



INSTITUT NATIONAL DES SCIENCES APPLIQUÉES DE LYON



INSA 1A

FORMATION

# Introduction aux bases de données – Part 3: SQL

# SQL: Structured Query Language

*SQL est un langage de manipulation de base de données développé par IBM (San Jose, 1981)*

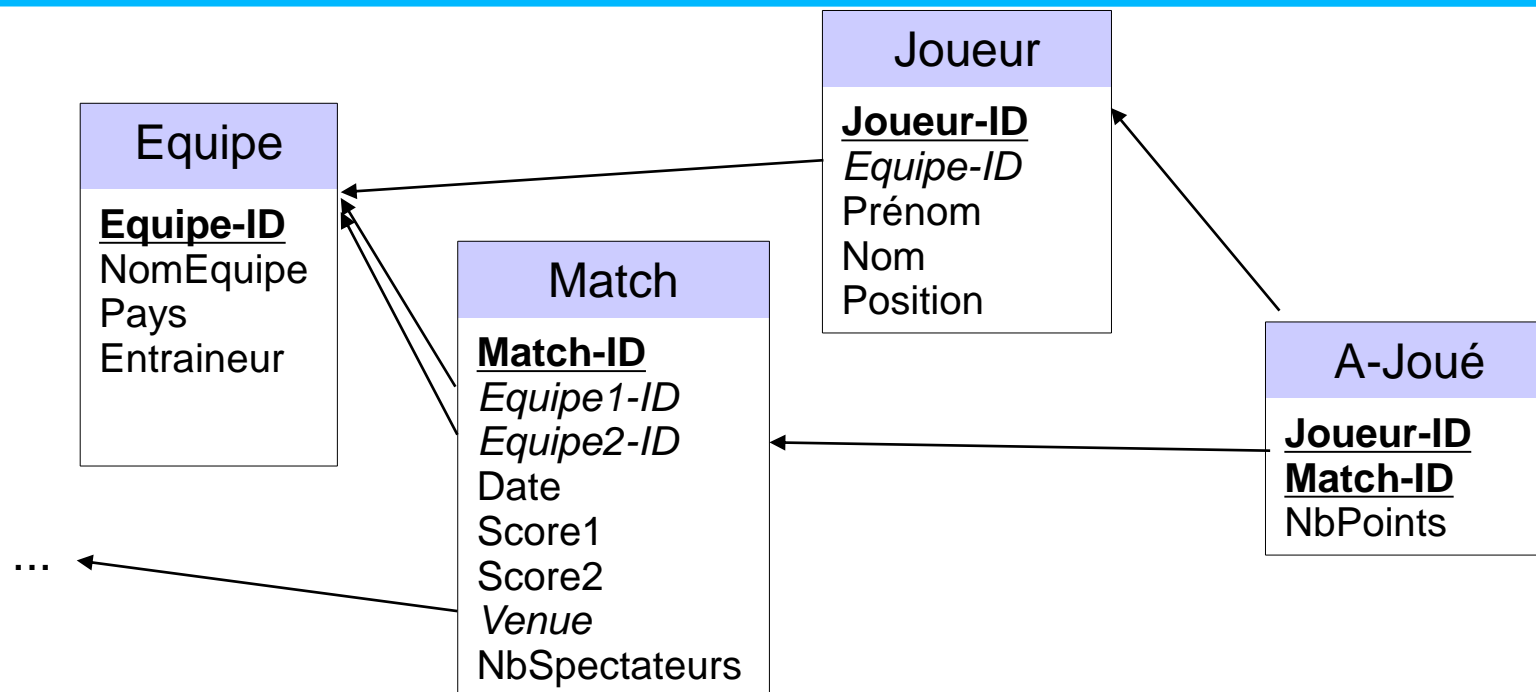
- ▶ Structured **Query Language**:

- ▶ Peut être écrit par des humains
- ▶ Peut être enregistré dans des fichiers
- ▶ Peut être échangé entre les applications

- ▶ **Structured** Query Language:

- ▶ Peuvent être traitées automatiquement par des machines
- ▶ SQL est un **standard**, mais il existe des dialectes (Access, MySQL), c'est à dire des variantes non standard

# Ex.: une Base de données Rugby



Equipe(Equipe-ID, NomEquipe, Pays, Entraîneur)

Joueur(Joueur-ID, Equipe-ID, Prénom, Nom, Position)

Match(Match-ID, Equipe1-ID, Equipe2-ID, Date, Score1, Score2, Venue, NbSpectateurs)

A-Joué(Joueur-ID, Match-ID, NbPoints)

# Extension (contenu de la BD)

## ► Table Equipe

Equipe-ID	NomEquipe	Pays	Entraîneur
FRA	XV de France	France	Laporte
NZL	All Blacks	New Zealand	Henry
ENG	XV of the Rose	England	Ashton
AUS	Wallabies	Australia	Connolly
RSA	Spring Boks	South Africa	White
ARG	Pumas	Argentina	Loffreda
SCO		Scotland	Hadden
FJI	Fidgi	Fidgi	Tabua

## ► Table Match

Match-ID	Equipe1-ID	Equipe2-ID	Date	Score1	Score2	Venue	NbSpectateurs
41	FRA	NZL	06/10/2007	20	18	Cardiff	73350
42	AUS	ENG	06/10/2007	10	12	Marseille	59500
43	RSA	FJI	07/10/2007	37	20	Marseille	59500
44	ARG	SCO	07/10/2007	19	13	Saint Denis	80000
45	ENG	FRA	13/10/2007	14	9	Saint Denis	80000
46	RSA	ARG	14/10/2007	37	13	Saint Denis	80000
47	FRA	ARG	19/10/2007	10	34	Paris	47870
48	RSA	ENG	20/10/2007	15	6	Saint Denis	80000

# Extension (contenu)

## ► Table Joueur

Joueur-ID	Equipe-ID	Prénom	Nom	Position
1	FRA	Jean-Baptiste	Elissalde	demi de mêlée
2	FRA	Vincent	Clerc	ailier
3	FRA	Lionel	Beauxis	demi d'ouverture
4	FRA	Lionel	Nallet	deuxième ligne
5	FRA	Sébastien	Chabal	troisième ligne
6	NZL	Joe	Rokocoko	ailier
7	NZL	Doug	Howlett	ailier
8	NZL	Dan	Carter	demi d'ouverture
9	NZL	Nick	Evans	demi d'ouverture
10	ARG	Felipe	Contepomi	demi d'ouverture
11	ARG	Juan Martin	Hernandez	arrière
12	ARG	Lucas	Borges	ailier
13	ARG	Ignacio	Corleto	ailier
14	AUS	Matt	Gitean	...
15	AUS	Stirling	Mortlock	...
16	AUS	Drew	Mitchel	...
17	ENG	Jonny	Wilkinson	...
18	RSA	Bryan	Habana	...
19	RSA	Percy	Montgomery	...
20	FJI	Nicky	Little	...
21	SCO	Alasdair	Dickinson	...

Joueur-ID	Match-ID	NbPoints
1	41	21
1	45	15
1	47	8
2	41	12
2	45	6
2	47	7
3	41	10
6	41	12
7	41	15
8	41	20
9	41	25
17	42	14
17	45	22
17	48	16
18	43	10
18	46	11
18	48	9
19	43	24
19	46	22
19	48	30
...	...	...

# Requêtes simples

## ► Forme générale:

```
SELECT <listOfColumns>  
FROM <listOfTables>;
```

-- Ceci est la clause du **SELECT** clause

-- Ceci est la clause du **FROM** clause



**A NE PAS OUBLIER!**

# Requêtes simples

## ► Forme générale:

```
SELECT <listOfColumns>  
FROM <listOfTables>;
```

## ► *Ex.: liste de tous les joueurs*

```
SELECT *  
FROM Joueur;
```

-- '\*' signifie "**toutes les colonnes**"

Joueur-ID	Equipe-ID	Prénom	Nom	Position
1	FRA	Jean-Baptiste	Elissalde	demi de mêlée
2	FRA	Vincent	Clerc	ailier
3	FRA	Lionel	Beauxis	demi d'ouverture
4	FRA	Lionel	Nallet	deuxième ligne
5	FRA	Sébastien	Chabal	troisième ligne
6	NZL	Joe	Rokocoko	ailier
7	NZL	Doug	Howlett	ailier
8	NZL	Dan	Carter	demi d'ouverture
9	NZL	Nick	Evans	demi d'ouverture
10	ARG	Felipe	Contepomi	demi d'ouverture
11	ARG	Juan Martin	Hernandez	arrière
12	ARG	Lucas	Borges	ailier
13	ARG	Ignacio	Corleto	ailier
14	AUS	Matt	Gitean	...
15	AUS	Stirling	Mortlock	...
16	AUS	Drew	Mitchel	...
17	ENG	Jonny	Wilkinson	...
18	RSA	Bryan	Habana	...
19	RSA	Percy	Montgomery	...
20	FJI	Nicky	Little	...
21	SCO	Alasdair	Dickinson	...



# Requêtes simples

## ► Forme générale:

```
SELECT <listOfColumns>  
FROM <listOfTables>;
```

## ► Ex.: liste de tous les joueurs

```
SELECT *  
FROM Joueur;
```

## ► En algèbre relationnelle: Joueur

Joueur-ID	Equipe-ID	Prénom	Nom	Position
1	FRA	Jean-Baptiste	Elissalde	demi de mêlée
2	FRA	Vincent	Clerc	ailier
3	FRA	Lionel	Beauxis	demi d'ouverture
4	FRA	Lionel	Nallet	deuxième ligne
5	FRA	Sébastien	Chabal	troisième ligne
6	NZL	Joe	Rokocoko	ailier
7	NZL	Doug	Howlett	ailier
8	NZL	Dan	Carter	demi d'ouverture
9	NZL	Nick	Evans	demi d'ouverture
10	ARG	Felipe	Contepomi	demi d'ouverture
11	ARG	Juan Martin	Hernandez	arrière
12	ARG	Lucas	Borges	ailier
13	ARG	Ignacio	Corleto	ailier
14	AUS	Matt	Gitean	...
15	AUS	Stirling	Mortlock	...
16	AUS	Drew	Mitchel	...
17	ENG	Jonny	Wilkinson	...
18	RSA	Bryan	Habana	...
19	RSA	Percy	Montgomery	...
20	FJI	Nicky	Little	...
21	SCO	Alasdair	Dickinson	...

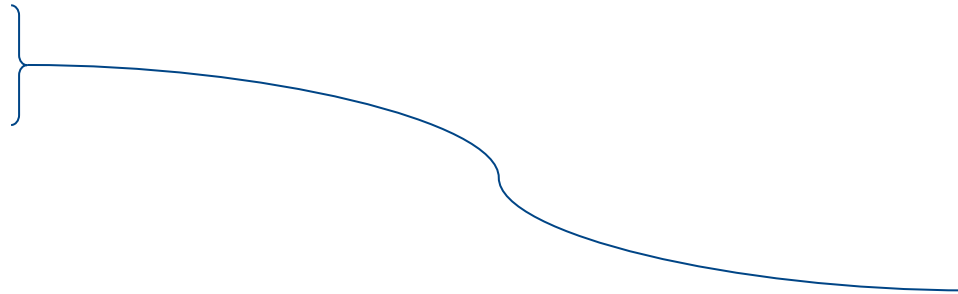


# Projection

- ▶ Projection sur les colonnes: nommez les colonnes dans la clause SELECT (**il peut y avoir des doublons**)

- ▶ *Ex.: liste de toutes les positions de Joueurs*

```
SELECT position  
FROM Joueur;
```



Position
demi de mêlée
ailier
demi d'ouverture
deuxième ligne
troisième ligne
ailier
ailier
demi d'ouverture
demi d'ouverture
demi d'ouverture
arrière
ailier
ailier
...
...
...
...
...
...
...
...
...

# Projection

- ▶ Projection sur les colonnes: nommez les colonnes dans la clause SELECT (**il peut y avoir des doublons**)

- ▶ *Ex.: liste de toutes les positions de Joueurs*

```
SELECT position  
FROM Joueur;
```

- ▶ En algèbre relationnelle:

$\pi_{\text{position}} (\text{Joueur})$

Position
demi de mêlée
ailier
demi d'ouverture
deuxième ligne
troisième ligne
ailier
ailier
demi d'ouverture
demi d'ouverture
demi d'ouverture
arrière
ailier
ailier
...
...
...
...
...
...
...
...
...

# Projection

- ▶ Projection sur les colonnes: nommez les colonnes dans la clause SELECT (**il peut y avoir des doublons**)

- ▶ *Ex.: liste de toutes les positions de Joueurs*

```
SELECT position  
FROM Joueur;
```

- ▶ En algèbre relationnelle:

$$\pi_{\text{position}} (\text{Joueur})$$

- ▶ Projection **sans doublons**

```
SELECT DISTINCT position  
FROM Joueur;
```

Position
demi de mêlée
ailier
demi d'ouverture
deuxième ligne
troisième ligne
ailier
ailier
demi d'ouverture
demi d'ouverture
demi d'ouverture
arrière
ailier
ailier
...
...
...
...
...
...
...
...

Position
demi de mêlée
ailier
demi d'ouverture
deuxième ligne
troisième ligne
arrière
...

# Sélection

```
SELECT <listOfColumns>  
FROM <listOfTables>  
WHERE <condition>;
```

-- Projection (clause du **SELECT**)

-- Sélection (clause du **WHERE**)

# Sélection

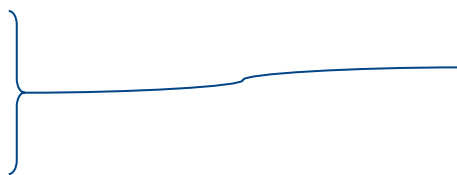
```
SELECT <listOfColumns>  
FROM <listOfTables>  
WHERE <condition>;
```

-- Projection (clause du **SELECT**)

-- Sélection (clause du **WHERE**)

► *Ex.: liste des noms des Joueurs français*

```
SELECT Prénom, Nom  
FROM Joueur  
WHERE Equipe-ID = 'FRA';
```



Prénom	Nom
Jean-Baptiste	Elissalde
Vincent	Clerc
Lionel	Beauxis
Lionel	Nallet
Sébastien	Chabal

# Sélection

```
SELECT <listOfColumns>  
FROM <listOfTables>  
WHERE <condition>;
```

-- Projection (clause du **SELECT**)

-- Sélection (clause du **WHERE**)

► Ex.: liste des noms des Joueurs français

```
SELECT Prénom, Nom  
FROM Joueur  
WHERE Equipe-ID = 'FRA';
```

Prénom	Nom
Jean-Baptiste	Elissalde
Vincent	Clerc
Lionel	Beauxis
Lionel	Nallet
Sébastien	Chabal

► En algèbre relationnelle:

$$\pi_{\text{Prénom, Nom}}(\sigma_{\text{Equipe-ID='FRA'}}(\text{Joueur}))$$

# Ordonner les résultats

```
SELECT <listofColumns>
FROM <listofTables>
WHERE <condition>
ORDER BY <column> [<order>] [, <column> [<order>]] ...
-- <order> est soit ASC ou DESC (ascendant ou descendant)
```

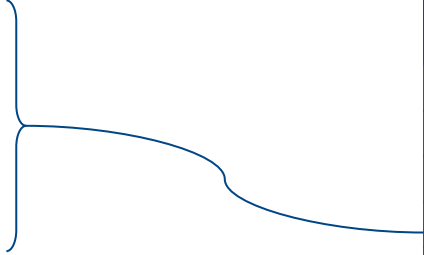


# Ordonner les résultats

```
SELECT <listOfColumns>
FROM <listOfTables>
WHERE <condition>
ORDER BY <column> [<order>] [, <column> [<order>]] ...
-- <order> est soit ASC ou DESC (ascendant ou descendant)
```

► *Ex.: liste des noms des Joueurs français, ordonnés par ordre alphabétique*

```
SELECT Prénom, Nom
FROM Joueur
WHERE Equipe-ID = 'FRA'
ORDER BY Nom ASC;
```



Prénom	Nom
Lionel	Beauxis
Sébastien	Chabal
Vincent	Clerc
Jean-Baptiste	Elissalde
Lionel	Nallet

# Combinaison de conditions

Les connecteurs booléens: **AND**, **OR**, **NOT** (negation)

Les opérateurs de comparaison: **=**, **<**, **>**, **<=**, **>=**, **<>**

► *Ex.: Matches disputés par l'Equipe française*

```
SELECT Match-ID, Equipe1-ID, Equipe2-ID, Date, NbSpectateurs  
FROM Match  
WHERE Equipe1-ID = 'FRA' OR Equipe2-ID = 'FRA';
```

Match-ID	Equipe1-ID	Equipe2-ID	Date	NbSpectateurs
41	FRA	NZL	06/10/2007	7335
45	ENG	FRA	13/10/2007	8000
47	FRA	ARG	19/10/2007	4787

# Combinaison de conditions

Les connecteurs booléens: **AND**, **OR**, **NOT** (negation)

Les opérateurs de comparaison: **=**, **<**, **>**, **<=**, **>=**, **<>**

► *Ex.: Matches disputés par l'Equipe française*

```
SELECT Match-ID, Equipe1-ID, Equipe2-ID, Date, NbSpectateurs
FROM Match
WHERE Equipe1-ID = 'FRA' OR Equipe2-ID = 'FRA';
```

Match-ID	Equipe1-ID	Equipe2-ID	Date	NbSpectateurs
41	FRA	NZL	06/10/2007	7335
45	ENG	FRA	13/10/2007	8000
47	FRA	ARG	19/10/2007	4787

► En algèbre relationnelle:

$\pi_{\text{Match-ID, Equipe1-ID, Equipe2-ID, Date, NbSpectateurs}}(\sigma_{\text{Equipe1-ID='FRA' or Equipe2-ID='FRA'}}(\text{Match}))$

# Combination de conditions

- *Ex.: Liste des matchs de l'Equipe française et ayant plus de 60000 spectateurs*

```
SELECT Match-ID, Equipe1-ID, Equipe2-ID, Date, NbSpectateurs  
FROM Match  
WHERE (Equipe1-ID = 'FRA' OR Equipe2-ID = 'FRA')  
AND NbSpectateurs > 60000;
```

Match-ID	Equipe1-ID	Equipe2-ID	Date	NbSpectateurs
41	FRA	NZL	06/10/2007	73350
45	ENG	FRA	13/10/2007	80000

# Combination de conditions

- *Ex.: Liste des matchs de l'Equipe française et ayant plus de 60000 spectateurs*

```
SELECT Match-ID, Equipe1-ID, Equipe2-ID, Date, NbSpectateurs
FROM Match
WHERE (Equipe1-ID = 'FRA' OR Equipe2-ID = 'FRA')
AND NbSpectateurs > 60000;
```

Match-ID	Equipe1-ID	Equipe2-ID	Date	NbSpectateurs
41	FRA	NZL	06/10/2007	73350
45	ENG	FRA	13/10/2007	80000

- En algèbre relationnelle:

$\pi$

Match-ID, Equipe1-ID, Equipe2-ID, Date, NbSpectateurs

$\left( \sigma_{(\text{Equipe1-ID}='FRA' \text{ or } \text{Equipe2-ID}='FRA') \text{ and } \text{NbSpectateurs}>60000}(\text{Match}) \right)$

# Conditions spéciales

- ▶ La valeur "vide" peut être testée avec une syntaxe spéciale:

```
SELECT <listOfColumns>  
FROM <listOfTables>  
WHERE <columnName> IS NULL;
```

ou

```
SELECT <listOfColumns>  
FROM <listOfTables>  
WHERE <columnName> IS NOT NULL;
```



Ne pas écrire:  
WHERE <columnName> ~~=~~ NULL;

# Produit cartésien

- Nommer les tables dans la clause FROM

```
SELECT *
FROM Joueur, A-Joué;
```

Premier Joueur

Joueur-ID	Equipe-ID	Prénom	Nom	Position	Joueur-ID	Match-ID	NbPoints
1	FRA	Jean-Baptiste	Elissalde	demi de mêlée	1	41	21
1	FRA	Jean-Baptiste	Elissalde	demi de mêlée	1	45	15
1	FRA	Jean-Baptiste	Elissalde	demi de mêlée	1	47	8
1	FRA	Jean-Baptiste	Elissalde	demi de mêlée	2	41	12
1	FRA	Jean-Baptiste	Elissalde	demi de mêlée	2	45	6
1	FRA	Jean-Baptiste	Elissalde	demi de mêlée	2	47	7
1	FRA	Jean-Baptiste	Elissalde	demi de mêlée	3	41	10
1	FRA	Jean-Baptiste	Elissalde	demi de mêlée	6	41	12
1	FRA	Jean-Baptiste	Elissalde	demi de mêlée	7	41	15
1	FRA	Jean-Baptiste	Elissalde	demi de mêlée	8	41	20
...	...	...	...	...	...	...	...
2	FRA	Vincent	Clerc	ailier	1	41	21
2	FRA	Vincent	Clerc	ailier	1	45	15
2	FRA	Vincent	Clerc	ailier	1	47	8
2	FRA	Vincent	Clerc	ailier	2	41	12

- En algèbre relationnelle : Joueur  $\otimes$  A-Joué

Second Joueur



# Jointure

*Produit cartésien précédé d'une sélection*

► *Ex.: Liste des Joueurs qui ont participé au match #41*

```
SELECT Equipe-ID, Nom, Match-ID  
FROM Joueur, A-Joué  
WHERE Joueur.Joueur-ID = A-Joué.Joueur-ID  
AND Match-ID = 41;
```

Equipe-ID	Nom	Match-ID
FRA	Elissalde	41
FRA	Clerc	41
FRA	Beauxis	41
NZL	Rokocoko	41
NZL	Howlett	41
NZL	Carter	41
NZL	Evans	41

# Jointure

*Produit cartésien précédé d'une sélection*

► *Ex.: Liste des Joueurs qui ont participé au match #41*

```
SELECT Equipe-ID, Nom, Match-ID  
FROM Joueur, A-Joué  
WHERE Joueur.Joueur-ID = A-Joué.Joueur-ID  
AND Match-ID = 41;
```

Equipe-ID	Nom	Match-ID
FRA	Elissalde	41
FRA	Clerc	41
FRA	Beauxis	41
NZL	Rokocoko	41
NZL	Howlett	41
NZL	Carter	41
NZL	Evans	41

► In relational algebra

$\pi_{\text{Equipe-ID, Nom, Match-ID}}(\text{Joueur} \bowtie_{\text{Joueur-ID} = \text{Joueur-ID}} \text{A-Joué})$

# Jointure de plus de deux tables

► *Ex.: Points marqués par les Joueurs des All Blacks*

```
SELECT    Nom, NomEquipe, Match-ID, Date,
          Equipe1-ID, Equipe2-ID, NbPoints
FROM      Match, Joueur, A-Joué, Equipe
WHERE     Joueur.Joueur-ID = A-Joué.Joueur-ID
AND       Match.Match-ID = A-Joué.Match-ID
AND       Equipe.Equipe-ID = Joueur.Equipe-ID
AND       NomEquipe = 'All Blacks';
```

Nom	NomEquipe	Match-ID	Date	Equipe1-ID	Equipe2-ID	NbPoints
Rokocoko	All Blacks	41	06/10/07	FRA	NZL	Rokocoko
Howlett	All Blacks	41	06/10/07	FRA	NZL	Howlett
Carter	All Blacks	41	06/10/07	FRA	NZL	Carter
Evans	All Blacks	41	06/10/07	FRA	NZL	Evans

# Jointure de plus de deux tables

► *Ex.: Points marqués par les Joueurs des All Blacks*

```
SELECT    Nom, NomEquipe, Match-ID, Date,
          Equipe1-ID, Equipe2-ID, NbPoints
FROM      Match, Joueur, A-Joué, Equipe
WHERE     Joueur.Joueur-ID = A-Joué.Joueur-ID
AND       Match.Match-ID = A-Joué.Match-ID
AND       Equipe.Equipe-ID = Joueur.Equipe-ID
AND       NomEquipe = 'All Blacks';
```

Nom	NomEquipe	Match-ID	Date	Equipe1-ID	Equipe2-ID	NbPoints
Rokocoko	All Blacks	41	06/10/07	FRA	NZL	Rokocoko
Howlett	All Blacks	41	06/10/07	FRA	NZL	Howlett
Carter	All Blacks	41	06/10/07	FRA	NZL	Carter
Evans	All Blacks	41	06/10/07	FRA	NZL	Evans

► En algèbre relationnelle

$\pi_{\text{Nom, NomEquipe, Match-ID, Date, Equipe1-ID, Equipe2-ID, NbPoints}}$   
 $(\sigma_{\text{Joueur.Joueur-ID=A-Joué.Joueur-ID}} (\text{Match} \otimes \text{Joueur} \otimes \text{A-Joué} \otimes \text{Equipe}))$   
and Match.Match-ID=A-Joué.Match-ID  
and Equipe.Equipe-ID=Joueur.Equipe-ID  
and NomEquipe="All Blacks"

# Requêtes imbriquées

► Par l'utilisation d'opérateurs ensemblistes:

**IN**            -- appartenance à l'ensemble

**NOT IN**      -- non appartenance à l'ensemble

**IN** est un opérateur qui détermine si la valeur d'une expression est égale à une ou plusieurs valeurs appartenant à un ensemble.

*Ex.: Liste des Joueurs qui ont participé au match #41, avec un projection sur le nom des Joueurs*

```
SELECT Nom
FROM Joueur
WHERE Joueur-ID IN
      (SELECT Joueur-ID
       FROM   A-Joué
       WHERE  Match-ID = 41);
```

Nom
Elissalde
Clerc
Beauxis
Rokocoko
Howlett
Carter
Evans

# Requêtes imbriquées

**IN**      -- appartenance à l'ensemble

**NOT IN**      est souvent utilisé à la place de l'opérateur de différence

► *Ex.: Joueurs qui n'ont participé à aucun match*

```
SELECT Joueur-ID
FROM Joueur
WHERE Joueur-ID NOT IN
      (SELECT Joueur-ID
       FROM   A-Joué);
```

Joueur-ID
4
5
10
11
12
13
14
15
16
20
21

# Requêtes imbriquées

**IN**      -- appartenance à l'ensemble

**NOT IN**      est souvent utilisé à la place de l'opérateur de différence

► *Ex.: Joueurs qui n'ont participé à aucun match*

```
SELECT Joueur-ID
FROM Joueur
WHERE Joueur-ID NOT IN
      (SELECT Joueur-ID
       FROM   A-Joué);
```

Joueur-ID
4
5
10
11
12
13
14
15
16
20
21

► En algèbre relationnelle

$\pi_{\text{Joueur-ID}}(\text{Joueur}) - \pi_{\text{Joueur-ID}}(\text{A-Joué})$



# D'autres opérations SQL

**UNION** sélection des lignes qui appartiennent à la première ou à la seconde relation (avec suppression de doublons).

*Ex.: Noms des Joueurs et Entraîneurs de l'Equipe française*

```
(SELECT Nom
FROM Joueur
WHERE Equipe-ID = 'FRA')
UNION
(SELECT Entraîneur
FROM Equipe
WHERE Equipe-ID = 'FRA');
```

Nom
Elissalde
Clerc
Beauxis
Nallet
Chabal
Laporte

# D'autres opérations SQL

**UNION** sélection des lignes qui appartiennent à la première ou à la seconde relation (avec suppression de doublons).

*Ex.: Noms des Joueurs et Entraîneurs de l'Equipe française*

```
(SELECT Nom
FROM Joueur
WHERE Equipe-ID = 'FRA')
UNION
(SELECT Entraîneur
FROM Equipe
WHERE Equipe-ID = 'FRA');
```

Nom
Elissalde
Clerc
Beauxis
Nallet
Chabal
Laporte

► D'autres opérateurs:

**INTERSECTION** sélection des lignes qui appartiennent à la fois aux deux relations

**MINUS** sélection des lignes de la première relation qui n'appartiennent pas à la deuxième relation (différence)

# Opérateurs d'agrégats

COUNT	Nombre de valeurs d'un ensemble
MAX	Valeur maximale d'un ensemble de valeurs
MIN	Valeur minimale d'un ensemble de valeurs
SUM	Somme d'une série de valeurs
AVG	Moyenne d'un ensemble de valeurs

► *Ex.: Nombre total de Joueurs*

```
SELECT COUNT(*) AS NbJoueur  
FROM Joueur;
```

NbJoueur
21

# Opérateurs d'agrégats

COUNT	Nombre de valeurs d'un ensemble
MAX	Valeur maximale d'un ensemble de valeurs
MIN	Valeur minimale d'un ensemble de valeurs
SUM	Somme d'une série de valeurs
AVG	Moyenne d'un ensemble de valeurs

► *Ex.: Nombre total de Joueurs*

```
SELECT COUNT(*) AS NbJoueur  
FROM Joueur;
```

NbJoueur
21

► *Ex.: Nombre maximum de spectateurs*

```
SELECT MAX(NbSpectateurs) AS MaxAttendance  
FROM Match;
```

MaxAttendance
80000

# Opérateurs d'agrégats

COUNT	Nombre de valeurs d'un ensemble
MAX	Valeur maximale d'un ensemble de valeurs
MIN	Valeur minimale d'un ensemble de valeurs
SUM	Somme d'une série de valeurs
AVG	Moyenne d'un ensemble de valeurs

► *Ex.: Nombre moyen de spectateurs*

```
SELECT AVG(NbSpectateurs) AS AvgSpectators  
FROM Match;
```

AvgSpectators
70027

Ou

```
SELECT SUM(NbSpectateurs) / COUNT(*)  
FROM Match;
```

**Des opérations  
mathématiques (+, -, \*, /)  
peuvent être utilisées**

# Partition de relations

**GROUP BY** effectue une **partition** du résultat, sur la base d'un attribut ; les agrégats sont calculés **pour chaque groupe** dans la partition

Joueur-ID	Equipe-ID	Prénom	Nom	Position
1	FRA	Jean-Baptiste	Elissalde	demi de mêlée
2	FRA	Vincent	Clerc	ailier
3	FRA	Lionel	Beauxis	demi d'ouverture
4	FRA	Lionel	Nallet	deuxième ligne
5	FRA	Sébastien	Chabal	troisième ligne
6	NZL	Joe	Rokocoko	ailier
7	NZL	Doug	Howlett	ailier
8	NZL	Dan	Carter	demi d'ouverture
9	NZL	Nick	Evans	demi d'ouverture
10	ARG	Felipe	Contepomi	demi d'ouverture
11	ARG	Juan Martin	Hernandez	arrière
12	ARG	Lucas	Borges	ailier
13	ARG	Ignacio	Corleto	ailier
14	AUS	Matt	Gitean	...
15	AUS	Stirling	Mortlock	...
16	AUS	Drew	Mitchel	...
17	ENG	Jonny	Wilkinson	...
18	RSA	Bryan	Habana	...
19	RSA	Percy	Montgomery	...
20	FJI	Nicky	Little	...
21	SCO	Alasdair	Dickinson	...

```
SELECT COUNT(*)  
AS NbJoueur  
FROM Joueur  
GROUP BY Equipe-ID;
```

# Partition de relations

**GROUP BY** effectue une **partition** du résultat, sur la base d'un attribut ; les agrégats sont calculés **pour chaque groupe** dans la partition

Joueur-ID	Equipe-ID	Prénom	Nom	Position
1	FRA	Jean-Baptiste	Elissalde	demi de mêlée
2	FRA	Vincent	Clerc	ailier
3	FRA	Lionel	Beauxis	demi d'ouverture
4	FRA	Lionel	Nallet	deuxième ligne
5	FRA	Sébastien	Chabal	troisième ligne
6	NZL	Joe	Rokocoko	ailier
7	NZL	Doug	Howlett	ailier
8	NZL	Dan	Carter	demi d'ouverture
9	NZL	Nick	Evans	demi d'ouverture
10	ARG	Felipe	Contepomi	demi d'ouverture
11	ARG	Juan Martin	Hernandez	arrière
12	ARG	Lucas	Borges	ailier
13	ARG	Ignacio	Corleto	ailier
14	AUS	Matt	Gitean	...
15	AUS	Stirling	Mortlock	...
16	AUS	Drew	Mitchel	...
17	ENG	Jonny	Wilkinson	...
18	RSA	Bryan	Habana	...
19	RSA	Percy	Montgomery	...
20	FJI	Nicky	Little	...
21	SCO	Alasdair	Dickinson	...

```
SELECT COUNT(*)  
AS NbJoueur  
FROM Joueur  
GROUP BY Equipe-ID;
```



# Partition de relations

**GROUP BY** effectue une **partition** du résultat, sur la base d'un attribut ; les agrégats sont calculés **pour chaque groupe** dans la partition

Joueur-ID	Equipe-ID	Prénom	Nom	Position
1	FRA	Jean-Baptiste	Elissalde	demi de mêlée
2	FRA	Vincent	Clerc	ailier
3	FRA	Lionel	Beauxis	demi d'ouverture
4	FRA	Lionel	Nallet	deuxième ligne
5	FRA	Sébastien	Chabal	troisième ligne
6	NZL	Joe	Rokocoko	ailier
7	NZL	Doug	Howlett	ailier
8	NZL	Dan	Carter	demi d'ouverture
9	NZL	Nick	Evans	demi d'ouverture
10	ARG	Felipe	Contepomi	demi d'ouverture
11	ARG	Juan Martin	Hernandez	arrière
12	ARG	Lucas	Borges	ailier
13	ARG	Ignacio	Corleto	ailier
14	AUS	Matt	Gitean	...
15	AUS	Stirling	Mortlock	...
16	ALIS	Drew	Mitchel	...
17	ENG	Jonny	Wilkinson	...
18	RSA	Bryan	Habana	...
19	RSA	Percy	Montgomery	...
20	FJI	Nicky	Little	...
21	SCO	Alasdair	Dickinson	...

```
SELECT COUNT(*)  
AS NbJoueur  
FROM Joueur  
GROUP BY Equipe-ID;
```

Ceci est  
la partition

# Partition de relations

**GROUP BY** effectue une **partition** du résultat, sur la base d'un attribut ; les agrégats sont calculés **pour chaque groupe** dans la partition

Joueur-ID	Equipe-ID	Prénom	Nom	Position
1	FRA	Jean-Baptiste	Elissalde	demi de mêlée
2	FRA	Vincent	Clerc	ailier
3	FRA	Lionel	Beauxis	demi d'ouverture
4	FRA	Lionel	Nallet	deuxième ligne
5	FRA	Sébastien	Chabal	troisième ligne
6	NZL	Joe	Rokocoko	ailier
7	NZL	Doug	Howlett	ailier
8	NZL	Dan	Carter	demi d'ouverture
9	NZL	Nick	Evans	demi d'ouverture
10	ARG	Felipe	Contepomi	demi d'ouverture
11	ARG	Juan Martin	Hernandez	arrière
12	ARG	Lucas	Borges	ailier
13	ARG	Ignacio	Corleto	ailier
14	AUS	Matt	Gitean	...
15	AUS	Stirling	Mortlock	...
16	ALIS	Drew	Mitchel	...
17	ENG	Jonny	Wilkinson	...
18	RSA	Bryan	Habana	...
19	RSA	Percy	Montgomery	...
20	FJI	Nicky	Little	...
21	SCO	Alasdair	Dickinson	...

```
SELECT COUNT(*)
AS NbJoueur
FROM Joueur
GROUP BY Equipe-ID;
```

5

4

4

3

1

2

1

1

NbJoueur
5
4
4
3
1
2
1
1

# Partition de relations

**GROUP BY** effectue une **partition** du résultat, sur la base d'un attribut ; les agrégats sont calculés **pour chaque groupe** dans la partition

► *Ex.: Pour chaque Joueur, lister la somme des points marqués*

```
SELECT Nom, SUM(NbPoint) AS Points
FROM A-Joué, Joueur
WHERE A-Joué.Joueur-ID = Joueur.Joueur-ID
GROUP BY Joueur-ID;
```

Nom	Points
Elissalde	45
Clerc	25
Beauxis	24
Rokocoko	12
Howlett	15
Carter	20
Evans	25
Wilkinson	52
Habana	30
Montgomery	76

# Partition de relations

**HAVING** applique une condition sur les agrégats

- Ex.: Liste des points marqués par les Joueurs ayant joué plus de 2 matchs (et dans l'ordre des points marqués)

```
SELECT Nom, SUM(NbPoint) AS Points
FROM A-Joué, Joueur
WHERE A-Joué.Joueur-ID = Joueur.Joueur-ID
GROUP BY Joueur-ID
HAVING COUNT(*) > 2
ORDER BY Points;
```

Nom	Points
Elissalde	45
Clerc	25
Wilkinson	52
Habana	30
Montgomery	76