

SQL 101

(cos everyone loves a DBA)

Such table.col1, table.col2

So table

Many join

Much group by

Wow



History of RDBMS

Relational **D**ata**B**ase **M**anagement **S**ystem

Invented in 1970 (ish) at IBM

Most popular type of database

Commercial



Open Source





20 years old (first release 1995)

second most widely used RDBMS

most widely used open-source RDBMS

written in C and C++

cross platform

open source (GPL v2)

Core Terms

- Server
- Databases / Schemas
- Tables
- Columns
- Rows
- Fields
- Users
- Permissions & Grants
- Root

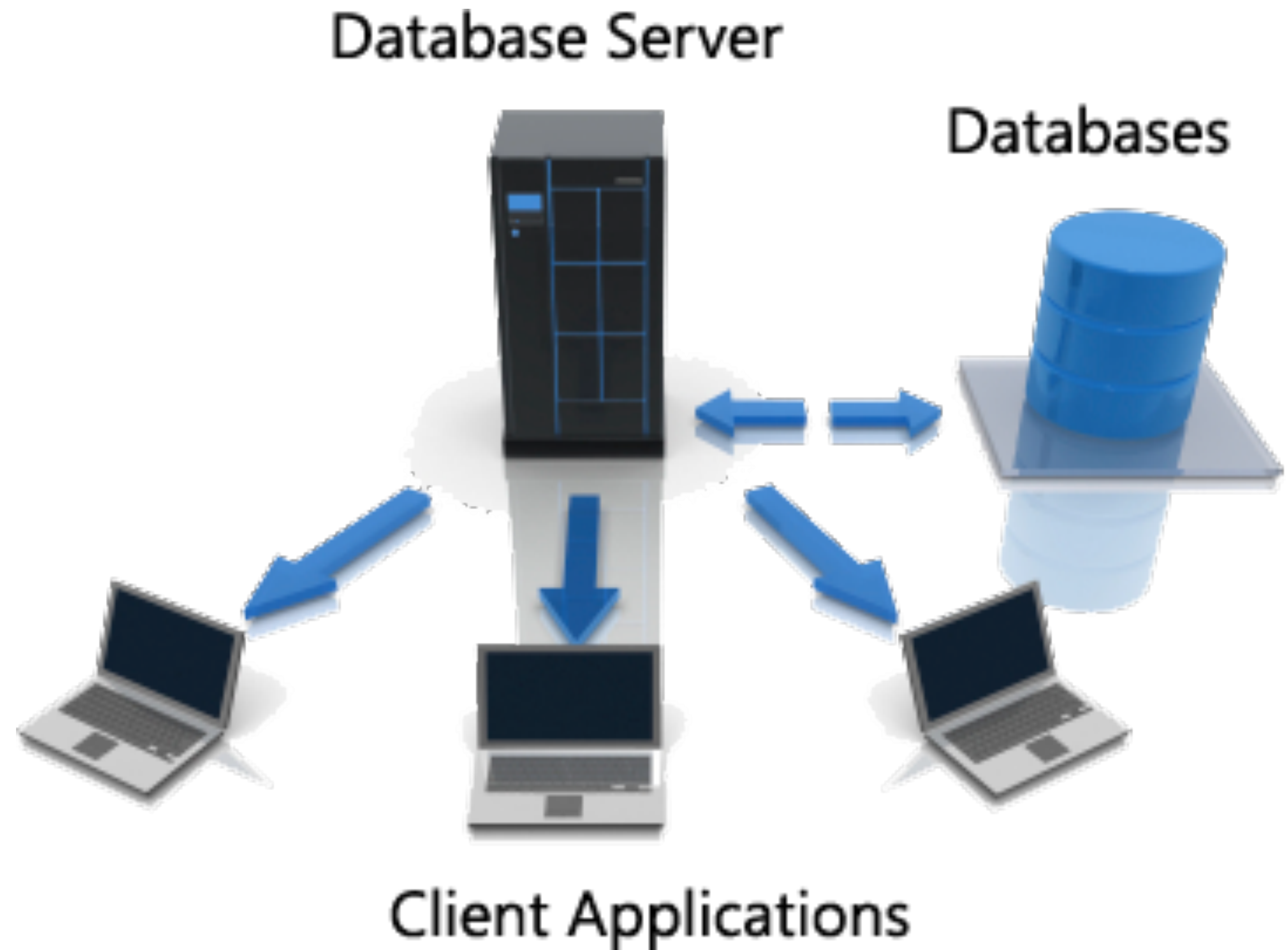
Server & Databases / Schemas

One server has many Databases

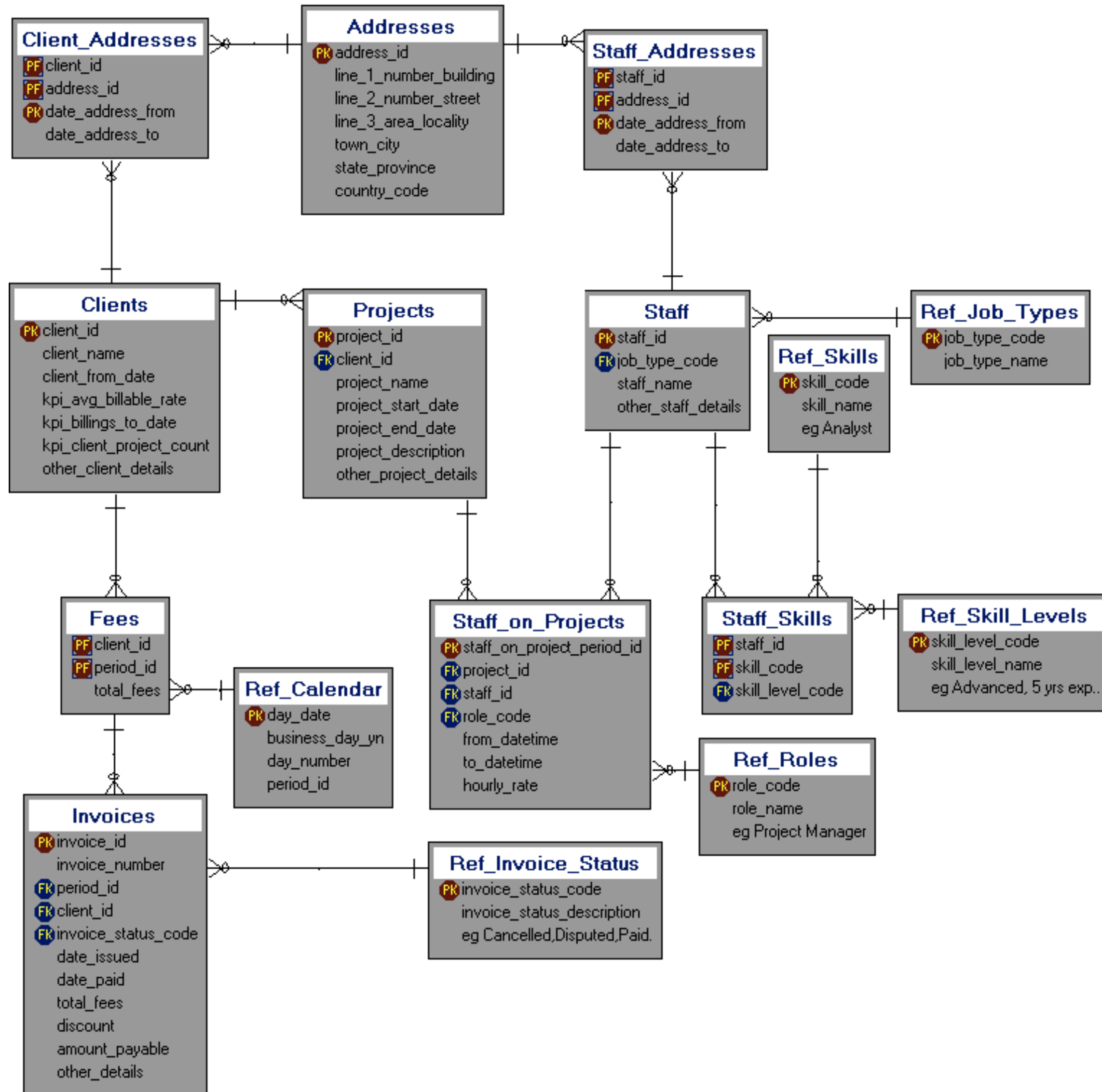
Also known as Schemas

User Schemas (yours)

System Schemas (don't touch!)



Tables



One Schema has many Tables

Tables are like an Excel Sheet

Columns & Rows

Big Schemas can have 100s of Tables

Columns, Rows & Fields

Each Table has Many Columns

And many Rows

One row per data item

Row & Column is a Field
not a cell! This isn't Excel

Each Column has a specific Type
and other properties

First Name	Last Name	Address	City	Age
Mickey	Mouse	123 Fantasy Way	Anaheim	73
Bat	Man	321 Cavern Ave	Gotham	54
Wonder	Woman	987 Truth Way	Paradise	39
Donald	Duck	555 Quack Street	Mallard	65
Bugs	Bunny	567 Carrot Street	Rascal	58
Wiley	Coyote	999 Acme Way	Canyon	61
Cat	Woman	234 Purrfect Street	Hairball	32
Tweety	Bird	543	Itotltaw	28

Users & Permissions & Grants

Very granular control over user access

Users can be restricted by:

- Host (IP Address)
- Database
- Table
- Privilege

Root user == god on the system

Do NOT use root!

<input checked="" type="checkbox"/> ALL PRIVILEGES	
<input checked="" type="checkbox"/> ALTER	<input checked="" type="checkbox"/> CREATE
<input checked="" type="checkbox"/> CREATE ROUTINE	<input checked="" type="checkbox"/> CREATE TEMPORARY TABLES
<input checked="" type="checkbox"/> CREATE VIEW	<input checked="" type="checkbox"/> DELETE
<input checked="" type="checkbox"/> DROP	<input checked="" type="checkbox"/> EXECUTE
<input checked="" type="checkbox"/> INDEX	<input checked="" type="checkbox"/> INSERT
<input checked="" type="checkbox"/> LOCK TABLES	<input checked="" type="checkbox"/> REFERENCES
<input checked="" type="checkbox"/> SELECT	<input checked="" type="checkbox"/> SHOW VIEW
<input checked="" type="checkbox"/> TRIGGER	<input checked="" type="checkbox"/> UPDATE

Data Types

- Text
- Numeric
- Dates & Times
- Nulls
- Auto Increment
- Keys

Data Types - Text

Char: Fixed Length Mixed Chars 123

Varchar: Variable Length Mixed Chars 123

Text: Long blocks of text

Blob: Long blocks of binary data (aka byte strings)

Data Types - Numeric

TinyInt / Int / BigInt: whole integers

Decimal / Numeric: exact decimal values to specified places

Float / Double: 4 / 8 byte precision floating point

Signed Unsigned = can have negative sign or not

Data Types - Integer Types

Type	Storage	Minimum Value	Maximum Value
	(Bytes)	(Signed/Unsigned)	(Signed/Unsigned)
TINYINT	1	-128	127
		0	255
SMALLINT	2	-32768	32767
		0	65535
MEDIUMINT	3	-8388608	8388607
		0	16777215
INT	4	-2147483648	2147483647
		0	4294967295
BIGINT	8	-9223372036854775808	9223372036854775807
		0	18446744073709551615

Data Types - Dates & Time

Date: 2015-06-14

Time: 21:32

Year: 2015 (or 15 which is bad!)

Datetime: 2015-06-14 21:32

Timestamp: 2015-06-14 21:32 (UTC)

PUBLIC SERVICE ANNOUNCEMENT:

OUR DIFFERENT WAYS OF WRITING DATES AS NUMBERS CAN LEAD TO ONLINE CONFUSION. THAT'S WHY IN 1988 ISO SET A GLOBAL STANDARD NUMERIC DATE FORMAT.

THIS IS ***THE*** CORRECT WAY TO WRITE NUMERIC DATES:

2013-02-27


THE FOLLOWING FORMATS ARE THEREFORE DISCOURAGED:

02/27/2013 02/27/13 27/02/2013 27/02/13

20130227 2013.02.27 27.02.13 27-02-13

27.2.13 2013.II.27. $27\frac{1}{2}$ -13 2013.158904109

MMXIII-II-XXVII MMXIII $\frac{\text{LVII}}{\text{CCCLXV}}$ 1330300800

$((3+3) \times (111+1) - 1) \times 3 / 3 - 1 / 3^3$ ~~2013~~  Hissss

10/11011/1101 02/27/20/13 $\begin{matrix} 2 & 3 & 1 & 4 \\ 0 & 1 & 2 & 3 & 7 \\ 5 & 6 & 7 & 8 \end{matrix}$

Nulls, Auto Increment and Keys

Nulls

An empty value (if the column allows or is nullable)

Not the same as "", 0, false or 'null'

Auto Increment

An int type that counts up by one each time

Primary Key

The main ID for a table - often an int with auto increment
(also creates an index on the column)

Foreign Key

A column in table that references the Primary Key in another

SQL

Structured **Q**uery **L**angauge

Query
SELECT

Manipulation
INSERT, UPDATE, DELETE

Joins
INNER, LEFT

SQL

Structured **Q**uery **L**angauge

Query
SELECT

Manipulation
INSERT, UPDATE, DELETE

Joins
INNER, LEFT

Anatomy of a SQL Statement / Query

Command	SELECT
Query Scope	first_name, last_name, date_of_birth
Table Selector	FROM people
Conditions	WHERE home_town = 'St. Helier'

Anatomy of a SQL Statement / Query

Command		SELECT
Query Scope		*
Table Selector		FROM people
Conditions		

Anatomy of a SQL Statement / Query

Command		UPDATE
Table Selector		people
Query Assignment		SET last_name = 'Crapaud'
Conditions		WHERE home_town = 'St. Helier'

Clients



MySQL Workbench
(Cross Platform)



Sequel Pro
(OSX)



Heidi SQL
(Windows)

Let's play!

Host: 178 . 62 . 245 . 199

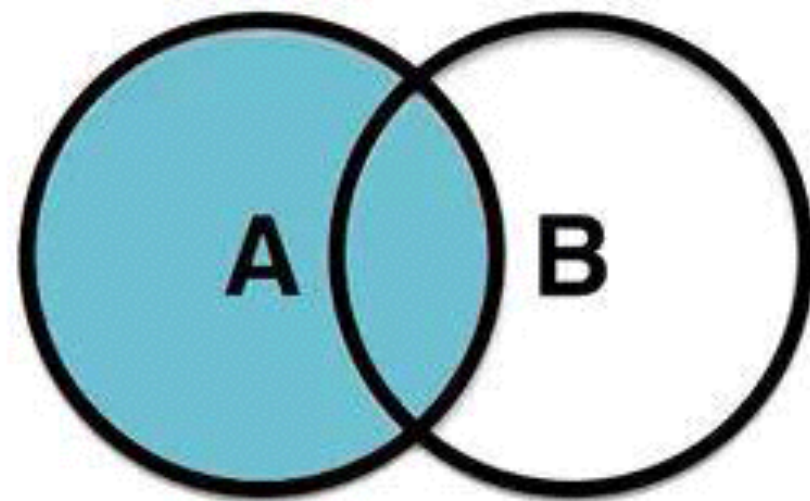
Port: 3306

User: dj1cpstudent

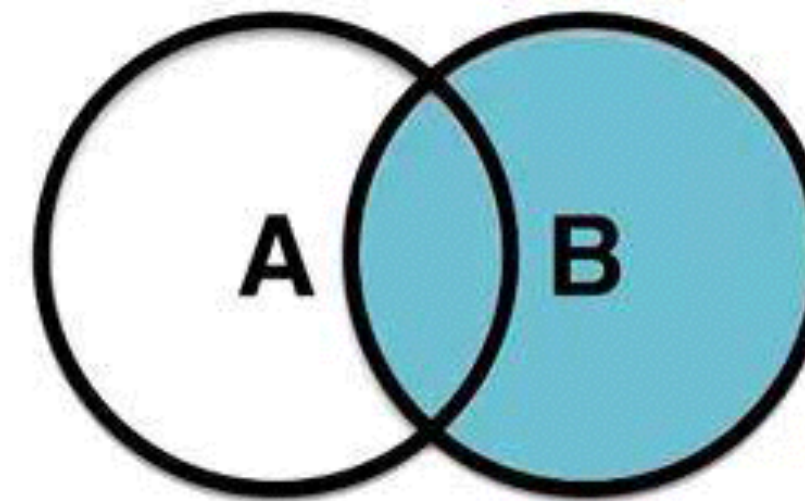
Pass: sqlstudy.2016

D/B : sakila

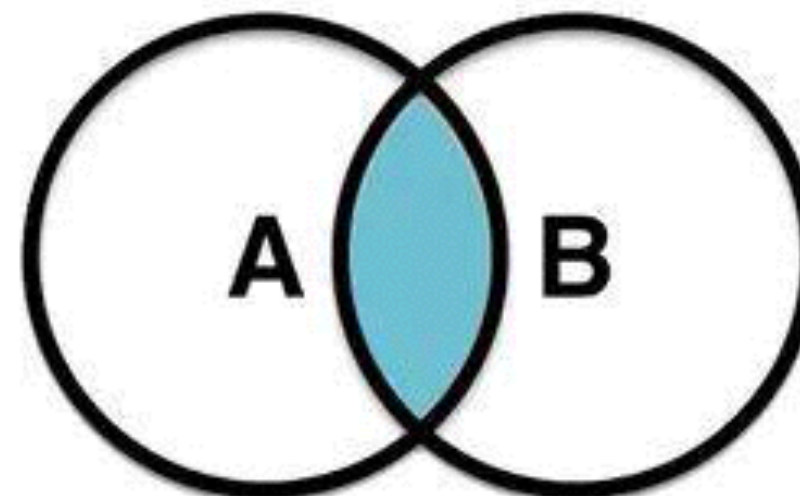
SQL JOINS



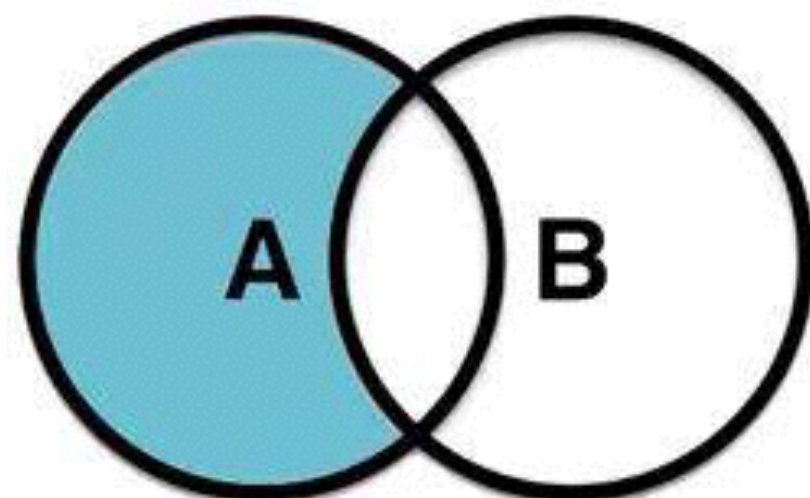
```
SELECT <fields list>  
FROM TableA A  
LEFT JOIN TableB B  
ON A.Key = B.Key
```



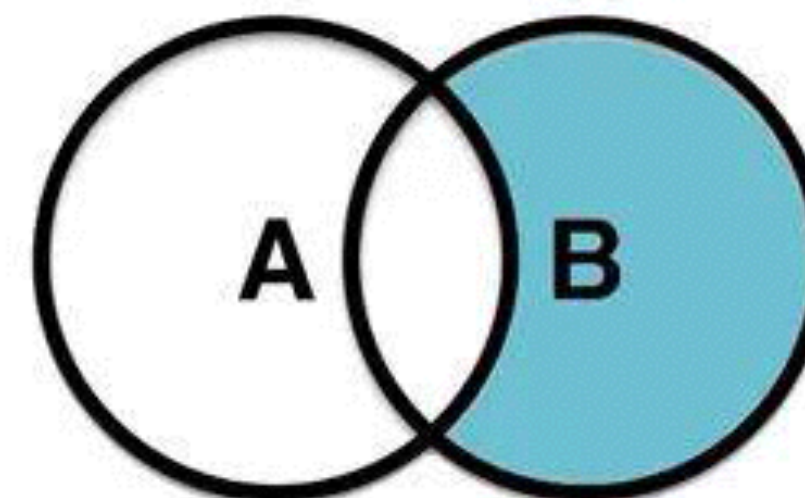
```
SELECT <fields list>  
FROM TableA A  
RIGHT JOIN TableB B  
ON A.Key = B.Key
```



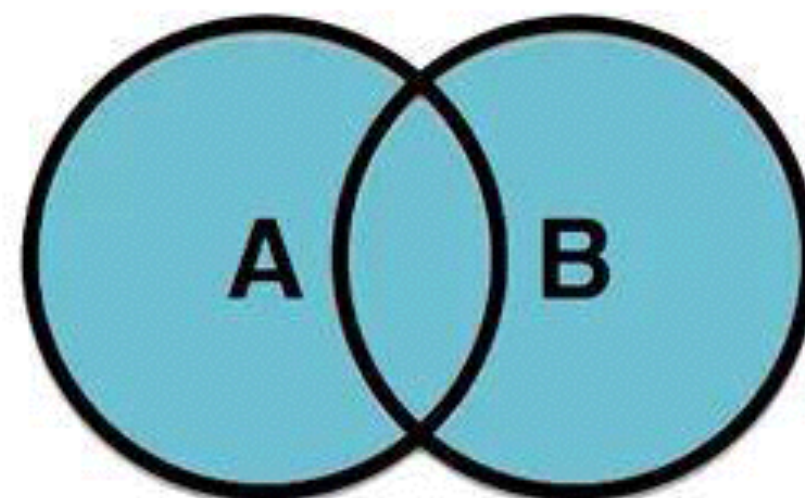
```
SELECT <fields list>  
FROM TableA A  
INNER JOIN TableB B  
ON A.Key = B.Key
```



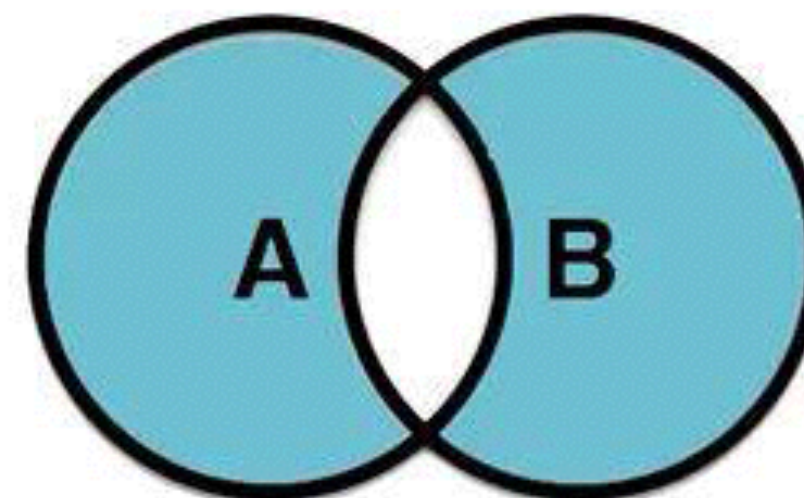
```
SELECT <fields list>  
FROM TableA A  
LEFT JOIN TableB B  
ON A.Key = B.Key  
WHERE B.Key IS NULL
```



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



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



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