

Introduction
aux BDRJean-Paul
Comet,
Nadia
Abchiche-
MimouniLes concepts
générauxNotions
essentiellesAlgèbre
Relationnelle
+ SQL

Opérateurs unaires

Traduction en SQL

Jointures

Opérateurs
ensemblistesOpérateurs hors de
l'Alg.

Autres ordres SQL

le modèle
Entités-
Associations

NoSQL

Normalisation
d'une relation

Différents types de jointures :

3 `[NATURAL] JOIN ... [USING ...]` : ✕

en spécifiant éventuellement les listes de attributs qui doivent exister dans les 2 tables.

```
SELECT * FROM t1 JOIN t2 USING (c1,c2,c3)
```

La clause USING (columnlist) recense la liste des attributs qui doivent exister dans les 2 tables. Les 2 jointures suivantes donnent le même résultat :

```
a JOIN b USING (c1,c2,c3)
```

```
a JOIN b ON a.c1=b.c1 AND a.c2=b.c2 AND a.c3=b.c3
```

Le terme **NATURAL** permet d'éviter de préciser les colonnes concernées par la jointure.

Lorsqu'on a plusieurs tables à joindre :

```
SELECT * FROM a JOIN b ON a.x=b.y JOIN c ON b.z=c.t
```

...

Introduction
aux BDRJean-Paul
Comet,
Nadia
Abchiche-
MimouniLes concepts
générauxNotions
essentiellesAlgèbre
Relationnelle
+ SQL

Opérateurs unaires

Traduction en SQL

Jointures

Opérateurs
ensemblistesOpérateurs hors de
l'Alg.

Autres ordres SQL

le modèle

Entités-
Associations

NoSQL

Normalisation
d'une relation

Différents types de jointures :

- ④ **LEFT /RIGHT /FULL [OUTER] JOIN ...** : jointures externes.

Exemple : $\left\{ \begin{array}{l} \text{dept}(\text{numdept}, \text{nomdept}) \\ \text{employeur}(\text{numdept}, \text{nom}, \text{adresse} \dots) \end{array} \right.$

Certains n -uplets de dept ont un n^o de dépt qui ne correspond à aucune ligne de employeur.

Ces n -uplets sont *à l'extérieur de la jointure* entre les 2 tables.

Une **jointure externe** permet de rajouter ces n -uplets.

LEFT JOIN $\equiv \left\{ \begin{array}{l} \text{les } n\text{-uplets de la } \boxtimes \text{ naturelle} \\ + \text{ les } n\text{-uplets de la table de gauche} \end{array} \right.$

- même syntaxe que le **[INNER] JOIN**.

- on peut utiliser une jointure externe pour trouver les enregistrements dans une table qui n'ont pas de correspondances dans une autre.

RIGHT JOIN $\equiv \left\{ \begin{array}{l} \text{les } n\text{-uplets de la } \boxtimes \text{ naturelle} + \\ \text{les } n\text{-uplets de la table de droite} \end{array} \right.$

FULL JOIN $\equiv \left\{ \begin{array}{l} \text{les } n\text{-uplets de la } \boxtimes \text{ naturelle} + \\ \text{les } n\text{-uplets de la table de gauche} \\ \text{les } n\text{-uplets de la table de droite} \end{array} \right.$

(le **FULL** n'est pas implémenté dans MySQL, mais est calculable)

Introduction
aux BDR

Jean-Paul
Comet,
Nadia
Abchiche-
Mimouni

Les concepts
généraux

Notions
essentielles

Algèbre
Relationnelle
+ SQL

Opérateurs unaires

Traduction en SQL

Jointures

Opérateurs
ensemblistes

Opérateurs hors de
l'Alg.

Autres ordres SQL

le modèle
Entités-
Associations

NoSQL

Normalisation
d'une relation

Différents types de jointures :

- ④ **LEFT /RIGHT /FULL [OUTER] JOIN ...** : jointures externes.

Les jointures externes sont extrêmement pratiques pour rapatrier le maximum d'information disponible, même si des lignes de table ne sont pas renseignées entre les différentes tables jointes.

Exemple d'utilisation.

```
{ Disque(Id,Nom_Disque,Commentaires,IdPays)
  Pays(IdPays,NPays)
```

Pour certains Disque, le code du pays n'a pas été précisé.

⇒ Lister tous les disques avec le nom du pays s'il existe.

Jointure naturelle :

```
SELECT * FROM Disque JOIN Pays USING (IdPays);
```

Produit cartésien :

```
SELECT * FROM Disque, Pays
      WHERE Disque.IdPays=Pays.IdPays;
```

Jointures externes :

```
SELECT * FROM Disque LEFT JOIN Pays USING (IdPays);
```

Ainsi, en utilisant cette jointure, on précise que la table Disque et la table Pays sont liées par le champs IdPays.

Et voilà !

Introduction
aux BDR

Jean-Paul
Comet,
Nadia
Abchiche-
Mimouni

Les concepts
généraux

Notions
essentielles

Algèbre
Relationnelle
+ SQL

Opérateurs unaires

Traduction en SQL

Jointures

Opérateurs
ensemblistes

Opérateurs hors de
l'Alg.

Autres ordres SQL

le modèle
Entités-
Associations

NoSQL

Normalisation
d'une relation

Différents types de jointures :

- ④ **LEFT /RIGHT /FULL [OUTER] JOIN ...** : jointures externes.

Autre exemple d'utilisation de la jointure externe :

```
SELECT table1.* FROM table1  
LEFT JOIN table2 ON table1.id=table2.id  
WHERE table2.id IS NULL;
```

Cet exemple retourne

- toutes les lignes trouvées dans **table1** avec une valeur de **id** qui n'est pas présente dans **table2**
 - (autrement dit) toutes les lignes de **table1** sans correspondance dans la table **table2**.
- ⇒ Cela demande que **table2.id** soit déclaré **NOT NULL**

Introduction aux BDR

Jean-Paul
Comet,
Nadia
Abchiche-
Mimouni

Les concepts généraux

Notions essentielles

Algèbre Relationnelle + SQL

Opérateurs unaires

Traduction en SQL

Jointures

Opérateurs ensemblistes

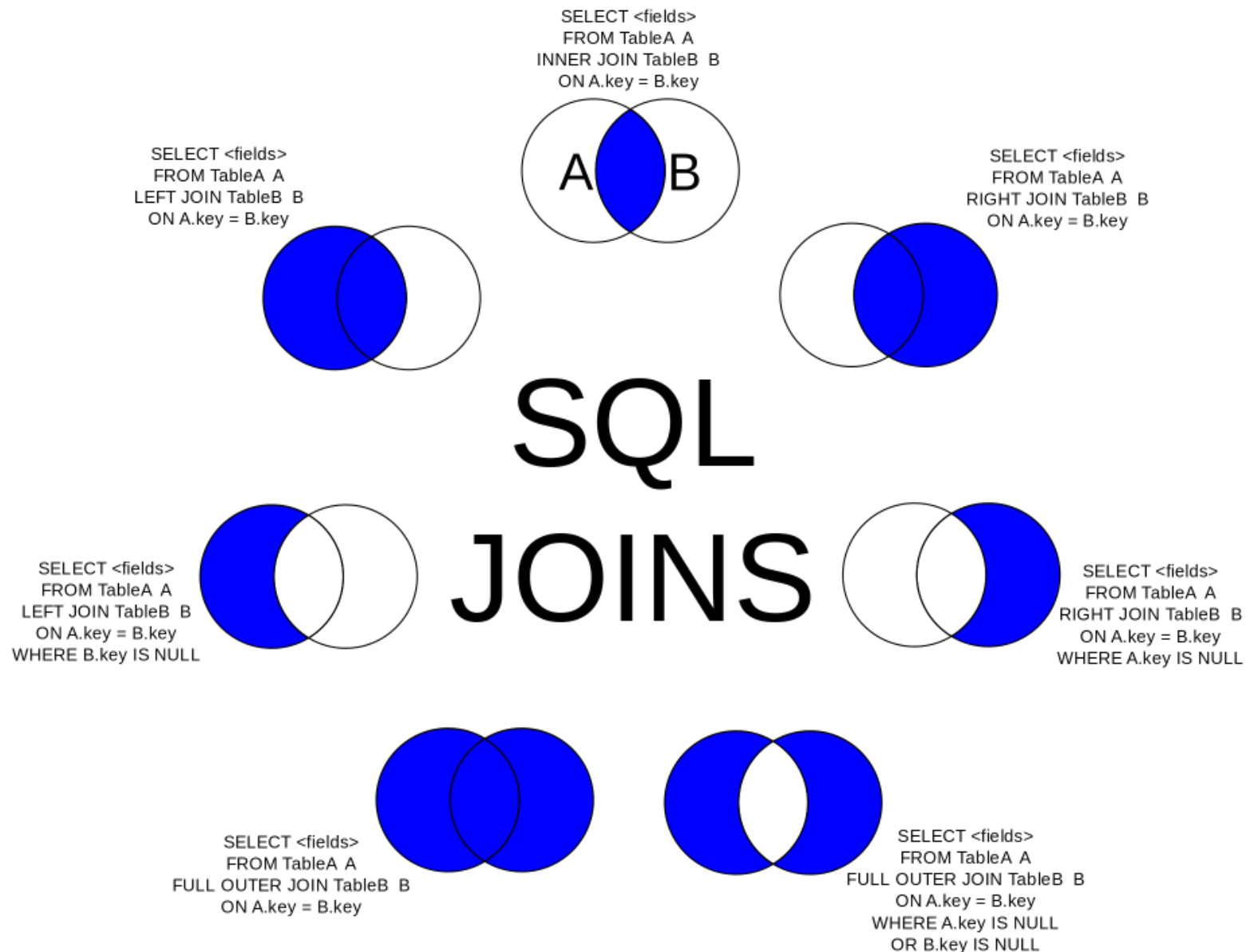
Opérateurs hors de l'Alg.

Autres ordres SQL

le modèle Entités-Associations

NoSQL

Normalisation d'une relation



This work is licensed under a Creative Commons Attribution 3.0 Unported License.
Author: <http://commons.wikimedia.org/wiki/User:Arbeck>

Introduction
aux BDRJean-Paul
Comet,
Nadia
Abchiche-
MimouniLes concepts
générauxNotions
essentiellesAlgèbre
Relationnelle
+ SQL

Opérateurs unaires

Traduction en SQL

Jointures

Opérateurs
ensemblistesOpérateurs hors de
l'Alg.

Autres ordres SQL

le modèle
Entités-
Associations

NoSQL

Normalisation
d'une relation

Soient deux relations de même schéma $R(A_1, A_2, \dots, A_n)$ et $S(A_1, A_2, \dots, A_n)$.

Union : $R \cup S$ est la relation temporaire de même schéma et de population égale à l'ensemble des n -uplets de R et de ceux de S (avec élimination des doublons éventuels).

Différence : $R \setminus S$ est la relation temporaire de même schéma et de population égale à l'ensemble des n -uplets de R moins ceux de S , c'est à dire à l'ensemble des n -uplets qui se trouvent dans R mais pas dans S .

Intersection : $R \cap S$ est la relation temporaire de même schéma et de population égale à l'ensemble des n -uplets qui sont présents à la fois dans R et dans S .

Introduction
aux BDRJean-Paul
Comet,
Nadia
Abchiche-
MimouniLes concepts
générauxNotions
essentiellesAlgèbre
Relationnelle
+ SQL

Opérateurs unaires

Traduction en SQL

Jointures

Opérateurs
ensemblistesOpérateurs hors de
l'Alg.

Autres ordres SQL

le modèle
Entités-
Associations

NoSQL

Normalisation
d'une relation

Exemple. Pour illustrer les opérateurs ensemblistes, nous reprenons la base de données relationnelle InstitutFormation dont le schéma est le suivant :

$$\mathcal{S} = \left\{ \begin{array}{l} \textit{personne}(\textit{numPers}, \textit{nom}, \textit{adresse}), \\ \textit{personnePrenom}(\textit{numPers}, \textit{numPrenom}, \textit{prenom}), \\ \textit{etudiant}(\textit{numPers}, \textit{numEtud}, \textit{dateN}), \\ \textit{etudiantEtude}(\textit{numEtud}, \textit{annee}, \textit{diplome}), \\ \textit{enseignant}(\textit{numPers}, \textit{tel}, \textit{status}, \textit{numEns}, \textit{banque}, \textit{agence}, \textit{compte}), \\ \textit{cours}(\textit{nomC}, \textit{cycle}, \textit{numEns}), \\ \textit{obtenu}(\textit{numEtud}, \textit{nomC}, \textit{note}), \\ \textit{inscrit}(\textit{numEtud}, \textit{nomC}), \\ \textit{prerequis}(\textit{nomC}, \textit{nomCpre}) \end{array} \right\}$$

Introduction
aux BDRJean-Paul
Comet,
Nadia
Abchiche-
MimouniLes concepts
générauxNotions
essentiellesAlgèbre
Relationnelle
+ SQL

Opérateurs unaires

Traduction en SQL

Jointures

Opérateurs
ensemblistesOpérateurs hors de
l'Alg.

Autres ordres SQL

le modèle
Entités-
Associations

NoSQL

Normalisation
d'une relation**Exemple.**

- Donnez la liste composée d'une part de la liste des numéros des enseignants en BD et d'autre part de la liste des numéros des étudiants inscrits en cours de BD.
- Donnez la liste des numéros d'étudiants qui ne sont pas inscrits au cours de BD.
- Donnez la liste des numéros des personnes qui sont à la fois enseignants et étudiants.

Introduction
aux BDRJean-Paul
Comet,
Nadia
Abchiche-
MimouniLes concepts
générauxNotions
essentiellesAlgèbre
Relationnelle
+ SQL

Opérateurs unaires

Traduction en SQL

Jointures

Opérateurs
ensemblistesOpérateurs hors de
l'Alg.

Autres ordres SQL

le modèle
Entités-
Associations

NoSQL

Normalisation
d'une relation**Exemple.**

- Donnez la liste composée d'une part de la liste des numéros des enseignants en BD et d'autre part de la liste des numéros des étudiants inscrits en cours de BD.

$$\pi[\text{numEns}] \sigma[\text{nomC} = \text{BD}] \text{Cours}$$

- Donnez la liste des numéros d'étudiants qui ne sont pas inscrits au cours de BD.
- Donnez la liste des numéros des personnes qui sont à la fois enseignants et étudiants.

Introduction
aux BDRJean-Paul
Comet,
Nadia
Abchiche-
MimouniLes concepts
générauxNotions
essentiellesAlgèbre
Relationnelle
+ SQL

Opérateurs unaires

Traduction en SQL

Jointures

Opérateurs
ensemblistesOpérateurs hors de
l'Alg.

Autres ordres SQL

le modèle
Entités-
Associations

NoSQL

Normalisation
d'une relation**Exemple.**

- Donnez la liste composée d'une part de la liste des numéros des enseignants en BD et d'autre part de la liste des numéros des étudiants inscrits en cours de BD.

$$\pi[\text{numEns}] \sigma[\text{nomC} = \text{BD}] \text{Cours}$$
$$\pi[\text{numEtud}] \sigma[\text{nomC} = \text{BD}] \text{inscrit}$$

- Donnez la liste des numéros d'étudiants qui ne sont pas inscrits au cours de BD.
- Donnez la liste des numéros des personnes qui sont à la fois enseignants et étudiants.

Introduction
aux BDRJean-Paul
Comet,
Nadia
Abchiche-
MimouniLes concepts
générauxNotions
essentiellesAlgèbre
Relationnelle
+ SQL

Opérateurs unaires

Traduction en SQL

Jointures

Opérateurs
ensemblistesOpérateurs hors de
l'Alg.

Autres ordres SQL

le modèle
Entités-
Associations

NoSQL

Normalisation
d'une relation**Exemple.**

- Donnez la liste composée d'une part de la liste des numéros des enseignants en BD et d'autre part de la liste des numéros des étudiants inscrits en cours de BD.

$$\begin{array}{c} \pi[\text{numEns}] \sigma[\text{nomC} = \text{BD}] \text{Cours} \\ \cup \\ \pi[\text{numEtud}] \sigma[\text{nomC} = \text{BD}] \text{inscrit} \end{array}$$

- Donnez la liste des numéros d'étudiants qui ne sont pas inscrits au cours de BD.
- Donnez la liste des numéros des personnes qui sont à la fois enseignants et étudiants.

Introduction
aux BDRJean-Paul
Comet,
Nadia
Abchiche-
MimouniLes concepts
générauxNotions
essentiellesAlgèbre
Relationnelle
+ SQL

Opérateurs unaires

Traduction en SQL

Jointures

Opérateurs
ensemblistesOpérateurs hors de
l'Alg.

Autres ordres SQL

le modèle
Entités-
Associations

NoSQL

Normalisation
d'une relation**Exemple.**

- Donnez la liste composée d'une part de la liste des numéros des enseignants en BD et d'autre part de la liste des numéros des étudiants inscrits en cours de BD.

$$H = \begin{cases} \alpha[\text{numEns} : \text{num}](\pi[\text{numEns}]\sigma[\text{nomC} = \text{BD}]\text{Cours}) \\ \cup \\ \alpha[\text{numEtud} : \text{num}](\pi[\text{numEtud}]\sigma[\text{nomC} = \text{BD}]\text{inscrit}) \end{cases}$$

- Donnez la liste des numéros d'étudiants qui ne sont pas inscrits au cours de BD.
- Donnez la liste des numéros des personnes qui sont à la fois enseignants et étudiants.

Introduction
aux BDRJean-Paul
Comet,
Nadia
Abchiche-
MimouniLes concepts
générauxNotions
essentiellesAlgèbre
Relationnelle
+ SQL

Opérateurs unaires

Traduction en SQL

Jointures

Opérateurs
ensemblistesOpérateurs hors de
l'Alg.

Autres ordres SQL

le modèle
Entités-
Associations

NoSQL

Normalisation
d'une relation**Exemple.**

- Donnez la liste composée d'une part de la liste des numéros des enseignants en BD et d'autre part de la liste des numéros des étudiants inscrits en cours de BD.

$$H = \begin{cases} \alpha[\text{numEns} : \text{num}](\pi[\text{numEns}]\sigma[\text{nomC} = \text{BD}]\text{Cours}) \\ \cup \\ \alpha[\text{numEtud} : \text{num}](\pi[\text{numEtud}]\sigma[\text{nomC} = \text{BD}]\text{inscrit}) \end{cases}$$

- Donnez la liste des numéros d'étudiants qui ne sont pas inscrits au cours de BD.

$$\pi[\text{numEtud}]\text{Etudiant} \setminus \pi[\text{numEtud}]\sigma[\text{nomC} = \text{BD}]\text{inscrit}$$

- Donnez la liste des numéros des personnes qui sont à la fois enseignants et étudiants.

Introduction
aux BDRJean-Paul
Comet,
Nadia
Abchiche-
MimouniLes concepts
générauxNotions
essentiellesAlgèbre
Relationnelle
+ SQL

Opérateurs unaires

Traduction en SQL

Jointures

Opérateurs
ensemblistesOpérateurs hors de
l'Alg.

Autres ordres SQL

le modèle
Entités-
Associations

NoSQL

Normalisation
d'une relation**Exemple.**

- Donnez la liste composée d'une part de la liste des numéros des enseignants en BD et d'autre part de la liste des numéros des étudiants inscrits en cours de BD.

$$H = \begin{cases} \alpha[\text{numEns} : \text{num}](\pi[\text{numEns}]\sigma[\text{nomC} = \text{BD}]\text{Cours}) \\ \cup \\ \alpha[\text{numEtud} : \text{num}](\pi[\text{numEtud}]\sigma[\text{nomC} = \text{BD}]\text{inscrit}) \end{cases}$$

- Donnez la liste des numéros d'étudiants qui ne sont pas inscrits au cours de BD.

$$\pi[\text{numEtud}]\text{Etudiant} \setminus \pi[\text{numEtud}]\sigma[\text{nomC} = \text{BD}]\text{inscrit}$$

- Donnez la liste des numéros des personnes qui sont à la fois enseignants et étudiants.

$$\pi[\text{numPers}]\text{Enseignant} \cap \pi[\text{numPers}]\text{Etudiant}$$

Introduction aux BDR

Jean-Paul
Comet,
Nadia
Abchiche-
Mimouni

Les concepts généraux

Notions essentielles

Algèbre Relationnelle + SQL

Opérateurs unaires

Traduction en SQL

Jointures

Opérateurs
ensemblistes

Opérateurs hors de
l'Alg.

Autres ordres SQL

le modèle Entités- Associations

NoSQL

Normalisation d'une relation

- syntaxe SQL : **UNION** , **INTERSECTION** , **EXCEPT** ou **MINUS** .
- Quels sont les noms et adresses des acteurs qui sont des producteurs ?
Solution en SQL :

Solution en algèbre relationnelle :

- Seul l'opérateur **UNION** est implémenté en MySQL
- Opérateur IN** . L'opérateur **IN** (resp. **NOT IN**) permet de simuler les autres opérateurs ensemblistes de l'algèbre relationnelle.

Introduction aux BDR

Jean-Paul
Comet,
Nadia
Abchiche-
Mimouni

Les concepts généraux

Notions essentielles

Algèbre Relationnelle + SQL

Opérateurs unaires

Traduction en SQL

Jointures

Opérateurs
ensemblistes

Opérateurs hors de
l'Alg.

Autres ordres SQL

le modèle Entités- Associations

NoSQL

Normalisation d'une relation

- syntaxe SQL : **UNION** , **INTERSECTION** , **EXCEPT** ou **MINUS** .
- Quels sont les noms et adresses des acteurs qui sont des producteurs ?

Solution en SQL :

```
(SELECT nom,adresse FROM Acteur)
```

```
(SELECT nom,adresse FROM Producteur)
```

Solution en algèbre relationnelle :

- Seul l'opérateur **UNION** est implémenté en MySQL
- Opérateur IN** . L'opérateur **IN** (resp. **NOT IN**) permet de simuler les autres opérateurs ensemblistes de l'algèbre relationnelle.

Introduction aux BDR

Jean-Paul
Comet,
Nadia
Abchiche-
Mimouni

Les concepts généraux

Notions essentielles

Algèbre Relationnelle + SQL

Opérateurs unaires

Traduction en SQL

Jointures

Opérateurs
ensemblistes

Opérateurs hors de
l'Alg.

Autres ordres SQL

le modèle Entités- Associations

NoSQL

Normalisation d'une relation

- syntaxe SQL : **UNION** , **INTERSECTION** , **EXCEPT** ou **MINUS** .
- Quels sont les noms et adresses des acteurs qui sont des producteurs ?

Solution en SQL :

```
(SELECT nom,adresse FROM Acteur)
```

```
INTERSECTION
```

```
(SELECT nom,adresse FROM Producteur)
```

Solution en algèbre relationnelle :

- Seul l'opérateur **UNION** est implémenté en MySQL
- Opérateur IN** . L'opérateur **IN** (resp. **NOT IN**) permet de simuler les autres opérateurs ensemblistes de l'algèbre relationnelle.

Introduction aux BDR

Jean-Paul
Comet,
Nadia
Abchiche-
Mimouni

Les concepts généraux

Notions essentielles

Algèbre Relationnelle + SQL

Opérateurs unaires

Traduction en SQL

Jointures

Opérateurs ensemblistes

Opérateurs hors de l'Alg.

Autres ordres SQL

le modèle Entités- Associations

NoSQL

Normalisation d'une relation

- syntaxe SQL : **UNION** , **INTERSECTION** , **EXCEPT** ou **MINUS** .
- Quels sont les noms et adresses des acteurs qui sont des producteurs ?

Solution en SQL :

```
(SELECT nom,adresse FROM Acteur)
INTERSECTION
(SELECT nom,adresse FROM Producteur)
```

Solution en algèbre relationnelle :

$$\pi \left[\begin{matrix} \textit{nom}, \\ \textit{adresse} \end{matrix} \right] (\textit{Acteur}) \cap \pi \left[\begin{matrix} \textit{nom}, \\ \textit{adresse} \end{matrix} \right] (\textit{Producteur})$$

- Seul l'opérateur **UNION** est implémenté en MySQL
- Opérateur IN** . L'opérateur **IN** (resp. **NOT IN**) permet de simuler les autres opérateurs ensemblistes de l'algèbre relationnelle.

Introduction aux BDR

Jean-Paul
Comet,
Nadia
Abchiche-
Mimouni

Les concepts généraux

Notions essentielles

Algèbre Relationnelle + SQL

Opérateurs unaires

Traduction en SQL

Jointures

Opérateurs ensemblistes

Opérateurs hors de l'Alg.

Autres ordres SQL

le modèle Entités- Associations

NoSQL

Normalisation d'une relation

- syntaxe SQL : **UNION** , **INTERSECTION** , **EXCEPT** ou **MINUS** .
- Quels sont les noms et adresses des acteurs qui **ne** sont **pas** des producteurs ?

Solution en SQL :

(SELECT nom,adresse FROM Acteur)

(SELECT nom,adresse FROM Producteur)

Solution en algèbre relationnelle :

$\pi \left[\begin{matrix} \textit{nom}, \\ \textit{adresse} \end{matrix} \right] (\textit{Acteur})$

$\pi \left[\begin{matrix} \textit{nom}, \\ \textit{adresse} \end{matrix} \right] (\textit{Producteur})$

- Seul l'opérateur **UNION** est implémenté en MySQL
- Opérateur IN** . L'opérateur **IN** (resp. **NOT IN**) permet de simuler les autres opérateurs ensemblistes de l'algèbre relationnelle.

Introduction aux BDR

Jean-Paul
Comet,
Nadia
Abchiche-
Mimouni

Les concepts généraux

Notions essentielles

Algèbre Relationnelle + SQL

Opérateurs unaires

Traduction en SQL

Jointures

Opérateurs ensemblistes

Opérateurs hors de l'Alg.

Autres ordres SQL

le modèle Entités- Associations

NoSQL

Normalisation d'une relation

- syntaxe SQL : **UNION** , **INTERSECTION** , **EXCEPT** ou **MINUS** .
- Quels sont les noms et adresses des acteurs qui **ne** sont **pas** des producteurs ?

Solution en SQL :

```
(SELECT nom,adresse FROM Acteur)
EXCEPT
(SELECT nom,adresse FROM Producteur)
```

Solution en algèbre relationnelle :

$$\pi \left[\begin{array}{c} \textit{nom}, \\ \textit{adresse} \end{array} \right] (\textit{Acteur}) \quad \setminus \quad \pi \left[\begin{array}{c} \textit{nom}, \\ \textit{adresse} \end{array} \right] (\textit{Producteur})$$

- Seul l'opérateur **UNION** est implémenté en MySQL
- Opérateur IN** . L'opérateur **IN** (resp. **NOT IN**) permet de simuler les autres opérateurs ensemblistes de l'algèbre relationnelle.

Introduction aux BDR

Jean-Paul
Comet,
Nadia
Abchiche-
Mimouni

Les concepts généraux

Notions essentielles

Algèbre Relationnelle + SQL

Opérateurs unaires

Traduction en SQL

Jointures

Opérateurs ensemblistes

Opérateurs hors de l'Alg.

Autres ordres SQL

le modèle Entités- Associations

NoSQL

Normalisation d'une relation

$$\mathcal{S} = \left\{ \begin{array}{l} \text{Film}(\text{Titre}, \text{date}, \text{longueur}, \text{couleur}, \text{nomStudio}, \text{IdProd}), \\ \text{Joue}(\text{TitreFilm}, \text{FilmDate}, \text{nomActeur}, \text{Paye}), \\ \text{Acteur}(\text{nom}, \text{adresse}, \text{genre}, \text{dateNaissance}), \\ \text{Studio}(\text{Nom}, \text{Adresse}), \text{Producteur}(\text{Nom}, \text{adresse}, \text{Id}) \end{array} \right\}$$

- ① Quels sont les noms des acteurs qui ne sont pas producteurs ?
- ② Quels sont les noms des producteurs de films dans lesquels joue Matt Damon qui sont sortis avant 2006 ?

Introduction aux BDR

Jean-Paul
Comet,
Nadia
Abchiche-
Mimouni

Les concepts généraux

Notions essentielles

Algèbre Relationnelle + SQL

Opérateurs unaires

Traduction en SQL

Jointures

Opérateurs ensemblistes

Opérateurs hors de l'Alg.

Autres ordres SQL

le modèle Entités- Associations

NoSQL

Normalisation d'une relation

$$\mathcal{S} = \left\{ \begin{array}{l} \text{Film}(\text{Titre}, \text{date}, \text{longueur}, \text{couleur}, \text{nomStudio}, \text{IdProd}), \\ \text{Joue}(\text{TitreFilm}, \text{FilmDate}, \text{nomActeur}, \text{Paye}), \\ \text{Acteur}(\text{nom}, \text{adresse}, \text{genre}, \text{dateNaissance}), \\ \text{Studio}(\text{Nom}, \text{Adresse}), \text{Producteur}(\text{Nom}, \text{adresse}, \text{Id}) \end{array} \right\}$$

- 1 Quels sont les noms des acteurs qui ne sont pas producteurs ?

```
SELECT nom FROM Acteur
```

```
WHERE nom NOT IN (SELECT Nom FROM Producteur)
```

- 2 Quels sont les noms des producteurs de films dans lesquels joue Matt Damon qui sont sortis avant 2006 ?

Introduction
aux BDRJean-Paul
Comet,
Nadia
Abchiche-
MimouniLes concepts
générauxNotions
essentiellesAlgèbre
Relationnelle
+ SQL

Opérateurs unaires

Traduction en SQL

Jointures

Opérateurs
ensemblistesOpérateurs hors de
l'Alg.

Autres ordres SQL

le modèle
Entités-
Associations

NoSQL

Normalisation
d'une relation

$$\mathcal{S} = \left\{ \begin{array}{l} \text{Film}(\text{Titre}, \text{date}, \text{longueur}, \text{couleur}, \text{nomStudio}, \text{IdProd}), \\ \text{Joue}(\text{TitreFilm}, \text{FilmDate}, \text{nomActeur}, \text{Paye}), \\ \text{Acteur}(\text{nom}, \text{adresse}, \text{genre}, \text{dateNaissance}), \\ \text{Studio}(\text{Nom}, \text{Adresse}), \text{Producteur}(\text{Nom}, \text{adresse}, \text{Id}) \end{array} \right\}$$

- ① Quels sont les noms des acteurs qui ne sont pas producteurs ?

```
SELECT nom FROM Acteur
```

```
WHERE nom NOT IN (SELECT Nom FROM Producteur)
```

$\pi[\text{nom}] \text{Acteur} \setminus \alpha[\text{Nom} : \text{nom}] \pi[\text{Nom}] \text{Producteur}$

- ② Quels sont les noms des producteurs de films dans lesquels joue Matt Damon qui sont sortis avant 2006 ?

Introduction
aux BDRJean-Paul
Comet,
Nadia
Abchiche-
MimouniLes concepts
générauxNotions
essentiellesAlgèbre
Relationnelle
+ SQL

Opérateurs unaires

Traduction en SQL

Jointures

Opérateurs
ensemblistesOpérateurs hors de
l'Alg.

Autres ordres SQL

le modèle
Entités-
Associations

NoSQL

Normalisation
d'une relation

$$\mathcal{S} = \left\{ \begin{array}{l} \text{Film}(\text{Titre}, \text{date}, \text{longueur}, \text{couleur}, \text{nomStudio}, \text{IdProd}), \\ \text{Joue}(\text{TitreFilm}, \text{FilmDate}, \text{nomActeur}, \text{Paye}), \\ \text{Acteur}(\text{nom}, \text{adresse}, \text{genre}, \text{dateNaissance}), \\ \text{Studio}(\text{Nom}, \text{Adresse}), \text{Producteur}(\text{Nom}, \text{adresse}, \text{Id}) \end{array} \right\}$$

- ① Quels sont les noms des acteurs qui ne sont pas producteurs ?

```
SELECT nom FROM Acteur
```

```
WHERE nom NOT IN (SELECT Nom FROM Producteur)
```

$\pi[\text{nom}] \text{Acteur} \setminus \alpha[\text{Nom} : \text{nom}] \pi[\text{Nom}] \text{Producteur}$

- ② Quels sont les noms des producteurs de films dans lesquels joue Matt Damon qui sont sortis avant 2006 ?

```
(SELECT TitreFilm, FilmDate FROM Joue  
WHERE nomActeur = 'Matt Damon' and FilmDate <= 2006)
```


Introduction aux BDR

Jean-Paul
Comet,
Nadia
Abchiche-
Mimouni

Les concepts généraux

Notions essentielles

Algèbre Relationnelle + SQL

Opérateurs unaires

Traduction en SQL

Jointures

Opérateurs ensemblistes

Opérateurs hors de l'Alg.

Autres ordres SQL

le modèle Entités- Associations

NoSQL

Normalisation d'une relation

$$\mathcal{S} = \left\{ \begin{array}{l} \text{Film}(\text{Titre}, \text{date}, \text{longueur}, \text{couleur}, \text{nomStudio}, \text{IdProd}), \\ \text{Joue}(\text{TitreFilm}, \text{FilmDate}, \text{nomActeur}, \text{Paye}), \\ \text{Acteur}(\text{nom}, \text{adresse}, \text{genre}, \text{dateNaissance}), \\ \text{Studio}(\text{Nom}, \text{Adresse}), \text{Producteur}(\text{Nom}, \text{adresse}, \text{Id}) \end{array} \right\}$$

- ① Quels sont les noms des acteurs qui ne sont pas producteurs ?

```
SELECT nom FROM Acteur
```

```
WHERE nom NOT IN (SELECT Nom FROM Producteur)
```

$\pi[\text{nom}] \text{Acteur} \setminus \alpha[\text{Nom} : \text{nom}] \pi[\text{Nom}] \text{Producteur}$

- ② Quels sont les noms des producteurs de films dans lesquels joue Matt Damon qui sont sortis avant 2006 ?

```
(SELECT IdProd FROM Film WHERE
(Titre,date) IN (SELECT TitreFilm,FilmDate FROM Joue
WHERE nomActeur = 'Matt Damon' and FilmDate<=2006) )
```

Introduction aux BDR

Jean-Paul
Comet,
Nadia
Abchiche-
Mimouni

Les concepts généraux

Notions essentielles

Algèbre Relationnelle + SQL

Opérateurs unaires

Traduction en SQL

Jointures

Opérateurs ensemblistes

Opérateurs hors de l'Alg.

Autres ordres SQL

le modèle Entités- Associations

NoSQL

Normalisation d'une relation

$$\mathcal{S} = \left\{ \begin{array}{l} \text{Film}(\text{Titre}, \text{date}, \text{longueur}, \text{couleur}, \text{nomStudio}, \text{IdProd}), \\ \text{Joue}(\text{TitreFilm}, \text{FilmDate}, \text{nomActeur}, \text{Paye}), \\ \text{Acteur}(\text{nom}, \text{adresse}, \text{genre}, \text{dateNaissance}), \\ \text{Studio}(\text{Nom}, \text{Adresse}), \text{Producteur}(\text{Nom}, \text{adresse}, \text{Id}) \end{array} \right\}$$

- 1 Quels sont les noms des acteurs qui ne sont pas producteurs ?

```
SELECT nom FROM Acteur
```

```
WHERE nom NOT IN (SELECT Nom FROM Producteur)
```

$\pi[\text{nom}] \text{Acteur} \setminus \alpha[\text{Nom} : \text{nom}] \pi[\text{Nom}] \text{Producteur}$

- 2 Quels sont les noms des producteurs de films dans lesquels joue Matt Damon qui sont sortis avant 2006 ?

```
SELECT Nom FROM Producteur WHERE
```

```
Id IN (SELECT IdProd FROM Film WHERE
```

```
(Titre,date) IN (SELECT TitreFilm,FilmDate FROM Joue
```

```
WHERE nomActeur = 'Matt Damon' and FilmDate<=2006) )
```

Introduction
aux BDRJean-Paul
Comet,
Nadia
Abchiche-
MimouniLes concepts
générauxNotions
essentiellesAlgèbre
Relationnelle
+ SQLOpérateurs unaires
Traduction en SQL

Jointures

Opérateurs
ensemblistesOpérateurs hors de
l'Alg.

Autres ordres SQL

le modèle
Entités-
Associations

NoSQL

Normalisation
d'une relation

$$\mathcal{S} = \left\{ \begin{array}{l} \text{Film}(\text{Titre}, \text{date}, \text{longueur}, \text{couleur}, \text{nomStudio}, \text{IdProd}), \\ \text{Joue}(\text{TitreFilm}, \text{FilmDate}, \text{nomActeur}, \text{Paye}), \\ \text{Acteur}(\text{nom}, \text{adresse}, \text{genre}, \text{dateNaissance}), \\ \text{Studio}(\text{Nom}, \text{Adresse}), \text{Producteur}(\text{Nom}, \text{adresse}, \text{Id}) \end{array} \right\}$$

- ① Quels sont les noms des acteurs qui ne sont pas producteurs ?

```
SELECT nom FROM Acteur
```

```
WHERE nom NOT IN (SELECT Nom FROM Producteur)
```

$$\pi[\text{nom}] \text{Acteur} \setminus \alpha[\text{Nom} : \text{nom}] \pi[\text{Nom}] \text{Producteur}$$

- ② Quels sont les noms des producteurs de films dans lesquels joue Matt Damon qui sont sortis avant 2006 ?

```
SELECT Nom FROM Producteur WHERE
```

```
Id IN (SELECT IdProd FROM Film WHERE
```

```
(Titre,date) IN (SELECT TitreFilm,FilmDate FROM Joue
```

```
WHERE nomActeur = 'Matt Damon' and FilmDate<=2006) )
```

$$\pi[\text{Nom}] \left(\text{Producteur} \bowtie_{\left[\begin{array}{l} \text{IdProd} \\ = \text{Id} \end{array} \right]} \text{Film} \bowtie_{\left[\begin{array}{l} \text{Titre} = \text{TitreFilm} \wedge \\ \text{Date} = \text{FilmDate} \end{array} \right]} \sigma_{\left[\begin{array}{l} \text{DateFilm} \leq 2006 \\ \wedge (\text{nomActeur} = \\ \text{"MattDamon"}) \end{array} \right]} \text{Joue} \right)$$

Introduction aux BDR

Jean-Paul
Comet,
Nadia
Abchiche-
Mimouni

Les concepts généraux

Notions essentielles

Algèbre Relationnelle + SQL

Opérateurs unaires

Traduction en SQL

Jointures

Opérateurs ensemblistes

Opérateurs hors de l'Alg.

Autres ordres SQL

le modèle Entités- Associations

NoSQL

Normalisation d'une relation

$$S = \left\{ \begin{array}{l} \textit{personne}(\textit{numPers}, \textit{nom}, \textit{adresse}), \\ \textit{personnePrenom}(\textit{numPers}, \textit{numPrenom}, \textit{prenom}), \\ \textit{etudiant}(\textit{numPers}, \textit{numEtud}, \textit{dateN}), \\ \textit{etudiantEtude}(\textit{numEtud}, \textit{annee}, \textit{diplome}), \\ \textit{enseignant}(\textit{numPers}, \textit{tel}, \textit{status}, \textit{numEns}, \textit{banque}, \textit{agence}, \textit{compte}), \\ \textit{cours}(\textit{nomC}, \textit{cycle}, \textit{numEns}), \\ \textit{obtenu}(\textit{numEtud}, \textit{nomC}, \textit{note}), \\ \textit{inscrit}(\textit{numEtud}, \textit{nomC}), \\ \textit{prerequis}(\textit{nomC}, \textit{nomCpre}) \end{array} \right\}$$

- 1 Quels sont les n^o des personnes qui sont enseignants mais pas étudiants ?

Introduction
aux BDRJean-Paul
Comet,
Nadia
Abchiche-
MimouniLes concepts
générauxNotions
essentiellesAlgèbre
Relationnelle
+ SQL

Opérateurs unaires

Traduction en SQL

Jointures

Opérateurs
ensemblistesOpérateurs hors de
l'Alg.

Autres ordres SQL

le modèle
Entités-
Associations

NoSQL

Normalisation
d'une relation

$$S = \left\{ \begin{array}{l} \textit{personne}(\textit{numPers}, \textit{nom}, \textit{adresse}), \\ \textit{personnePrenom}(\textit{numPers}, \textit{numPrenom}, \textit{prenom}), \\ \textit{etudiant}(\textit{numPers}, \textit{numEtud}, \textit{dateN}), \\ \textit{etudiantEtude}(\textit{numEtud}, \textit{annee}, \textit{diplome}), \\ \textit{enseignant}(\textit{numPers}, \textit{tel}, \textit{status}, \textit{numEns}, \textit{banque}, \textit{agence}, \textit{compte}), \\ \textit{cours}(\textit{nomC}, \textit{cycle}, \textit{numEns}), \\ \textit{obtenu}(\textit{numEtud}, \textit{nomC}, \textit{note}), \\ \textit{inscrit}(\textit{numEtud}, \textit{nomC}), \\ \textit{prerequis}(\textit{nomC}, \textit{nomCpre}) \end{array} \right\}$$

- 1 Quels sont les n^o des personnes qui sont enseignants mais pas étudiants ?

```
SELECT numPers FROM enseignant
WHERE numPers NOT IN
  (SELECT numPers FROM etudiant)
```

Introduction aux BDR

Jean-Paul
Comet,
Nadia
Abchiche-
Mimouni

Les concepts généraux

Notions essentielles

Algèbre Relationnelle + SQL

Opérateurs unaires

Traduction en SQL

Jointures

Opérateurs ensemblistes

Opérateurs hors de l'Alg.

Autres ordres SQL

le modèle Entités- Associations

NoSQL

Normalisation d'une relation

$$S = \left\{ \begin{array}{l} \textit{personne}(\textit{numPers}, \textit{nom}, \textit{adresse}), \\ \textit{personnePrenom}(\textit{numPers}, \textit{numPrenom}, \textit{prenom}), \\ \textit{etudiant}(\textit{numPers}, \textit{numEtud}, \textit{dateN}), \\ \textit{etudiantEtude}(\textit{numEtud}, \textit{annee}, \textit{diplome}), \\ \textit{enseignant}(\textit{numPers}, \textit{tel}, \textit{status}, \textit{numEns}, \textit{banque}, \textit{agence}, \textit{compte}), \\ \textit{cours}(\textit{nomC}, \textit{cycle}, \textit{numEns}), \\ \textit{obtenu}(\textit{numEtud}, \textit{nomC}, \textit{note}), \\ \textit{inscrit}(\textit{numEtud}, \textit{nomC}), \\ \textit{prerequis}(\textit{nomC}, \textit{nomCpre}) \end{array} \right\}$$

- 1 Quels sont les n^o des personnes qui sont enseignants mais pas étudiants ?

```
SELECT numPers FROM enseignant
WHERE numPers NOT IN
  (SELECT numPers FROM etudiant)
 $\pi[\textit{numPers}]\textit{enseignant} \setminus \pi[\textit{numPers}]\textit{etudiant}$ 
```

Introduction aux BDR

Jean-Paul
Comet,
Nadia
Abchiche-
Mimouni

Les concepts généraux

Notions essentielles

Algèbre Relationnelle + SQL

Opérateurs unaires

Traduction en SQL

Jointures

Opérateurs
ensemblistes

Opérateurs hors de
l'Alg.

Autres ordres SQL

le modèle Entités- Associations

NoSQL

Normalisation d'une relation

$$S = \left\{ \begin{array}{l} \textit{personne}(\textit{numPers}, \textit{nom}, \textit{adresse}), \\ \textit{personnePrenom}(\textit{numPers}, \textit{numPrenom}, \textit{prenom}), \\ \textit{etudiant}(\textit{numPers}, \textit{numEtud}, \textit{dateN}), \\ \textit{etudiantEtude}(\textit{numEtud}, \textit{annee}, \textit{diplome}), \\ \textit{enseignant}(\textit{numPers}, \textit{tel}, \textit{status}, \textit{numEns}, \textit{banque}, \textit{agence}, \textit{compte}), \\ \textit{cours}(\textit{nomC}, \textit{cycle}, \textit{numEns}), \\ \textit{obtenu}(\textit{numEtud}, \textit{nomC}, \textit{note}), \\ \textit{inscrit}(\textit{numEtud}, \textit{nomC}), \\ \textit{prerequis}(\textit{nomC}, \textit{nomCpre}) \end{array} \right\}$$

- 2 Quels sont les n^o de téléphone des enseignants qui enseignent dans un cours validé par un étudiant dont le nom est Gautier ?

Introduction aux BDR

Jean-Paul
Comet,
Nadia
Abchiche-
Mimouni

Les concepts généraux

Notions essentielles

Algèbre Relationnelle + SQL

Opérateurs unaires

Traduction en SQL

Jointures

Opérateurs
ensemblistes

Opérateurs hors de
l'Alg.

Autres ordres SQL

le modèle Entités- Associations

NoSQL

Normalisation d'une relation

$$S = \left\{ \begin{array}{l} \text{personne}(\text{numPers}, \text{nom}, \text{adresse}), \\ \text{personnePrenom}(\text{numPers}, \text{numPrenom}, \text{prenom}), \\ \text{etudiant}(\text{numPers}, \text{numEtud}, \text{dateN}), \\ \text{etudiantEtude}(\text{numEtud}, \text{annee}, \text{diplome}), \\ \text{enseignant}(\text{numPers}, \text{tel}, \text{status}, \text{numEns}, \text{banque}, \text{agence}, \text{compte}), \\ \text{cours}(\text{nomC}, \text{cycle}, \text{numEns}), \\ \text{obtenu}(\text{numEtud}, \text{nomC}, \text{note}), \\ \text{inscrit}(\text{numEtud}, \text{nomC}), \\ \text{prerequis}(\text{nomC}, \text{nomCpre}) \end{array} \right\}$$

- 2 Quels sont les n^o de téléphone des enseignants qui enseignent dans un cours validé par un étudiant dont le nom est Gautier ?

`(SELECT numPers FROM personne WHERE nom = 'Gautier')`

Introduction
aux BDRJean-Paul
Comet,
Nadia
Abchiche-
MimouniLes concepts
générauxNotions
essentiellesAlgèbre
Relationnelle
+ SQL

Opérateurs unaires

Traduction en SQL

Jointures

Opérateurs
ensemblistesOpérateurs hors de
l'Alg.

Autres ordres SQL

le modèle
Entités-
Associations

NoSQL

Normalisation
d'une relation

$$S = \left\{ \begin{array}{l} \text{personne}(\text{numPers}, \text{nom}, \text{adresse}), \\ \text{personnePrenom}(\text{numPers}, \text{numPrenom}, \text{prenom}), \\ \text{etudiant}(\text{numPers}, \text{numEtud}, \text{dateN}), \\ \text{etudiantEtude}(\text{numEtud}, \text{annee}, \text{diplome}), \\ \text{enseignant}(\text{numPers}, \text{tel}, \text{status}, \text{numEns}, \text{banque}, \text{agence}, \text{compte}), \\ \text{cours}(\text{nomC}, \text{cycle}, \text{numEns}), \\ \text{obtenu}(\text{numEtud}, \text{nomC}, \text{note}), \\ \text{inscrit}(\text{numEtud}, \text{nomC}), \\ \text{prerequis}(\text{nomC}, \text{nomCpre}) \end{array} \right\}$$

- 2 Quels sont les n^o de téléphone des enseignants qui enseignent dans un cours validé par un étudiant dont le nom est Gautier ?

```
(SELECT numEtud FROM Etudiants WHERE numPers IN  
  (SELECT numPers FROM personne WHERE nom = 'Gautier')  
)
```

Introduction aux BDR

Jean-Paul
Comet,
Nadia
Abchiche-
Mimouni

Les concepts généraux

Notions essentielles

Algèbre Relationnelle + SQL

Opérateurs unaires

Traduction en SQL

Jointures

Opérateurs
ensemblistes

Opérateurs hors de
l'Alg.

Autres ordres SQL

le modèle Entités- Associations

NoSQL

Normalisation d'une relation

$$S = \left\{ \begin{array}{l} \text{personne}(\text{numPers}, \text{nom}, \text{adresse}), \\ \text{personnePrenom}(\text{numPers}, \text{numPrenom}, \text{prenom}), \\ \text{etudiant}(\text{numPers}, \text{numEtud}, \text{dateN}), \\ \text{etudiantEtude}(\text{numEtud}, \text{annee}, \text{diplome}), \\ \text{enseignant}(\text{numPers}, \text{tel}, \text{status}, \text{numEns}, \text{banque}, \text{agence}, \text{compte}), \\ \text{cours}(\text{nomC}, \text{cycle}, \text{numEns}), \\ \text{obtenu}(\text{numEtud}, \text{nomC}, \text{note}), \\ \text{inscrit}(\text{numEtud}, \text{nomC}), \\ \text{prerequis}(\text{nomC}, \text{nomCpre}) \end{array} \right\}$$

- 2 Quels sont les n^o de téléphone des enseignants qui enseignent dans un cours validé par un étudiant dont le nom est Gautier ?

```
(SELECT nomC FROM obtenu WHERE numEtud IN
  (SELECT numEtud FROM Etudiants WHERE numPers IN
    (SELECT numPers FROM personne WHERE nom = 'Gautier')
  )
)
```

Introduction aux BDR

Jean-Paul
Comet,
Nadia
Abchiche-
Mimouni

Les concepts généraux

Notions essentielles

Algèbre Relationnelle + SQL

Opérateurs unaires

Traduction en SQL

Jointures

Opérateurs
ensemblistes

Opérateurs hors de
l'Alg.

Autres ordres SQL

le modèle Entités- Associations

NoSQL

Normalisation d'une relation

$$S = \left\{ \begin{array}{l} \text{personne}(\text{numPers}, \text{nom}, \text{adresse}), \\ \text{personnePrenom}(\text{numPers}, \text{numPrenom}, \text{prenom}), \\ \text{etudiant}(\text{numPers}, \text{numEtud}, \text{dateN}), \\ \text{etudiantEtude}(\text{numEtud}, \text{annee}, \text{diplome}), \\ \text{enseignant}(\text{numPers}, \text{tel}, \text{status}, \text{numEns}, \text{banque}, \text{agence}, \text{compte}), \\ \text{cours}(\text{nomC}, \text{cycle}, \text{numEns}), \\ \text{obtenu}(\text{numEtud}, \text{nomC}, \text{note}), \\ \text{inscrit}(\text{numEtud}, \text{nomC}), \\ \text{prerequis}(\text{nomC}, \text{nomCpre}) \end{array} \right\}$$

- 2 Quels sont les n^o de téléphone des enseignants qui enseignent dans un cours validé par un étudiant dont le nom est Gautier ?

```
(SELECT numEns FROM cours WHERE nomC IN
  (SELECT nomC FROM obtenu WHERE numEtud IN
    (SELECT numEtud FROM Etudiants WHERE numPers IN
      (SELECT numPers FROM personne WHERE nom = 'Gautier')
    )
  )
)
```

Introduction aux BDR

Jean-Paul
Comet,
Nadia
Abchiche-
Mimouni

Les concepts généraux

Notions essentielles

Algèbre Relationnelle + SQL

Opérateurs unaires

Traduction en SQL

Jointures

Opérateurs
ensemblistes

Opérateurs hors de
l'Alg.

Autres ordres SQL

le modèle Entités- Associations

NoSQL

Normalisation d'une relation

$$S = \left\{ \begin{array}{l} \text{personne}(\text{numPers}, \text{nom}, \text{adresse}), \\ \text{personnePrenom}(\text{numPers}, \text{numPrenom}, \text{prenom}), \\ \text{etudiant}(\text{numPers}, \text{numEtud}, \text{dateN}), \\ \text{etudiantEtude}(\text{numEtud}, \text{annee}, \text{diplome}), \\ \text{enseignant}(\text{numPers}, \text{tel}, \text{status}, \text{numEns}, \text{banque}, \text{agence}, \text{compte}), \\ \text{cours}(\text{nomC}, \text{cycle}, \text{numEns}), \\ \text{obtenu}(\text{numEtud}, \text{nomC}, \text{note}), \\ \text{inscrit}(\text{numEtud}, \text{nomC}), \\ \text{prerequis}(\text{nomC}, \text{nomCpre}) \end{array} \right\}$$

- 2 Quels sont les n^o de téléphone des enseignants qui enseignent dans un cours validé par un étudiant dont le nom est Gautier ?

```
SELECT tel FROM enseignant WHERE numEns IN
  (SELECT numEns FROM cours WHERE nomC IN
    (SELECT nomC FROM obtenu WHERE numEtud IN
      (SELECT numEtud FROM Etudiants WHERE numPers IN
        (SELECT numPers FROM personne WHERE nom = 'Gautier')
      )
    )
  )
)
```

Introduction aux BDR

Jean-Paul
Comet,
Nadia
Abchiche-
Mimouni

Les concepts généraux

Notions essentielles

Algèbre Relationnelle + SQL

Opérateurs unaires

Traduction en SQL

Jointures

Opérateurs
ensemblistes

Opérateurs hors de
l'Alg.

Autres ordres SQL

le modèle Entités- Associations

NoSQL

Normalisation d'une relation

- Les variables de n -uplets vont être utiles lorsque l'on cherche à raisonner sur plusieurs n -uplets d'une même relation et à les comparer.
- L'idée est de prendre deux n -uplets dans une relation (autrement dit un n -uplet de $R \times R$) et de sélectionner les couples de n -uplets satisfaisant certaines contraintes.

Exemple : Quels acteurs ont la même adresse ?

```
SELECT Star1.Nom, Star2.Nom FROM Acteur Star1, Acteur Star2
```

Ici Star1 et Star2 sont des variables pour les n -uplets. On aurait pu nommé ces variables différemment par exemple t1 et t2.

Introduction aux BDR

Jean-Paul
Comet,
Nadia
Abchiche-
Mimouni

Les concepts généraux

Notions essentielles

Algèbre Relationnelle + SQL

Opérateurs unaires

Traduction en SQL

Jointures

Opérateurs
ensemblistes

Opérateurs hors de
l'Alg.

Autres ordres SQL

le modèle Entités- Associations

NoSQL

Normalisation d'une relation

- Les variables de n -uplets vont être utiles lorsque l'on cherche à raisonner sur plusieurs n -uplets d'une même relation et à les comparer.
- L'idée est de prendre deux n -uplets dans une relation (autrement dit un n -uplet de $R \times R$) et de sélectionner les couples de n -uplets satisfaisant certaines contraintes.

Exemple : Quels acteurs ont la même adresse ?

```
SELECT Star1.Nom, Star2.Nom FROM Acteur Star1, Acteur Star2  
WHERE Star1.adresse = Star2.adresse
```

Ici Star1 et Star2 sont des variables pour les n -uplets. On aurait pu nommé ces variables différemment par exemple t1 et t2.

Introduction aux BDR

Jean-Paul
Comet,
Nadia
Abchiche-
Mimouni

Les concepts généraux

Notions essentielles

Algèbre Relationnelle + SQL

Opérateurs unaires

Traduction en SQL

Jointures

Opérateurs
ensemblistes

Opérateurs hors de
l'Alg.

Autres ordres SQL

le modèle Entités- Associations

NoSQL

Normalisation d'une relation

- Les variables de n -uplets vont être utiles lorsque l'on cherche à raisonner sur plusieurs n -uplets d'une même relation et à les comparer.
- L'idée est de prendre deux n -uplets dans une relation (autrement dit un n -uplet de $R \times R$) et de sélectionner les couples de n -uplets satisfaisant certaines contraintes.

Exemple : Quels acteurs ont la même adresse ?

```
SELECT Star1.Nom, Star2.Nom FROM Acteur Star1, Acteur Star2  
WHERE Star1.adresse = Star2.adresse
```

Ici Star1 et Star2 sont des variables pour les n -uplets. On aurait pu nommé ces variables différemment par exemple t1 et t2.

- 1 Problème 1 : on obtient des couples dont les deux valeurs sont identiques.

Introduction
aux BDRJean-Paul
Comet,
Nadia
Abchiche-
MimouniLes concepts
générauxNotions
essentiellesAlgèbre
Relationnelle
+ SQL

Opérateurs unaires

Traduction en SQL

Jointures

Opérateurs
ensemblistesOpérateurs hors de
l'Alg.

Autres ordres SQL

le modèle
Entités-
Associations

NoSQL

Normalisation
d'une relation

- Les variables de n -uplets vont être utiles lorsque l'on cherche à raisonner sur plusieurs n -uplets d'une même relation et à les comparer.
- L'idée est de prendre deux n -uplets dans une relation (autrement dit un n -uplet de $R \times R$) et de sélectionner les couples de n -uplets satisfaisant certaines contraintes.

Exemple : Quels acteurs ont la même adresse ?

```
SELECT Star1.Nom, Star2.Nom FROM Acteur Star1, Acteur Star2
```

```
WHERE Star1.adresse = Star2.adresse AND Star1.nom <> Star2.nom
```

Ici Star1 et Star2 sont des variables pour les n -uplets. On aurait pu nommé ces variables différemment par exemple t1 et t2.

- 1 Problème 1 : on obtient des couples dont les deux valeurs sont identiques. $\Rightarrow$ Star1.nom <> Star2.nom

Introduction aux BDR

Jean-Paul
Comet,
Nadia
Abchiche-
Mimouni

Les concepts généraux

Notions essentielles

Algèbre Relationnelle + SQL

Opérateurs unaires

Traduction en SQL

Jointures

Opérateurs
ensemblistes

Opérateurs hors de
l'Alg.

Autres ordres SQL

le modèle Entités- Associations

NoSQL

Normalisation d'une relation

- Les variables de n -uplets vont être utiles lorsque l'on cherche à raisonner sur plusieurs n -uplets d'une même relation et à les comparer.
- L'idée est de prendre deux n -uplets dans une relation (autrement dit un n -uplet de $R \times R$) et de sélectionner les couples de n -uplets satisfaisant certaines contraintes.

Exemple : Quels acteurs ont la même adresse ?

```
SELECT Star1.Nom, Star2.Nom FROM Acteur Star1, Acteur Star2
```

```
WHERE Star1.adresse = Star2.adresse AND Star1.nom <> Star2.nom
```

Ici Star1 et Star2 sont des variables pour les n -uplets. On aurait pu nommé ces variables différemment par exemple t1 et t2.

- ❶ Problème 1 : on obtient des couples dont les deux valeurs sont identiques. $\Rightarrow$ Star1.nom $\neq$ Star2.nom
- ❷ Problème 2 : on a les couples en double !

Introduction
aux BDRJean-Paul
Comet,
Nadia
Abchiche-
MimouniLes concepts
générauxNotions
essentiellesAlgèbre
Relationnelle
+ SQL

Opérateurs unaires

Traduction en SQL

Jointures

Opérateurs
ensemblistesOpérateurs hors de
l'Alg.

Autres ordres SQL

le modèle
Entités-
Associations

NoSQL

Normalisation
d'une relation

- Les variables de n -uplets vont être utiles lorsque l'on cherche à raisonner sur plusieurs n -uplets d'une même relation et à les comparer.
- L'idée est de prendre deux n -uplets dans une relation (autrement dit un n -uplet de $R \times R$) et de sélectionner les couples de n -uplets satisfaisant certaines contraintes.

Exemple : Quels acteurs ont la même adresse ?

```
SELECT Star1.Nom, Star2.Nom FROM Acteur Star1, Acteur Star2
```

```
WHERE Star1.adresse = Star2.adresse AND Star1.nom < Star2.nom
```

Ici Star1 et Star2 sont des variables pour les n -uplets. On aurait pu nommé ces variables différemment par exemple t1 et t2.

- 1 Problème 1 : on obtient des couples dont les deux valeurs sont identiques. $\Rightarrow \text{Star1.nom} \neq \text{Star2.nom}$
- 2 Problème 2 : on a les couples en double !
 $\Rightarrow \text{Star1.nom} < \text{Star2.nom}$

Introduction
aux BDR

Jean-Paul
Comet,
Nadia
Abchiche-
Mimouni

Les concepts
généraux

Notions
essentielles

Algèbre
Relationnelle
+ SQL

Opérateurs unaires
Traduction en SQL

Jointures

Opérateurs
ensemblistes

Opérateurs hors de
l'Alg.

Autres ordres SQL

le modèle
Entités-
Associations

NoSQL

Normalisation
d'une relation

Opérateur **ANY** $\equiv$ **SOME**

Quels sont les titres et dates des films dont le titre a été réutilisés par la suite pour au moins un autre film ?

```
SELECT Titre, date FROM Film t1
```

```
WHERE date < ANY
```

```
(SELECT date FROM Film WHERE Titre=t1.Titre)
```

- On sélectionne les titre et date des n -uplets
- chaque n -uplet sera appelé provisoirement $t1$
- pour lesquels la date est strictement antérieure à une date quelconque (**ANY**) de la table obtenue en sélectionnant toutes les dates associées à un film dont le titre est égal au titre du n -uplet $t1$.

Introduction
aux BDR

Jean-Paul
Comet,
Nadia
Abchiche-
Mimouni

Les concepts
généraux

Notions
essentielles

Algèbre
Relationnelle
+ SQL

Opérateurs unaires
Traduction en SQL

Jointures

Opérateurs
ensemblistes

Opérateurs hors de
l'Alg.

Autres ordres SQL

le modèle
Entités-
Associations

NoSQL

Normalisation
d'une relation

Opérateur ALL

Quels acteurs n'ont joué dans aucun film paru avant 2000 ?

```
SELECT nomActeur FROM Joue t1
```

```
WHERE t1.nomActeur <> ALL
```

```
(SELECT nomActeur FROM Joue WHERE FilmDate<=2000)
```

On sélectionne les noms d'acteurs qui sont différents de tous les noms d'acteurs (**ALL**) obtenus en sélectionnant les noms d'acteurs ayant joué dans un film dont la date est antérieure à 2000.

Introduction
aux BDRJean-Paul
Comet,
Nadia
Abchiche-
MimouniLes concepts
générauxNotions
essentiellesAlgèbre
Relationnelle
+ SQL

Opérateurs unaires

Traduction en SQL

Jointures

Opérateurs
ensemblistesOpérateurs hors de
l'Alg.

Autres ordres SQL

le modèle
Entités-
Associations

NoSQL

Normalisation
d'une relation**Rappel :**

- l'implémentation des relations dans SQL est basée sur des multi-ensembles (dans un multi-ensemble, chaque élément peut apparaître plusieurs fois contrairement au cas d'un ensemble).

L'utilisation de **DISTINCT** permet alors de supprimer les doublons.

Exemple. Quel est l'ensemble des producteurs de films dans lesquels Matt Damon joue ?

```
SELECT DISTINCT Nom FROM Producteur, Film, Joue
WHERE IdProd = Id AND
      Titre = TitreFilm AND
      date = FilmDate AND
      nomActeur = "Matt Damon";
```

Introduction
aux BDR

Jean-Paul
Comet,
Nadia
Abchiche-
Mimouni

Les concepts
généraux

Notions
essentielles

Algèbre
Relationnelle
+ SQL

Opérateurs unaires

Traduction en SQL

Jointures

Opérateurs
ensemblistes

Opérateurs hors de
l'Alg.

Autres ordres SQL

le modèle
Entités-
Associations

NoSQL

Normalisation
d'une relation

Exemple. A combien s'élèvent les gains de Matt Damon ?

```
SELECT SUM(Paye) FROM Joue  
WHERE nomActeur=' 'Matt Damon' ';
```

Introduction
aux BDRJean-Paul
Comet,
Nadia
Abchiche-
MimouniLes concepts
générauxNotions
essentiellesAlgèbre
Relationnelle
+ SQL

Opérateurs unaires

Traduction en SQL

Jointures

Opérateurs
ensemblistesOpérateurs hors de
l'Alg.

Autres ordres SQL

le modèle
Entités-
Associations

NoSQL

Normalisation
d'une relation

Exemple. A combien s'élèvent les gains de Matt Damon ?

```
SELECT SUM(Paye) FROM Joue  
WHERE nomActeur=' 'Matt Damon' ';
```

Exemple. Quel est la moyenne des cachets de Matt Damon ?

```
SELECT AVG(Paye) FROM Joue  
WHERE nomActeur=' 'Matt Damon' ';
```

Introduction
aux BDRJean-Paul
Comet,
Nadia
Abchiche-
MimouniLes concepts
générauxNotions
essentiellesAlgèbre
Relationnelle
+ SQL

Opérateurs unaires

Traduction en SQL

Jointures

Opérateurs
ensemblistesOpérateurs hors de
l'Alg.

Autres ordres SQL

le modèle
Entités-
Associations

NoSQL

Normalisation
d'une relation

Exemple. A combien s'élèvent les gains de Matt Damon ?

```
SELECT SUM(Paye) FROM Joue  
WHERE nomActeur=' 'Matt Damon' ';
```

Exemple. Quel est la moyenne des cachets de Matt Damon ?

```
SELECT AVG(Paye) FROM Joue  
WHERE nomActeur=' 'Matt Damon' ';
```

Exemple. Quel est le cachet minimal que Matt Damon a accepté ?

```
SELECT MIN(Paye) FROM Joue  
WHERE nomActeur=' 'Matt Damon' ';
```


Introduction
aux BDRJean-Paul
Comet,
Nadia
Abchiche-
MimouniLes concepts
générauxNotions
essentiellesAlgèbre
Relationnelle
+ SQL

Opérateurs unaires

Traduction en SQL

Jointures

Opérateurs
ensemblistesOpérateurs hors de
l'Alg.

Autres ordres SQL

le modèle
Entités-
Associations

NoSQL

Normalisation
d'une relation

Exemple. A combien s'élèvent les gains de Matt Damon ?

```
SELECT SUM(Paye) FROM Joue  
WHERE nomActeur=' 'Matt Damon' ';
```

Exemple. Quel est la moyenne des cachets de Matt Damon ?

```
SELECT AVG(Paye) FROM Joue  
WHERE nomActeur=' 'Matt Damon' ';
```

Exemple. Quel est le cachet minimal que Matt Damon a accepté ?

```
SELECT MIN(Paye) FROM Joue  
WHERE nomActeur=' 'Matt Damon' ';
```

Exemple. Quel est le cachet maximal de Matt Damon ?

```
SELECT MAX(Paye) FROM Joue  
WHERE nomActeur=' 'Matt Damon' ';
```

Introduction aux BDR

Jean-Paul
Comet,
Nadia
Abchiche-
Mimouni

Les concepts généraux

Notions essentielles

Algèbre Relationnelle + SQL

Opérateurs unaires

Traduction en SQL

Jointures

Opérateurs ensemblistes

Opérateurs hors de l'Alg.

Autres ordres SQL

le modèle Entités- Associations

NoSQL

Normalisation d'une relation

Exemple.

- ① nombre de valeurs non nulles dans une colonne :
`SELECT COUNT(nomActeur) FROM joue`
- ② nombre de valeurs distinctes non nulles :
`SELECT COUNT(DISTINCT nomActeur) FROM joue`
- ③ nombre total de lignes (même s'il y a des valeurs nulles) :
`SELECT COUNT(*) FROM joue`