INTRODUCTION TO SQL

BACKEND-WORKSHOP

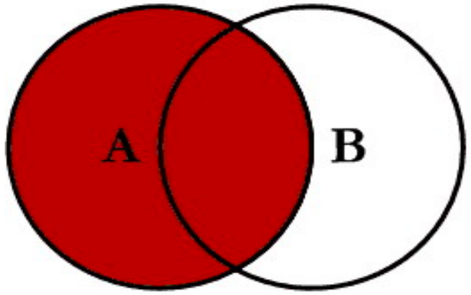
Michael Fröhlich - michael-froehlich@cdtm.de

Tobias Dümmling - tobias.duemmling@cdtm.de

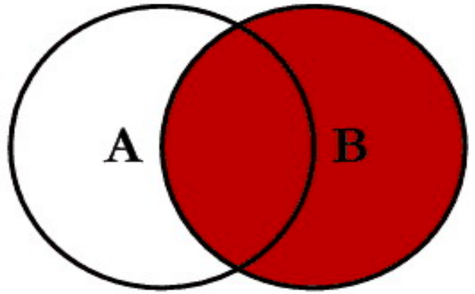
HOW TO GET DATA FROM DIFFERENT TABLES?

- E.g. All lists owned by a user OR where a user is a collaborator

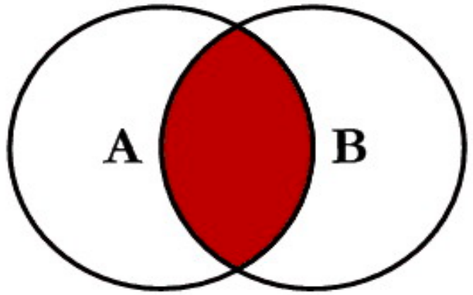
SQL JOINS



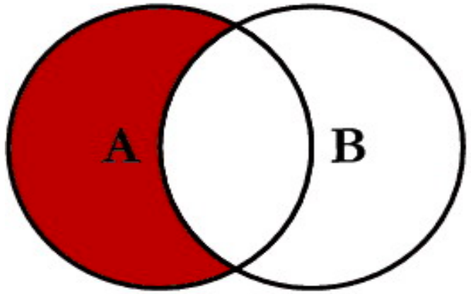

```
SELECT <select_list>
FROM TableA A
LEFT JOIN TableB B
ON A.Key = B.Key
```



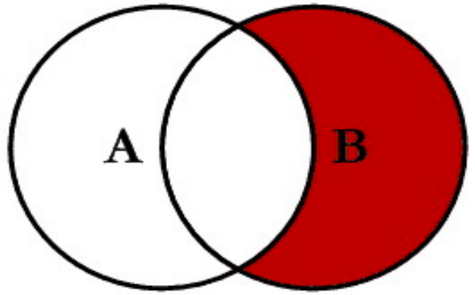

```
SELECT <select_list>
FROM TableA A
RIGHT JOIN TableB B
ON A.Key = B.Key
```



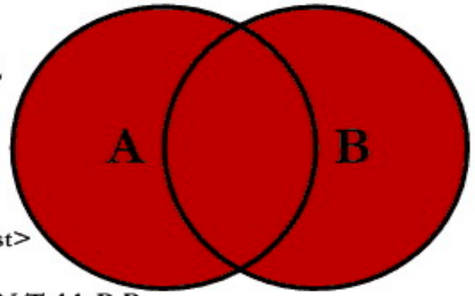

```
SELECT <select_list>
FROM TableA A
INNER JOIN TableB B
ON A.Key = B.Key
```



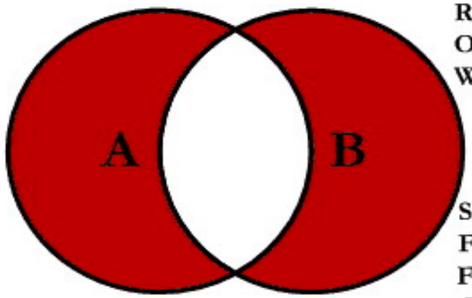

```
SELECT <select_list>
FROM TableA A
LEFT JOIN TableB B
ON A.Key = B.Key
WHERE B.Key IS NULL
```




```
SELECT <select_list>
FROM TableA A
RIGHT JOIN TableB B
ON A.Key = B.Key
WHERE A.Key IS NULL
```




```
SELECT <select_list>
FROM TableA A
FULL OUTER JOIN TableB B
ON A.Key = B.Key
```




```
SELECT <select_list>
FROM TableA A
FULL OUTER JOIN TableB B
ON A.Key = B.Key
WHERE A.Key IS NULL
OR B.Key IS NULL
```


SQL Joins explained in detail: http://www.sql-join.com/

HOW TO GET DATA FROM DIFFERENT TABLES?

- Joins help to access data stored in different tables
- They merge tables based on a defined condition
- For this workshop we will focus on the left outer join

THE JOIN COMMAND

matrnr	name	semester
24002	Xenokrates	18
25403	Jonas	12
26120	Fichte	10
26830	Aristoxenos	8
27550	Schopenhauer	6
28106	Carnap	3
29120	Theophrastos	2
29555	Feuerbach	2

matrnr	vorlnr	persnr	note
28106	5001	2126	1.0
25403	5041	2125	2.0
27550	4630	2137	2.0

- Required is a list of every student enrolled containing his name. For the ones that check exams the grade should also be added to the table.

LEFT OUTER JOIN

- Combines two tables based on a common Attribute

<AttrName> = Attribute Name (use * to get all attributes of a table)
<TableName> = Name of the Table
<Conditions> = Define the Attribute the tables are joined on
<Direction> = ASC (ascending) or DESC(descending)

```
SELECT  <AttrName1>, <AttrName2>
FROM <TableName1> LEFT JOIN <TableName2>
ON <Conditions>
ORDER BY <AttrName> <Direction>
```

```
SELECT name, note
FROM Studenten LEFT JOIN pruefen
ON studenten.matrnr = pruefen.matrnr
ORDER BY studenten.name;
```

LEFT OUTER JOIN – THE RESULT

name	note
Aristoxenos	null
Carnap	1.0
Feuerbach	null
Fichte	null
Jonas	2.0
Schopenhauer	2.0
Theophrastos	null
Xenokrates	null

matrnr	vorlnr	persnr	note
28106	5001	2126	1.0
25403	5041	2125	2.0
27550	4630	2137	2.0

matrnr	name	semester
24002	Xenokrates	18
25403	Jonas	12
26120	Fichte	10
26830	Aristoxenos	8
27550	Schopenhauer	6
28106	Carnap	3
29120	Theophrastos	2
29555	Feuerbach	2


A SQL query walks up to two tables in a restaurant and asks: “Mind if I join you?”