The Great Summary

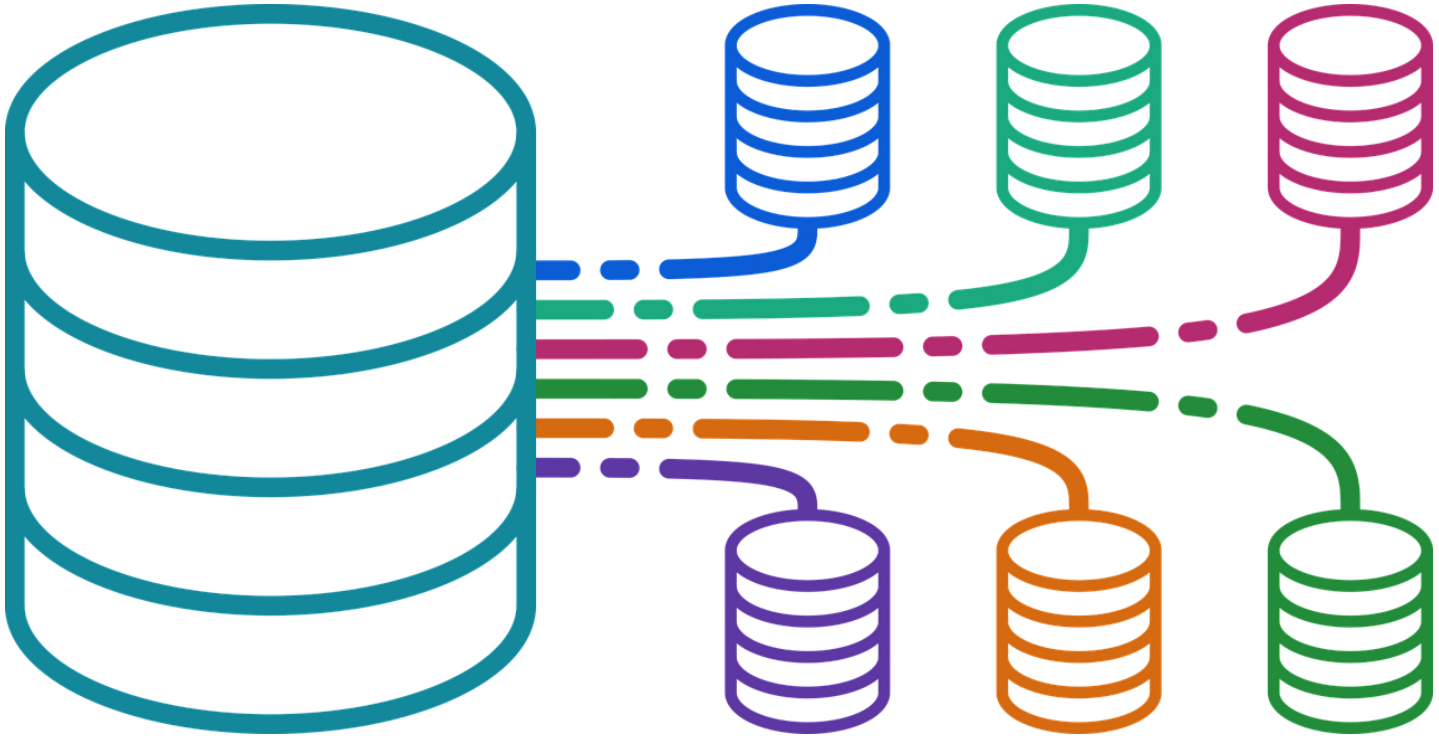
CMPSC 305 – Database Systems

ALLEGHENY COLLEGE

What has this class covered?

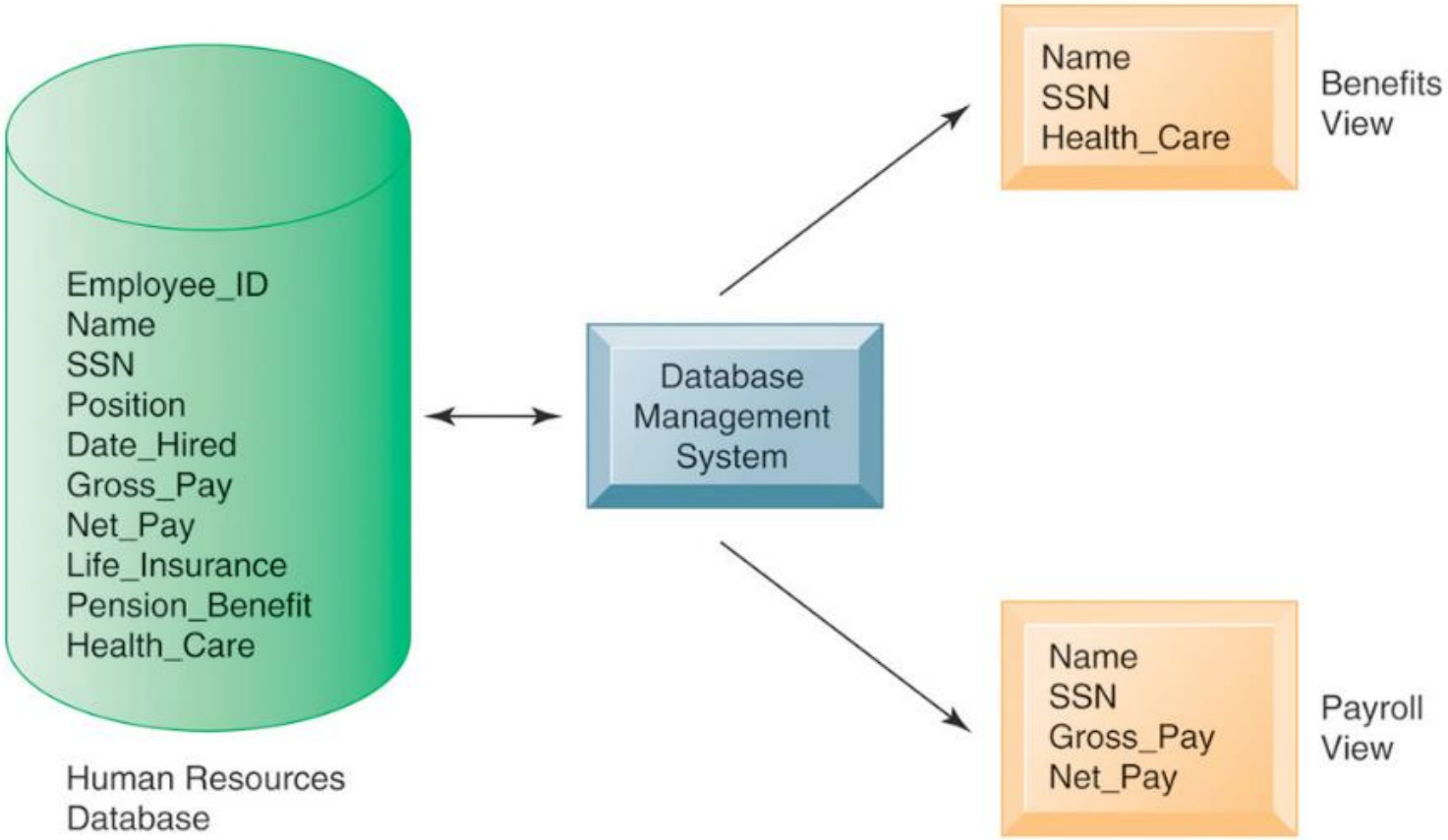
- Some of the fundamental theory and methods behind modern databases systems
    - SQL: Sqlite3, DBBrowser
    - NoSQL: Mongo,
    - Graphical: Neo4J
- Building schemas with integrity constrains for data management
- Manipulating data, populating bases and extracted out filtered information
- Programming queries across all DB systems
- Management and Automation: Programming for abstraction
- How to pull information (knowledge) from raw data

SQL Databases

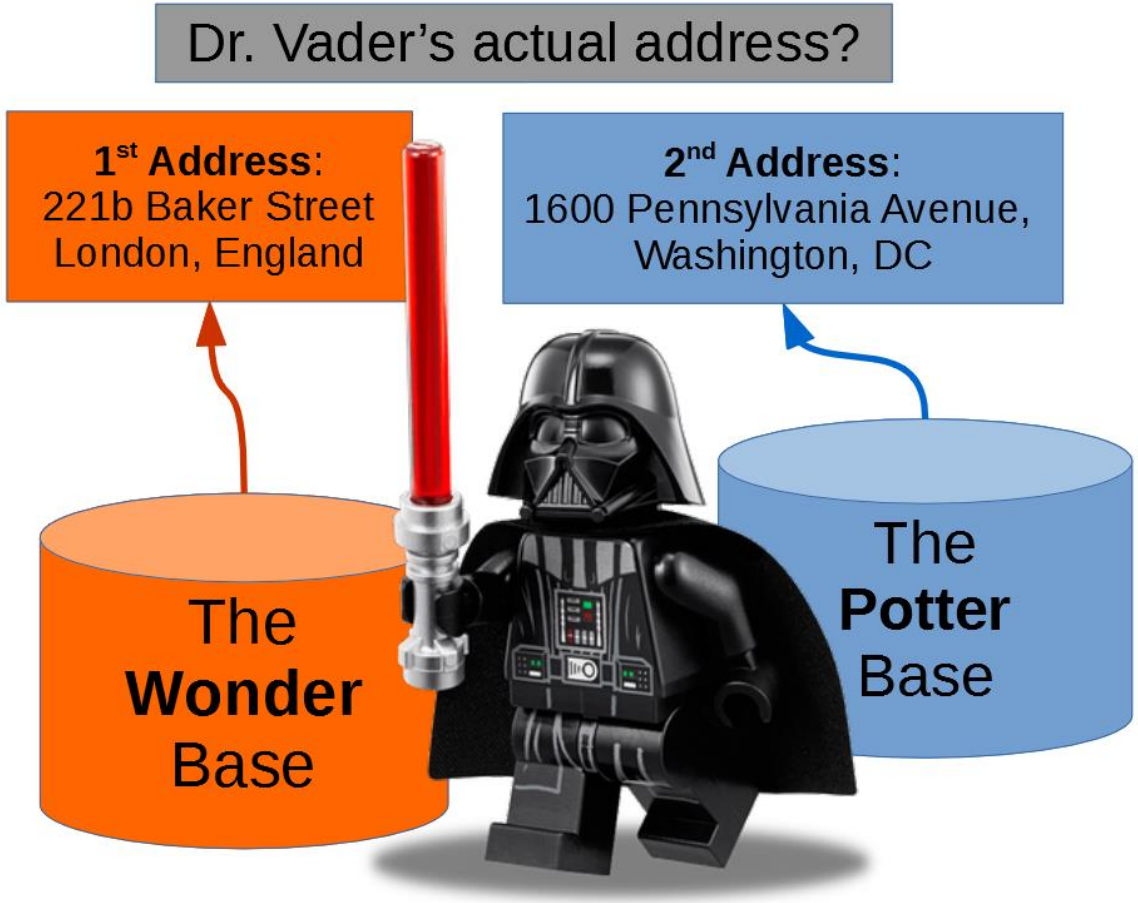


- What is the function of a database?

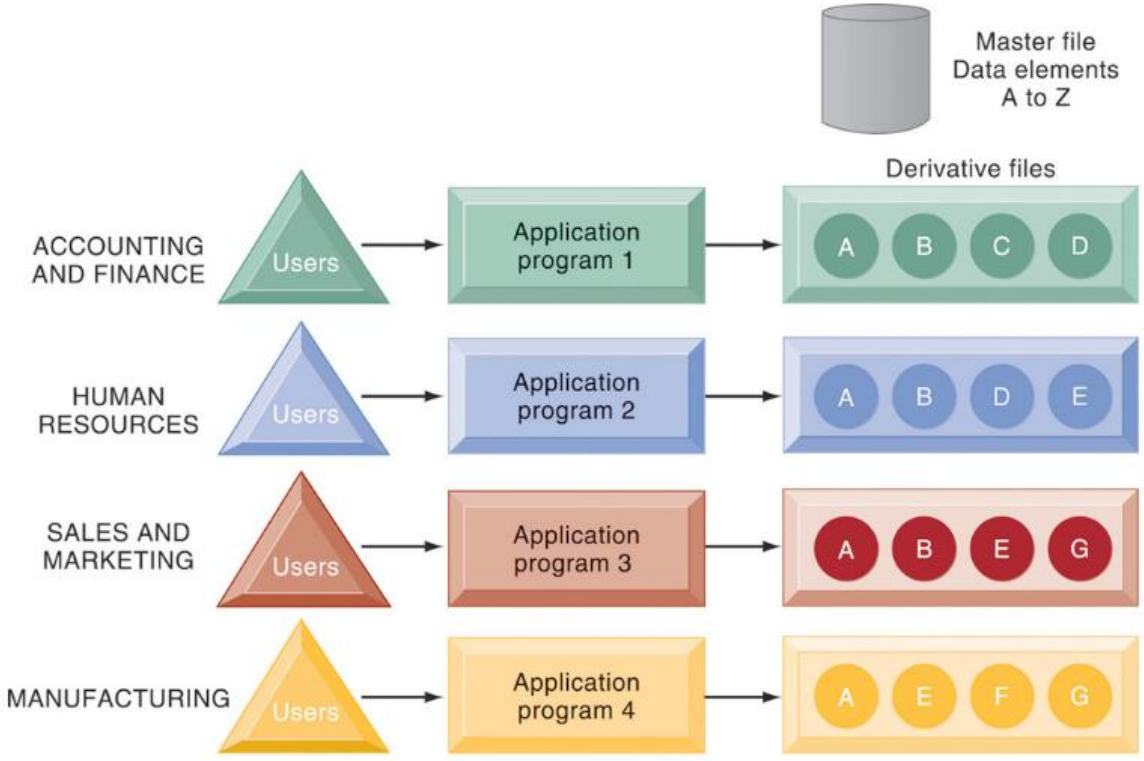
To Connect Data



Data Disconnections at Batman and Associates
Two different address? :-(



How Many Databases Do You Need??



- A firm may have managed several information sources at the same time
- Anything wrong with disconnecting databases?

Specific Information For Each Table

Combine and connect the data in one base

ID	*name*	*dept_name*	*salary*
22222	Einstein	Physics	95000
12121	Wu	Finance	90000
32343	El Said	History	60000
45565	Katz	Comp. Sci.	75000
98345	Kim	Elec. Eng.	80000
76766	Crick	Biology	72000
10101	Srinivasan	Comp. Sci.	65000
58583	Califieri	History	62000
83821	Brandt	Comp. Sci.	92000
15151	Mozart	Music	40000
33456	Gold	Physics	87000
76543	Singh	Finance	80000

(a) The *instructor* table

dept_name	*building*	*budget*
Comp. Sci.	Taylor	100000
Biology	Watson	90000
Elec. Eng.	Taylor	85000
Music	Packard	80000
Finance	Painter	120000
History	Painter	50000
Physics	Watson	70000

(b) The *department* table

- Specific tables for types of data

- Pronounced “ess-que-el” stands for Structured Query Language.
- Used to communicate with a database.
- According to ANSI (American National Standards Institute), it is the standard language for relational database management systems.
- The standard computer language for relational database management and data manipulation.
    - Used to query, insert, update and modify data

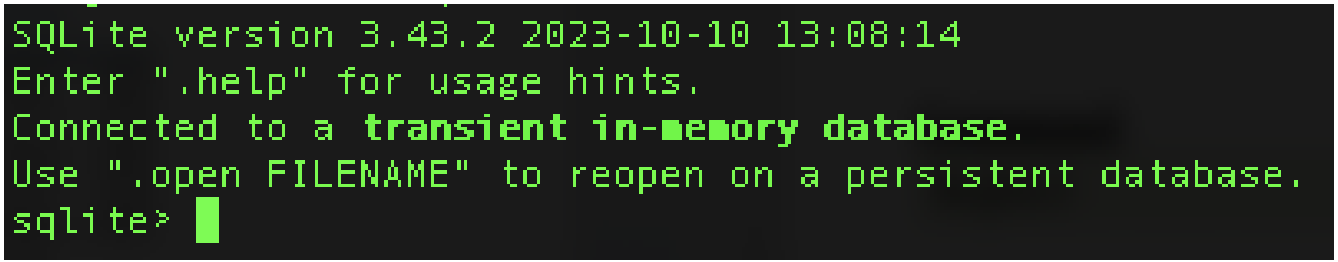
SQLite3

A practical Open-Source Database

Command

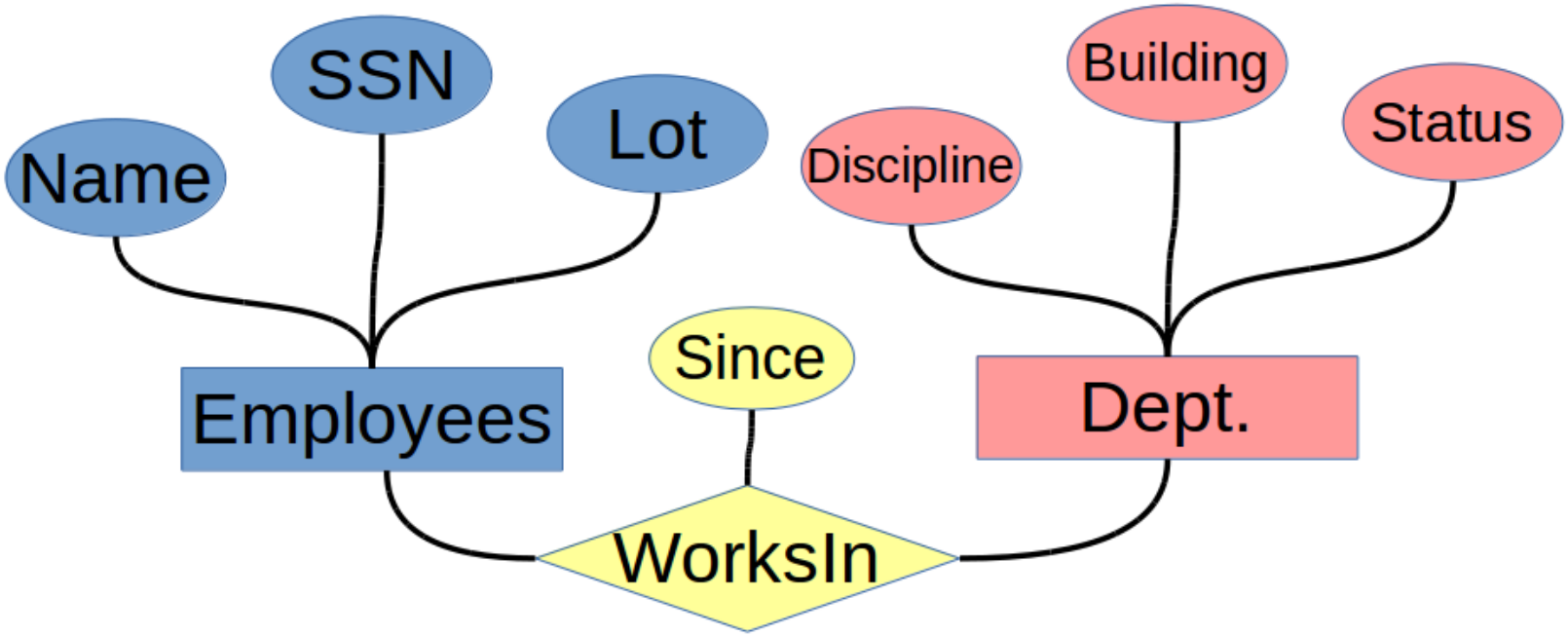
```
$sqlite3
```

You should see this



ER Model Basics

Schemas and Relationships



ER Model Basics

Schemas and Relationships

- A schema resembles a subroutine and describes the table and the data that it contains.
- Relationship: An association among two of more entities
- Relationship Set: A collection of similar relationships for entities
- Relationship sets can also have descriptive attributes (i.e., the “since” attribute of WorksIn)

Entity sets

ID	Tea	Sandwich
JJ	**1**	**Ruban**
OBC	1	PBJ
AM	1	Chicken
GK	1	Chicken
JJ	**1**	**Ruban**
DW	0	PBJ
MC	1	Ruban
JJ	**1**	**Ruban**
SR	1	Ruban
JJ	**1**	**Ruban**
KT	1	Ruban

- **Entity set**: a collection of entities of the same kind (i.e., the preferred sandwiches.)
- Strong Entity sets: Each row is unique in the table.

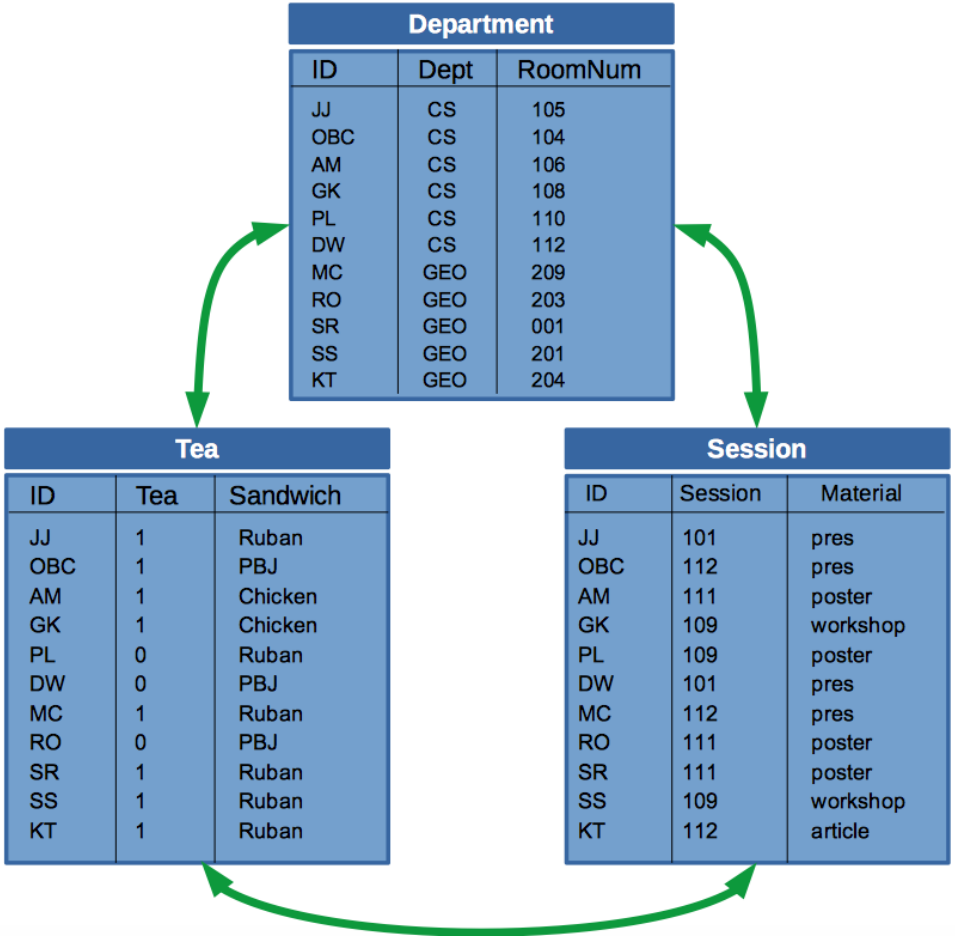
Keys for SQL

- **Primary keys:** Unique identifiers for the row of information sharing a relation (n-tuple).
- **Super keys:** A super key is a set of attributes within a table whose values can be used to uniquely identify a n-tuple.
- **Candidate keys:** is a minimal set of attributes necessary to identify a n-tuple.

Keys

You will note the importance of keys once you start storing your data in your own databases!

Linking the tables by queries



Department

ID	Dept	RoomNum
JJ	CS	105
OBC	CS	104
AM	CS	106
GK	CS	108
PL	CS	110
DW	CS	112
MC	GEO	209
RO	GEO	203
SR	GEO	001
SS	GEO	201
KT	GEO	204

Tea

ID	Tea	Sandwich
JJ	1	Ruban
OBC	1	PBJ
AM	1	Chicken
GK	1	Chicken
PL	0	Ruban
DW	0	PBJ
MC	1	Ruban
RO	0	PBJ
SR	1	Ruban
SS	1	Ruban
KT	1	Ruban

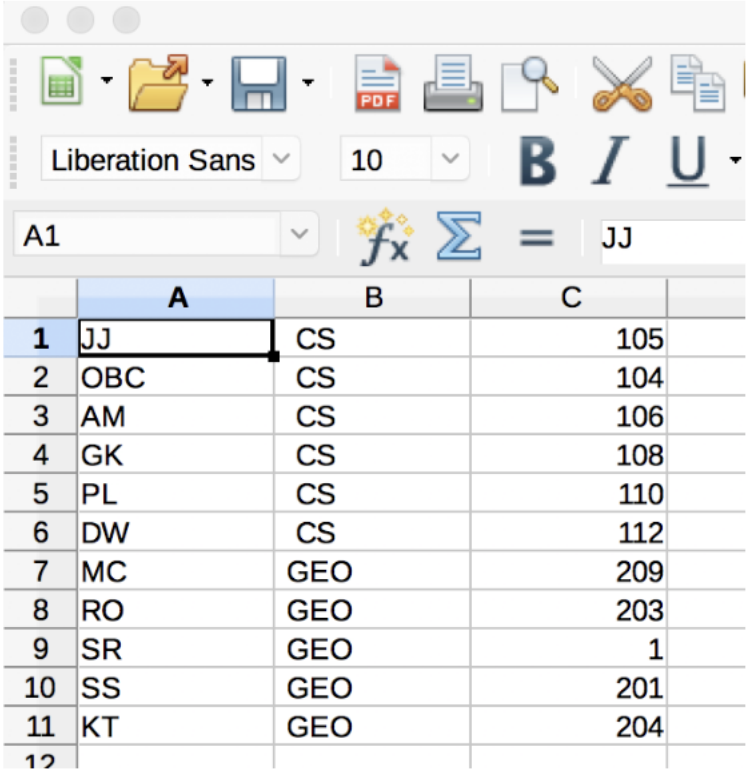
Session

ID	Session	Material
JJ	101	pres
OBC	112	pres
AM	111	poster
GK	109	workshop
PL	109	poster
DW	101	pres
MC	112	pres
RO	111	poster
SR	111	poster
SS	109	workshop
KT	112	article

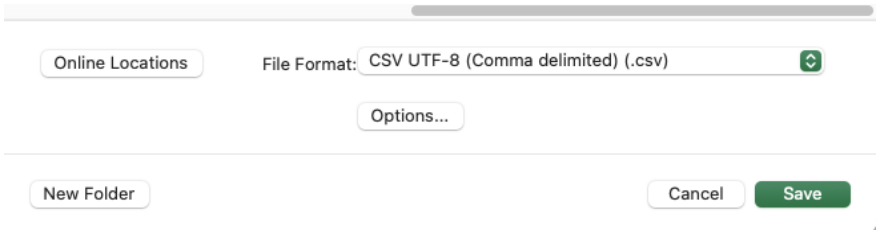
Putting Data into CSV format

- Data as Comma-Separated Values

Data in Spreadsheet Form to Files
Use: “save a copy”



	A	B	C
1	JJ	CS	105
2	OBC	CS	104
3	AM	CS	106
4	GK	CS	108
5	PL	CS	110
6	DW	CS	112
7	MC	GEO	209
8	RO	GEO	203
9	SR	GEO	1
10	SS	GEO	201
11	KT	GEO	204
12			



Making files of CSV's (Comma-separated values)...?

```
JJ,CS,105
OBC,CS,104
AM,CS,106
GK,CS,108
PL,CS,110
DW,CS,112
MC,GEO,209
RO,GEO,203
SR,GEO,001
SS,GEO,201
KT,GEO,204
```

```
JJ,1,Ruban
OBC,1,PBJ
AM,1,Chicken
GK,1,Chicken
PL,0,Ruban
DW,0,PBJ
MC,1,Ruban
RO,0,PBJ
SR,1,Ruban
SS,1,Ruban
KT,1,Ruban
```

```
JJ,101,pres
OBC,112,pres
AM,111,poster
GK,109,workshop
PL,109,poster
DW,101,pres
MC,112,pres
RO,111,poster
SR,111,poster
SS,109,workshop
KT,112,article
```

- Tables: department, tea, session
- Once your file is in this CSV format, it can be easily loaded into the database

The select Clause

The SELECT clause filters out particular data from a table.

- SQL allows duplicates in relations as well as in query results.
- The SELECT statement has many optional clauses:
  - WHERE specifies which rows to retrieve.
  - GROUP BY groups rows sharing a property so that an aggregate function can be applied to each group.
  - HAVING selects among the groups defined by the GROUP BY clause.
  - ORDER BY specifies an order in which to return the rows.
  - AS provides an alias which can be used to temporarily rename tables or columns..

Given table ‘T’

SELECT

Table "T"	Query	Result
C1 | C2 1 | a 2 | b	`SELECT * FROM T;`	C1 | C2 1 | a 2 | b
C1 | C2 1 | a 2 | b	`SELECT C1 FROM T;`	C1 1 2
C1 | C2 1 | a 2 | b	`SELECT * FROM T WHERE C1 = 1;`	C1 | C2 1 | a
C1 | C2 1 | a 2 | b	`SELECT * FROM T ORDER BY C1 DESC;`	C1 | C2 2 | b 1 | a

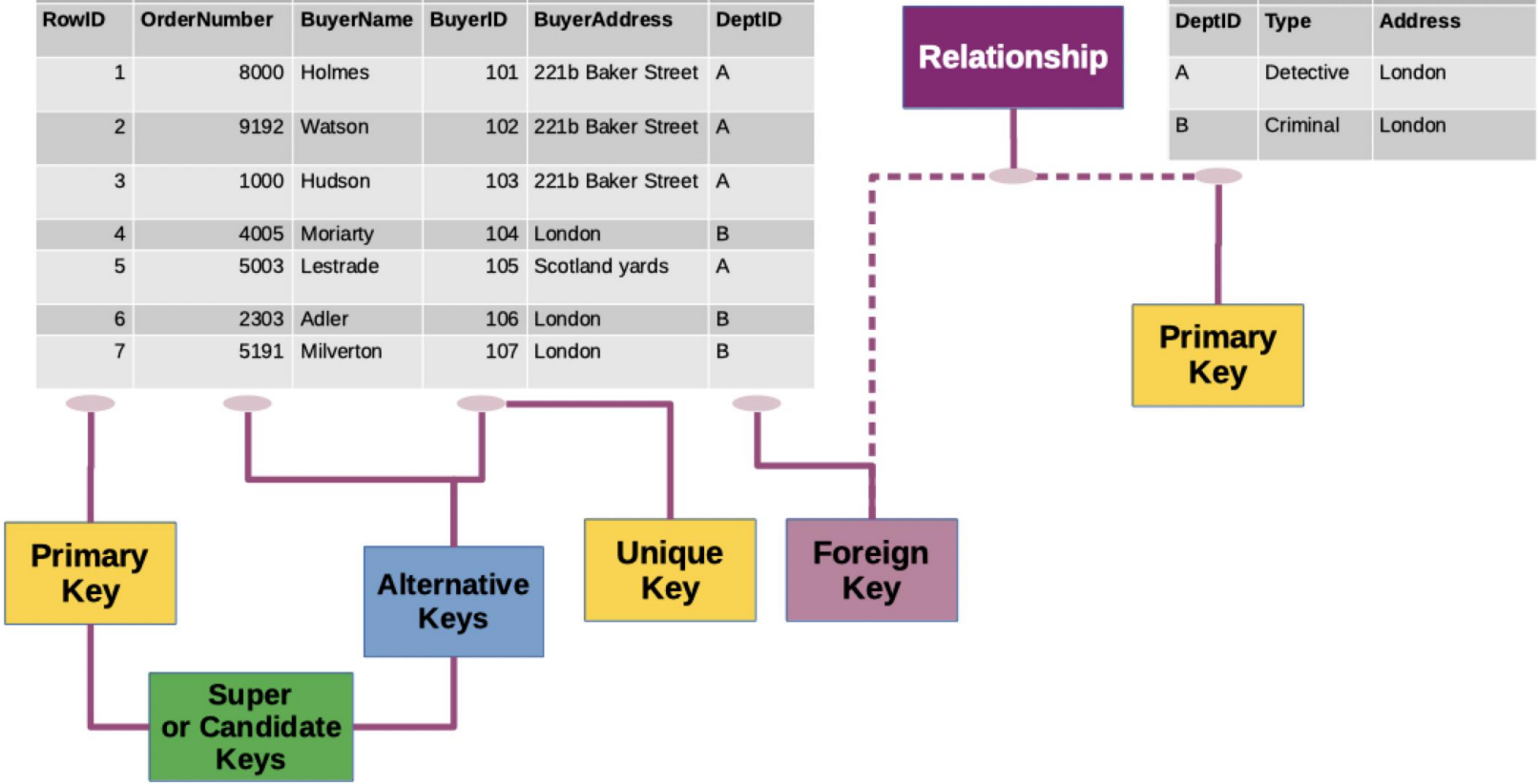
Integrity Constraints

- The CONSTRAINTS are an integrity which defines some conditions that restrict the column to contain the true data while inserting or updating or deleting.
- Integrity constraints provide a mechanism for ensuring that data conforms to guidelines specified by the database administrator. The most common types of constraints include:
    - UNIQUE constraints: To ensure that a given column is unique
    - NOT NULL constraints: To ensure that no null values are allowed
    - FOREIGN KEY constraints: To ensure that two keys share a primary key to foreign key relationship
    - Ensure that a link exists between two tables.

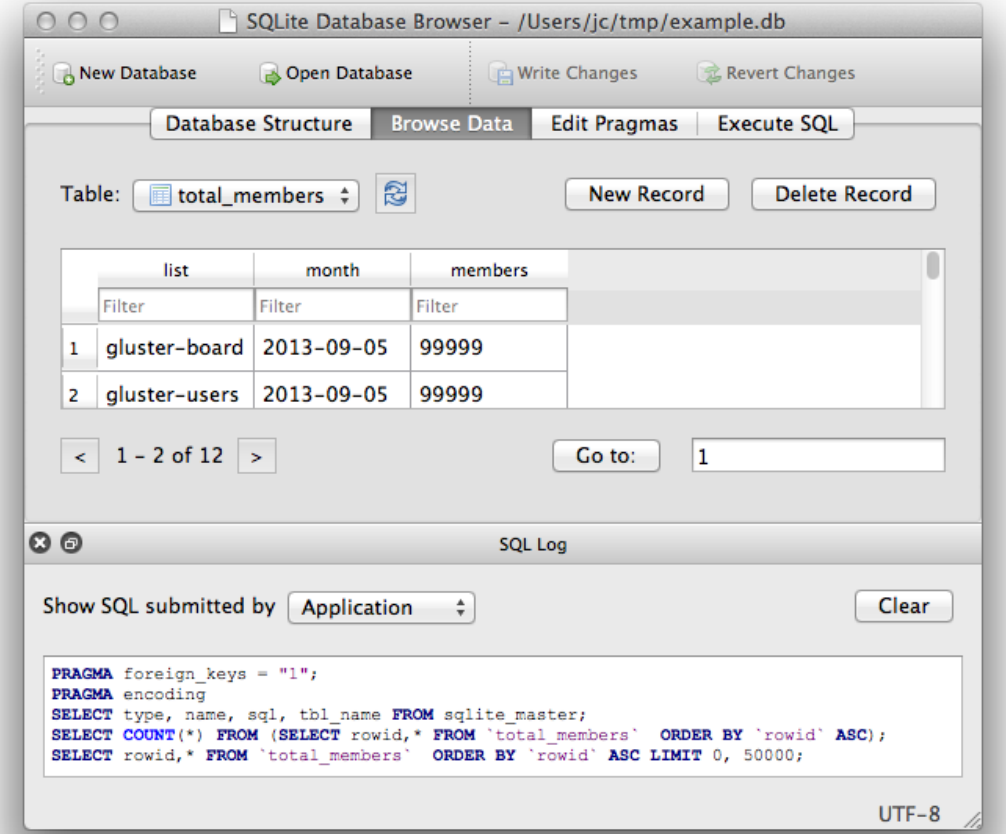
Keys!

RowID	OrderNumber	BuyerName	BuyerID	BuyerAddress	DeptID
1	8000	Holmes	101	221b Baker Street	A
2	9192	Watson	102	221b Baker Street	A
3	1000	Hudson	103	221b Baker Street	A
4	4005	Moriarty	104	London	B
5	5003	Lestrade	105	Scotland yards	A
6	2303	Adler	106	London	B
7	5191	Milverton	107	London	B

DeptID	Type	Address
A	Detective	London
B	Criminal	London

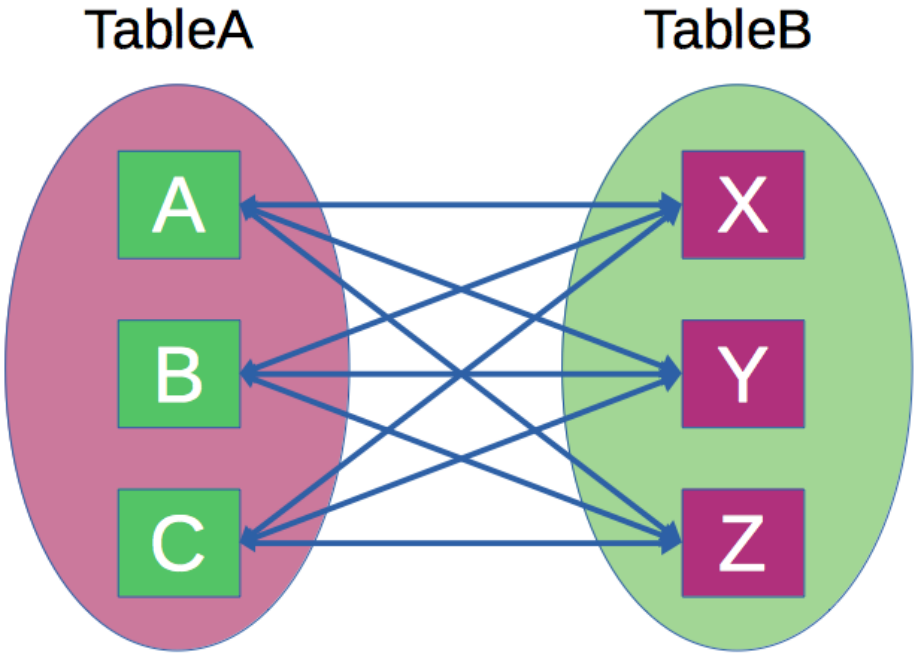


DB Browser


SQLite Database Browser – /Users/jc/tmp/example.db
New Database
Open Database
Write Changes
Revert Changes
Database Structure
Browse Data
Edit Pragmas
Execute SQL
Table: total_members
New Record
Delete Record
list
month
members
Filter
Filter
Filter
1 gluster-board 2013-09-05 99999
2 gluster-users 2013-09-05 99999
< 1 - 2 of 12 >
Go to: 1
SQL Log
Show SQL submitted by Application
Clear
PRAGMA foreign_keys = "1";
PRAGMA encoding
SELECT type, name, sql, tbl_name FROM sqlite_master;
SELECT COUNT(*) FROM (SELECT rowid,* FROM `total_members` ORDER BY `rowid` ASC);
SELECT rowid,* FROM `total_members` ORDER BY `rowid` ASC LIMIT 0, 50000;
UTF-8

Joins

CROSS JOIN: Cartesian Products



SELECT * FROM tableA CROSS JOIN tableB

- SELECT * from TableA CROSS JOIN TableB;
- SELECT * from tableA, TableB;

Steps to run a command in SQL using Python

Connecting to a database with Python code

Five basic steps to using a database according to the Python Database API Specification v2.0

- Step 1: Defining the query
- Step 2: Connecting to the database
- Step 3: Execute the query
- Step 4i, (SELECT): Analyze the result
- Step 4ii, or (UPDATE): Commit the change
- Step 5: Cleaning up; close the database connection

Django

An easy-to-create web site and online database server

- Putting a database on a website!
- https://www.djangoproject.com/

Yes! It Worked!

The install worked successfully! Congratulations!

View release notes for Django 5.2

You are seeing this page because DEBUG=True is in your settings file and you have not configured any URLs.

django

The Files of Your Project

- **Notable Files**
  - **apps.py:** The main file for the hello App
  - **models.py**: A blueprint for how data will be used in the site
  - **tests.py**: For adding tests for bug checking the hello part of the project
  - **views.py**: A request-handler for connecting the URL to the displayed website
  - **mysite/mysite/urls.py**: Requests for apps are all directed using this file.
  - **mysite/hello/urls.py**: Requests for the hello apps are all directed using this file.

NoSQL: Another Type of Database

"Not only SQL" (so much more to offer!)

- Different types of NoSQL databases

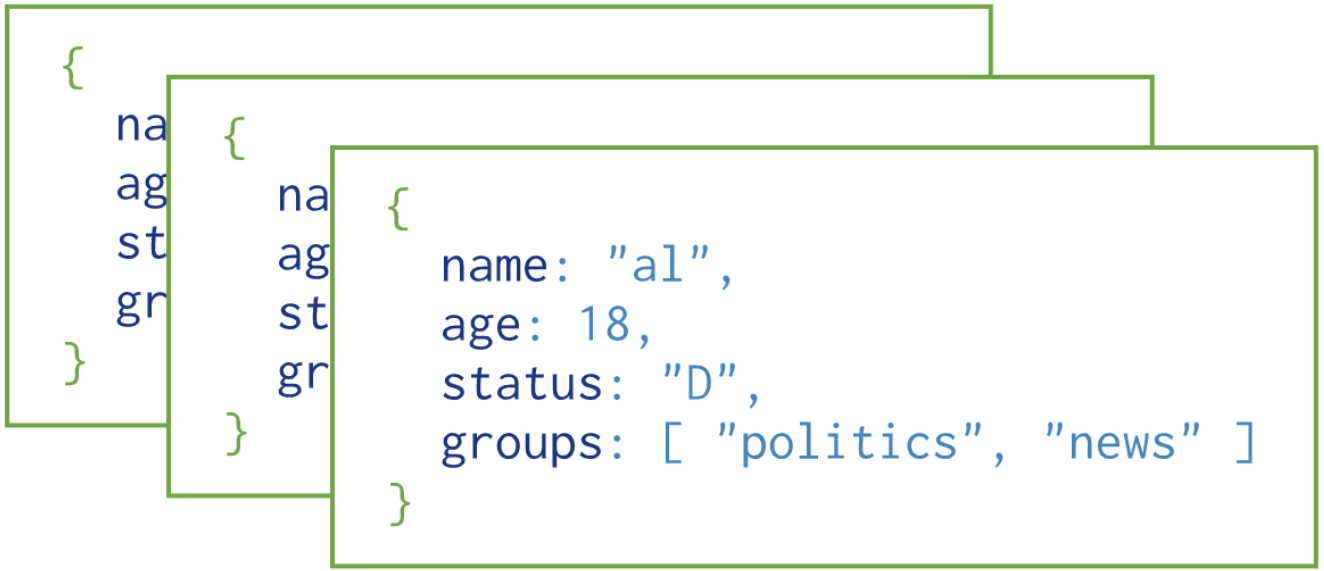
A NoSQL Database Management System

(SQLite3 cannot operate here.)

- https://www.mongodb.com/

Database Language Guide

SQL systems versus NoSQL



- No pre-defined data schema
  - Data may be entered in absence of a defined schema
- Documents (rows) of collections (DB's) may have different types of data

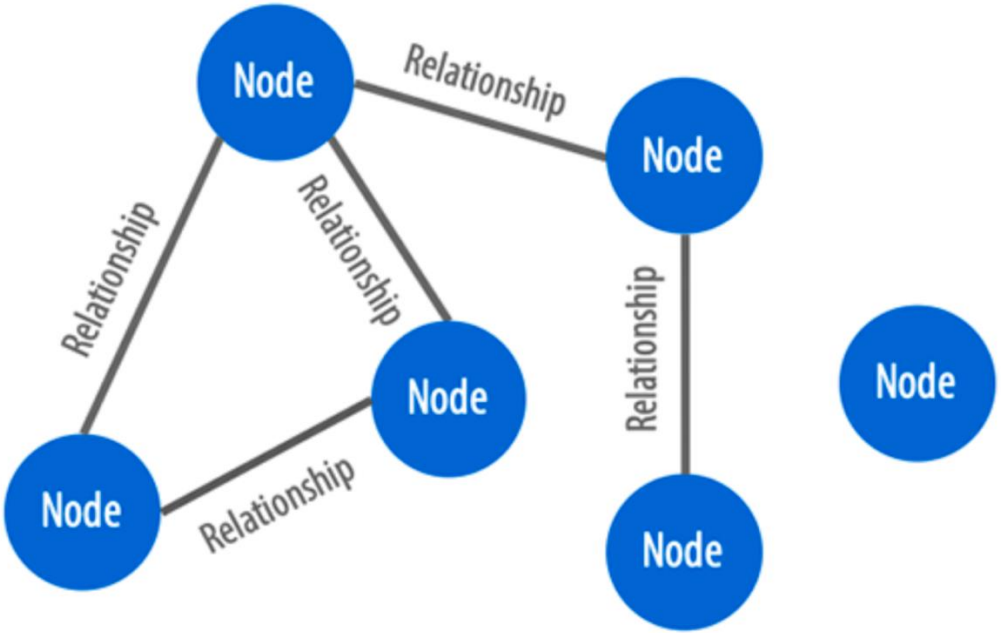
Databases, Visually

- A visual database system using methods from graph theory to use networks to determine relationships (edges) and discover meaning from connected data-points (nodes). Users are able to interact with the data in a network.

- https://neo4j.com/
- Graphgists Projects: https://neo4j.com/graphgists/

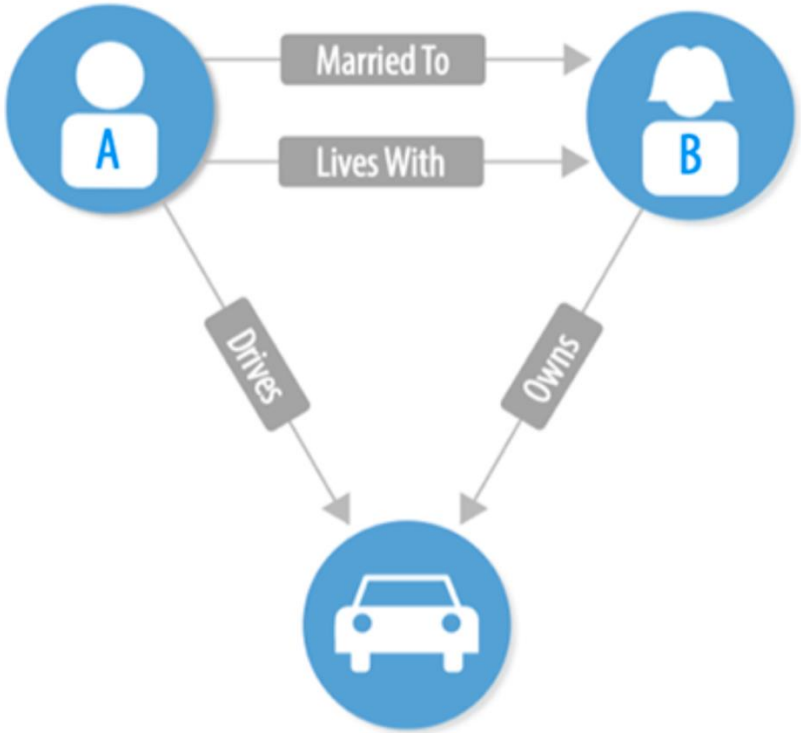
Networks Of Data

Relationships exist by connectivity



- Nodes and edges represent inter-relationships
- Relationships are described by connections between nodes
- Single nodes have no immediate relationships with the others

Networks In Neo4J

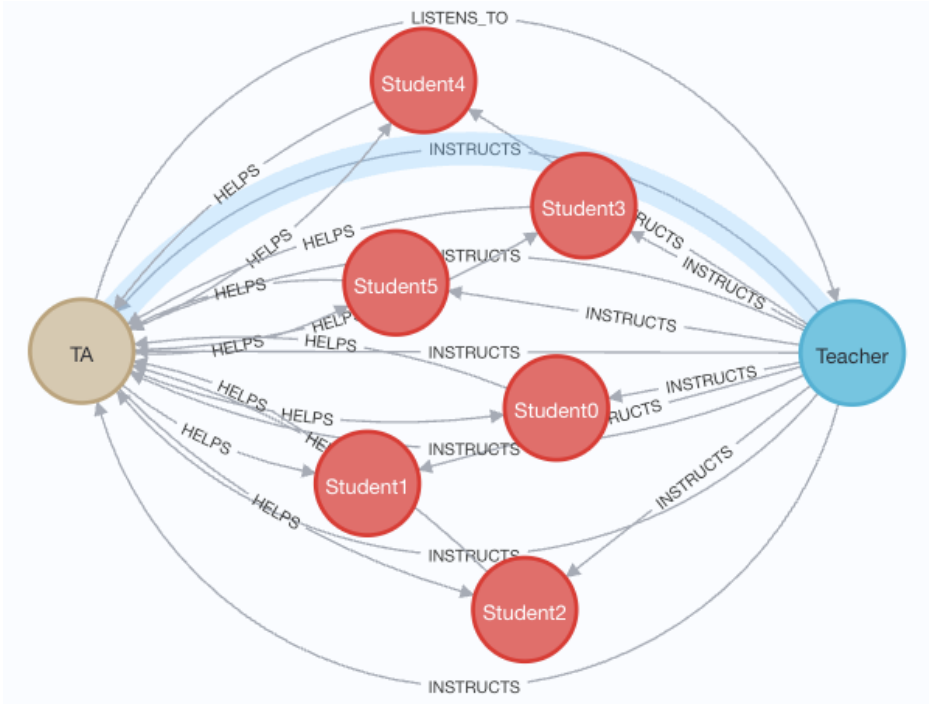


- An acting schema: The relationships between nodes are built into the network

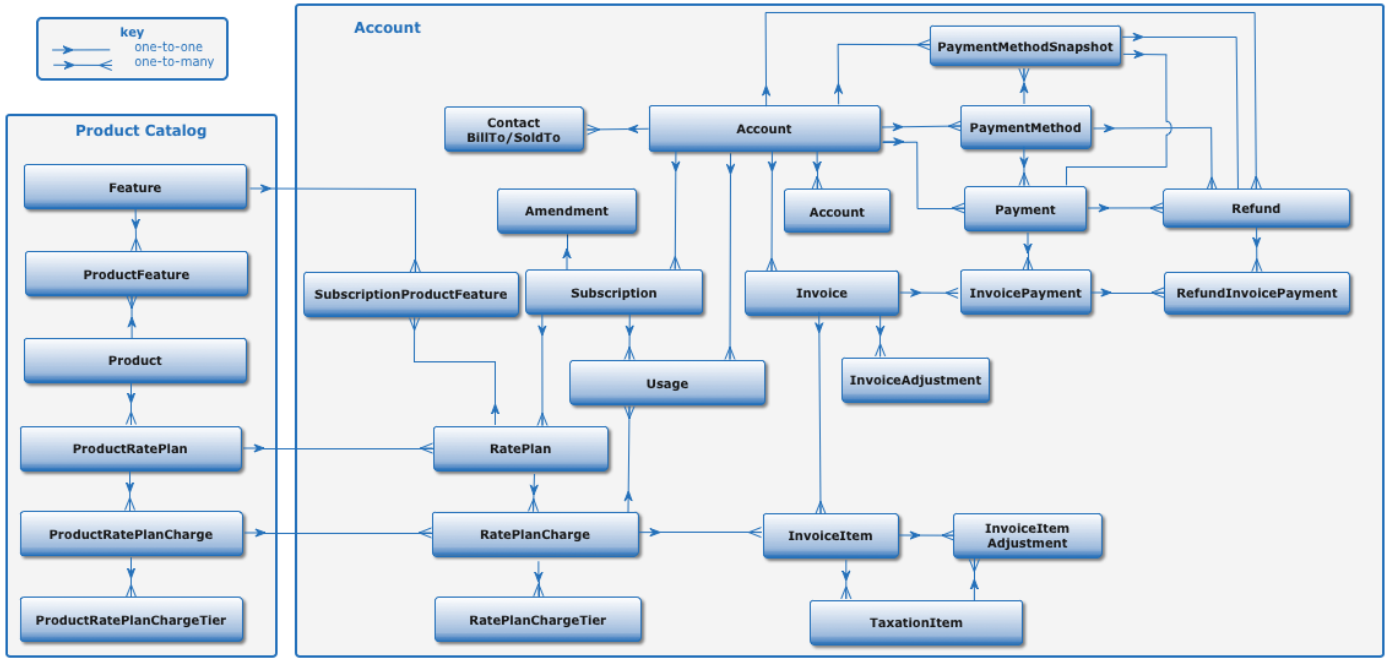
Relationship Queries

Who instructs whom?

```
MATCH t=()-[s:INSTRUCTS]->() RETURN t
```



What Has This Class Covered?



What has this class not covered?!

(Now go update the Skills section of your resum´e!)