

# Base de données

---

## Séance 6

Requêtes avec SELECT (suite)

# Plan

---

- Rappel :
  1. Jointure avec pivot
  2. Jointure interne
  3. Jointure naturelle
- Fonction d'agrégat (GROUP BY)

# Exemple complet

---

**Individu**(num\_ind, nom, prenom)

**Jouer**(#num\_ind, #num\_film, role)

**Film**(num\_film, #num\_ind, titre, genre, annee)

**Projection**(#num\_cine, #num\_film, pdate)

**Cinéma**(num\_cine, nom, adresse)

---

# Individu

---

num_ind	nom	prenom
01	Kidman	Nicole
02	Bettany	Paul
03	Watson	Emily
04	Skarsgard	Stellan
05	Travolta	John
06	L. Jackson	Samuel
07	Willis	Bruce
08	Irons	Jeremy
09	Spader	James
10	Hunter	Holly
11	Arquette	Rosanna
12		John
13	von Trier	Lars
14	Tarantino	Quentin
15	Cronenberg	David
16	Mazursky	Paul
17	Jones	Grace
18	Glen	John

# Film

---

num_film	num_ind	titre	genre	année
05	13	Dogville	Drame	2002
04	13	Breaking the waves	Drame	1996
03	14	Pulp Fiction	Policier	1994
02	15	Faux-Semblants	Epouvante	1988
01	15	Crash	Drame	1996
06	12	Alamo	Western	1960
07	18	Dangereusement vôtre	Espionnage	1985

---

# Jouer

---

num_ind	num_film	role
01	05	Grace
02	05	Tom Edison
03	04	Bess
04	04	Jan
05	03	Vincent Vega
06	03	Jules Winnfield
07	03	Butch Coolidge
08	02	Beverly & Elliot Mantle
09	01	James Ballard
10	01	Helen Remington
11	01	Gabrielle
04	05	Chuck
16	07	May Day

---

# Exemple complet

---

**Individu**(num\_ind, nom, prenom)

**Jouer**(#num\_ind, #num\_film, role)

**Film**(num\_film, #num\_ind, titre, genre, annee)

**Projection**(#num\_cine, #num\_film, pdate)

**Cinéma**(num\_cine, nom, adresse)

- Quels sont les **cinémas** qui ont **projeté** des **films** dans lesquels a **joué Travolta** ?
-

# Exemple complet

---

**Individu**(num\_ind, nom, prenom)

**Jouer**(#num\_ind, #num\_film, role)

**Film**(num\_film, #num\_ind, titre, genre, annee)

**Projection**(#num\_cine, #num\_film, pdate)

**Cinéma**(num\_cine, nom, adresse)

- Quels sont les **cinémas** qui ont **projeté** des **films** dans lesquels a **joué Travolta** ?
-



# 1. Jointure avec pivot

---

```
SELECT DISTINCT c.nom
FROM cinema c, projection p, film f, jouer j, individu i
WHERE      c.num_cine = p.num_cine
AND        p.num_film = f.num_film
AND        f.num_film = j.num_film
AND        j.num_ind = i.num_ind
AND        i.nom = 'Travolta';
```

4 **pivots** + 1 **sélection** + 1 **projection**

---

## 2. Jointure interne

---

```
SELECT DISTINCT c.nom
FROM cinema c
JOIN projection p ON c.num_cine = p.num_cine
JOIN film f ON p.num_film = f.num_film
JOIN jouer j ON f.num_film = j.num_film
JOIN individu i ON j.num_ind = i.num_ind
WHERE i.nom = 'Travolta';
```

4 **jointures** + 1 **sélection** + 1 **projection**

---

### 3. Jointure naturelle

---

```
SELECT DISTINCT nom
FROM    cinema NATURAL JOIN projection
        NATURAL JOIN film
        NATURAL JOIN jouer
        NATURAL JOIN individu
WHERE   nom = 'Travolta';
```

*Problème : ambiguïtés sur les colonnes qui portent le même nom, on peut préciser celle(s) à utiliser avec la clause USING mais à EVITER !*

---

# Ambigüité jointure naturelle

---

**Individu**(num\_ind, nom, prenom)

**Jouer**(#num\_ind, #num\_film, role)

**Film**(num\_film, #num\_ind, titre, genre, annee)

**Projection**(#num\_cine, #num\_film, pdate)

**Cinéma**(num\_cine, nom, adresse)

- Quels sont les **cinémas** qui ont **projeté** des **films** dans lesquels a **joué Travolta** ?
-

# Agrégats

---

- Partitionnement horizontal d'une table selon les valeurs d'un groupe d'attributs appelés facteurs de groupage
  - Suivi d'un regroupement par une fonction de calcul en colonne (SUM, MIN, MAX, AVG, COUNT, ...)
  - La table résultat contient une ligne par partition avec :
    - des facteurs de groupage (valeur unique par partition)
    - le résultat du calcul
-

# Exemple 1 : calculer le degré moyen de vins par cru

---

Vins	cru	mill	degre	quantite
	CHABLIS	1977	10.9	100
	CHABLIS	1987	11.	250
	VOLNA	1977	10.8	400
	Y VOLNA	1986	11.	300
	Y MEDOC	1985	11.	200

# Exemple 1 : calculer le degré moyen de vins par cru

Vins	cru	mill	degre	quantite
	CHABLIS	1977	10.9	100
	CHABLIS	1987	11.	250
	VOLNA	1977	10.8	400
	VOLNA	1986	11.	300
	MEDOC	1985	11.	200

AVG	cru	AVG(degre)
	CHABLIS	11.
	VOLNA	11.
	MEDOC	11.

## Exemple 2 : calculer la somme de quantité pour chaque cru

---

Vins	cru	mill	degre	quantite
	CHABLIS	1977	10.9	100
	CHABLIS	1987	11.	250
	VOLNA	1977	10.8	400
	VOLNA	1986	11.	300
	MEDOC	1985	11.	200

2

SUM	cru	SUM(quantite)
	CHABLIS	350
	VOLNA	700
	MEDOC	200



# Syntaxe SQL de groupage

---

- Pour regrouper des données, il faut utiliser la clause **GROUP BY** suivi du facteur de groupage.

```
SELECT
    <col1>, <col2>, <fonction_agrégat>(<col3>)
FROM
    ...
WHERE
    ...
GROUP BY <col1>, <col2>;
```


## Groupage : exemple 1

---

- Calculer les quantités récoltées pour chaque producteur.

```
SELECT
    numprod, SUM(quantite)
FROM
    recolte
GROUP BY
    numprod ;
```

Groupeur



## Groupage : exemple 2

---

- Calculer le nombre de films dramatiques réalisés par chaque réalisateur.

Film(num\_film, *num\_ind*, titre, genre, annee)

```
SELECT num_ind, COUNT(*)  
FROM Film  
WHERE genre = 'Drame'  
GROUP BY num_ind;
```

## Groupage : la clause HAVING

---

- Elle permet d'opposer un prédicat à un groupe.

```
SELECT num_client, SUM(montant)
FROM commandes
GROUP BY num_client
HAVING SUM(montant) > 1000;
```

- HAVING s'applique sur le résultat de la requête.
- *La clause **HAVING** élimine des groupes comme la clause **WHERE** élimine des lignes !*

# WHERE ou HAVING ?

---

- **Question 1** : Calculer pour chaque cinéma le nombre de projections dont la date est antérieure à '2000-01-01'.
- **Question 2** : Quels sont les cinémas dont le nombre de projections est supérieur ou égal à 4 ?

Projection(num\_cine, num\_film, pdate)

---

# WHERE ou HAVING ?

---

## □ Réponse 1 :

```
SELECT num_cine, COUNT(*)  
FROM Projection  
WHERE pdate < '2000-01-01'  
GROUP BY num_cine;
```

## □ Réponse 2 :

```
SELECT num_cine, COUNT(*)  
FROM Projection  
GROUP BY num_cine  
HAVING COUNT(*) >= 4;
```

---