SQL

2ÈME INFO

CHRISTOPHE DAMAS

DÉTAILS PRATIQUES

- **1 séance d’exercice (2h) présentiel le mercredi**
  - Stéphanie Ferneeuw, Nicolas Burny
- **1 séance de théorie (1h) distanciel asynchrone**

- **Programme**
  - S1 – S5: Perfectionnement des requêtes SQL
  - S6 – S12: Exercice complet

- **Support**
  - Syllabus + Slides (Moodle)
  - Pendant l’examen, accès au contenu de moodle

EVALUATION

- **Janvier**
  - Examen intégré (90%)
  - Interro Queries semaine 6 (5%)
  - Evaluation continue conception (5%)

- **Septembre**
  - Examen intégré (100%)

POSTGRESQL

Un des leaders de l’open-source

- Concurrent de MySQL

http://www.postgresql.org/

- Cette année, nous utiliserons Data Grip comme client

POSTGRESQL À LA MAISON

- Vidéo sur moodle

- En gros:

1. Téléchargement serveur postgres (sur https://www.postgresql.org/)
2. Téléchargement Data Grip
3. Création de la base de données pubs2

ET MAINTENANT…

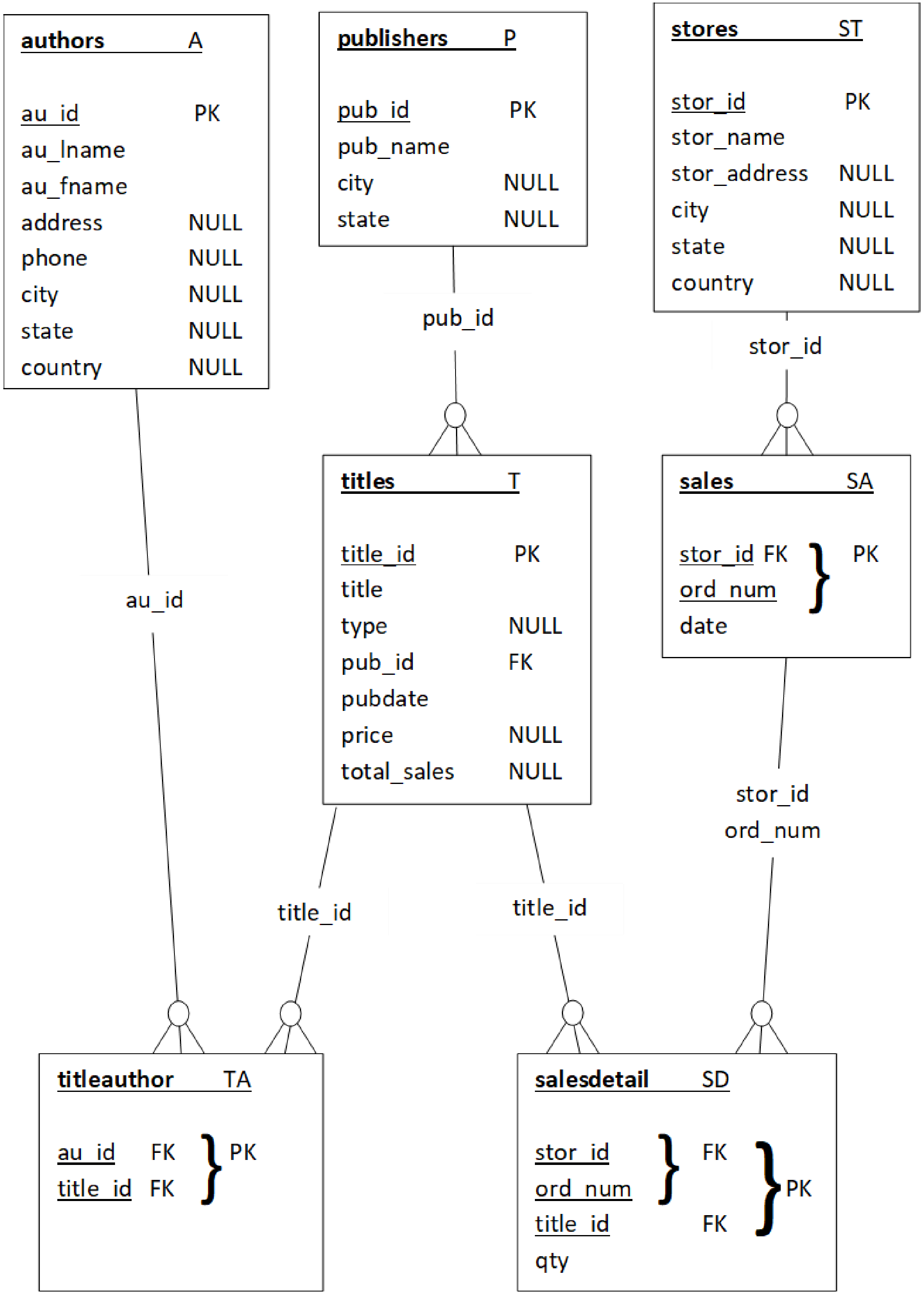
Le cours !

QU’EST-CE QUE SQL ?

- **Structured Query Language**
- **Emerge d’un besoin**
  - Fin des années 60
  - Stocker les données, les organiser, les interroger, les faire évoluer d’une manière consistante
- **Edgar F. Codd en 1970 (IBM research) :**
  - A Relational Model of Data for Large Shared Data Banks

MODÈLE RELATIONNEL

- **Le SQL se base sur le modèle relationnel**
- **Une base de donnée relationnelle est constituée d’un ensemble de tables**
    - Les colonnes sont les attributs
    - Les lignes sont appelées tuples
        - Chaque tuple est unique !
        - Il n’y a pas de notion d’ordre entre les tuples



authors A

au_id PK
au_lname
au_fname
address NULL
phone NULL
city NULL
state NULL
country NULL

publishers P

pub_id PK
pub_name
city NULL
state NULL

stores ST

stor_id PK
stor_name
stor_address NULL
city NULL
state NULL
country NULL

pub_id

stor_id

titles T

title_id PK
title
type NULL
pub_id FK
pubdate
price NULL
total_sales NULL

sales SA

stor_id FK } PK
ord_num
date

au_id

stor_id
ord_num

title_id

title_id

titleauthor TA

au_id FK } PK
title_id FK

salesdetail SD

stor_id } FK } PK
ord_num
title_id FK
qty

RELATIONS

Clef primaire

- Identifiant unique pour un tuple
    - Peut être une combinaison de plusieurs colonnes

Clef étrangère

- Référence à une clef primaire d’une autre table
    - Prend la valeur de la clef primaire à laquelle on fait référence

OPÉRATION : UNION

R

Nom	Destination	Code-dépl
Dufour	Paris	321
Dufour	Milan	325
Durand	Paris	360
Dutoit	Paris	322
Dutoit	Paris	312
Dutoit	Oslo	319

S

Nom	Destination	Code-dépl
Dufour	Paris	321
Dufour	Milan	325
Durand	Paris	588
Janssens	Prague	322

$R \cup S$

Nom	Destination	Code-dépl
Dufour	Paris	321
Dufour	Milan	325
Durand	Paris	360
Dutoit	Paris	322
Dutoit	Paris	312
Dutoit	Oslo	319
Durand	Paris	588
Janssens	Prague	322

OPÉRATION : DIFFÉRENCE

R

Nom	Destination	Code-dépl
Dufour	Paris	321
Dufour	Milan	325
Durand	Paris	360
Dutoit	Paris	322
Dutoit	Paris	312
Dutoit	Oslo	319

S

Nom	Destination	Code-dépl
Dufour	Paris	321
Dufour	Milan	325
Durand	Paris	588
Janssens	Prague	322

R-S

Nom	Destination	Code-dépl
Durand	Paris	360
Dutoit	Paris	322
Dutoit	Paris	312
Dutoit	Oslo	319

OPÉRATION : PRODUIT CARTÉSIEN

R

Nom	Destination	Code-dépl
Dufour	Paris	321
Dufour	Milan	325
Durand	Paris	360
Dutoit	Paris	322
Dutoit	Paris	312
Dutoit	Oslo	319

T

Nom	Rembours
Dufour	2
Dutoit	4
Janssens	0
Albrecht	2
Fanuel	3

R x T

R.Nom	Destination	Code-dépl	T.Nom	Rembours
Dufour	Paris	321	Dufour	2
Dufour	Paris	321	Dutoit	4
Dufour	Paris	321	Janssens	0
Dufour	Paris	321	Albrecht	2
Dufour	Paris	321	Fanuel	3
Dufour	Milan	325	Dufour	2
…	…	…	…	…
Dutoit	Oslo	319	Fanuel	3

OPÉRATION : PROJECTION

R

Nom	Destination	Code-dépl
Dufour	Paris	321
Dufour	Milan	325
Durand	Paris	360
Dutoit	Paris	322
Dutoit	Paris	312
Dutoit	Oslo	319

$\pi_{\text{Nom,Destination}}(R)$

Nom	Destination
Dufour	Paris
Dufour	Milan
Durand	Paris
Dutoit	Paris
Dutoit	Oslo

OPÉRATION : SÉLECTION

T

Nom	**Rembours**
Dufour	2
Dutoit	4
Janssens	0
Albrecht	2
Fanuel	3

$\sigma_{\text{Rembours}<3}(T)$

Nom	**Rembours**
Dufour	2
Janssens	0
Albrecht	2

QUELLES SONT LES DESTINATIONS DES PERSONNES AVEC AU MOINS 2 REMBOURS ?

T

Nom	Rembours
Dufour	2
Dutoit	4
Janssens	0
Albrecht	2
Fanuel	3

$\sigma_{Rembours>=2}(T)$

Nom	Destination
Dufour	2
Dutoit	4
Albrecht	2
Fanuel	3

QUELLES SONT LES DESTINATIONS DES PERSONNES AVEC AU MOINS 2 REMBOURS ?

$\sigma_{Rembours>=2}(T)$

Nom	Rembours
Dufour	2
Dutoit	4
Albrecht	2
Fanuel	3

S

Nom	Destination	Code-dépl
Dufour	Paris	321
Dufour	Milan	325
Durand	Paris	588
Janssens	Prague	322

$\sigma_{Rembours>=2}(T) \text{ x } S$

S.Nom	Destination	Code-dépl	σ.Nom	σ.Rembours
Dufour	Paris	321	Dufour	2
Dufour	Paris	321	Dutoit	4
Dufour	Paris	321	Albrecht	2
Dufour	Paris	321	Fanuel	3
…	…	…	…	…
Janssens	Prague	322	Dutoit	4
Janssens	Prague	322	Albrecht	2
Janssens	Prague	322	Fanuel	3

QUELLES SONT LES DESTINATIONS DES PERSONNES AVEC AU MOINS 2 REMBOURS ?

$\sigma_{\text{Rembours>=2}}(\text{T}) \text{ x S}$

S.Nom	**Destination**	**Code-dépl**	**σ.Nom**	**σ.Rembours**
Dufour	Paris	321	Dufour	2
Dufour	Paris	321	Dutoit	4
Dufour	Paris	321	Albrecht	2
Dufour	Paris	321	Fanuel	3
…	…	…	…	…
Janssens	Prague	322	Dutoit	4
Janssens	Prague	322	Albrecht	2
Janssens	Prague	322	Fanuel	3

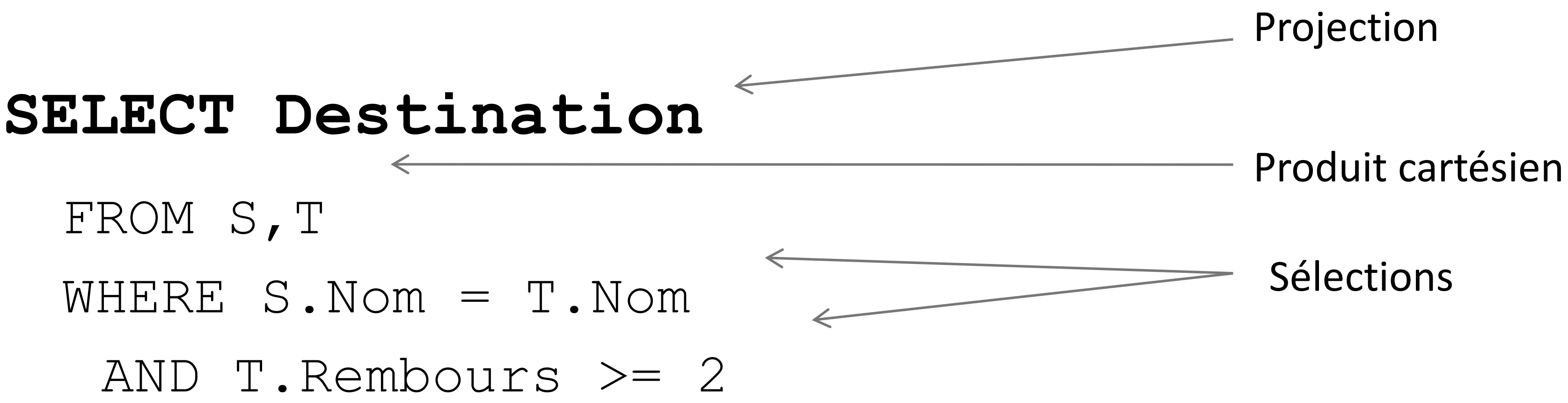
$\sigma_{\text{S.Nom=}\sigma\text{.Nom}}(\sigma_{\text{Rembours>=2}}(\text{T}) \text{ x S})$

S.Nom	**Destination**	**Code-dépl**	**σ.Nom**	**σ.Rembours**
Dufour	Paris	321	Dufour	2
Dufour	Milan	325	Dufour	2

$\pi_{\text{Destination}}(\sigma_{\text{S.Nom=}\sigma\text{.Nom}}(\sigma_{\text{Rembours>=2}}(\text{T}) \text{ x S}))$

Destination
Paris
Milan

EN SQL



```
SELECT Destination
  FROM S,T
  WHERE S.Nom = T.Nom
   AND T.Rembours >= 2
```

Projection

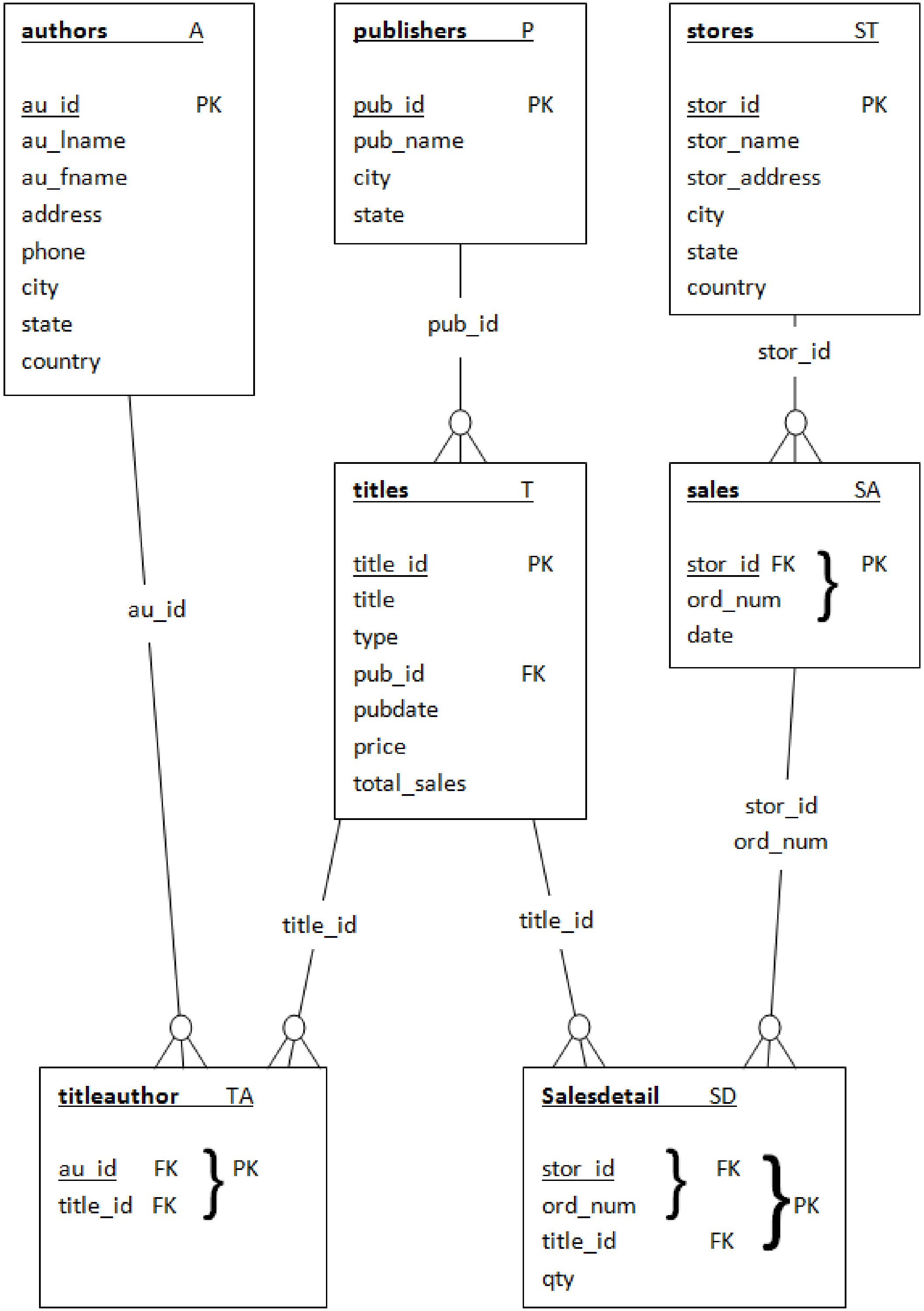
Produit cartésien

Sélections

INTERROGATION : SELECT

```
SELECT [ ALL | DISTINCT]
  * | nom_colonne
  [ [ AS ] nom_d_affichage ] [, ...]
  [ FROM nom_table ]
  [ WHERE condition ]
  [ ORDER BY nom_colonne [ ASC | DESC ] [, ...] ]
```

Lister la table des auteurs



authors A

au_id PK
au_lname
au_fname
address
phone
city
state
country

publishers P

pub_id PK
pub_name
city
state

stores ST

stor_id PK
stor_name
stor_address
city
state
country

titles T

title_id PK
title
type
pub_id FK
pubdate
price
total_sales

sales SA

stor_id FK } PK
ord_num
date

titleauthor TA

au_id FK } PK
title_id FK

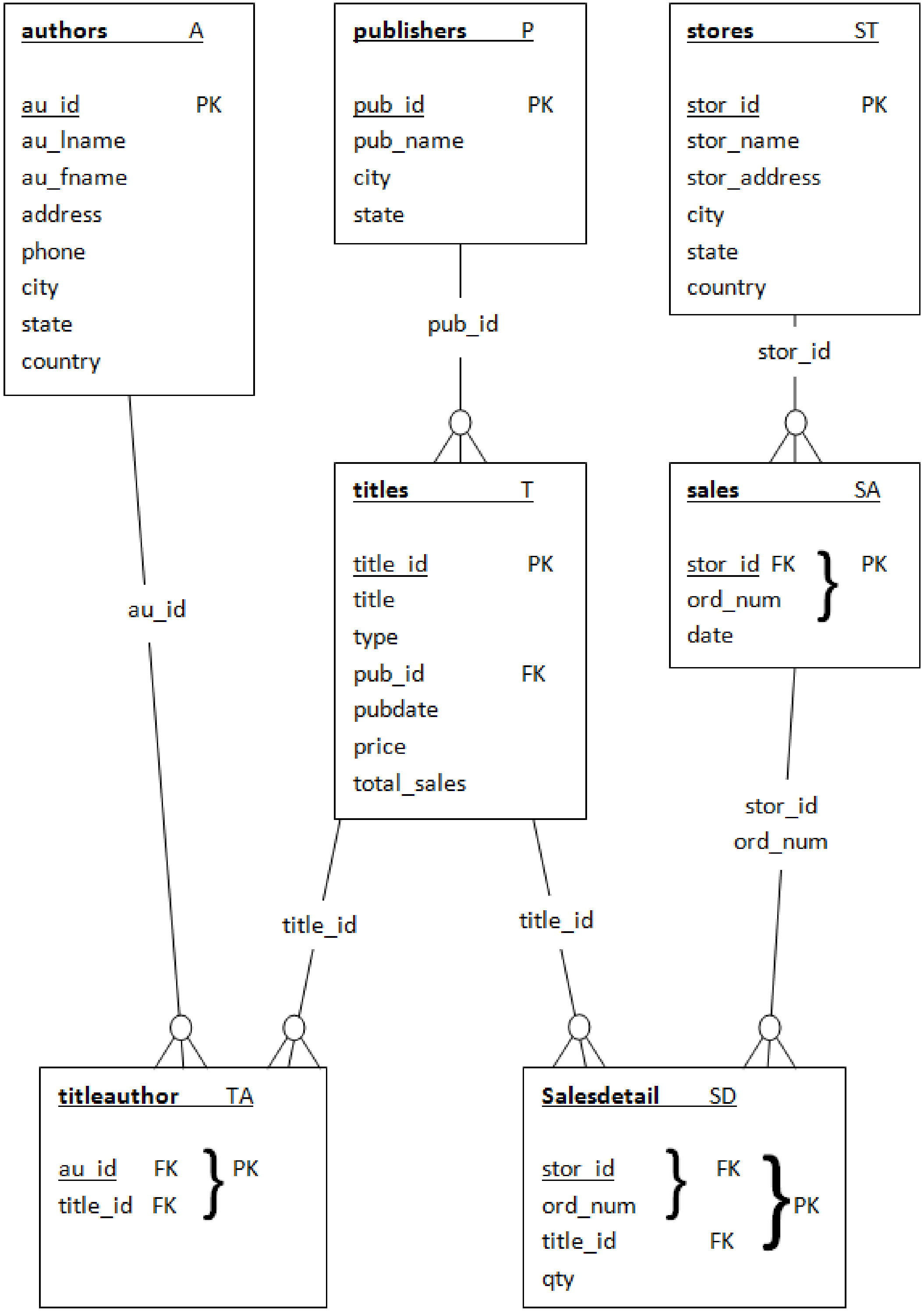
Salesdetail SD

stor_id } FK } PK
ord_num
title_id FK
qty

```
SELECT *
FROM authors;
```

```
SELECT authors.*
FROM authors;
```

```
SELECT au_id, au_lname,
  au_fname, address, phone,
  city, state, country
FROM authors;
```



authors A

au_id PK
au_lname
au_fname
address
phone
city
state
country

publishers P

pub_id PK
pub_name
city
state

stores ST

stor_id PK
stor_name
stor_address
city
state
country

titles T

title_id PK
title
type
pub_id FK
pubdate
price
total_sales

sales SA

stor_id FK } PK
ord_num }
date

titleauthor TA

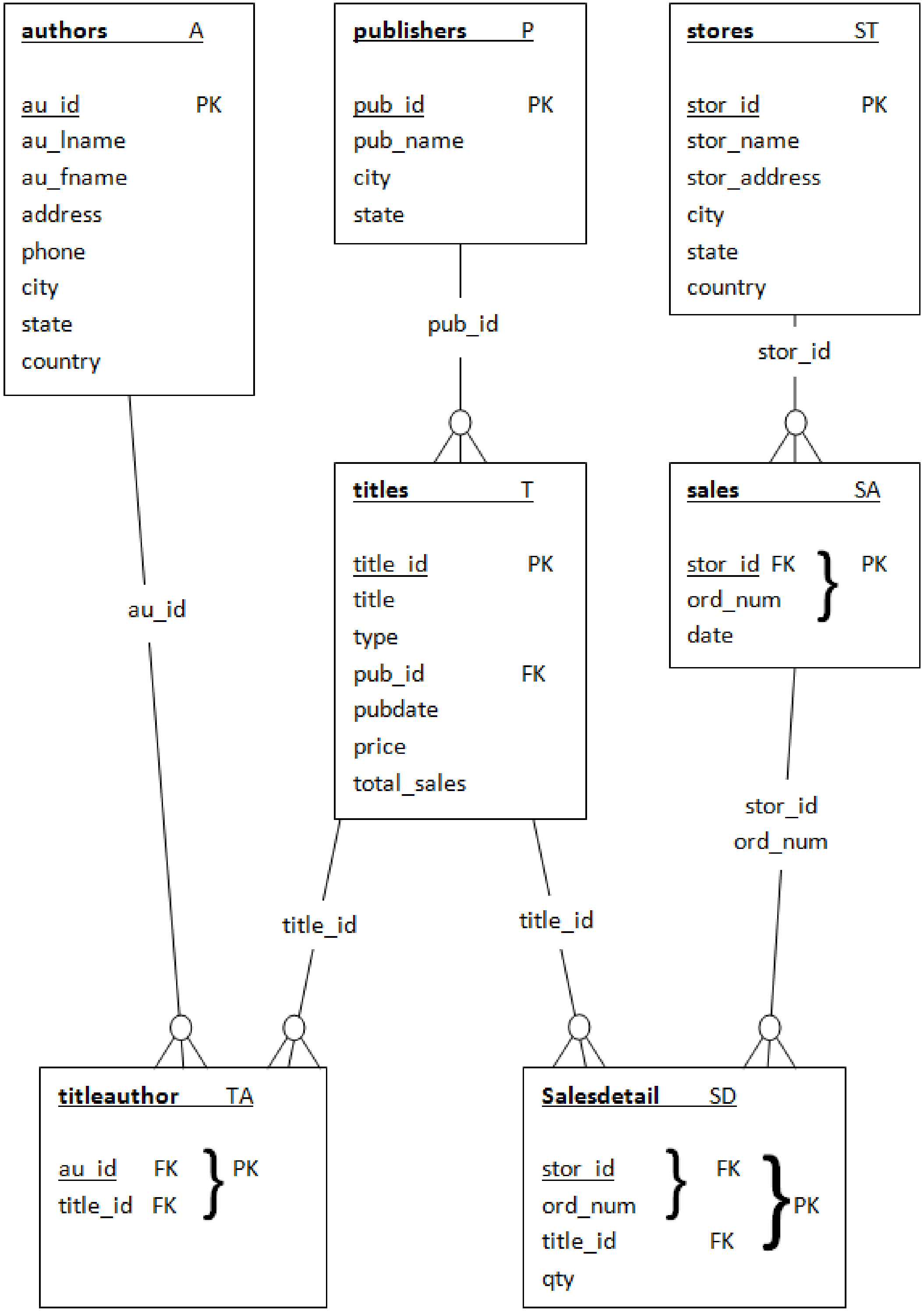
au_id FK } PK
title_id FK }

Salesdetail SD

stor_id } FK } PK
ord_num } }
title_id FK }
qty

Lister la table des auteurs

```
SELECT a.au_id, a.au_lname,
  a.au_fname, a.address,
  a.phone, a.city, a.state,
  a.country
FROM authors a;
```



authors A
au_id PK
au_lname
au_fname
address
phone
city
state
country

publishers P
pub_id PK
pub_name
city
state

stores ST
stor_id PK
stor_name
stor_address
city
state
country

pub_id

stor_id

titles T
title_id PK
title
type
pub_id FK
pubdate
price
total_sales

sales SA
stor_id FK } PK
ord_num
date

au_id

stor_id
ord_num

title_id

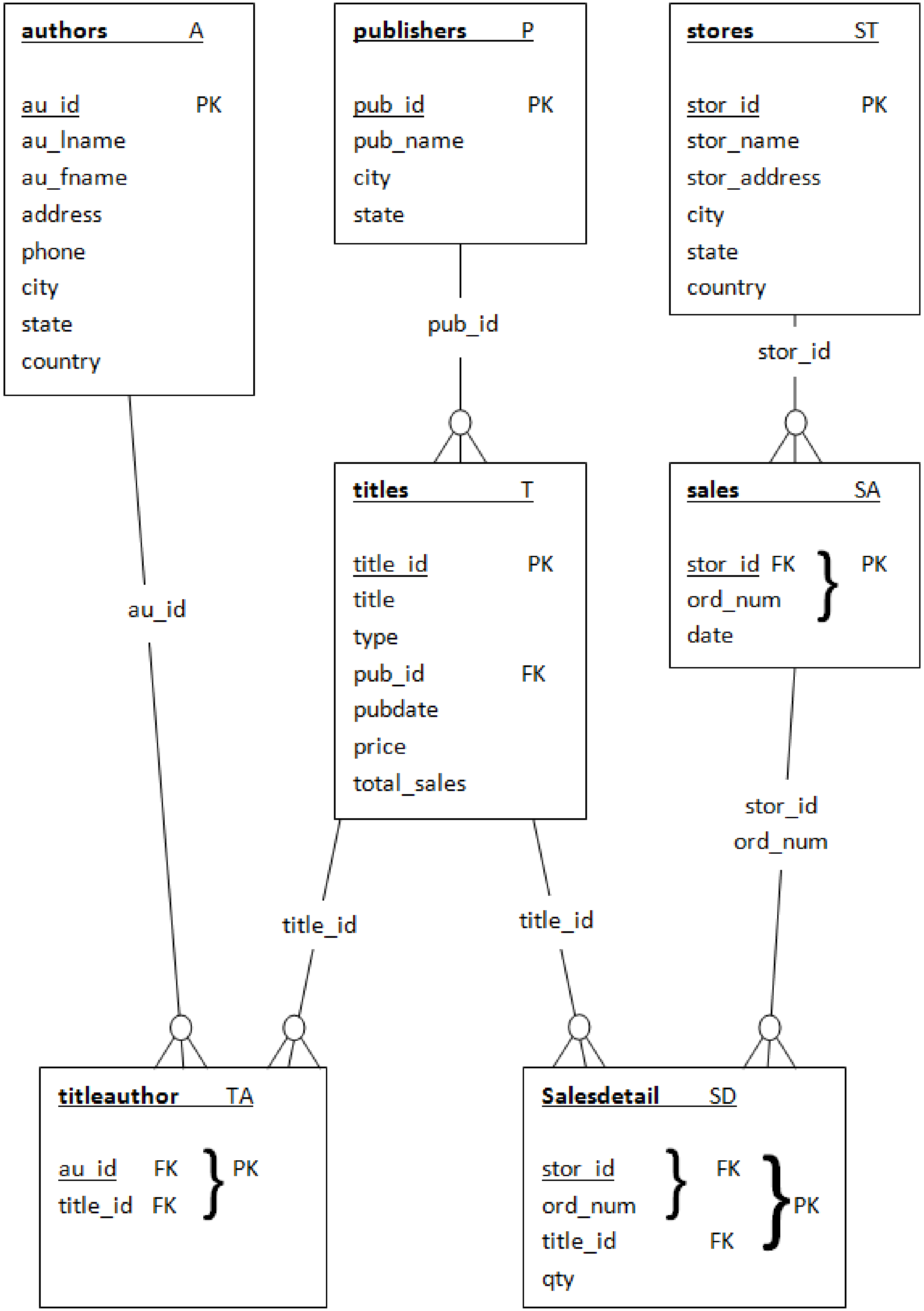
title_id

titleauthor TA
au_id FK } PK
title_id FK

Salesdetail SD
stor_id } FK } PK
ord_num
title_id FK
qty

Lister les noms et prénoms des auteurs californiens

```
SELECT a.au_lname,
  a.au_fname
FROM authors a
WHERE a.state = 'CA';
```



authors A

au_id PK
au_lname
au_fname
address
phone
city
state
country

publishers P

pub_id PK
pub_name
city
state

stores ST

stor_id PK
stor_name
stor_address
city
state
country

pub_id

stor_id

au_id

titles T

title_id PK
title
type
pub_id FK
pubdate
price
total_sales

sales SA

stor_id FK } PK
ord_num
date

stor_id
ord_num

title_id

title_id

titleauthor TA

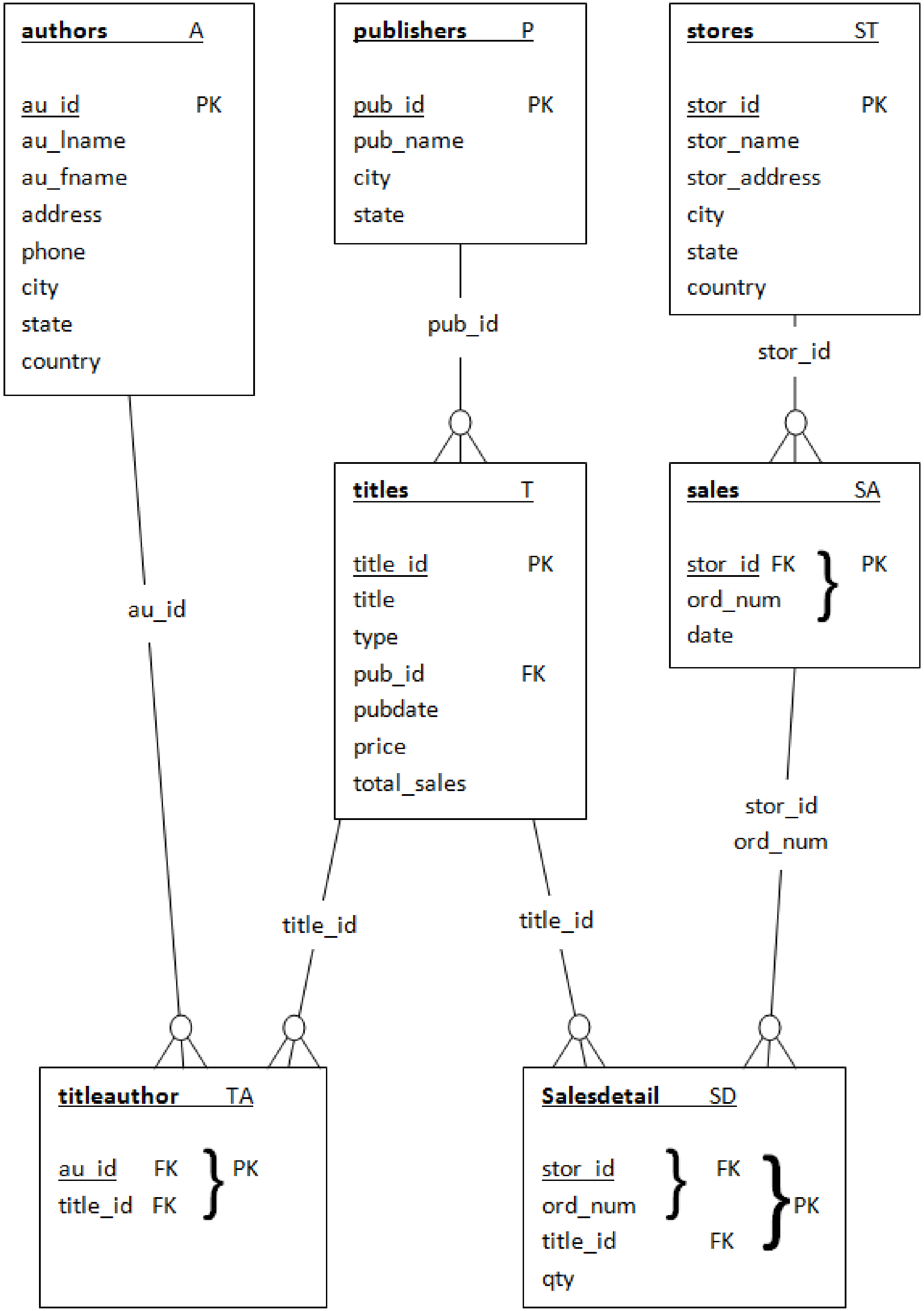
au_id FK } PK
title_id FK

Salesdetail SD

stor_id } FK } PK
ord_num
title_id FK
qty

Lister les noms et prénoms des auteurs dont la 2ème lettre du nom est e

```
SELECT a.au_lname, a.au_fname
FROM authors a
WHERE a.au_lname LIKE  '_e%';
```



authors A

au_id PK
au_lname
au_fname
address
phone
city
state
country

publishers P

pub_id PK
pub_name
city
state

stores ST

stor_id PK
stor_name
stor_address
city
state
country

pub_id

stor_id

titles T

title_id PK
title
type
pub_id FK
pubdate
price
total_sales

sales SA

stor_id FK } PK
ord_num
date

au_id

stor_id
ord_num

title_id

title_id

titleauthor TA

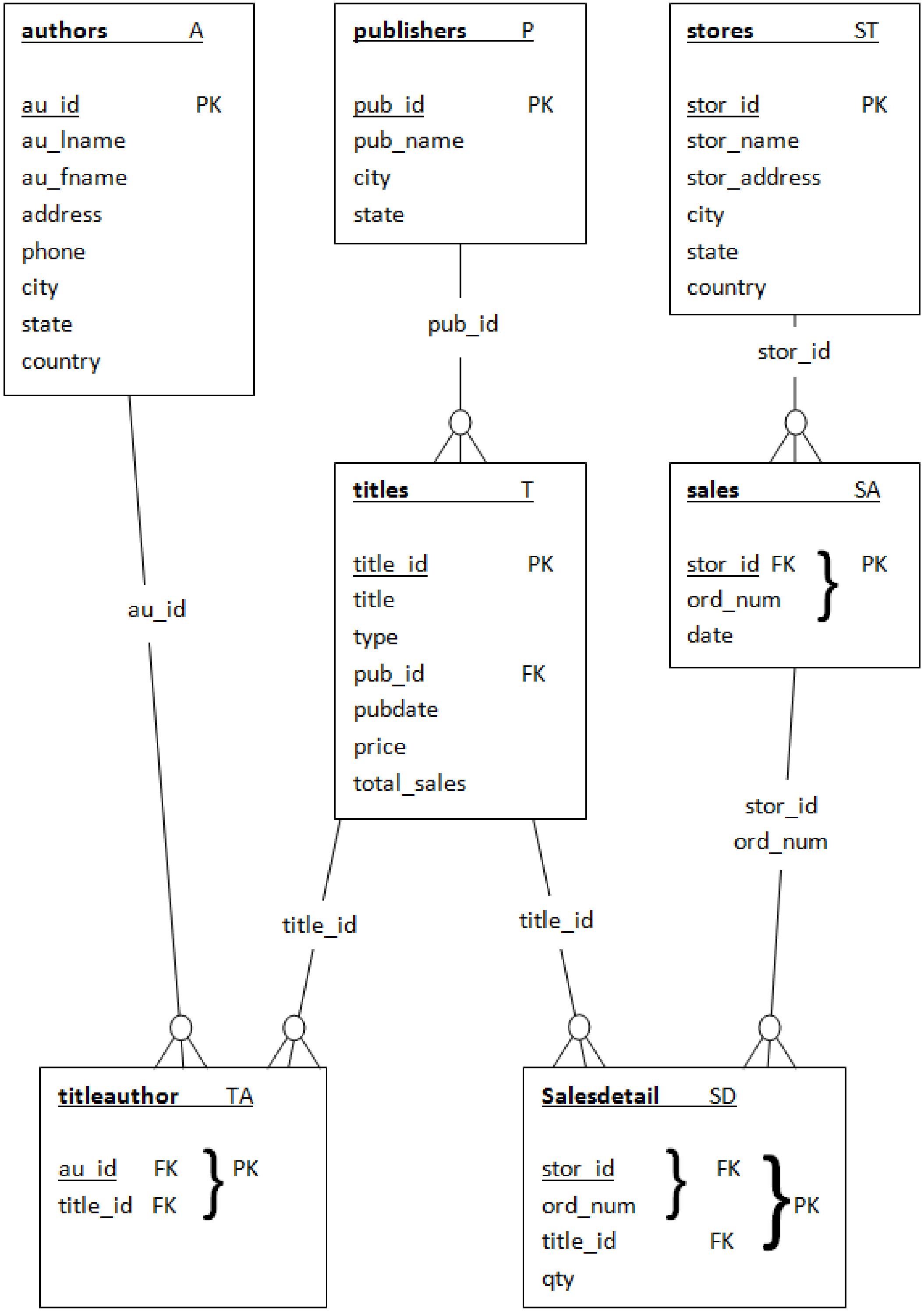
au_id FK } PK
title_id FK

Salesdetail SD

stor_id } FK } PK
ord_num
title_id FK
qty

Lister les noms des auteurs dont le nom termine par er

```
SELECT a.au_lname
FROM authors a
WHERE a.au_lname LIKE
   '%er'
```



authors A
au_id PK
au_lname
au_fname
address
phone
city
state
country

publishers P
pub_id PK
pub_name
city
state

stores ST
stor_id PK
stor_name
stor_address
city
state
country

titles T
title_id PK
title
type
pub_id FK
pubdate
price
total_sales

sales SA
stor_id FK } PK
ord_num
date

titleauthor TA
au_id FK } PK
title_id FK

Salesdetail SD
stor_id } FK } PK
ord_num
title_id FK
qty

au_id

pub_id

stor_id

title_id

title_id

stor_id
ord_num

Lister les noms des auteurs dont le nom termine par er

```
SELECT a.au_lname
FROM authors a
WHERE a.au_lname LIKE '%er'
```

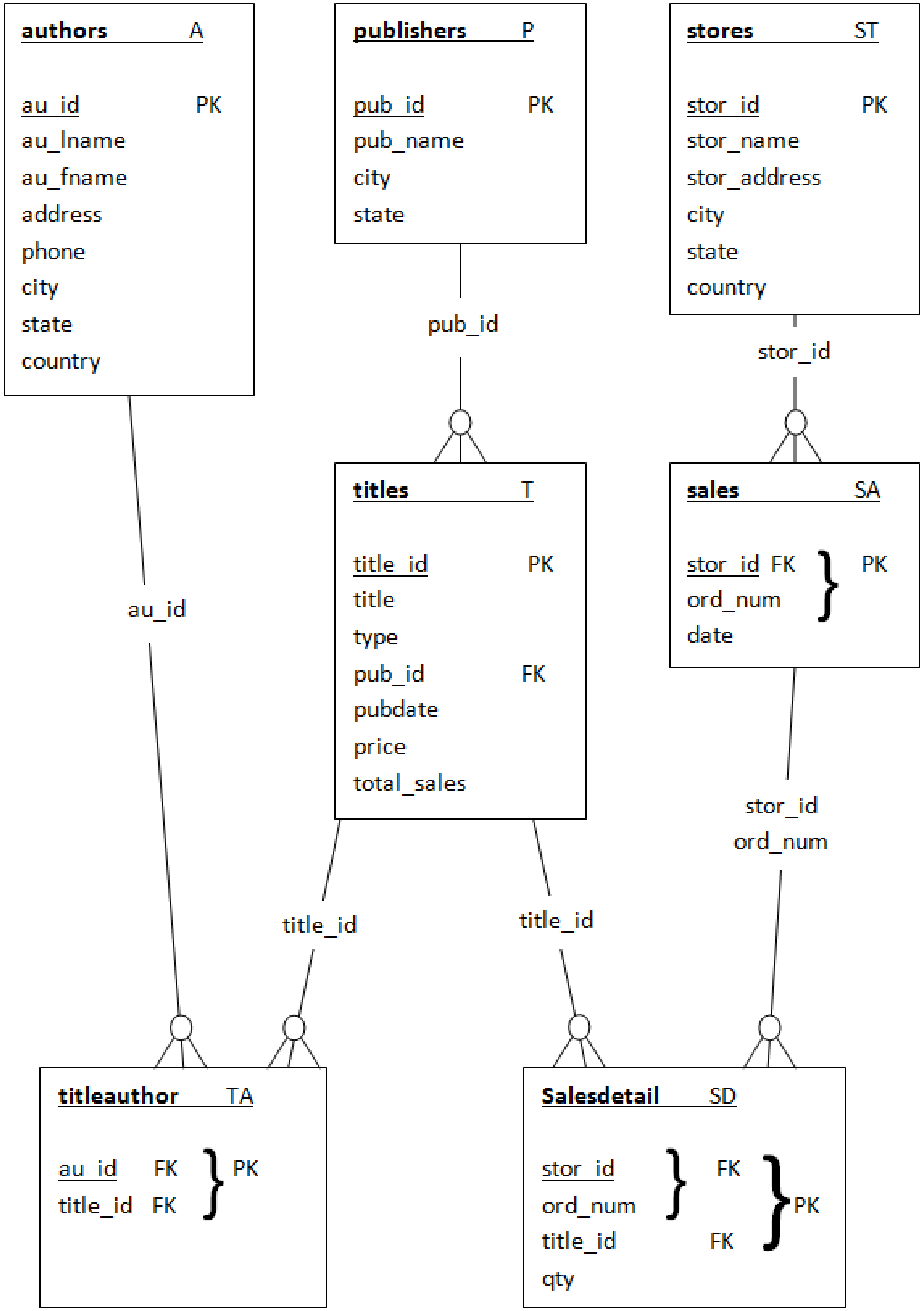
Ringer
Ringer
Stringer
MacFeather
Hunter

MacFeather
Stringer
Hunter
Ringer

```
SELECT DISTINCT a.au_lname
FROM authors a
WHERE a.au_lname LIKE '%er'
```

Lister les noms et prénoms des auteurs dont le nom commence par d

```
SELECT a.au_id, a.au_lname,
  a.au_fname
FROM authors a
WHERE a.au_lname SIMILAR TO
  '[dD]%';
```



authors A

au_id PK
au_lname
au_fname
address
phone
city
state
country

publishers P

pub_id PK
pub_name
city
state

stores ST

stor_id PK
stor_name
stor_address
city
state
country

titles T

title_id PK
title
type
pub_id FK
pubdate
price
total_sales

sales SA

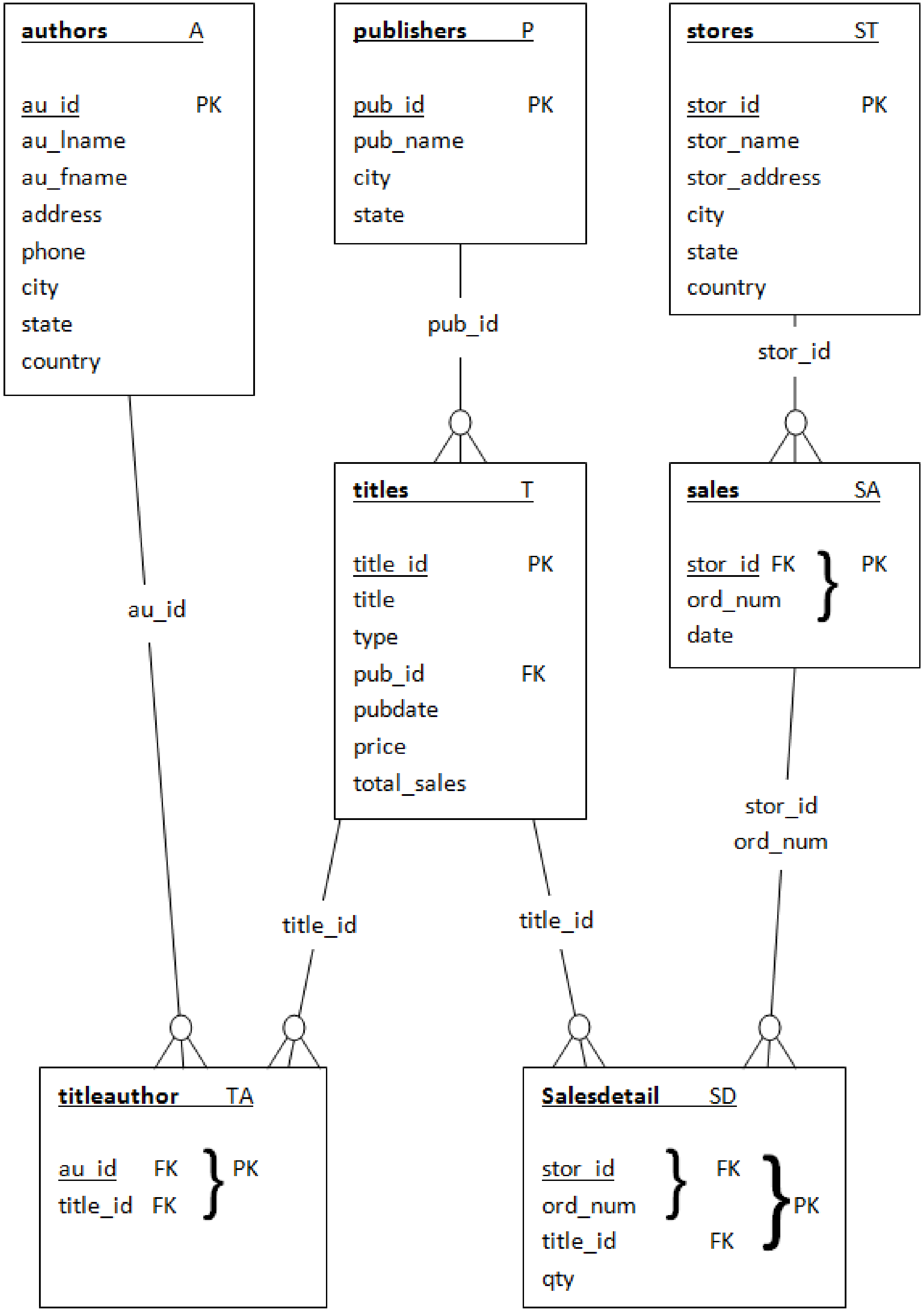
stor_id FK } PK
ord_num }
date

titleauthor TA

au_id FK } PK
title_id FK }

Salesdetail SD

stor_id } FK } PK
ord_num }
title_id FK }
qty



authors A
au_id PK
au_lname
au_fname
address
phone
city
state
country

publishers P
pub_id PK
pub_name
city
state

stores ST
stor_id PK
stor_name
stor_address
city
state
country

au_id

pub_id

stor_id

titles T
title_id PK
title
type
pub_id FK
pubdate
price
total_sales

sales SA
stor_id FK } PK
ord_num
date

stor_id
ord_num

title_id

title_id

titleauthor TA
au_id FK } PK
title_id FK

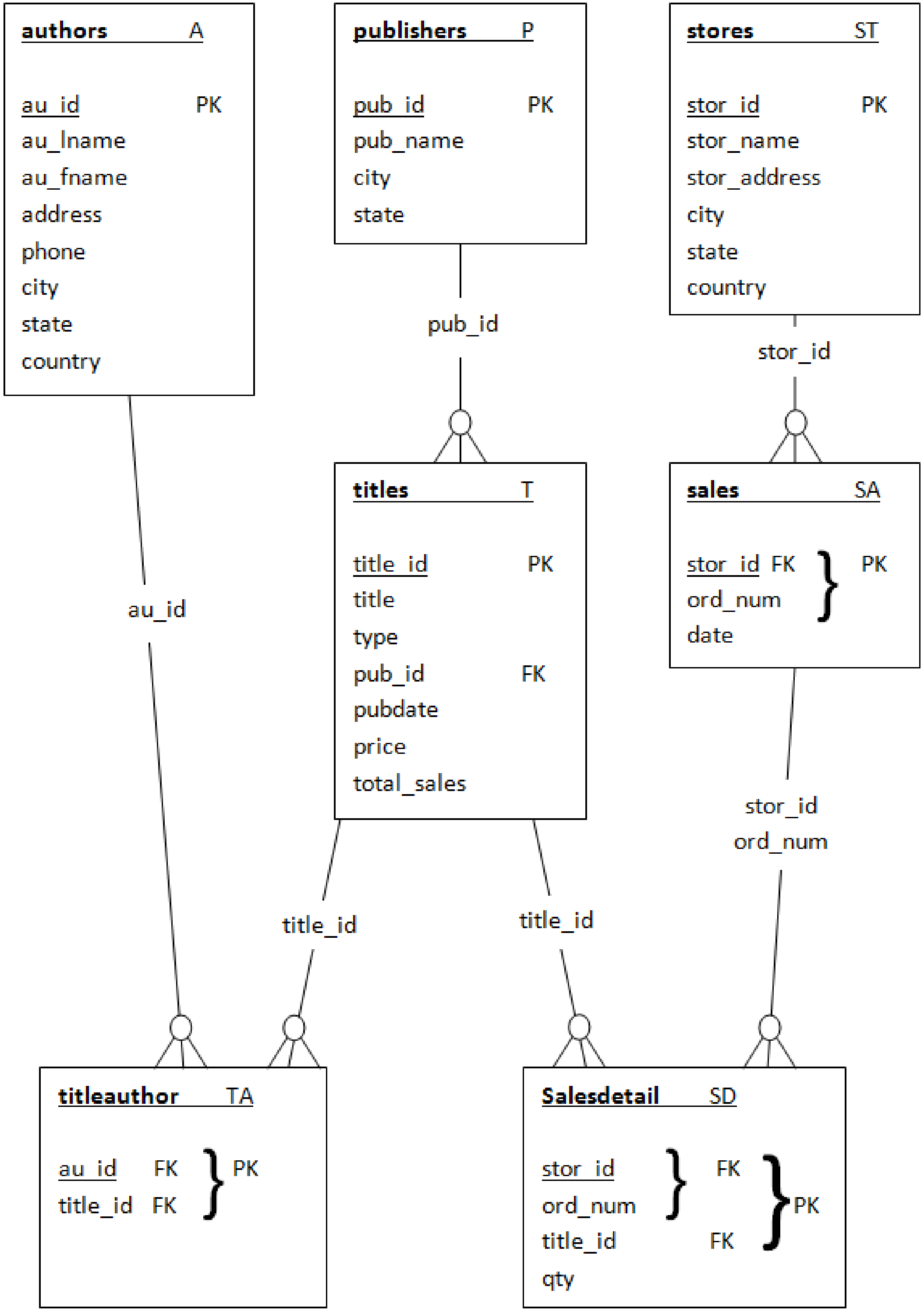
Salesdetail SD
stor_id } FK } PK
ord_num
title_id FK
qty

Lister les noms et prénoms des auteurs triés par ordre alphabétique

```
SELECT a.au_id, a.au_lname,
  a.au_fname

FROM authors a

ORDER BY a.au_lname ASC,
  a.au_fname ASC;
```

JOINTURE

```
SELECT [ ALL | DISTINCT]
    * | expression [ [ AS ] nom_d_affichage ] [, ...]
    [ FROM éléments_from [, ...] ]
    [ WHERE condition ]
    [ ORDER BY expression [ ASC | DESC ] [, ...] ]
```



authors A
au_id PK
au_lname
au_fname
address
phone
city
state
country

publishers P
pub_id PK
pub_name
city
state

stores ST
stor_id PK
stor_name
stor_address
city
state
country

titles T
title_id PK
title
type
pub_id FK
pubdate
price
total_sales

sales SA
stor_id FK } PK
ord_num
date

titleauthor TA
au_id FK } PK
title_id FK

Salesdetail SD
stor_id } FK } PK
ord_num
title_id FK
qty

pub_id

stor_id

au_id

title_id

title_id

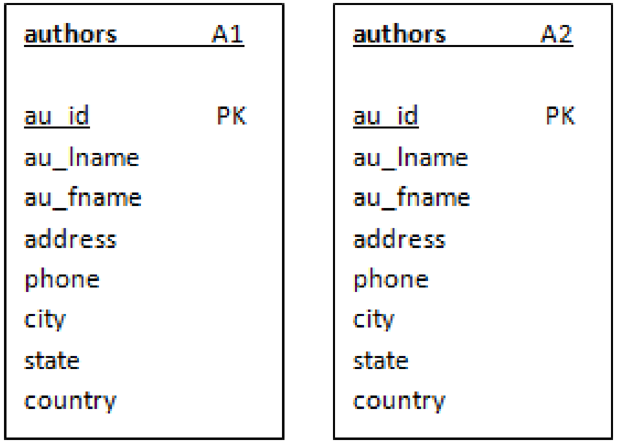
stor_id
ord_num

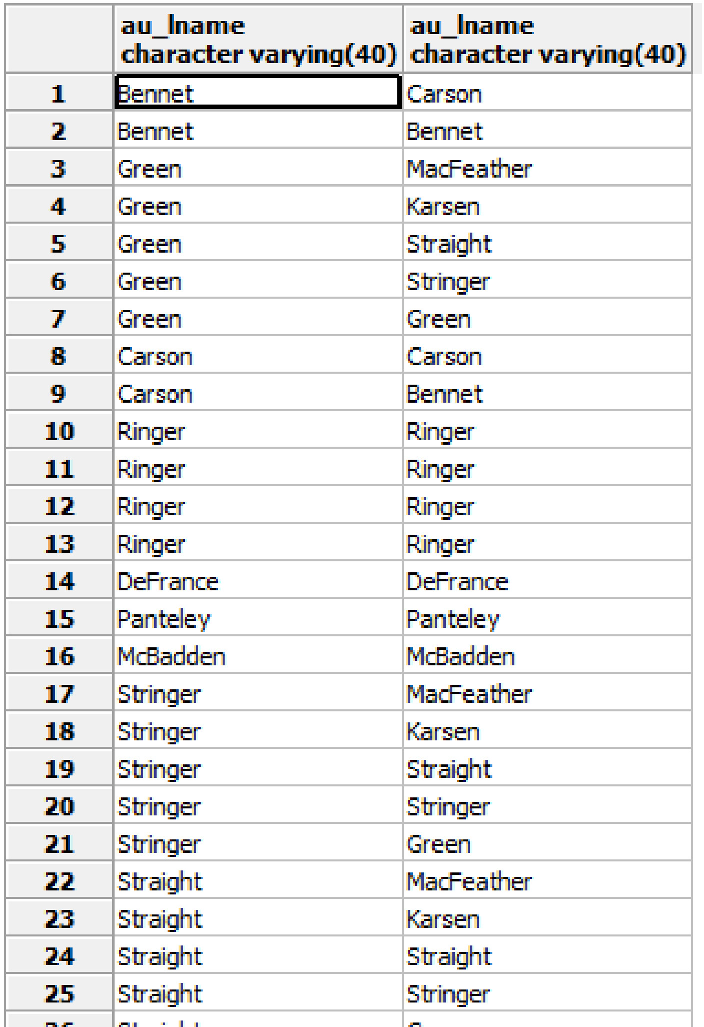
Lister les livres et leurs éditeurs

```
SELECT t.title_id,
  t.title,p.pub_name
FROM titles t,
  publishers p
WHERE t.pub_id=p.pub_id
```

Lister les paires d’auteurs qui vivent dans la même ville

Lister les paires d’auteurs qui vivent dans la même ville





	au_lname character varying(40)	au_lname character varying(40)
1	Bennet	Carson
2	Bennet	Bennet
3	Green	MacFeather
4	Green	Karsen
5	Green	Straight
6	Green	Stringer
7	Green	Green
8	Carson	Carson
9	Carson	Bennet
10	Ringer	Ringer
11	Ringer	Ringer
12	Ringer	Ringer
13	Ringer	Ringer
14	DeFrance	DeFrance
15	Panteley	Panteley
16	McBadden	McBadden
17	Stringer	MacFeather
18	Stringer	Karsen
19	Stringer	Straight
20	Stringer	Stringer
21	Stringer	Green
22	Straight	MacFeather
23	Straight	Karsen
24	Straight	Straight
25	Straight	Stringer

```
SELECT a1.au_lname,
  a2.au_lname
FROM authors a1, authors
  a2
WHERE a1.city=a2.city
```

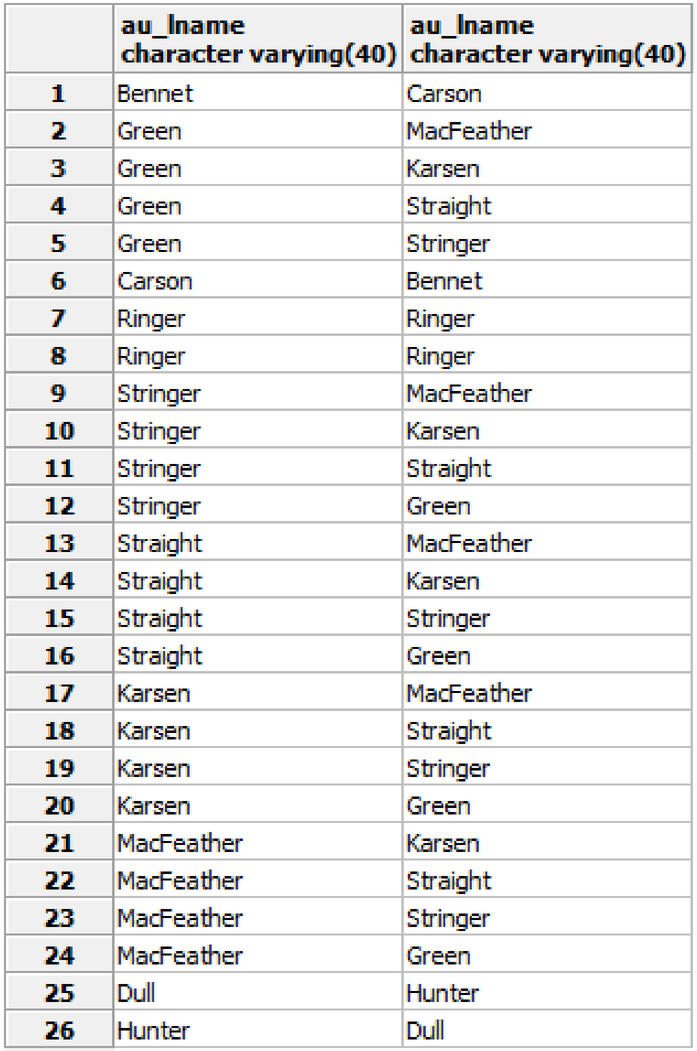
⇒ 49 résultats

authors A1

au_id PK
au_lname
au_fname
address
phone
city
state
country

authors A2

au_id PK
au_lname
au_fname
address
phone
city
state
country

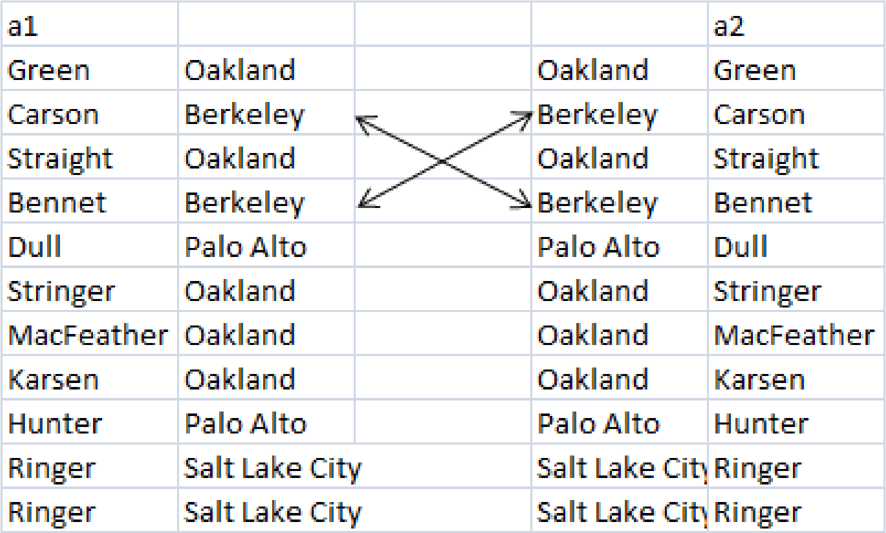


	au_lname character varying(40)	au_lname character varying(40)
1	Bennet	Carson
2	Green	MacFeather
3	Green	Karsen
4	Green	Straight
5	Green	Stringer
6	Carson	Bennet
7	Ringer	Ringer
8	Ringer	Ringer
9	Stringer	MacFeather
10	Stringer	Karsen
11	Stringer	Straight
12	Stringer	Green
13	Straight	MacFeather
14	Straight	Karsen
15	Straight	Stringer
16	Straight	Green
17	Karsen	MacFeather
18	Karsen	Straight
19	Karsen	Stringer
20	Karsen	Green
21	MacFeather	Karsen
22	MacFeather	Straight
23	MacFeather	Stringer
24	MacFeather	Green
25	Dull	Hunter
26	Hunter	Dull

Lister les paires d'auteurs qui vivent dans la même ville

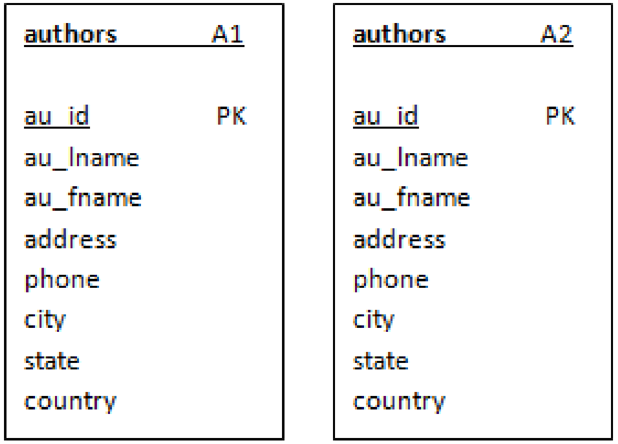
```
SELECT a1.au_lname,
  a2.au_lname

FROM authors a1, authors
  a2

WHERE a1.city=a2.city

AND a1.au_id<>a2.au_id
```

⇒ 26 résultats



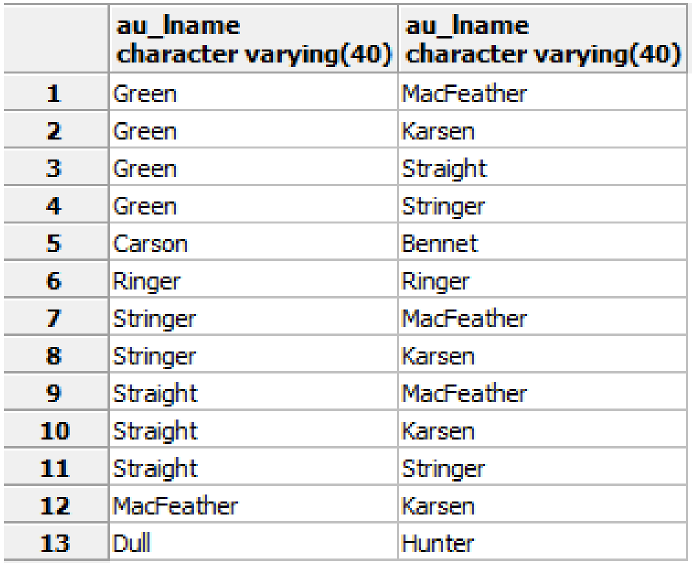
a1				a2
Green	Oakland		Oakland	Green
Carson	Berkeley		Berkeley	Carson
Straight	Oakland		Oakland	Straight
Bennet	Berkeley		Berkeley	Bennet
Dull	Palo Alto		Palo Alto	Dull
Stringer	Oakland		Oakland	Stringer
MacFeather	Oakland		Oakland	MacFeather
Karsen	Oakland		Oakland	Karsen
Hunter	Palo Alto		Palo Alto	Hunter
Ringer	Salt Lake City		Salt Lake Cit	Ringer
Ringer	Salt Lake City		Salt Lake Cit	Ringer

Lister les paires d'auteurs qui vivent dans la même ville

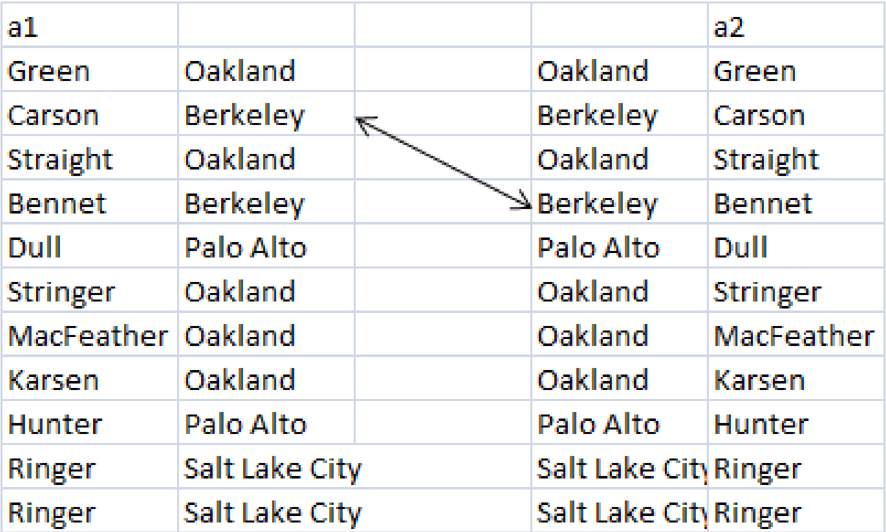


```
SELECT a1.au_lname,
  a2.au_lname
FROM authors a1, authors
  a2
WHERE a1.city=a2.city
AND a1.au_id<a2.au_id
```

⇒ 13 résultats



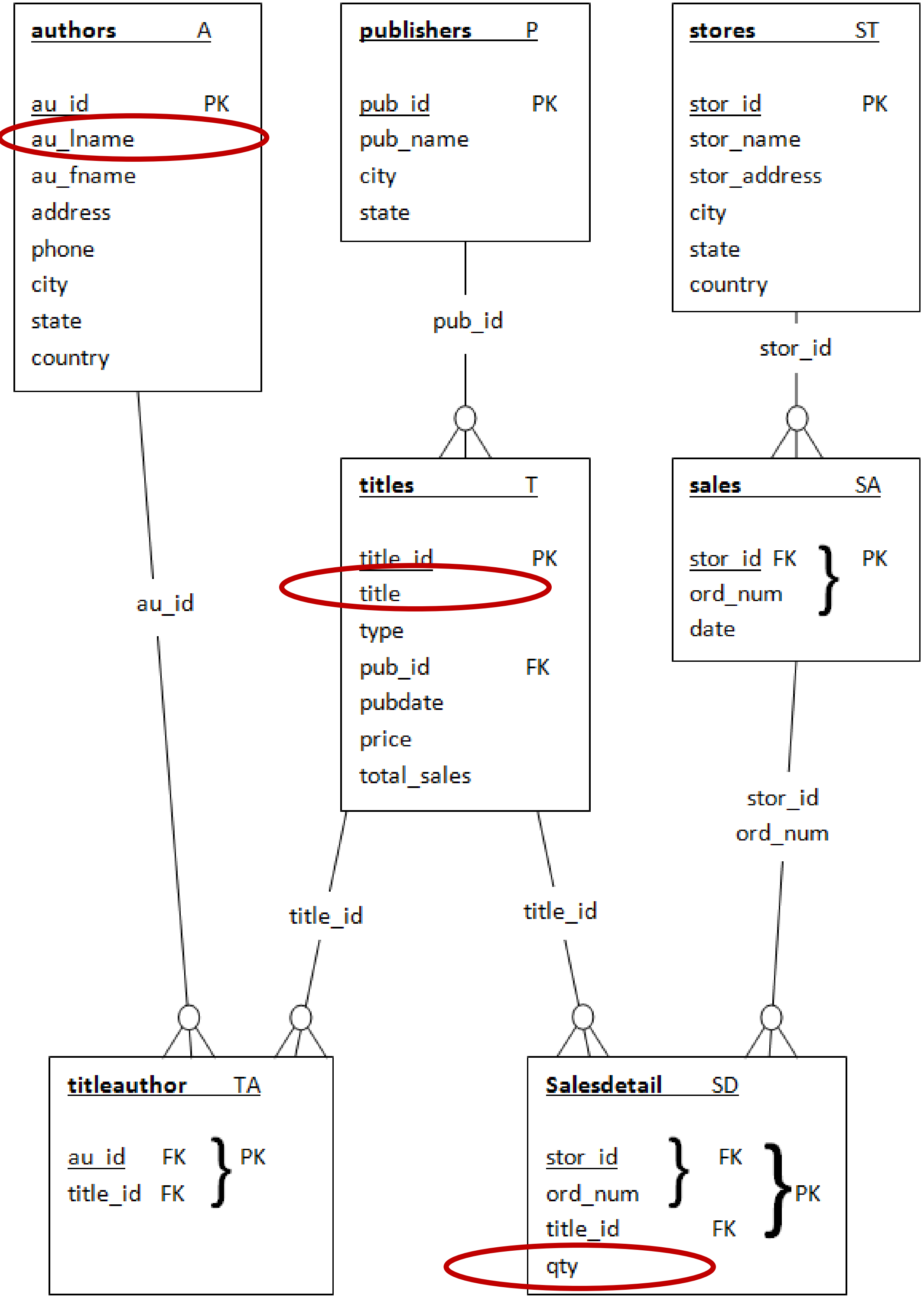
	au_lname character varying(40)	au_lname character varying(40)
1	Green	MacFeather
2	Green	Karsen
3	Green	Straight
4	Green	Stringer
5	Carson	Bennet
6	Ringer	Ringer
7	Stringer	MacFeather
8	Stringer	Karsen
9	Straight	MacFeather
10	Straight	Karsen
11	Straight	Stringer
12	MacFeather	Karsen
13	Dull	Hunter



a1				a2
Green	Oakland		Oakland	Green
Carson	Berkeley		Berkeley	Carson
Straight	Oakland		Oakland	Straight
Bennet	Berkeley		Berkeley	Bennet
Dull	Palo Alto		Palo Alto	Dull
Stringer	Oakland		Oakland	Stringer
MacFeather	Oakland		Oakland	MacFeather
Karsen	Oakland		Oakland	Karsen
Hunter	Palo Alto		Palo Alto	Hunter
Ringer	Salt Lake City		Salt Lake Cit	Ringer
Ringer	Salt Lake City		Salt Lake Cit	Ringer

Quels sont les livres vendus par Green, et en quelle quantité ?

Etape 1 : quels sont les champs dont on a besoin ?



authors A
au_id PK
au_lname
au_fname
address
phone
city
state
country

publishers P
pub_id PK
pub_name
city
state

stores ST
stor_id PK
stor_name
stor_address
city
state
country

pub_id

stor_id

au_id

titles T
title_id PK
title
type
pub_id FK
pubdate
price
total_sales

sales SA
stor_id FK } PK
ord_num
date

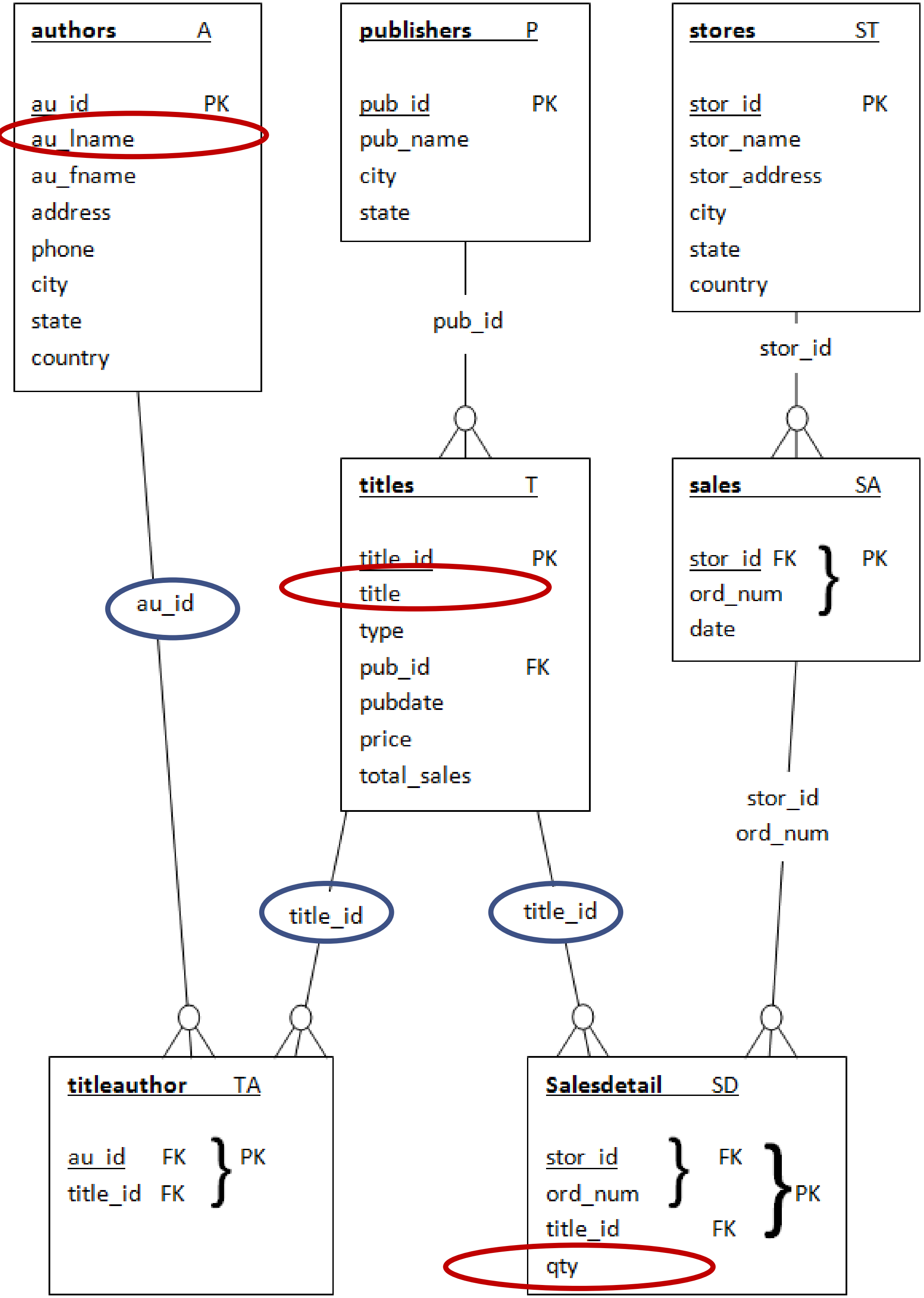
stor_id
ord_num

title_id

title_id

titleauthor TA
au_id FK } PK
title_id FK

Salesdetail SD
stor_id } FK } PK
ord_num
title_id FK
qty



authors A

au_id PK
au_lname
au_fname
address
phone
city
state
country

publishers P

pub_id PK
pub_name
city
state

stores ST

stor_id PK
stor_name
stor_address
city
state
country

pub_id

stor_id

au_id

titles T

title_id PK
title
type
pub_id FK
pubdate
price
total_sales

sales SA

stor_id FK } PK
ord_num
date

stor_id
ord_num

title_id

title_id

titleauthor TA

au_id FK } PK
title_id FK

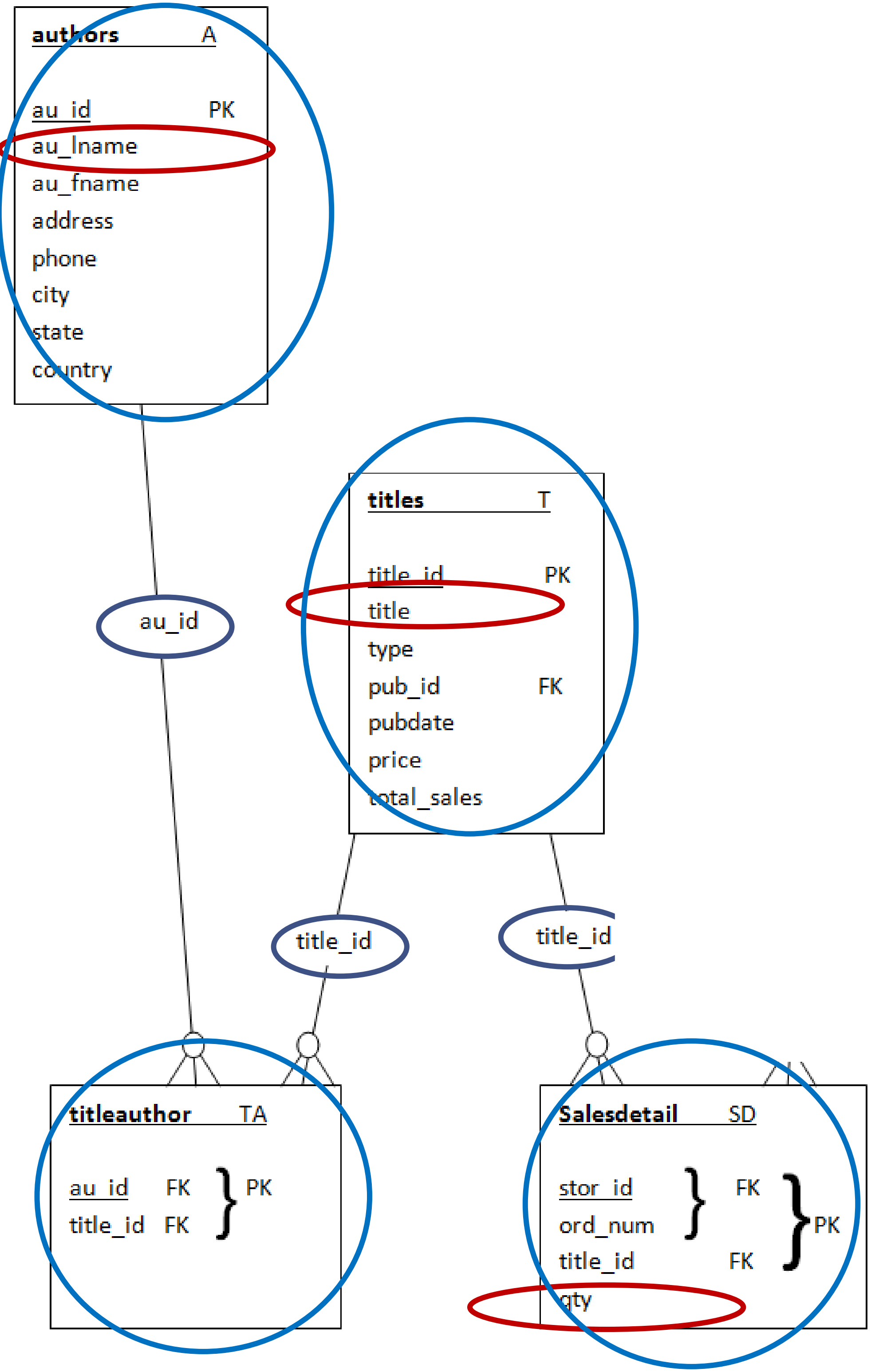
Salesdetail SD

stor_id } FK } PK
ord_num
title_id FK
qty

Quels sont les livres vendus par Green, et en quelle quantité ?

Etape 1 : quels sont les champs dont on a besoin ?

Etape 2 : comment les mettre en relation ?

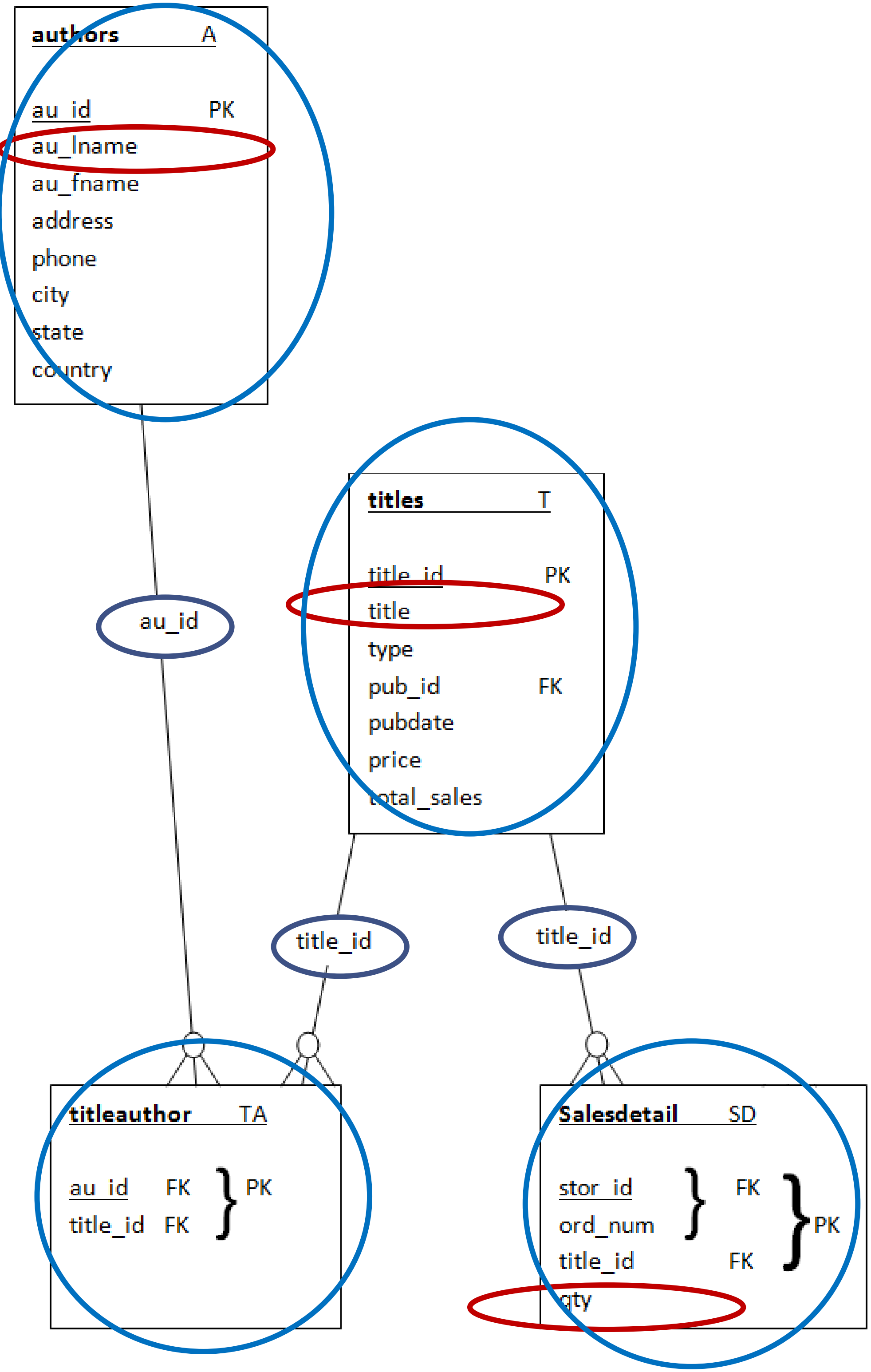


Quels sont les livres *vendus par Green, et en quelle quantité ?*

Etape 1 : quels sont les champs dont on a besoin ?

Etape 2 : comment les mettre en relation ?

=> jointures + tables



Quels sont les livres vendus par Green, et en quelle quantité ?

```
SELECT t.title,sd.qty

FROM authors a, titleauthor
   ta, salesdetail sd,
   titles t

WHERE a.au_lname = 'Green'

   AND a.au_id=ta.au_id

   AND
   ta.title_id=t.title_id

   AND
   ta.title_id=sd.title_id
```

The Busy Executive's Database Guide	320
The Busy Executive's Database Guide	136
The Busy Executive's Database Guide	345
The Busy Executive's Database Guide	94
The Busy Executive's Database Guide	1500
The Busy Executive's Database Guide	300
The Busy Executive's Database Guide	200
The Busy Executive's Database Guide	1000
The Busy Executive's Database Guide	200
You Can Combat Computer Stress!	135
You Can Combat Computer Stress!	200
You Can Combat Computer Stress!	4000
You Can Combat Computer Stress!	230
You Can Combat Computer Stress!	200
You Can Combat Computer Stress!	30
You Can Combat Computer Stress!	35
You Can Combat Computer Stress!	42
You Can Combat Computer Stress!	2200
You Can Combat Computer Stress!	3000
You Can Combat Computer Stress!	3000
You Can Combat Computer Stress!	2000
You Can Combat Computer Stress!	150
You Can Combat Computer Stress!	500

Pas très utile comme information !

GROUP BY HAVING

```
SELECT [ ALL | DISTINCT]
* | expression [ [ AS ] nom_d_affichage ] [, ...]
    [ FROM éléments_from [, ...] ]
    [ WHERE condition ]
    [ GROUP BY expression [, ...] ]
    [ HAVING condition [, ...] ]
    [ ORDER BY expression [ ASC | DESC ] [, ...] ]
```

The Busy Executive's Database Guide	320
The Busy Executive's Database Guide	136
The Busy Executive's Database Guide	345
The Busy Executive's Database Guide	94
The Busy Executive's Database Guide	1500
The Busy Executive's Database Guide	300
The Busy Executive's Database Guide	200
The Busy Executive's Database Guide	1000
The Busy Executive's Database Guide	200
You Can Combat Computer Stress!	135
You Can Combat Computer Stress!	200
You Can Combat Computer Stress!	4000
You Can Combat Computer Stress!	230
You Can Combat Computer Stress!	200
You Can Combat Computer Stress!	30
You Can Combat Computer Stress!	35
You Can Combat Computer Stress!	42
You Can Combat Computer Stress!	2200
You Can Combat Computer Stress!	3000
You Can Combat Computer Stress!	3000
You Can Combat Computer Stress!	2000
You Can Combat Computer Stress!	150
You Can Combat Computer Stress!	500

GROUP BY t.title

Chaque groupe doit être réduit à un seul élément dans la sortie du SELECT

The Busy Executive's Database Guide	320
The Busy Executive's Database Guide	136
The Busy Executive's Database Guide	345
The Busy Executive's Database Guide	94
The Busy Executive's Database Guide	1500
The Busy Executive's Database Guide	300
The Busy Executive's Database Guide	200
The Busy Executive's Database Guide	1000
The Busy Executive's Database Guide	200
You Can Combat Computer Stress!	135
You Can Combat Computer Stress!	200
You Can Combat Computer Stress!	4000
You Can Combat Computer Stress!	230
You Can Combat Computer Stress!	200
You Can Combat Computer Stress!	30
You Can Combat Computer Stress!	35
You Can Combat Computer Stress!	42
You Can Combat Computer Stress!	2200
You Can Combat Computer Stress!	3000
You Can Combat Computer Stress!	3000
You Can Combat Computer Stress!	2000
You Can Combat Computer Stress!	150
You Can Combat Computer Stress!	500

```
SELECT t.title,SUM(sd.qty)
FROM authors a, titleauthor ta,
salesdetail sd, titles t
WHERE a.au_lname = 'Green'
    AND a.au_id=ta.au_id
    AND ta.title_id=sd.title_id
    AND t.title_id=ta.title_id
GROUP BY t.title
```

The Busy Executive's Database Guide	4095
You Can Combat Computer Stress!	15722

Que se passe-t-il si deux livres portent le même titre ?

The Busy Executive's Database Guide	320
The Busy Executive's Database Guide	136
The Busy Executive's Database Guide	345
The Busy Executive's Database Guide	94
The Busy Executive's Database Guide	1500
The Busy Executive's Database Guide	300
The Busy Executive's Database Guide	200
The Busy Executive's Database Guide	1000
The Busy Executive's Database Guide	200
You Can Combat Computer Stress!	135
You Can Combat Computer Stress!	200
You Can Combat Computer Stress!	4000
You Can Combat Computer Stress!	230
You Can Combat Computer Stress!	200
You Can Combat Computer Stress!	30
You Can Combat Computer Stress!	35
You Can Combat Computer Stress!	42
You Can Combat Computer Stress!	2200
You Can Combat Computer Stress!	3000
You Can Combat Computer Stress!	3000
You Can Combat Computer Stress!	2000
You Can Combat Computer Stress!	150
You Can Combat Computer Stress!	500

```
SELECT t.title,SUM(sd.qty)
FROM authors a, titleauthor
ta, salesdetail sd, titles t
WHERE a.au_lname = 'Green'
    AND a.au_id=ta.au_id
    AND ta.title_id=sd.title_id
    AND t.title_id=ta.title_id
GROUP BY t.title_id
```

The Busy Executive's Database Guide	4095
You Can Combat Computer Stress!	15722

C'est la PK qui garantit l'identité du livre, pas son titre !

The Busy Executive's Database Guide	320
The Busy Executive's Database Guide	136
The Busy Executive's Database Guide	345
The Busy Executive's Database Guide	94
The Busy Executive's Database Guide	1500
The Busy Executive's Database Guide	300
The Busy Executive's Database Guide	200
The Busy Executive's Database Guide	1000
The Busy Executive's Database Guide	200
You Can Combat Computer Stress!	135
You Can Combat Computer Stress!	200
You Can Combat Computer Stress!	4000
You Can Combat Computer Stress!	230
You Can Combat Computer Stress!	200
You Can Combat Computer Stress!	30
You Can Combat Computer Stress!	35
You Can Combat Computer Stress!	42
You Can Combat Computer Stress!	2200
You Can Combat Computer Stress!	3000
You Can Combat Computer Stress!	3000
You Can Combat Computer Stress!	2000
You Can Combat Computer Stress!	150
You Can Combat Computer Stress!	500

```
SELECT t.title,SUM(sd.qty)
FROM authors a, titleauthor
ta, salesdetail sd, titles t
WHERE a.au_lname = 'Green'
   AND a.au_id=ta.au_id
   AND ta.title_id=sd.title_id
   AND t.title_id=ta.title_id
GROUP BY t.title_id
HAVING SUM(sd.qty)>5000
```

You Can Combat Computer Stress!	15722

OPÉRATEURS D'AGGRÉGATION

- `COUNT`
- `SUM`
- `MIN`
- `MAX`
- `AVG`