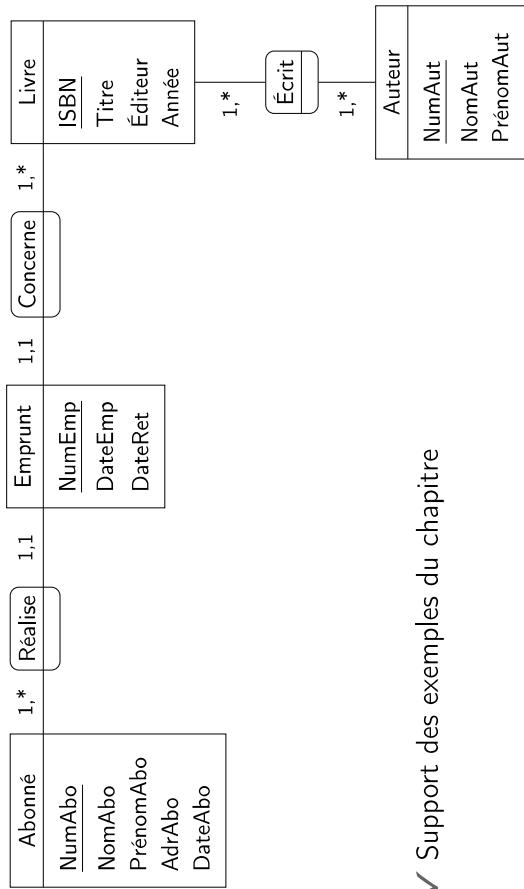


Exemple BD Bibliothèque : schéma E-A



✓ Support des exemples du chapitre

Exemple BD Bibliothèque : schéma relationnel

- Schéma relationnel de la BD

```
ABONNÉ(NumAbo, NomAbo, PrénomAbo, AdrAbo, DateAbo)
LIVRE(ISBN, Titre, Éditeur, Année)
AUTEUR(NumAut, NomAut, PrénomAut)
ÉCRIT(ISBN, NumAut)
EMPRUNT(NumEmp, NumAbo, ISBN, DateEmp, DateRet)
```

Exemple BD Bibliothèque : tables

- Tables (relations).

LIVRE				AUTEUR			
ISBN	Titre	Éditeur	Année	NumAut	NomAut	PrénomAut	
9782212112818	Bases de Données	Eyrolles	1989	1	Gardarin	Georges	
978222286805158	Le Langage C	Masson	1985	2	Kernighan	Brian	
97822207257357	Fondation	Dencél	2006	3	Ritchie	Dennis	
				4	Asimov	Isaac	

ÉCRIT				ABONNE			
NumAbo	NumAut	PrénomAbo	DateAbo	NumAbo	NomAbo	PrénomAbo	DateAbo
1	1	Dupont	2008-06-18	1	Philippe	Philippe	2008-06-18
2	2	Durand	2009-01-02	2	Arthur	Arthur	2009-01-02
3	3	Dupont	2015-05-03	3	Charlie	Charlie	2015-05-03
4	4	Ducros	2020-07-04	4	Marie	Marie	2020-07-04
5	5	Vernier	2021-09-15	5	Alain	Alain	2021-09-15

EMPRUNT				
NumEmp	ISBN	NomAbo	DateEmp	DateRet
1	978222286805158	2	2021-09-06	2021-09-20
2	978222286805158	3	2021-09-25	2021-10-11
3	9782212112818	1	2021-10-28	2021-11-10
4	9782212112818	1	2021-11-08	NULL

- Mots-clés de SQL : caractères COURIER majuscules
- Paramètres des requêtes : caractères courier minuscules
- Paramètres optionnels : [option]
- Valeurs multiples possibles : valeur₁|valeur₂
- Options multiples : [option₁|option₂]

Format général des requêtes :

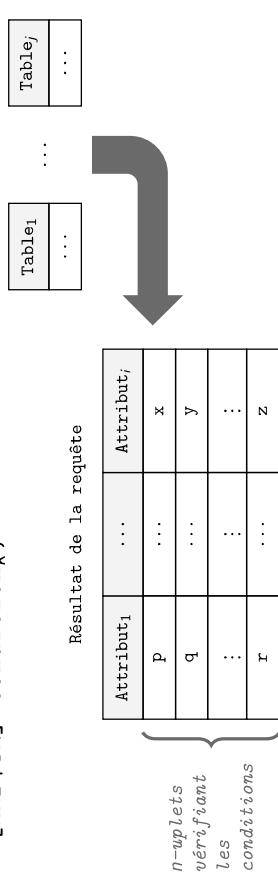
- SELECT spécifie les colonnes qui doivent apparaître dans les résultats
- FROM spécifie la table ou les tables à utiliser
- WHERE filtre les lignes selon une condition donnée
- GROUP BY forme des groupes de lignes de même valeur de colonne
- HAVING filtre les groupes sujets à une certaine condition
- ORDER BY spécifie l'ordre d'apparition des données dans le résultat

Langage d'interrogation de données

Requête simple de sélection

- Retrouver les attributs des n -uplets (lignes/tuples) vérifiant certains critères.

```
SELECT attribut1, ..., attributi
FROM table1, ..., tablej
WHERE condition1
[AND | OR] ...
[AND | OR] conditionk;
```



Requête simple de sélection

- Exemple :

```
SELECT NomAbo, PrénomAbo, DateAbo
FROM Abonné;
```
- Sélection des noms, prénoms et dates d'abonnement de tous les abonnés.

NomAbo	PrénomAbo	DateAbo
Dupont	Philippe	2008-06-18
Durand	Arthur	2009-01-02
Dupont	Charlie	2015-05-03
Ducros	Marie	2020-07-04
Vernier	Alain	2021-09-15

Sélection de tous les attributs

- Le symbole * est synonyme de « tous les attributs » :

```
SELECT *
FROM Abonné;
```

- Sélection de tous les attributs pour tous les abonnés.

NumAbo	NomAbo	PrénomAbo	DateAbo
1	Dupont	Philippe	2008-06-18
2	Durand	Arthur	2009-01-02
3	Dupont	Charlie	2015-05-03
4	Ducros	Marie	2020-07-04
5	Vernier	Alain	2021-09-15

La clause WHERE

- La clause WHERE permet de définir des critères de sélection des n-uplets :

```
SELECT *  
FROM Abonné  
WHERE NumAbo > 2;
```

- Sélection de tous les attributs pour tous les abonnés dont le numéro est supérieur (strictement) à 2.

NumAbo	NomAbo	PrénomAbo	DateAbo
3	Dupont	Charlie	2015-05-03
4	Ducros	Marie	2020-07-04
5	Vernier	Alain	2021-09-15

Requêtes multi-critères

- Combinaison de critères de sélection des n-uplets :
- ```
SELECT *
FROM Abonné
WHERE NumAbo >= 2 AND NumAbo <=4;
```
- Sélection de tous les attributs pour les abonnés dont le numéro est compris entre 2 et 4.

| NumAbo | NomAbo | PrénomAbo | DateAbo    |
|--------|--------|-----------|------------|
| 2      | Durand | Arthur    | 2009-01-02 |
| 3      | Dupont | Charlie   | 2015-05-03 |
| 4      | Ducros | Marie     | 2020-07-04 |

## Requêtes multi-critères (2)

- Combinaison de critères de sélection des n-uplets :

```
SELECT *
FROM Abonné
WHERE NumAbo <= 2 OR NumAbo > 4;

Selection de tous les attributs pour les abonnés dont le numéro est
inférieur ou égal à 2 ou supérieur strictement à 4.
```

| NumAbo | NomAbo  | PrénomAbo | DateAbo    |
|--------|---------|-----------|------------|
| 1      | Dupont  | Philippe  | 2008-06-18 |
| 2      | Durand  | Arthur    | 2009-01-02 |
| 5      | Vernier | Alain     | 2021-09-15 |

## Disjonctions et conjonctions de critères

- Si la requête mélange conjonctions (AND) et disjonctions (OR) de critères, il faut faire attention à la priorité des opérateurs.
- Exemple :

```
SELECT *
FROM Emprunt
WHERE NumAbo = 1 AND DateEmp = '2021-10-28'
OR DateEmp = '2021-09-06';
est équivalent à
```

```
SELECT *
FROM Emprunt
WHERE (NumAbo = 1 AND DateEmp = '2021-10-28')
OR DateEmp = '2021-09-06';
```

Résultat : liste des emprunts de l'abonné n°1 le 28 octobre 2021 et de tous les emprunts du 6 septembre 2021.

- AND est prioritaire sur OR.

57 / 69

## Opérateurs de comparaison

- Opérateurs de comparaison :
  - =, <>, <, >, >=, <= : comparaisons classiques
  - BETWEEN val<sub>1</sub> AND val<sub>2</sub> : valeurs comprises entre val<sub>1</sub> et val<sub>2</sub>
  - IS NULL : valeur NULL
  - IS NOT NULL : valeurs non NULL
  - IN (listeValeurs) : valeurs appartenant à la liste
  - NOT IN (listeValeurs) : valeurs n'appartenant pas à la liste
- Exemple :

```
SELECT *
FROM Abonnés
WHERE NumAbo BETWEEN 2 AND 4;
```

## Sélection sur les chaînes de caractères

- Sélection des n-uplets contenant une chaîne

```
SELECT *
FROM Abonnés
WHERE NomAbo = 'Durand';
```

- Sélection des abonnés dont le nom est Durand.

| NumAbo | NomAbo | PrénomAbo | DateAbo    |
|--------|--------|-----------|------------|
| 2      | Durand | Arthur    | 2009-01-02 |

## Sélection de sous-chaines de caractères

- Motifs de sélection des chaînes attribut [NOT] LIKE expr
  - L'attribut attribut doit / ne doit pas correspondre à l'expression expr
  - Jokers utilisables dans expr
    - Symbole % : n'importe quelle chaîne de caractères, vide y compris
    - Symbole \_ : n'importe quel caractère (un seul, pas vide).
  - Exemple :
- ```
SELECT *  
FROM Abonné  
WHERE NomAbo LIKE 'Dupon_';
```
- Sélection de tous les attributs pour les abonnés dont le nom est Dupond, Dupont, ... (mais pas Dupon).

60 / 69

Sélection sur les chaînes de caractères

- Exemple :

```
SELECT *
FROM Abonné
WHERE NomAbo LIKE 'Du%n_';
```
- Sélection des abonnés dont le nom est Dupont, Dupond, Durant, Durand, Dune, etc.

NumAbo	NomAbo	PrénomAbo	DateAbo
1	Dupont	Philippe	2008-06-18
2	Durand	Arthur	2009-01-02
3	Dupont	Charlie	2015-05-03

Sélections sur les dates

- Dates entourées de ' '
- Valeurs séparées par - -

```
SELECT *  
FROM Abonné  
WHERE DateAbo > '2007-10-01';
```

- Sélection des abonnés inscrits après le 1^{er} octobre 2007.

NumAbo	NomAbo	PrenomAbo	DateAbo
1	Dupont	Philippe	2008-06-18
2	Durand	Arthur	2009-01-02
3	Dupont	Charlie	2015-05-03
4	Ducros	Marie	2020-07-04
5	Vernier	Alain	2021-09-15

Opérateur BETWEEN

- Applicable aux mêmes types que les opérateurs =, <, >
- Sélection des abonnés inscrits entre le 1^{er} et le 31 octobre inclus :

```
SELECT *  
FROM Abonné  
WHERE DateAbo BETWEEN '2007-10-01' AND '2007-10-31';  
  
• Sélection des abonnés de nom supérieur à « Dupond » et inférieur à  
« Ferrand » selon l'ordre lexicographique (alphabétique) :  
  
SELECT *  
FROM Abonné  
WHERE NomAbo BETWEEN 'Dupond' AND 'Ferrand';
```

Valeurs NULL

- Opérateur IS NULL
 - Vrai si la valeur est NULL (absente, inappropriée, valeur par défaut)
 - Liste des emprunts en cours
- SELECT *
FROM Emprunt
WHERE DateRet IS NULL;
- Opérateur IS NOT NULL
 - Vrai si la valeur est différente de NULL
 - Liste des emprunts terminés (livres restitués)
- SELECT *
FROM Emprunt
WHERE DateRet IS NOT NULL;

Sélection dans une liste de valeurs

- Opérateur IN
 - Sélectionne les valeurs dans la liste
 - Sélection des éditeurs Denoël et Eyrolles :

```
SELECT *
FROM Livre
WHERE Editeur IN 'Denoël', 'Eyrolles';
```

- Opérateur NOT IN
 - Sélectionne les valeurs qui ne sont pas dans la liste
 - Sélection des livres édités par d'autres éditeurs que Masson et Springer

```
SELECT *
FROM Livre
WHERE Editeur NOT IN 'Masson', 'Springer';
```

Ordre des n-uplets

- Ordonner la sortie selon les attributs spécifiés
ORDER BY att₁, ..., att_i [ASC|DESC]
- Affiche la sortie en triant les n-uplets selon les attributs att₁, ..., att_i dans l'ordre croissant (ASC), par défaut, ou décroissant (DESC)
- Exemple :

```
SELECT * FROM Abonné
WHERE DateAbo > '2007-10-01'
ORDER BY NomAbo, PrénomAbo;
```

NumAbo	NomAbo	PrénomAbo	DateAbo
4	Ducros	Marie	2020-07-04
3	Dupont	Charlie	2015-05-03
1	Dupont	Philippe	2008-06-18
2	Durand	Arthur	2009-01-02
5	Vernier	Alain	2021-09-15

Renommage de colonnes et valeurs distinctes

- Renommage des colonnes du résultat :

```
SELECT NomAbo AS "Nom de l'abonné",
      DateAbo AS Date
     FROM Abonné;
```

- Notez l'utilisation des guillemets ("") car nouvel identifiant (nouveau nom d'attribut) et pas simplement chaîne de caractères. Beaucoup de SGBD se servent également de l'accent grave (`) à la place des guillemets.

- Sélection de valeurs distinctes

```
SELECT DISTINCT NomAbo
     FROM Abonné
    WHERE NumAbo <= 4;
```

NomAbo
Dupont
Durand
Dupont



67 / 69