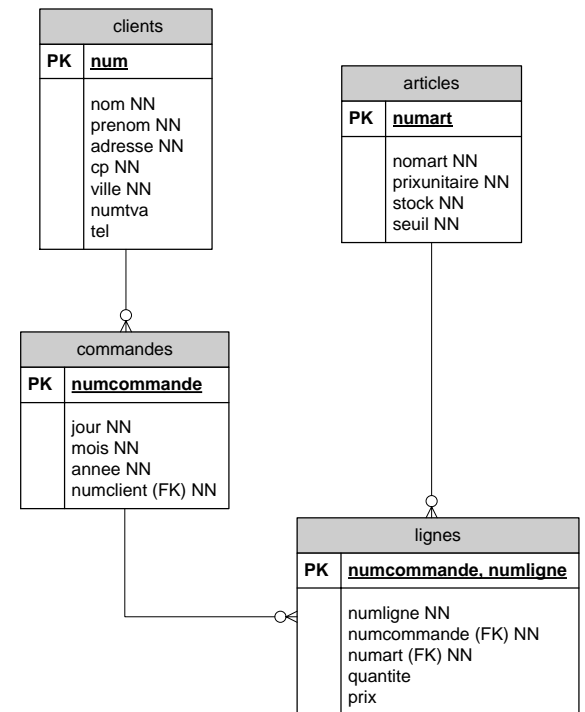
# Exemple de sous-requête (1)

- Comment trouver les commandes de tous les clients habitant Namur ?
- On peut d'abord identifier les clients habitant Namur :

```
SELECT num  
FROM clients  
WHERE ville = 'Namur'
```

- Ce qui nous donnerait comme résultat :

num
1
3
5





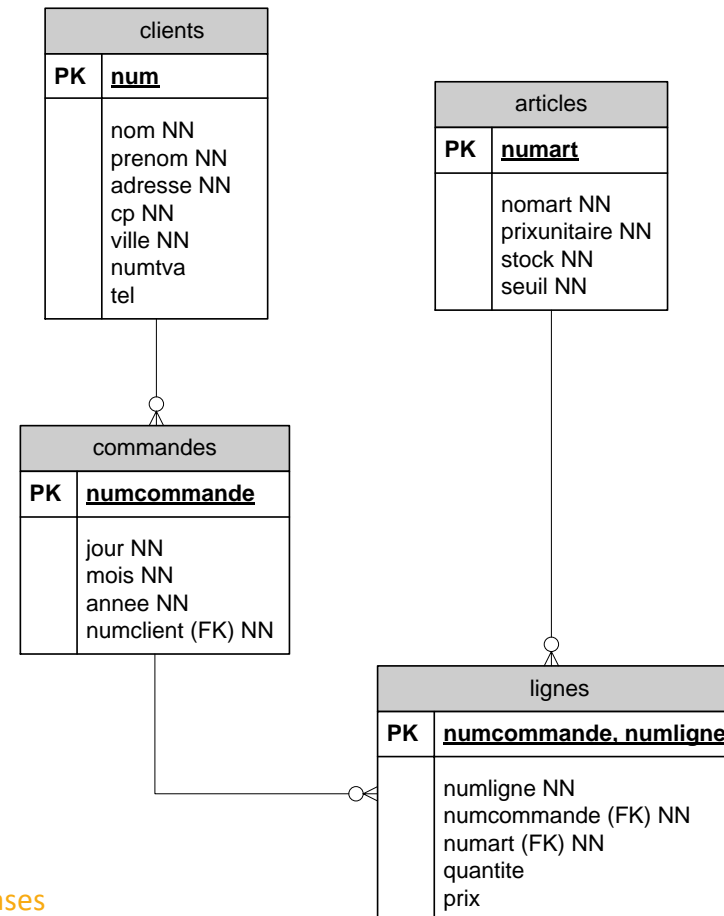


## Exemple de sous-requête (2)

- Il est alors aisé de retrouver les commandes de ces clients :

```
SELECT numcommande  
FROM commandes  
WHERE numclient IN (1, 3, 5)
```

- Mais tout ceci n'est pas très pratique ...





## Exemple de sous-requête (3)

- Remplaçons la liste de valeurs par l'expression qui a permis de les extraire de clients :

```
SELECT numcommande
FROM commandes
WHERE numclient IN (SELECT num
                    FROM clients
                    WHERE ville = 'Namur')
```



# Autre exemple de sous-requête

- Quelle est la liste de toutes les commandes du ou des client(s) dont le nom est 'Legrand' ?

```
SELECT co.numcommande  
FROM commandes co, clients cl  
WHERE co.numclient = cl.num  
AND cl.nom = 'Legrand'
```

```
SELECT co.numcommande  
FROM commandes co  
WHERE co.numclient IN (  
    SELECT cl.num  
    FROM clients cl  
    WHERE cl.nom = 'Legrand' )
```

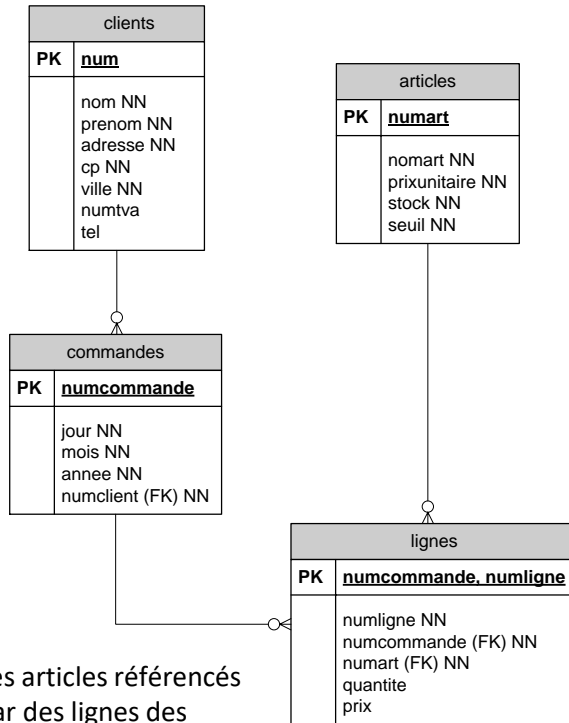


# Plusieurs sous-requêtes imbriquées

- Quels sont les articles qui ont été commandés par au moins 1 client de Namur ?

```
SELECT *  
FROM articles  
WHERE numart IN  
    (SELECT numart  
     FROM lignes  
     WHERE numcommande IN  
         (SELECT nucommande  
          FROM commandes  
          WHERE numclient IN  
              (SELECT num  
               FROM clients  
               WHERE ville = 'Namur'))))
```

Gestion de données : bases



Les articles référencés  
par des lignes des  
commandes des  
clients de Namur

Les lignes des  
commandes des  
clients de Namur

Les commandes des  
clients de Namur

Les clients de Namur



# Conditions d'association négatives

- Permet de retenir les lignes qui **ne** sont **pas** associées aux éléments d'un ensemble déterminé de lignes
- Quelles sont les commandes qui **ne** contiennent **pas** l'article 4 ?

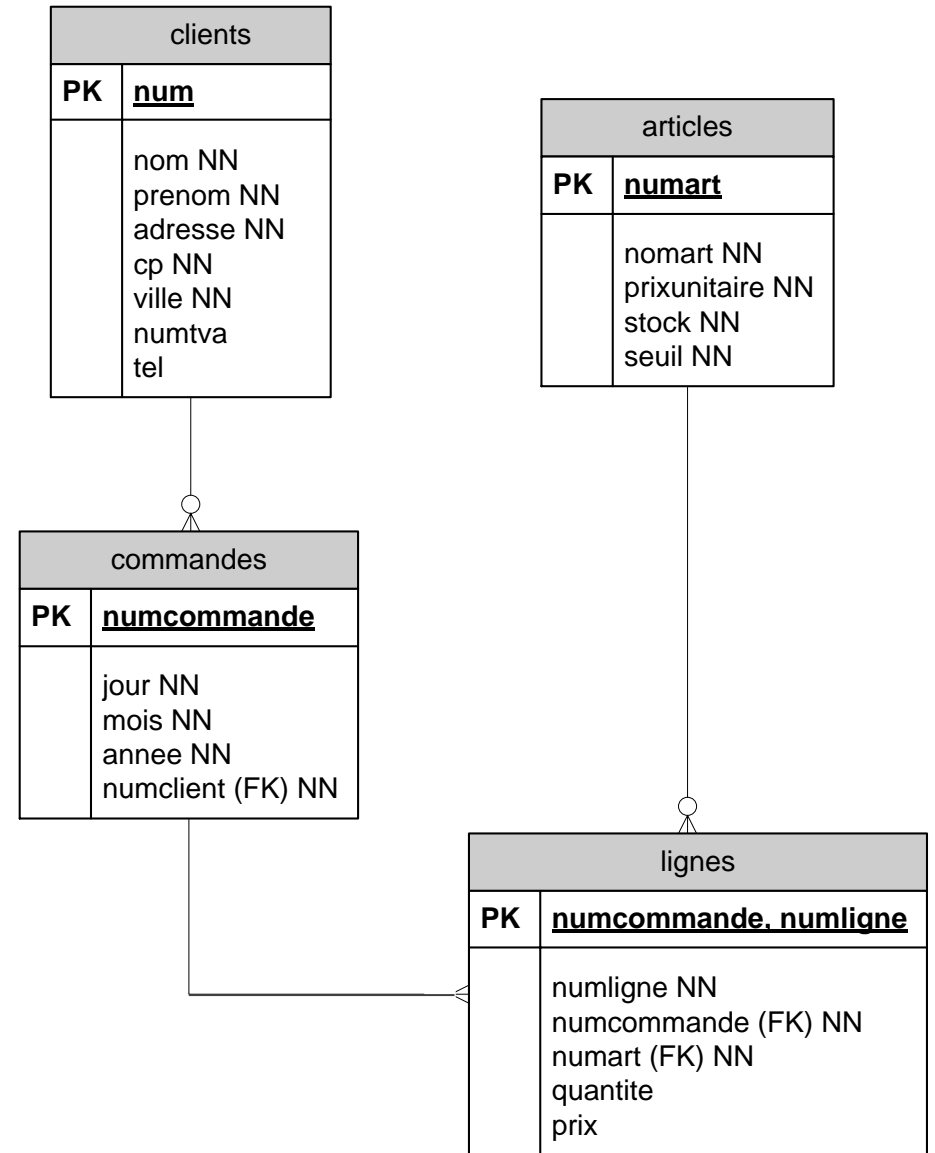
```
SELECT *  
FROM commandes  
WHERE numcommande NOT IN  
      (SELECT numcommande  
       FROM lignes  
       WHERE numarticle = 4)
```

- La sous-requête désigne ici les numéros de commande dont au moins 1 ligne contient l'article 4. Ces commandes sont à rejeter (NOT IN)



# Exercice 1

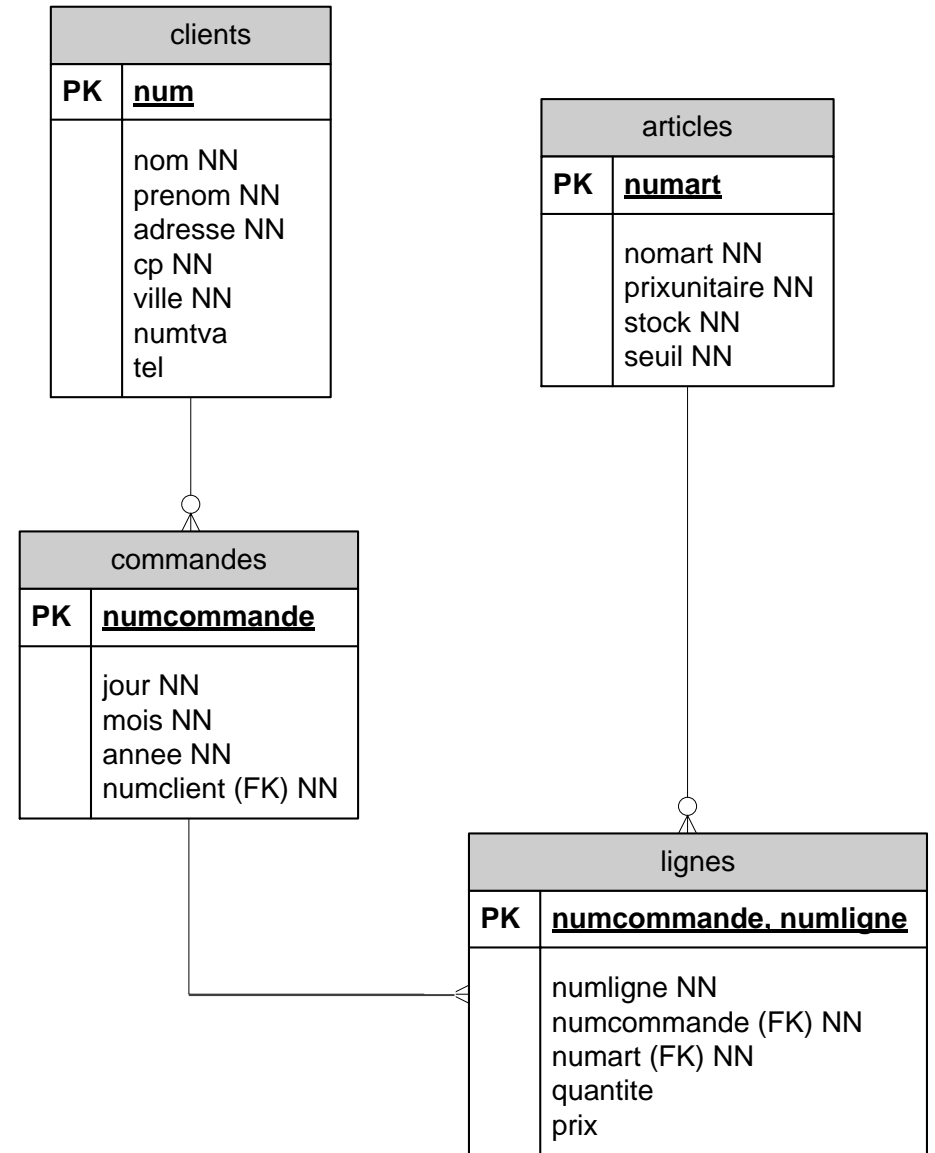
- Quels sont les articles les plus chers ?  
Indiquez pour chacun le numart et le nomart.





# Sub-SELECT : exercice

- Quelles sont les commandes qui ont été passées par les clients habitant au code postal 1200 ? Répondez à cette question à l'aide d'une sous-requête





# GROUP BY - HAVING





# Fonctions agrégées

- Valeurs globales
- Ne correspondent pas à un tuple en particulier
- Calcul sur **tous** les tuples **du résultat**



# Fonctions agrégées

➤ Compter le nombre de commandes :

```
SELECT count(numcommande)
```

ou

```
SELECT count(*)
```

```
FROM commandes
```

```
FROM commandes
```

➔ **Compter toutes les commandes de la table**



# GROUP BY (p. 26)

- Regroupement des tuples selon un critère
  - Compter le nombre de lignes de commande par commande :

```
SELECT numcommande, count(*) AS nombre_lignes  
FROM lignes  
GROUP BY numcommande
```

numcommande	nombre_lignes
105	10
106	7
108	2



# Fonctions agrégées et GROUP BY : à retenir

- Dans le SELECT, on ne peut faire appel à une fonction agrégée que si
  - ✓ toutes les autres colonnes résultent elles aussi d'une fonction agrégée (→ résultat = 1 seule ligne)
  - OU
  - ✓ toutes les autres colonnes sont présentes dans la clause GROUP BY



# GROUP BY - HAVING (p. 27)

- **SÉLECTION DE CERTAINS TUPLES DU RÉSULTAT**
- Ajouter un critère de sélection
- Donner le nombre de lignes de commande par commande, lorsque celui-ci est supérieur à 5 :

```
SELECT numcommande, count(*) AS nombre_lignes
```

```
FROM lignes
```

```
GROUP BY numcommande
```

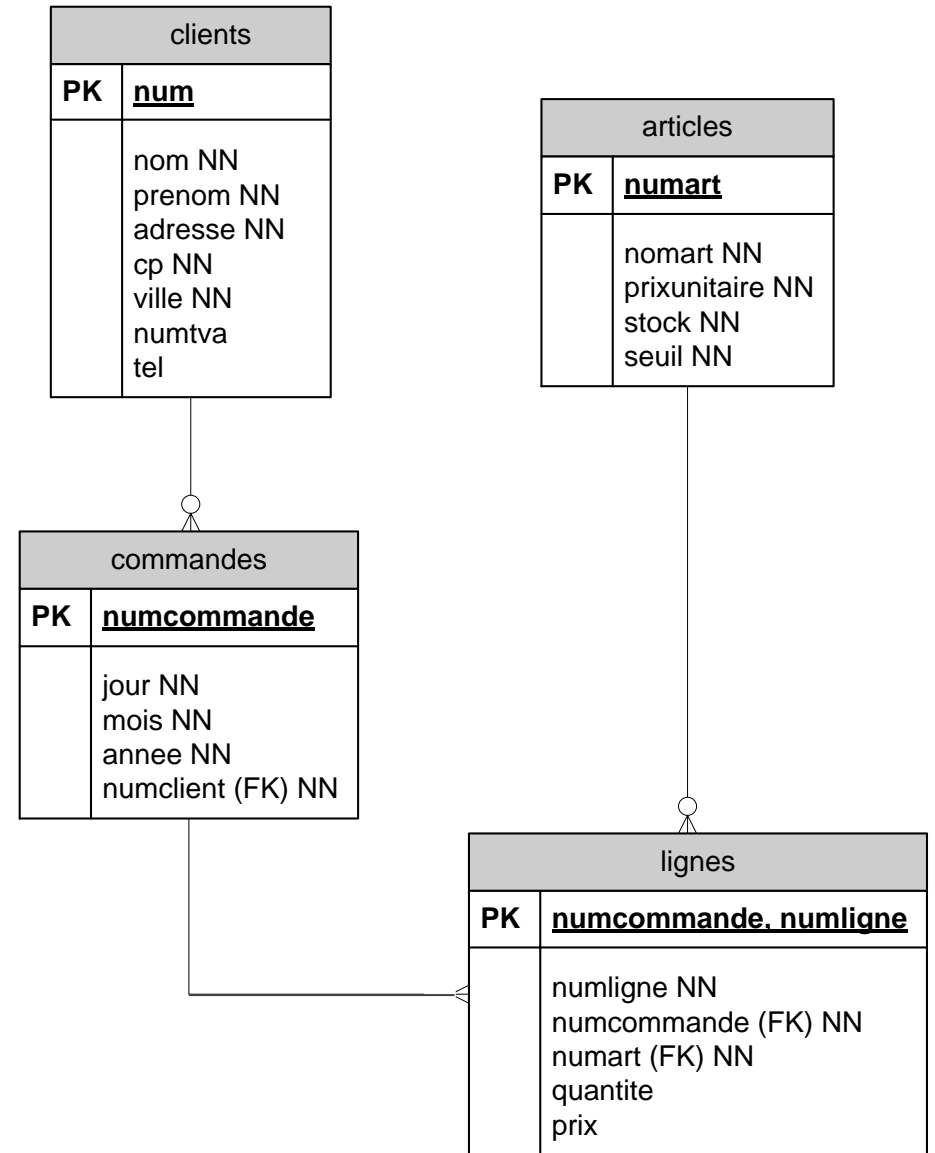
```
HAVING count(*) > 5
```

numcommande	nombre_lignes
105	10
106	7



# Exercice 1

Quel est le montant total payé par chaque client ?  
Donnez le num de chaque client.

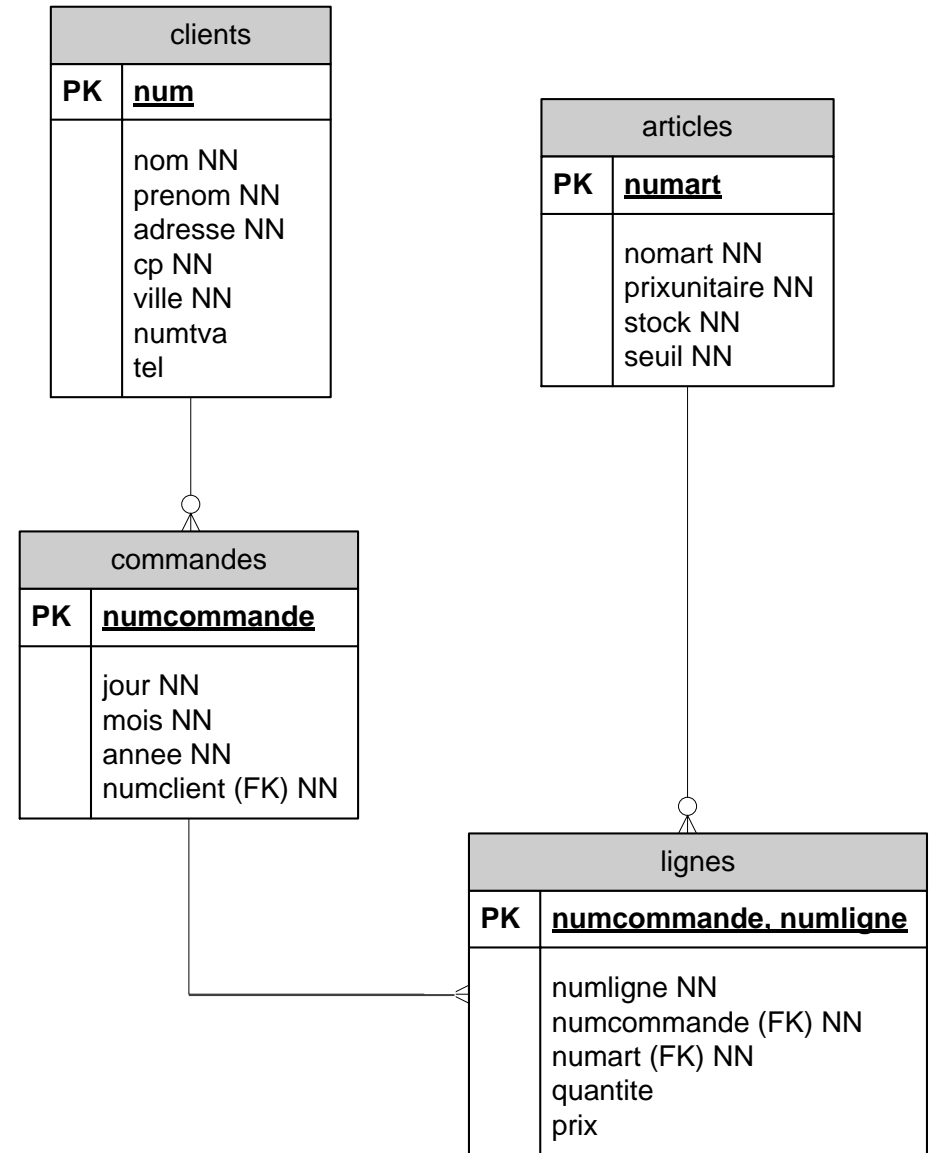




## Exercice 2

Quels sont les clients qui ont passé au moins 2 commandes ?

Donnez le numéro et le nom de ces clients.

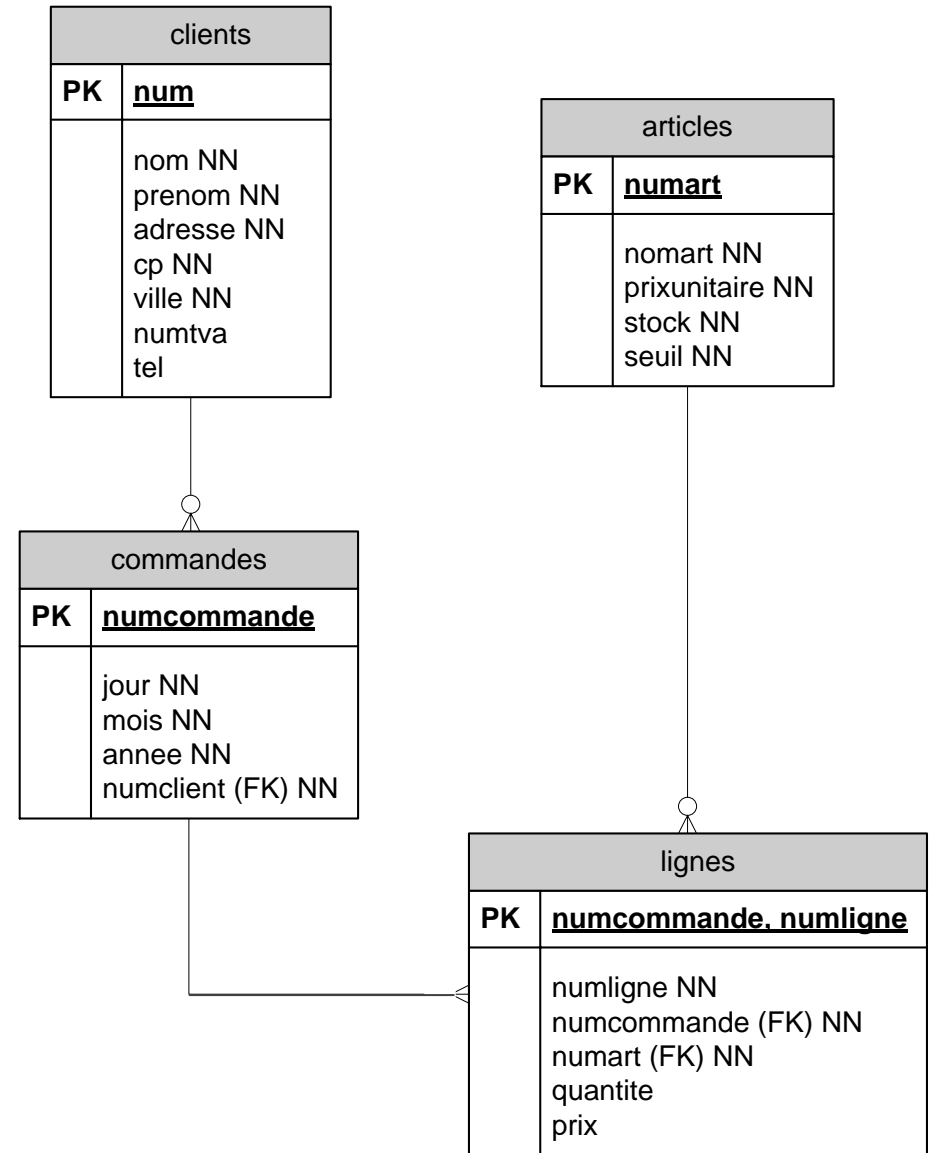




## Exercice 2

Quels sont les clients qui ont passé au moins 2 commandes ?

Donnez le numéro et le nom de ces clients.







# Ordre des instructions

- En SQL, l'ordre des instructions a de l'importance et doit être respecté

SELECT

FROM

WHERE

GROUP BY

HAVING

ORDER BY

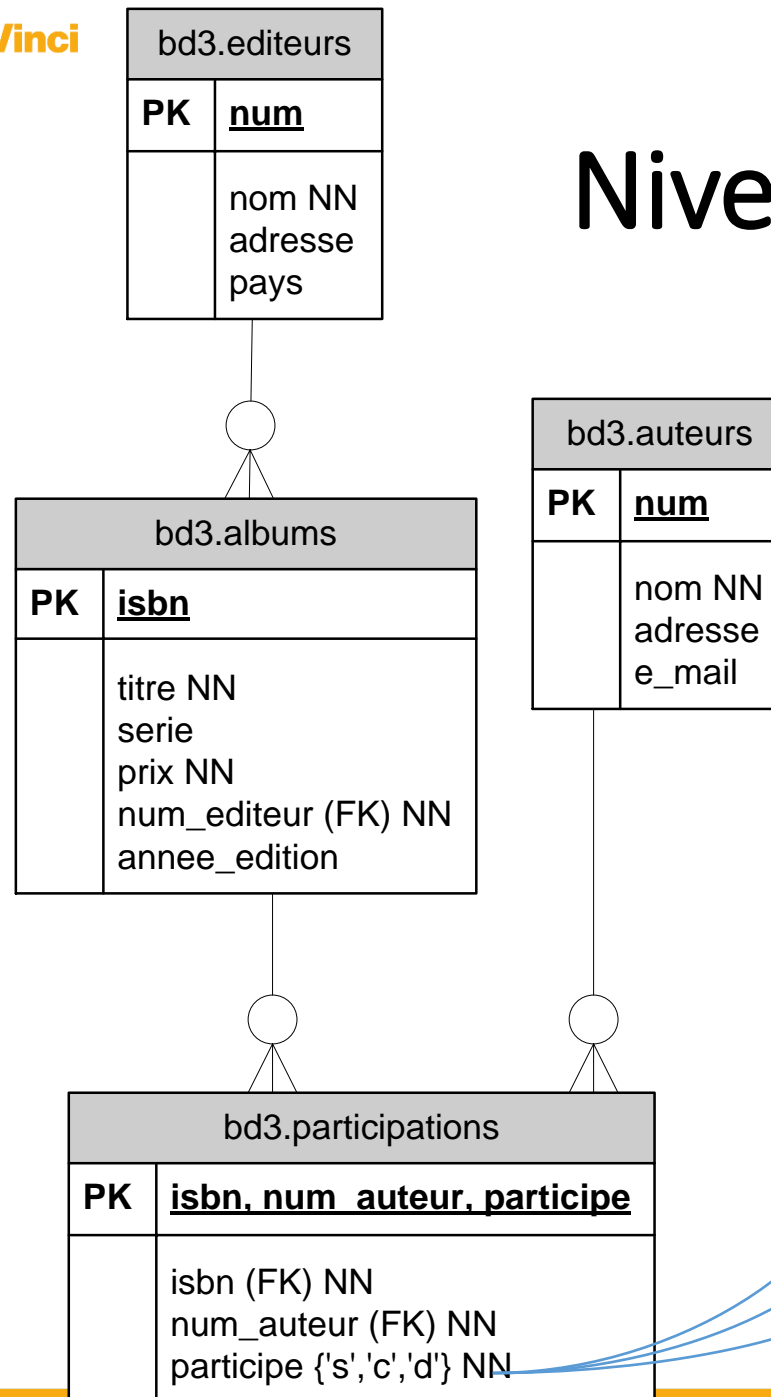
Facultatifs, dépend de ce qui est demandé dans la requête



# bd3.albums



## Niveau 3



DESSINATEUR  
SCÉNARISTE  
COLORISTE



# bd3.albums

bd3.albums	
PK	<u>isbn</u>
	titre serie num_editeur (FK) annee_edition prix

	isbn	titre	serie	num_editeur	annee_edition	prix
1	2-87097-055-9	Joyeux Noël, May		4	1998	7
2	2-205-03043-4	La fiancée de Lucky Luke	Lucky Luke	2	1985	5
3	2-88257-000-4	Nitroglycérine	Lucky Luke	2	1987	5
4	2-205-00585-5	Ma Dalton	Lucky Luke	2	1980	5
5	2-205-00920-6	La guérison des Dalton	Lucky Luke	2	1978	5
6	2-203-00105-4	L'Oreille Cassée	Tintin	7	1979	8
7	2-203-00103-8	Les Cigares du Pharaon	Tintin	7	1955	8
8	2-203-00107-0	Le Sceptre d'Ottokar	Tintin	7	1947	8
9	2-908-46271-0	FAITES GAFFE A LAGAFFE	Gaston	4	1996	6
10	2-8001-0955-6	LA SAGA DES GAFFES	Gaston	4	1982	6
11	2-8001-0091-5	LE CAS LAGAFFE	Gaston	4	1977	6
12	2-8001-0019-2	Spirou et les hommes-bulles	Spirou	4	1989	7
13	2-8001-0018-4	L'ombre du Z	Spirou	4	1976	7
14	2-8001-0017-6	Z comme Zorglub	Spirou	4	1977	7
15	2-8001-0015-X	Le voyageur du météore	Spirou	4	1977	7



# bd3.auteurs

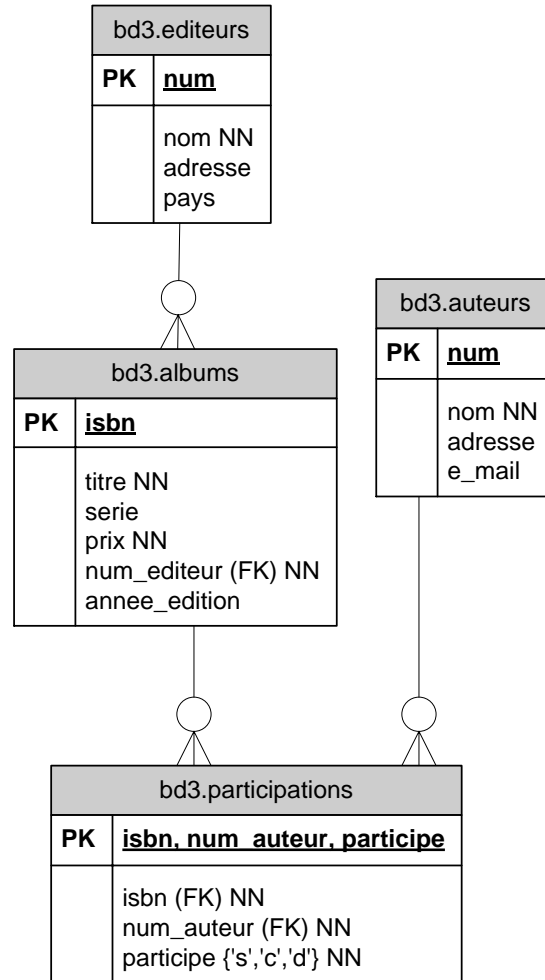
bd3.auteurs	
PK	<u>num</u>
	nom adresse e_mail

	num	nom	adresse	e_mail
1	1	Uderzo		uderzo@hotmail.com
2	2	Gosciny		gosciny@gmail.com
3	3	Hergé		Tintin@hotmail.be
4	4	Franquin		franquin@yahoo.fr
5	5	Jacobs		jacobs@yahoo.fr
6	6	de Moor		de_moor@yahoo.fr
7	7	Sente		sente@yahoo.fr
8	8	Juilliard		juilliard@gmail.com
9	9	DeMille		de_mille@gmail.fr
10	10	Benoît		benoit@yahoo.fr
11	12	Fanny		fanny@gmail.com
12	13	Jidehem		jidehem@gmail.com
13	14	Cosey		cosey@hotmail.com
14	15	Morris		morris@gmail.com
15	16	Guy Vidal		guy.vidal@hotmail.be
16	17	Dupuis		Dupuis@gmail.com
17	11	Van Hamme	Bruxelles	van_hamme@hotmail.com





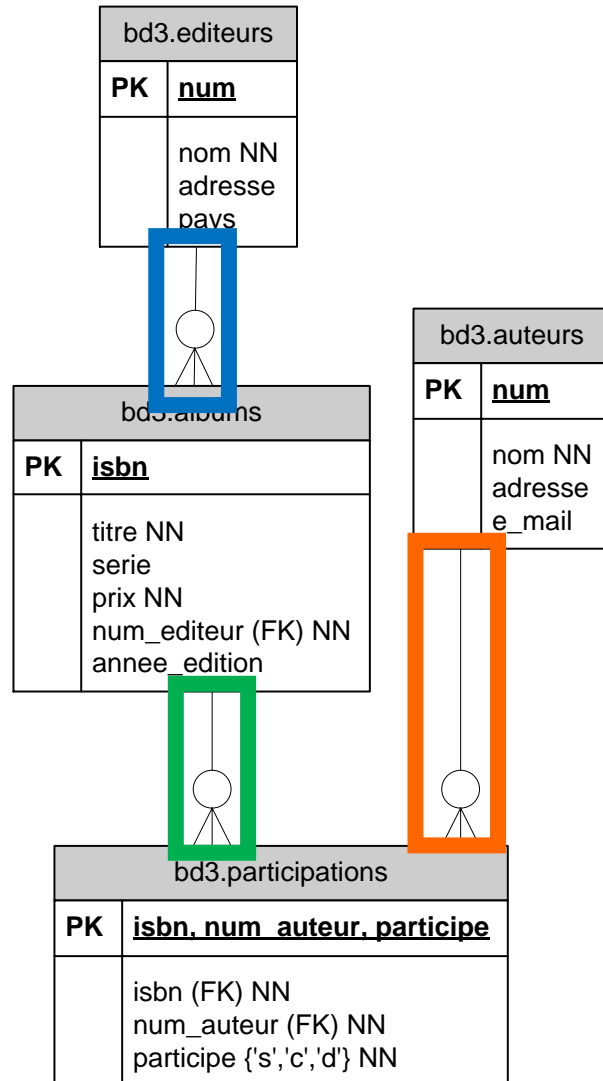
# bd3.participation



	isbn	num_auteur	participe
1	2-87097-055-9	14	s
2	2-87097-055-9	14	d
3	2-87097-055-9	14	c
4	2-205-03043-4	16	s
5	2-205-03043-4	15	d
6	2-88257-000-4	16	s
7	2-88257-000-4	15	d
8	2-205-00585-5	2	s
9	2-205-00585-5	15	d
10	2-205-00920-6	2	s
11	2-205-00920-6	15	d
12	2-203-00105-4	3	s
13	2-203-00105-4	3	d
14	2-203-00105-4	3	c
15	2-203-00103-8	3	s
16	2-203-00103-8	3	d
17	2-203-00103-8	3	c
18	2-203-00107-0	3	s
19	2-203-00107-0	3	d
20	2-203-00107-0	3	c
21	2-908-46271-0	4	s
22	2-908-46271-0	4	d
23	2-908-46271-0	12	c
24	2-8001-0955-6	4	s
25	2-8001-0955-6	4	d
26	2-8001-0091-5	4	d
27	2-8001-0019-2	4	s
28	2-8001-0019-2	4	d
29	2-8001-0018-4	13	s
30	2-8001-0018-4	4	d



# Jointures ?



```
SELECT *  
FROM bd3.editeurs ed, bd3.albums  
al, bd3.auteurs au, bd3.participations  
pa  
WHERE ed.num = al.num_editeur  
AND al.isbn = pa.isbn  
AND pa.num_auteur = au.num
```



# Conclusion

- Toute la matière du premier syllabus a été vue !
- Cette semaine (semaine 4) : exercices niveau 3 + **interro n°2**
- Semaine 5 : remédiation pendant la séance de théorie. Toutes les réponses à vos questions (sur le cours de BD1 bien sûr)
- Semaine 5 : exercices niveau 4
- Semaine 6 : **conception** de bases de données + **interro SQL n°3**