



INF 4401 - SQL 1

Julien Romero - Télécom SudParis

Previously...

The Relational Model

- We organize data into **tables** composed of **columns** (attributes) and **rows** (tuples)
- Each column has a name and a **domain**
- The content of a table is called a **relation**
- A **key** identifies uniquely a row
- A **relation schema** is a description of a table
- A **database schema** is the description of all the tables in the database

The Relational Model - Example

Students

StudentID	Name	Age
0202	John	10
3453	Bob	20
0192	John	42

- Attributes with domains?
- Key?
- Relation schema?

The Relational Model - Example

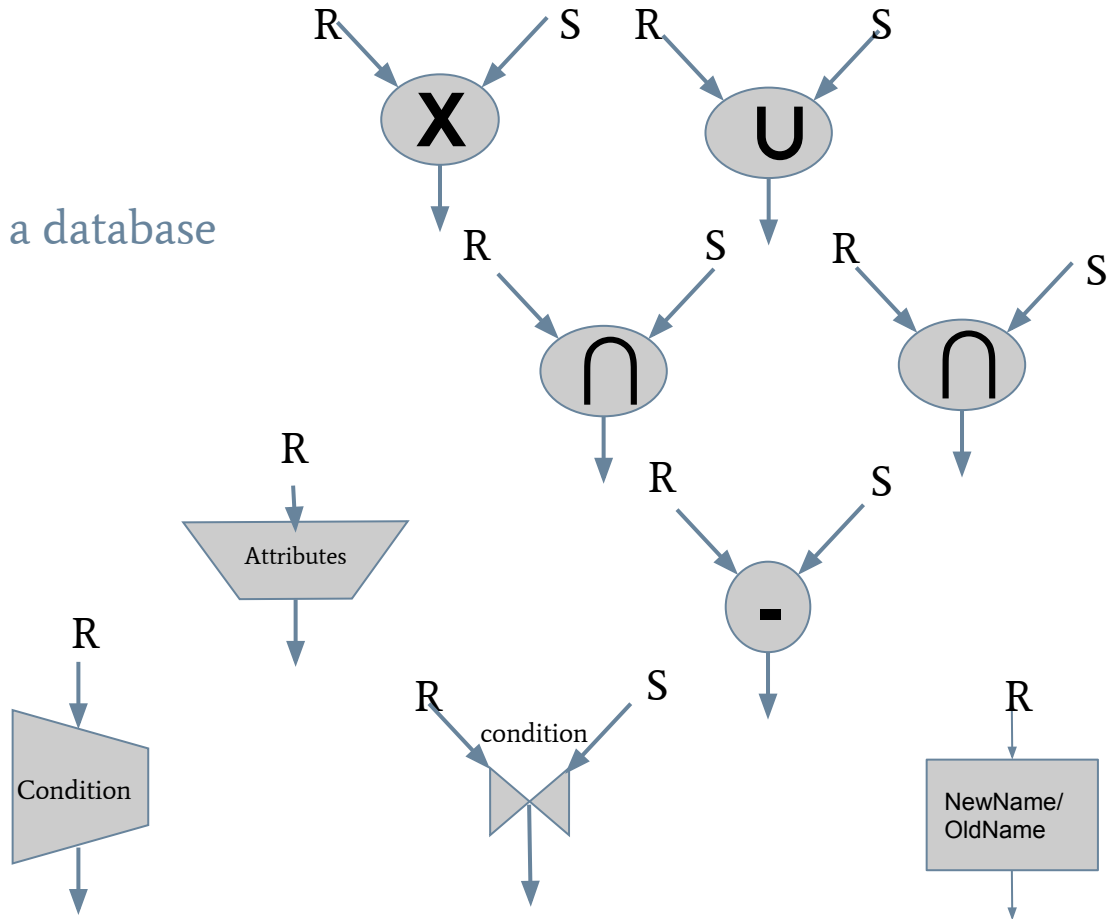
Students

StudentID	Name	Age
0202	John	10
3453	Bob	20
0192	John	42

- Attributes with domains:
 - StudentID: Integer
 - Name: String
 - Age: Integer
- Key: StudentID
- Relation schema:
 - Students(StudentID: Integer, Name: String, Age: Integer)

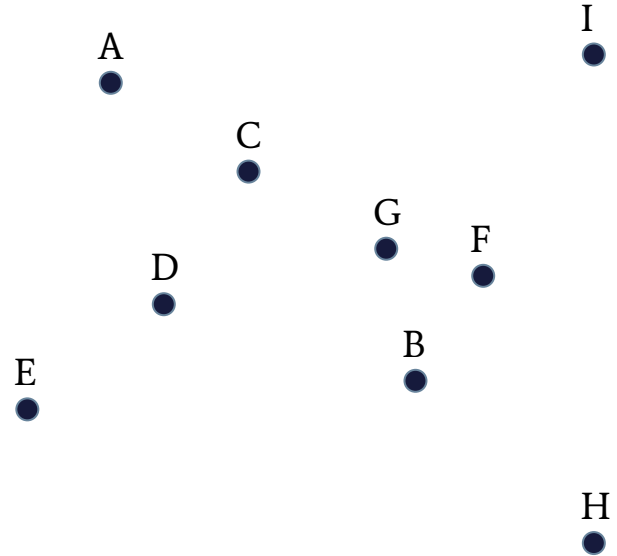
Relational Algebra

- A formalism to query data in a database
- Operations:
 - Project
 - Select
 - Product
 - Union, Intersection, Difference
 - Renaming
 - Join
- Join = Product + Select



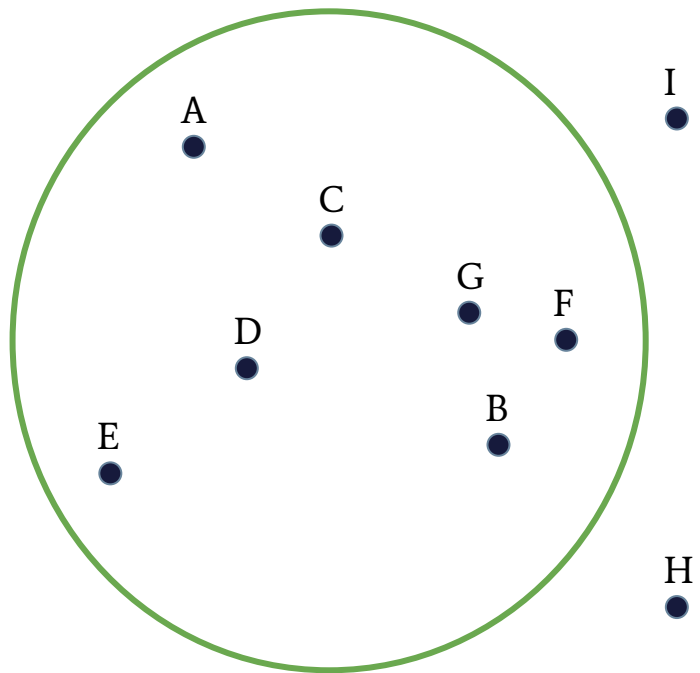
Set Theory

- We have a set of data points
 - They can represent people, planets, students, images, ...



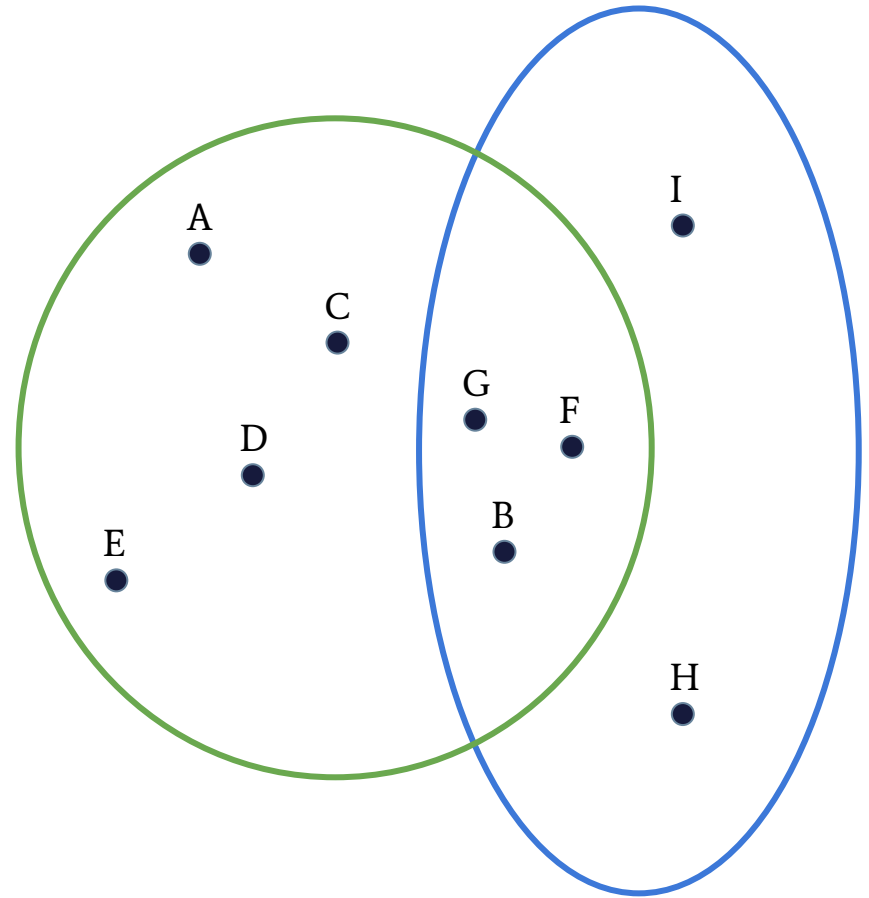
Set Theory

- We have a set of data points
 - They can represent people, planets, students, images, ...
- A group of data points is called a set
 - Here the set is $\{A, B, C, D, E, F, G\}$



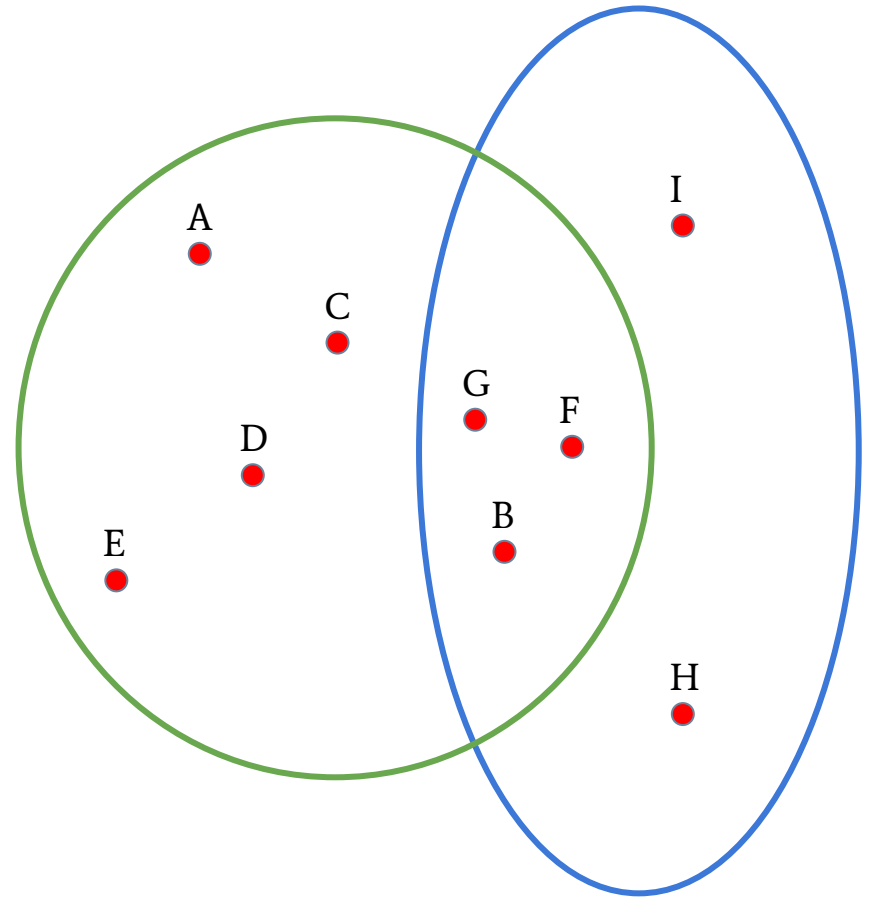
Set Theory

- We have a set of data points
 - They can represent people, planets, students, images, ...
- A group of data points is called a set
 - Here the set is $\{A, B, C, D, E, F, G\}$
- When we have several sets, we can compare them.



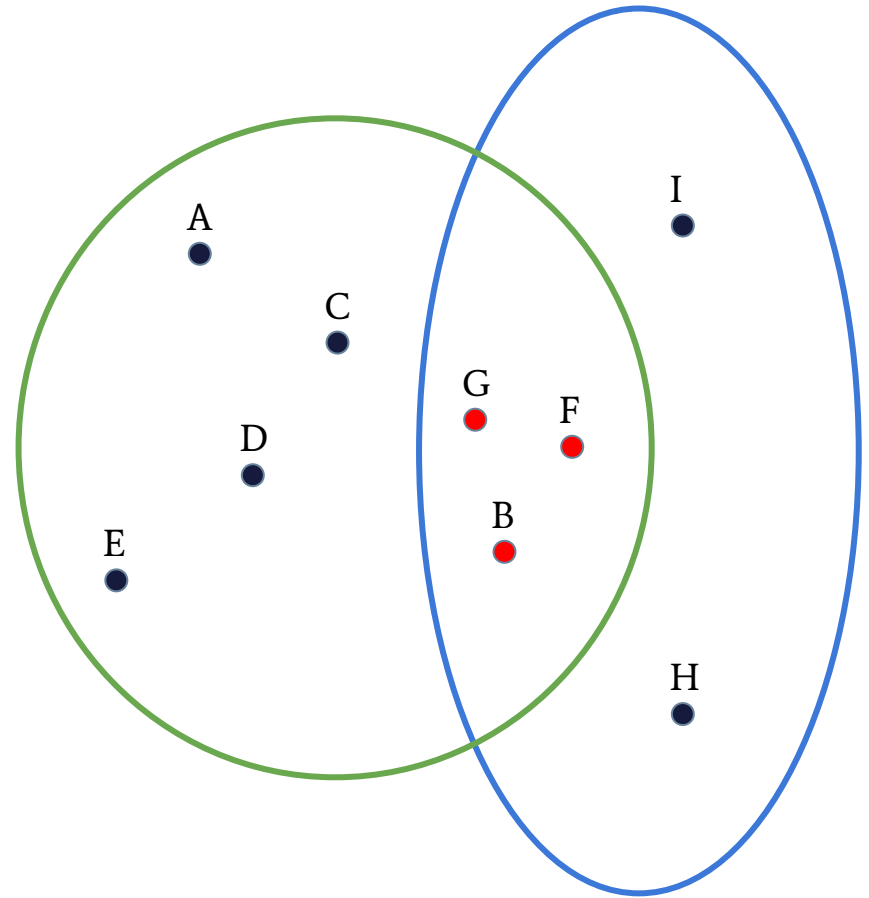
Set Theory

- We have a set of data points
 - They can represent people, planets, students, images, ...
- A group of data points is called a set
 - Here the set is $\{A, B, C, D, E, F, G\}$
- When we have several sets, we can compare them.
 - Union: Elements in at least one set (OR)



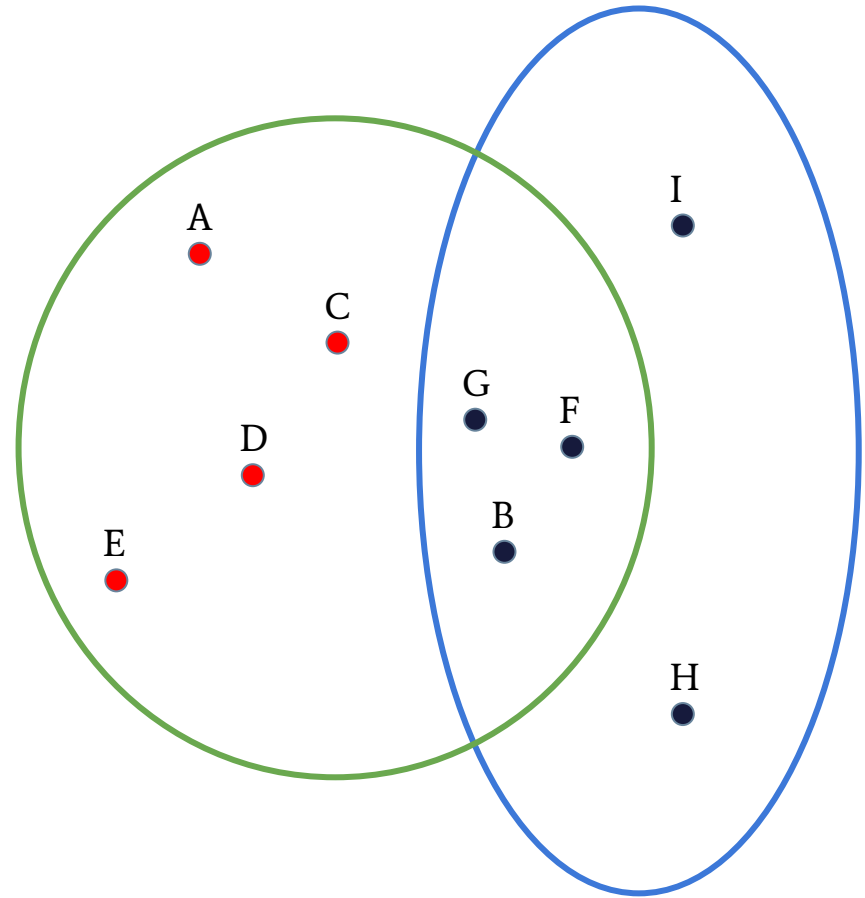
Set Theory

- We have a set of data points
 - They can represent people, planets, students, images, ...
- A group of data points is called a set
 - Here the set is $\{A, B, C, D, E, F, G\}$
- When we have several sets, we can compare them.
 - Union: Elements in at least one set (OR)
 - Intersection: Elements in all sets (AND)



Set Theory

- We have a set of data points
 - They can represent people, planets, students, images, ...
- A group of data points is called a set
 - Here the set is $\{A, B, C, D, E, F, G\}$
- When we have several sets, we can compare them.
 - Union: Elements in at least one set (OR)
 - Intersection: Elements in all sets (AND)
 - Difference: Elements in a first set but not in a second one (here Green - Blue) (AND NOT)



SQL

What is SQL?

- SQL (Structured Query Language) is a **declarative** language used to **query**, **manipulate**, **define**, and **secure** databases.
- Pronounced S-Q-L or sequel.
- Declarative = Express the logic of a program without defining its control flow
 - Example: Get the list of all the cats' names such that the cats birth year is less than 2010

Python

```
if __name__ == "__main__":  
    result = []  
    for cat in cats:  
        if cat.birth_year < 2010:  
            result.append(cat.name)  
    print("\n".join(result))
```

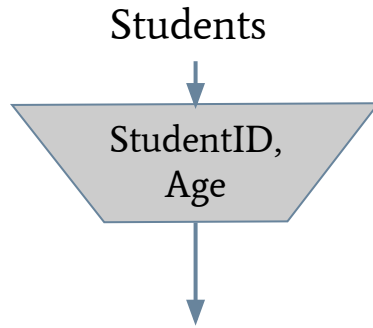
SQL

```
SELECT name  
FROM cats  
WHERE birth_year < 2000
```

From Relational Algebra To SQL

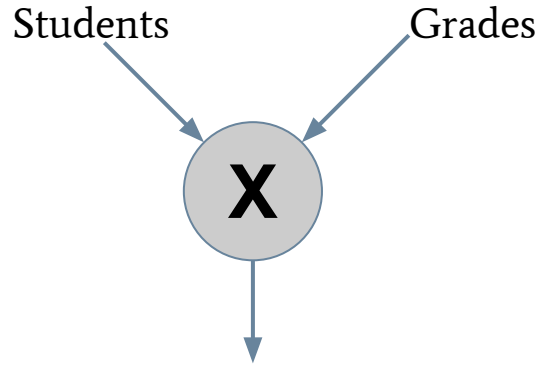
- SQL is a practical implementation of relational algebra (but it does much more)
- The three main keywords of SQL are **SELECT**, **FROM** and **WHERE**. They match operator in relational algebra:
 - **SELECT** is a projection (!) and attribute renaming
 - **PROJECT**(A1, ..., An in the table R) \leftrightarrow **SELECT** A1, ..., An **FROM** R;
 - **FROM** is a cartesian product and table renaming
 - **PRODUCT**(R1, ..., Rn) \leftrightarrow **SELECT** * **FROM** R1, ..., Rn;
 - **WHERE** is a selection
 - **SELECT**(condition in R) \leftrightarrow **SELECT** * **FROM** R **WHERE** condition;

From Relational Algebra To SQL - Project



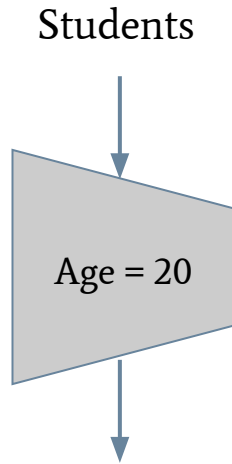
```
SELECT StudentID, Age  
FROM Students
```


From Relational Algebra To SQL - Product



```
SELECT *  
FROM Students, Grades
```

From Relational Algebra To SQL - SELECT



```
SELECT *  
FROM Students  
WHERE Age = 20
```

SQL - SFW Queries

- We want to select information from one or several tables that match given conditions.
 - What are the movies from 2014?
 - Who directed Avatar and in which year?
 - What is the list of directors?

Structure (keywords are not case sensitive but values are!):

```
SELECT <list of attribute(s) to select>  
FROM <list of table(s) to consider>  
[WHERE <condition> ];
```

This command returns a new table.

Table movies

title	director	year
Avatar	James Cameron	2009
The Grand Budapest Hotel	Wes Anderson	2014
Whiplash	Damien Chazelle	2014
Gran Torino	Clint Eastwood	2008
The Godfather	Francis Ford Coppola	1972
Moonrise Kingdom	Wes Anderson	2012

The brackets [...] mean that the command is optional.

SQL - SFW Queries - Semantics

- The execution of a SFW query is equivalent to a cartesian product, followed by a selection, and ending with a projection
 - We talk about *semantics* and not execution order because the execution order depends on the implementation: The DBMS must guarantee the equivalence.
- Important to understand some queries

SQL - SFW Queries - Semantics - Example

- `SELECT P.A FROM P, Q WHERE P.A = Q.B`

A
1
3

Table P

B	C
2	2
3	3
3	4

Table Q

SQL - SFW Queries - Semantics - Cartesian Product

- `SELECT P.A FROM P, Q WHERE P.A = Q.B`

A	B	C
1	2	3
1	3	4
1	3	5
3	2	3
3	3	4
3	3	5

SQL - SFW Queries - Semantics - Selection

- `SELECT P.A FROM P, Q WHERE P.A = Q.B`

A	B	C
3	3	4
3	3	5

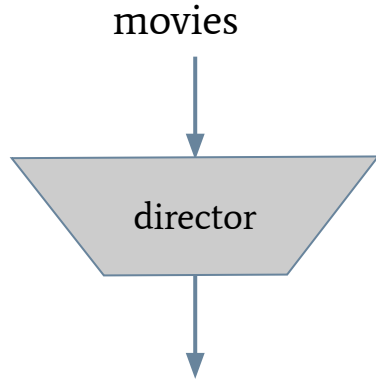
SQL - SFW Queries - Semantics - Projection

- **SELECT P.A** FROM P, Q WHERE P.A = Q.B

A
3
3

SQL - SFW Queries - distinct

- What is the list of directors?



```
SELECT DISTINCT(director)
FROM movies;
```

Table movies

title	director	year
Avatar	James Cameron	2009
The Grand Budapest Hotel	Wes Anderson	2014
Whiplash	Damien Chazelle	2014
Gran Torino	Clint Eastwood	2008
The Godfather	Francis Ford Coppola	1972
Moonrise Kingdom	Wes Anderson	2012

distinct get the set of unique elements.

SQL - SFW Queries- Examples

- What are the titles of the movies from 2014?

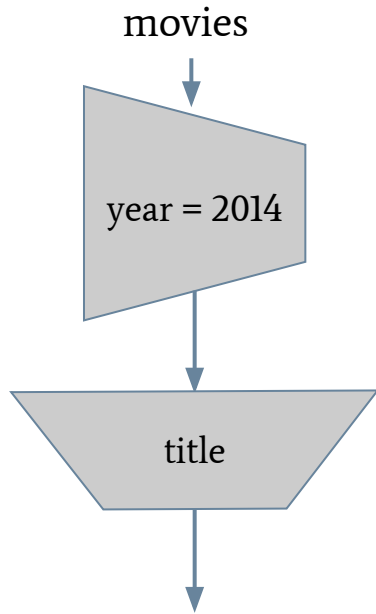
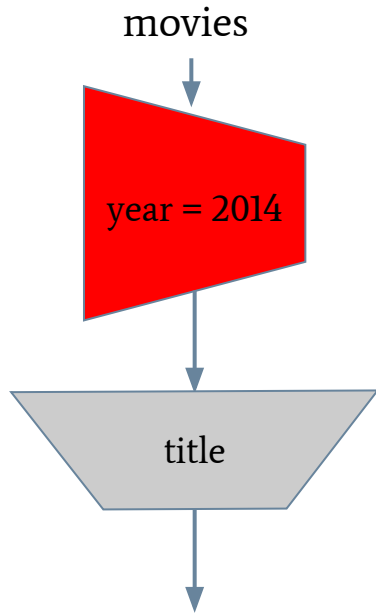


Table movies

title	director	year
Avatar	James Cameron	2009
The Grand Budapest Hotel	Wes Anderson	2014
Whiplash	Damien Chazelle	2014
Gran Torino	Clint Eastwood	2008
The Godfather	Francis Ford Coppola	1972
Moonrise Kingdom	Wes Anderson	2012

SQL - SFW Queries- Examples

- What are the titles of the movies from 2014?



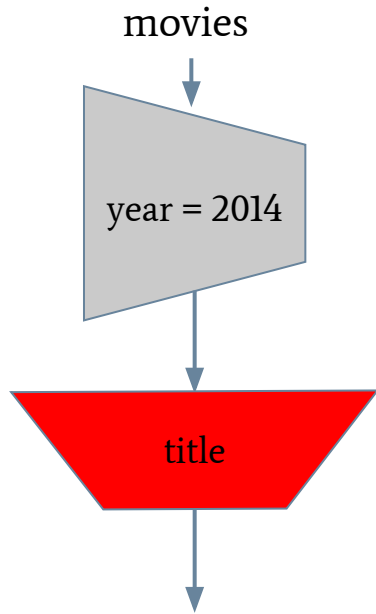
```
SELECT *  
FROM movies  
WHERE year = 2014;
```

Table movies

title	director	year
Avatar	James Cameron	2009
The Grand Budapest Hotel	Wes Anderson	2014
Whiplash	Damien Chazelle	2014
Gran Torino	Clint Eastwood	2008
The Godfather	Francis Ford Coppola	1972
Moonrise Kingdom	Wes Anderson	2012

SQL - SFW Queries- Examples

- What are the titles of the movies from 2014?



```
SELECT title
FROM movies
WHERE year = 2014;
```

Table movies

title	director	year
Avatar	James Cameron	2009
The Grand Budapest Hotel	Wes Anderson	2014
Whiplash	Damien Chazelle	2014
Gran Torino	Clint Eastwood	2008
The Godfather	Francis Ford Coppola	1972
Moonrise Kingdom	Wes Anderson	2012

SQL - SFW Queries- Examples

- What are the titles of the movies from 2014?

```
SELECT title
FROM movies
WHERE year = 2014;
```

Table movies

title	director	year
Avatar	James Cameron	2009
The Grand Budapest Hotel	Wes Anderson	2014
Whiplash	Damien Chazelle	2014
Gran Torino	Clint Eastwood	2008
The Godfather	Francis Ford Coppola	1972
Moonrise Kingdom	Wes Anderson	2012

SQL - SFW Queries- Examples

- Who directed *Avatar* and in which year

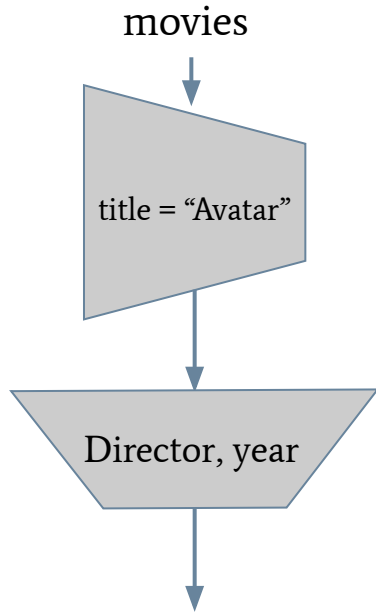
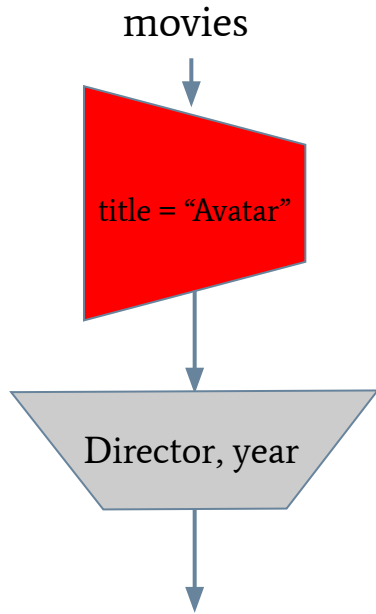


Table movies

title	director	year
Avatar	James Cameron	2009
The Grand Budapest Hotel	Wes Anderson	2014
Whiplash	Damien Chazelle	2014
Gran Torino	Clint Eastwood	2008
The Godfather	Francis Ford Coppola	1972
Moonrise Kingdom	Wes Anderson	2012

SQL - SFW Queries- Examples

- Who directed *Avatar* and in which year



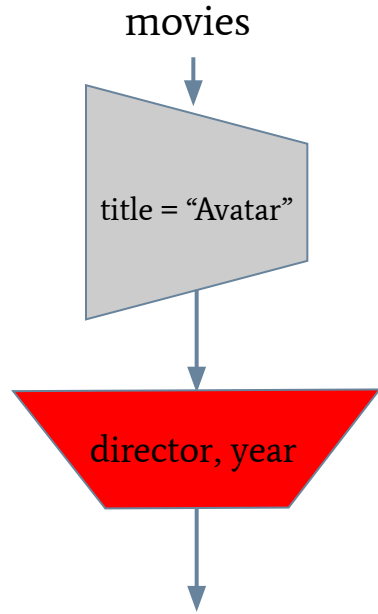
```
SELECT *  
FROM movies  
WHERE title = 'Avatar';
```

Table movies

title	director	year
Avatar	James Cameron	2009
The Grand Budapest Hotel	Wes Anderson	2014
Whiplash	Damien Chazelle	2014
Gran Torino	Clint Eastwood	2008
The Godfather	Francis Ford Coppola	1972
Moonrise Kingdom	Wes Anderson	2012

SQL - SFW Queries- Examples

- Who directed *Avatar* and in which year



```
SELECT director, year
FROM movies
WHERE title = 'Avatar';
```

Table movies

title	director	year
Avatar	James Cameron	2009
The Grand Budapest Hotel	Wes Anderson	2014
Whiplash	Damien Chazelle	2014
Gran Torino	Clint Eastwood	2008
The Godfather	Francis Ford Coppola	1972
Moonrise Kingdom	Wes Anderson	2012

SQL - SFW Queries- Examples

- Who directed Avatar and in which year

```
SELECT director, year
FROM movies
WHERE title = 'Avatar';
```

Table movies

title	director	year
Avatar	James Cameron	2009
The Grand Budapest Hotel	Wes Anderson	2014
Whiplash	Damien Chazelle	2014
Gran Torino	Clint Eastwood	2008
The Godfather	Francis Ford Coppola	1972
Moonrise Kingdom	Wes Anderson	2012

SQL - SFW Queries - Select All

- Get all data about Wes Anderson.

```
SELECT *  
FROM movies  
WHERE director = 'Wes Anderson';
```

* selects all columns.

Table movies

title	director	year
Avatar	James Cameron	2009
The Grand Budapest Hotel	Wes Anderson	2014
Whiplash	Damien Chazelle	2014
Gran Torino	Clint Eastwood	2008
The Godfather	Francis Ford Coppola	1972
Moonrise Kingdom	Wes Anderson	2012

SQL - SFW Queries - Multiple Tables

- Get all movies titles from Damien Chazelle and their note.

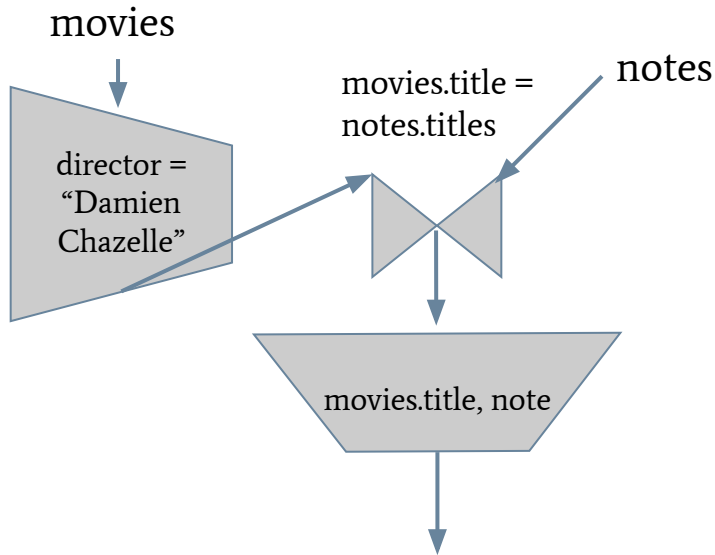


Table movies

title	director	year
Avatar	James Cameron	2009
The Grand Budapest Hotel	Wes Anderson	2014
Whiplash	Damien Chazelle	2014
Gran Torino	Clint Eastwood	2008
The Godfather	Francis Ford Coppola	1972
Moonrise Kingdom	Wes Anderson	2012

Table notes

title	note
Avatar	9
Whiplash	10
Whiplash	8

SQL - SFW Queries - Multiple Tables

- Get all movies titles from Damien Chazelle and their note.

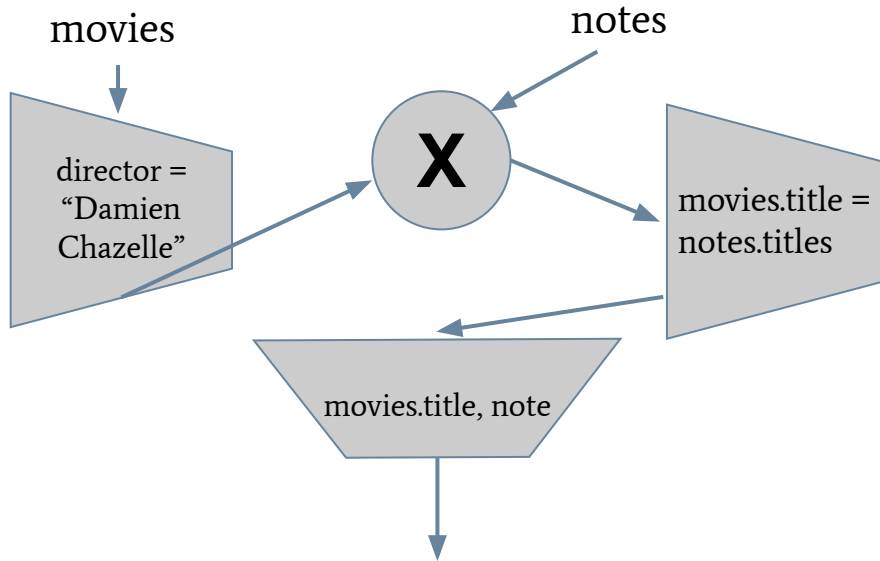


Table movies

title	director	year
Avatar	James Cameron	2009
The Grand Budapest Hotel	Wes Anderson	2014
Whiplash	Damien Chazelle	2014
Gran Torino	Clint Eastwood	2008
The Godfather	Francis Ford Coppola	1972
Moonrise Kingdom	Wes Anderson	2012

Table notes

title	note
Avatar	9
Whiplash	10
Whiplash	8

SQL - SFW Queries - Multiple Tables

- Get all movies titles from Damien Chazelle and their note.

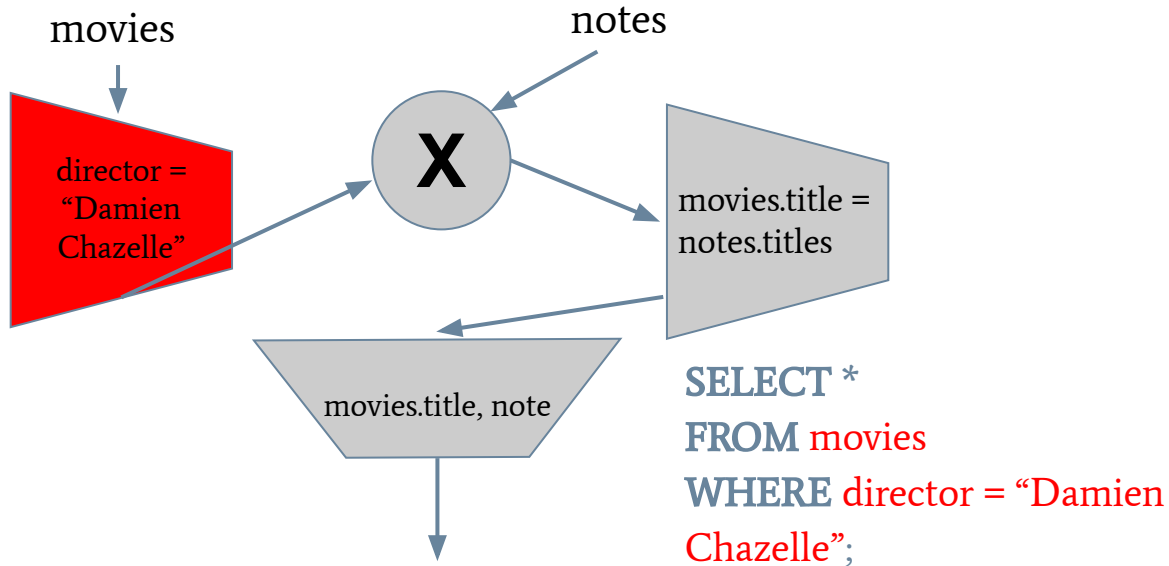


Table movies

title	director	year
Avatar	James Cameron	2009
The Grand Budapest Hotel	Wes Anderson	2014
Whiplash	Damien Chazelle	2014
Gran Torino	Clint Eastwood	2008
The Godfather	Francis Ford Coppola	1972
Moonrise Kingdom	Wes Anderson	2012

Table notes

title	note
Avatar	9
Whiplash	10
Whiplash	8

SQL - SFW Queries - Multiple Tables

- Get all movies titles from Damien Chazelle and their note.

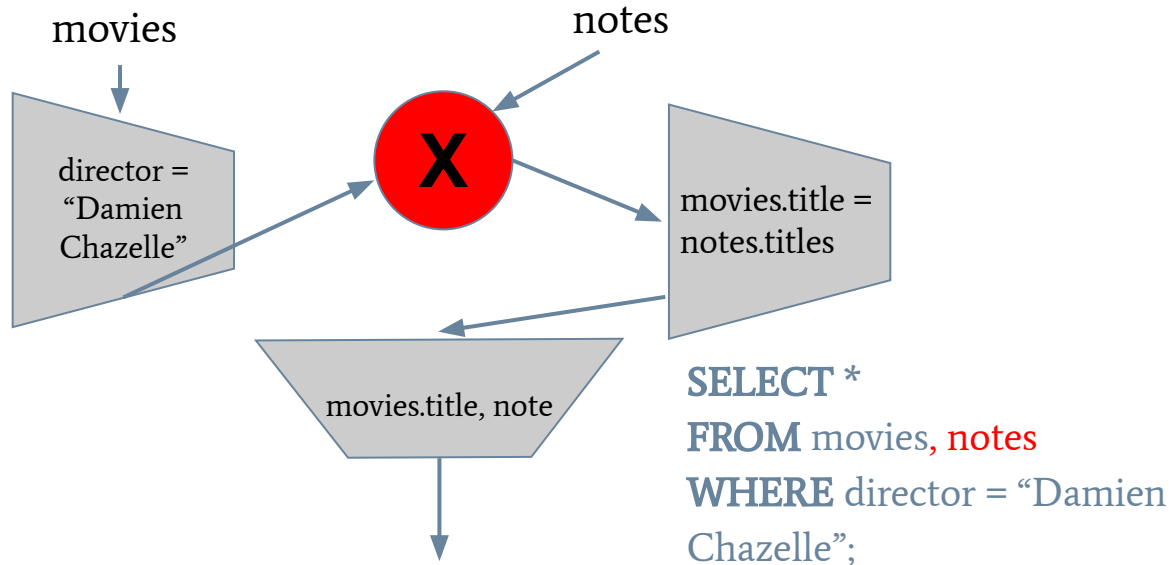


Table movies

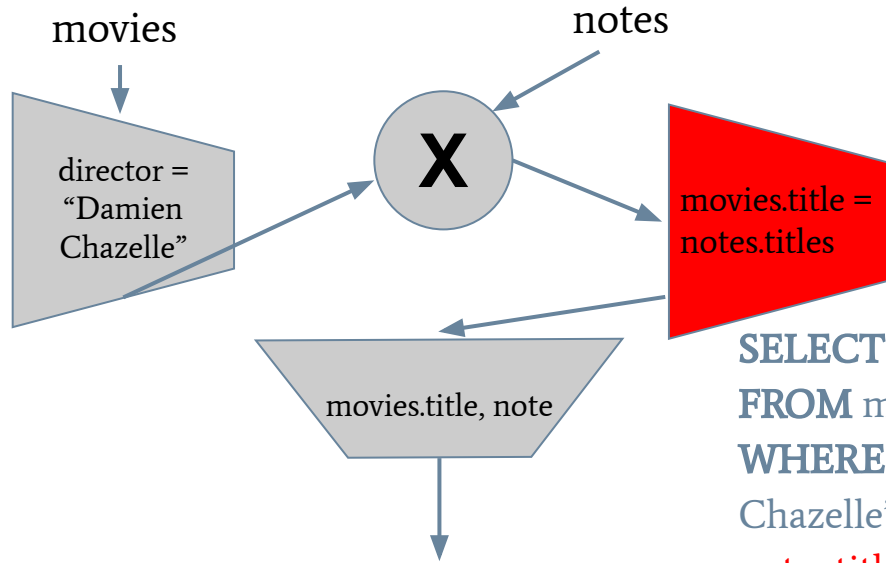
title	director	year
Avatar	James Cameron	2009
The Grand Budapest Hotel	Wes Anderson	2014
Whiplash	Damien Chazelle	2014
Gran Torino	Clint Eastwood	2008
The Godfather	Francis Ford Coppola	1972
Moonrise Kingdom	Wes Anderson	2012

Table notes

title	note
Avatar	9
Whiplash	10
Whiplash	8

SQL - SFW Queries - Multiple Tables

- Get all movies titles from Damien Chazelle and their note.



```
SELECT *  
FROM movies, notes  
WHERE director = "Damien  
Chazelle" AND movies.title =  
notes.title;
```

Table movies

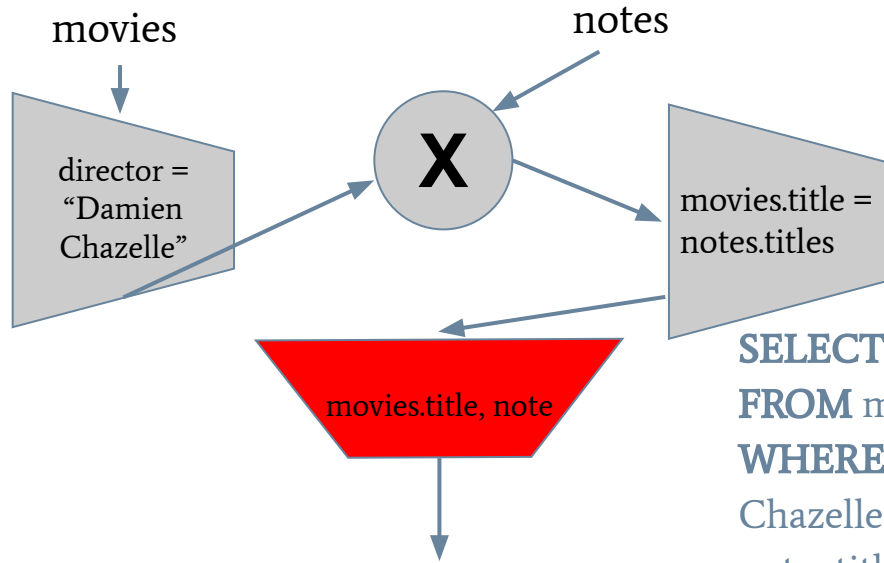
title	director	year
Avatar	James Cameron	2009
The Grand Budapest Hotel	Wes Anderson	2014
Whiplash	Damien Chazelle	2014
Gran Torino	Clint Eastwood	2008
The Godfather	Francis Ford Coppola	1972
Moonrise Kingdom	Wes Anderson	2012

Table notes

title	note
Avatar	9
Whiplash	10
Whiplash	8

SQL - SFW Queries - Multiple Tables

- Get all movies titles from Damien Chazelle and their note.



```
SELECT movies.title, note
FROM movies, notes
WHERE director = "Damien Chazelle" AND movies.title = notes.title;
```

Table movies

title	director	year
Avatar	James Cameron	2009
The Grand Budapest Hotel	Wes Anderson	2014
Whiplash	Damien Chazelle	2014
Gran Torino	Clint Eastwood	2008
The Godfather	Francis Ford Coppola	1972
Moonrise Kingdom	Wes Anderson	2012

Table notes

title	note
Avatar	9
Whiplash	10
Whiplash	8

SQL - SFW Queries - Multiple Tables

- Get all movies titles from Damien Chazelle and their note.

```
SELECT movies.title, note
FROM movies, notes
WHERE director = "Damien Chazelle" AND
movies.title = notes.title;
```

When the name of a column appears in two tables, we can mention explicitly the table.

Table movies

title	director	year
Avatar	James Cameron	2009
The Grand Budapest Hotel	Wes Anderson	2014
Whiplash	Damien Chazelle	2014
Gran Torino	Clint Eastwood	2008
The Godfather	Francis Ford Coppola	1972
Moonrise Kingdom	Wes Anderson	2012

Table notes

title	note
Avatar	9
Whiplash	10
Whiplash	8

SQL - SFW Queries - Conditions in WHERE

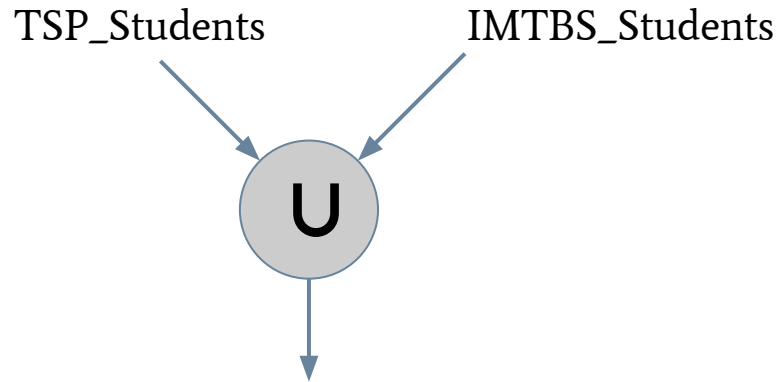
- Comparison operators:
 - Classical: =, <> and != (both for “different”), <, >, <=, >=, IS NULL, IS NOT NULL
 - **IN**: Value in a list
 - **SELECT * FROM** movies **WHERE** director **IN** ('Damien Chazelle', 'Wes Anderson');
 - **BETWEEN**: Values in an interval (limits included)
 - **SELECT * FROM** movies **WHERE** year **BETWEEN** 2012 **AND** 2014;
 - **LIKE**: Value follows a pattern
 - % represents any string (may be empty)
 - _ represents any single character.
 - Example: 'The %' represents all strings starting by “The “ like “The Godfather”
 - **SELECT * FROM** movies **WHERE** title **LIKE** 'The %';
- Logical operators:
 - **AND, OR, NOT**

Set Operators

The set operators from the relational algebra have equivalents in SQL.

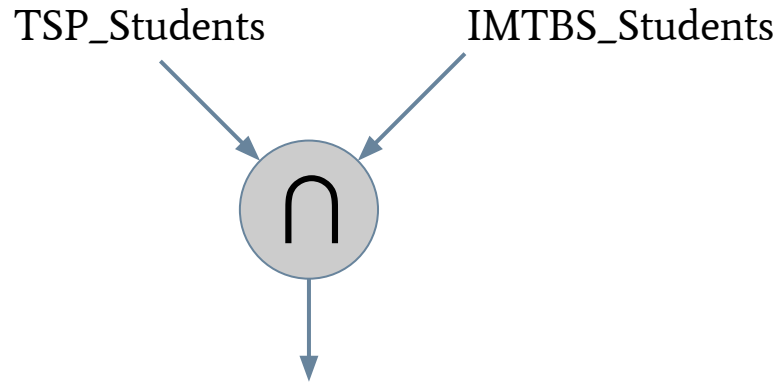
- **UNION** \leftrightarrow **SELECT * FROM T1 UNION SELECT * FROM T2**
- **INTERSECTION** \leftrightarrow **SELECT * FROM T1 INTERSECT SELECT * FROM T2**
 - or **SELECT value FROM T1 WHERE value IN (SELECT value FROM T2);**
- **DIFFERENCE** \leftrightarrow **SELECT * FROM T1 EXCEPT SELECT * FROM T2**
 - or **SELECT value FROM T1 WHERE value NOT IN (SELECT value FROM T2);**
 - Sometimes **MINUS** instead of **EXCEPT**

Set Operators - Union



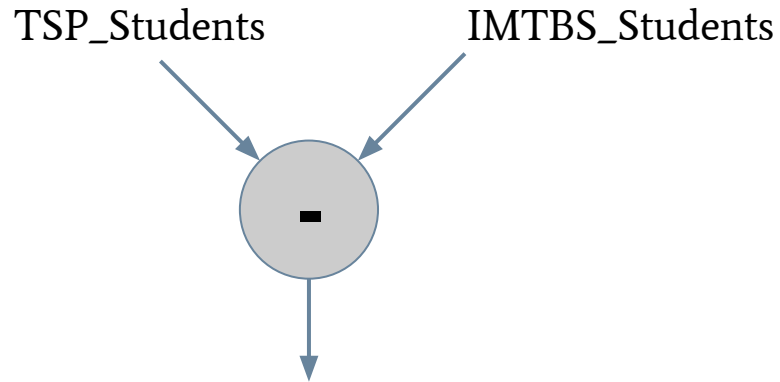
```
SELECT *  
FROM TSP_Students  
UNION  
SELECT *  
FROM IMTBS_Students
```

Set Operators - Intersection



```
SELECT *  
FROM TSP_Students  
INTERSECT  
SELECT *  
FROM IMTBS_Students
```

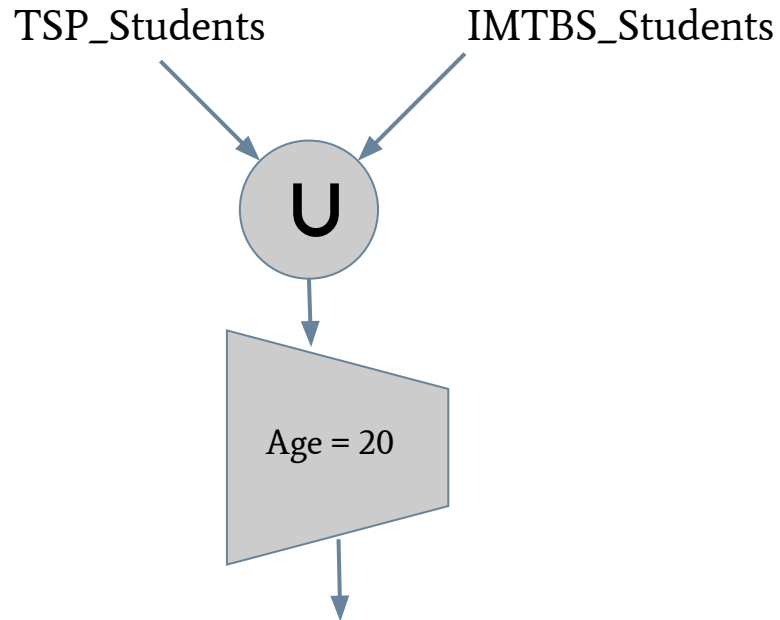
Set Operators - Difference



```
SELECT *  
FROM TSP_Students  
EXCEPT  
SELECT *  
FROM IMTBS_Students
```

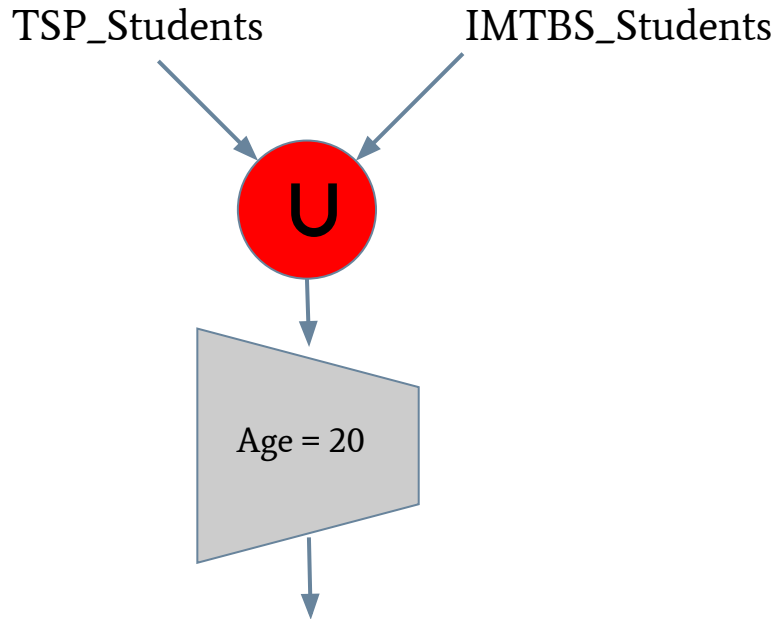
Set Operators - Example

Get the list of students from TSP or IMTBS who are 20 years old



Set Operators - Example

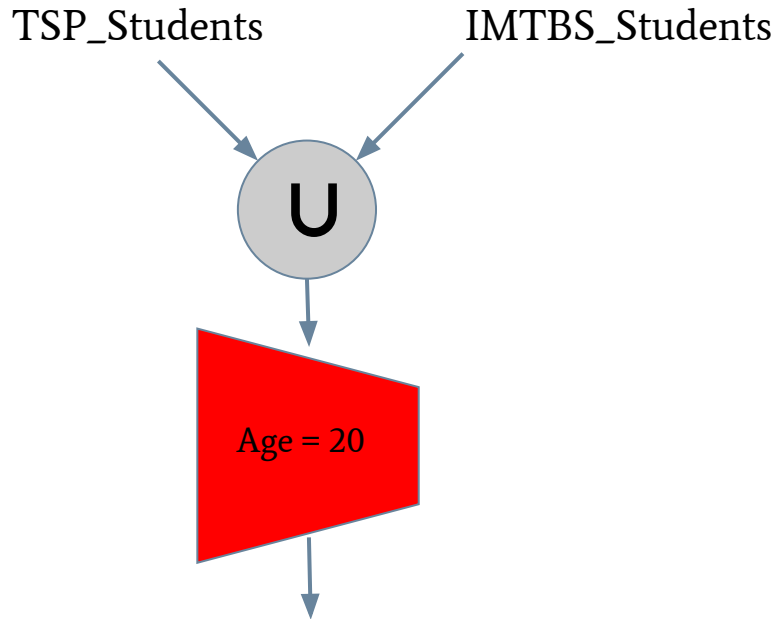
Get the list of students from TSP or IMTBS who are 20 years old



```
SELECT *  
FROM TSP_Students  
UNION  
SELECT *  
FROM IMTBS_Students
```


Set Operators - Example

Get the list of students from TSP or IMTBS who are 20 years old



```
SELECT *  
FROM TSP_Students  
WHERE Age = 20  
UNION  
SELECT *  
FROM IMTBS_Students  
WHERE Age = 20
```

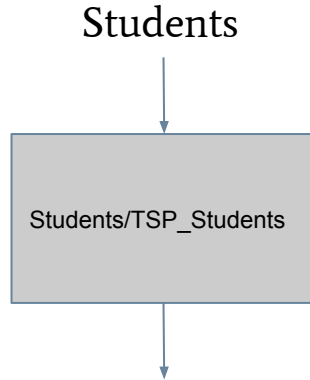
SQL - Renaming

With SQL, it is possible to rename tables and columns with the keyword **AS**

Examples:

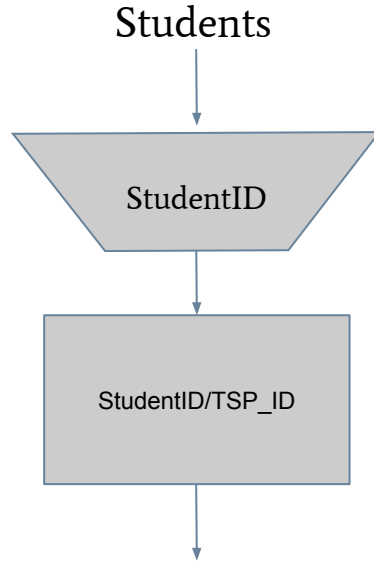
- **SELECT** title **AS** titre **FROM** movies;
- **SELECT** M1.title, M2.title **FROM** movies **AS** M1, movies **AS** M2;

SQL - Table Renaming



```
SELECT *  
FROM Students AS TSP_Students
```

SQL - Column Renaming



```
SELECT StudentID AS TSP_ID  
FROM Students
```

How To Convert From Relational Algebra To SQL

- Read the diagram from the top to the bottom. If you read:
 - A **Project**, change the previous list of attributes with the new one in the **SELECT**
 - A **Selection**, add the condition to the **WHERE**. If there was no **WHERE**, simply add it. Otherwise, add a **AND** and add the new condition.
 - A **Cartesian Product**, add the table names in the **FROM**. **/!** The new attributes list in the **SELECT** is the equal to the two inputs attributes lists (e.g: table1.StudentID, table1.year X table2.StudentID, table2.grade = table1.StudentID, table1.year, table2.StudentID, table2.grade). If both are *****, you can keep it, otherwise, you need to transform the ***** to the real list of attributes.
 - A **Join**, perform the transformation to a Cartesian Product and a Selection, i.e. add both the table to the **FROM** and the condition to the **WHERE**
 - A **Table Rename**, add the renaming in the **FROM** with **AS**
 - A **Column Rename**, add the renaming in the **SELECT** with **AS**
 - A **Set Operator**, add a **UNION/INTERSECT/EXCEPT** between the SFW queries of the two input tables. **/!** The next symbols need to be applied to all SFW queries.

Summary

```
SELECT <list of attributes/columns to select + AS>  
FROM <list of tables to consider + AS>  
[WHERE <condition without aggregation>];  
  
+ UNION, INTERSECT, EXCEPT
```