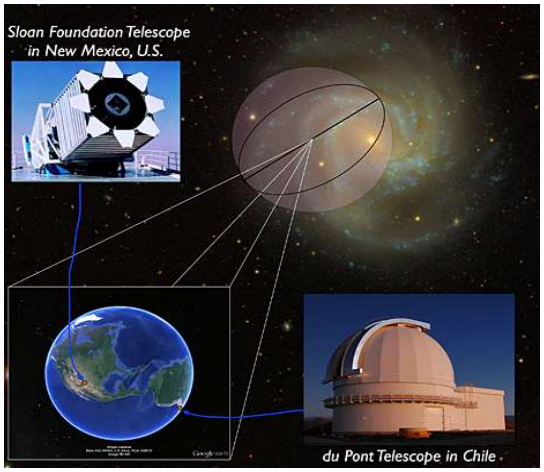
Data Management for Data Science DATA 514

Lecture 1: Introduction, Data Models

Gradience token on the whiteboard: please write it down

IBM MERGE Healthcare

Class Goals



- The world is drowning in data!
- Needed: data scientists to help manage this data
  - Help domain scientists achieve new discoveries
  - Help companies provide better services
  - Help governments become more efficient
- Welcome to 514
  - Existing tools PLUS data management principles

Staff

- Instructor: Babak Salimi
  - Office hours: TBD, CSE 282

- TA: Ee (Isaac) Ahn
- Office hours:
  - Monday: 5:00 pm - 6:00 pm
  - Wednesday: 5:00 pm - 6:00 pm ??

About Me

- Postdoctoral Researcher at UW since 2016
- PhD from Carleton University, Canada, Ottawa
- Born in Iran
- Research Interests: Decision Making System, Causal Inference from Big Data, Database Repair and Approximate Query Processing

Course Format

- Lectures Tuesdays, 5pm-7:50pm
  - Location: here!
- Sections: Tuesdays, 8-8:50pm
  - Content: exercises, tutorials, questions
  - Locations: here!
- 6 homework assignments
- 7 web quizzes
- Midterm and final

Communications

- Web page: https://courses.cs.washington.edu/courses/csed514/18wi/
  - Syllabus is there
  - Lectures will be available there (see calendar)
  - Homework assignments will be available there
  - Link to web quizzes is there
- Mailing list
  - Announcements (low traffic – must read)
  - Registered students automatically subscribed
- Discussion board
  - **THE** place to ask course-related questions
  - Today, go to board and enable notifications

Textbook

Main textbook, available at the bookstore:

- *Database Systems: The Complete Book*,
  Hector Garcia-Molina,
  Jeffrey Ullman,
  Jennifer Widom
  Second edition.

Most important: COME TO CLASS ! ASK QUESTIONS !

Other Texts

Available at the Engineering Library
(some on reserve):

- *Database Management Systems*, Ramakrishnan
- *Fundamentals of Database Systems*, Elmasri, Navathe
- *Foundations of Databases*, Abiteboul, Hull, Vianu
- *Data on the Web,* Abiteboul, Buneman, Suciu

Grading

- Homeworks 30%
- Web quizzes 20%
- Midterm 20%
- Final 30%

Eight Homework Assignments

H1: Sqlite

H2: Basic SQL with SQLite

H3: Advanced SQL with SQL Server

H4: Conceptual Design

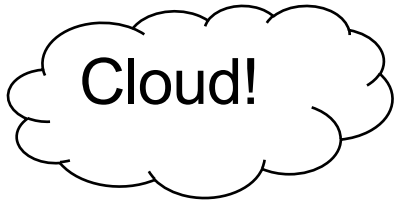
H5: JSon

H6: SQL in Java (JDBC)

Check calendar for due dates -- Submit via gitlab!

About the Assignments

- Homework assignments will take time but most time should be spent *learning*
- Do them on your own
- Very practical assignments
- Put everything on your resume!!!
  - SQL, SQLite, SQL Server, SQL Azure JDBC, JSon,…



Deadlines and Late Days

- Assignments are expected to be done on time, but things happen, so…
- You have up to 4 late days
  - No more than 2 on any one assignment
  - Use in 24-hour chunks
- Late days = safety net, not convenience!
  - You should not plan on using them
  - If you use all 4 you are doing it wrong

Six Web Quizzes

- http://newgradiance.com/
- Create account, provide token
- Class token: on the white board, write it down
- No late days – closes at 11:00 deadline
- Provides explanations for wrong answers
- Short tests, take many times, best score counts

Exams

- Midterm and Final
  - See course calendar for dates and times
- **May bring 1 letter-size, double-side piece of paper with notes**
- Closed book. No computers, phones, watches, etc.!
- Check course website for dates
- Location: in class

Academic Integrity

Anything you submit for credit is expected to be your own work

- Of course OK to exchange ideas, but not detailed solutions
- We all know difference between collaboration and cheating
- Attempt to gain credit for work you did not do is misconduct

Now onto the real stuff…

Outline of Today's Lecture

- Overview of database management systems
- Course content

- Data Models
- SQL

Database Management System

What is a DBMS ?

Give examples of DBMSs

Database Management System

What is a DBMS ?

- *A big program written by someone else that allows us to manage efficiently a large database and allows it to persist over long periods of time*

Give examples of DBMSs

- Oracle, IBM DB2, Microsoft SQL Server, Vertica
- Open source: MySQL (Sun/Oracle), PostgreSQL, AsterixDB
- Open source library: SQLite

We will focus on relational DBMSs most quarter

An Example: Online Bookseller

- What data do we need?
  -
  -
  -
  -
- What capabilities on the data do we need?
  -
  -
  -

An Example: Online Bookseller

- What data do we need?
  - Data about books, customers, pending orders, order histories, trends, preferences, etc.
  - Data about sessions (clicks, pages, searches)
  - Note: data must be persistent! Outlive application
  - Also note that data is large… won't fit all in memory
- What capabilities on the data do we need?
  -
  -
  -

An Example: Online Bookseller

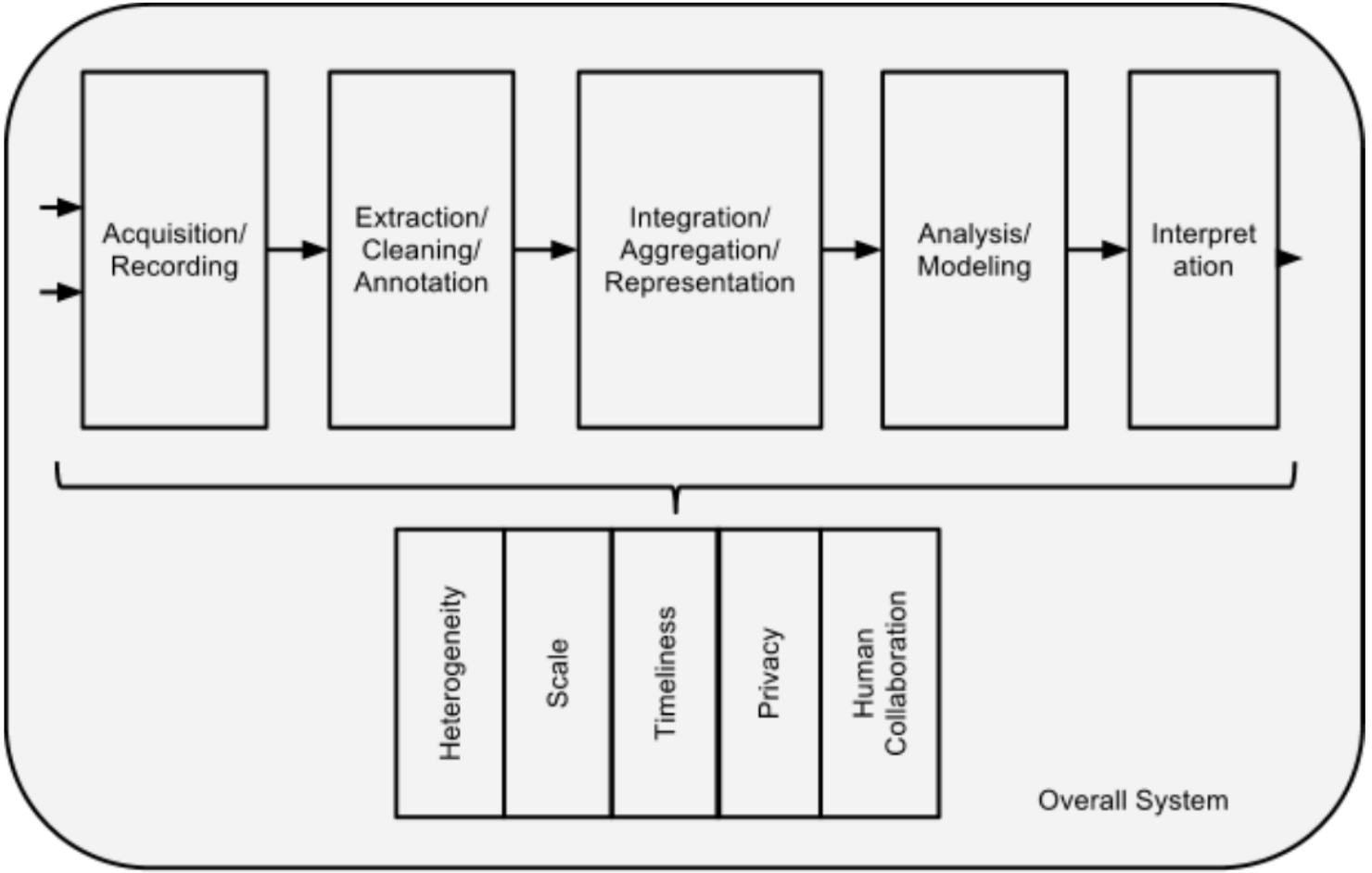
- What data do we need?
  - Data about books, customers, pending orders, order histories, trends, preferences, etc.
  - Data about sessions (clicks, pages, searches)
  - Note: data must be persistent! Outlive application
  - Also note that data is large… won't fit all in memory
- What capabilities on the data do we need?
  - Insert/remove books, find books by author/title/etc., analyze past order history, recommend books, …
  - Data must be accessed efficiently, by many users
  - Data must be safe from failures and malicious users

Multi-user

- Jane and John both have ID number for gift certificate (credit) of $200 they got as a wedding gift
  - Jane @ her office orders "The Selfish Gene, R. Dawkins" ($80)
  - John @ his office orders "Guns and Steel, J. Diamond" ($100)

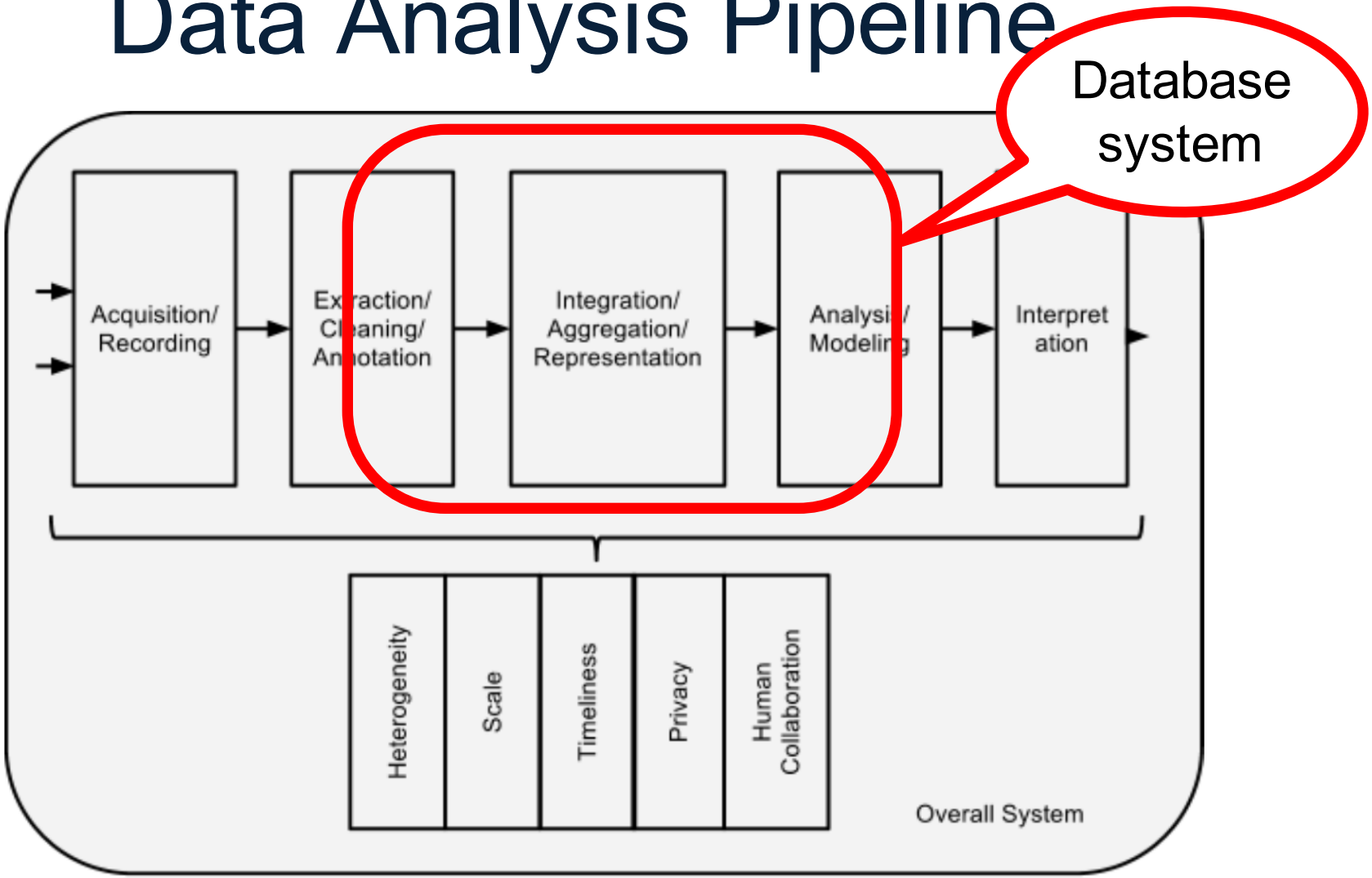
- Questions:
  - What is the ending credit?
  - What if second book costs $130?
  - What if system crashes?

Data Analysis Pipeline*



*Challenges and Opportunities with Big Data, 2012

Data Analysis Pipeline*



*Challenges and Opportunities with Big Data, 2012

DBMS Benefits

- Expensive to implement all these features inside the application
- DBMS provides these features (and more)
- DBMS simplifies application development

Client/Server Architecture

- One *server* that stores the database (DBMS):
  - Usually a beefy system
  - But can be your own desktop…
  - … or a huge cluster running a parallel DBMS
- Many *clients* run apps and connect to DBMS
  - E.g. Microsoft's Management Studio
  - Or psql (for PostgreSQL)
  - Or some Java/C++ program (very typical)
- Clients "talk" to server using JDBC protocol

People

- **DB designer**: establishes schema
- **DB application developer**: writes programs that query and modify data
- **DB administrator**: loads data, tunes system, keeps whole thing running
- **Data analyst**: data mining, data integration
- **Data Scientist**: analyst, designer, developer, administrator
- **DBMS implementer**: builds the DBMS

Key Data Mngmt Concepts

- **Data models**: how to describe real-world data
  - Relational, XML, graph data (RDF)
- **Schema v.s. data**
- **Declarative query language**
  - Say what you want not how to get it
- **Data independence**
  - Physical independence: Can change how data is stored on disk without maintenance to applications
  - Logical independence: can change schema w/o affecting apps
- **Query optimizer** and compiler
- **Transactions**: isolation and atomicity

Review this slide throughout the quarter!

What This Course Contains

- **Focus: Using DBMSs**
- Relational Data Model
  - SQL, Relational Algebra, Relational Calculus, datalog
- Conceptual design
  - E/R diagrams, Views, and Database normalization
- Query execution
- Semistructured Data Model
  - SQL++,JSon,
- Transactions
- Data integration and data cleaning

Data Models

- Recall our example: want to design a database of books:
  - author, title, publisher, pub date, price, etc
  - How should we describe this data?
- **Data model** = mathematical formalism (or conceptual way) for describing the data

Data Models

- Relational
  - Data represented as relations
- Semi-structured (JSon)
  - Data represented as trees
- Key-value pairs
  - Used by NoSQL systems
- Graph
- Object-oriented

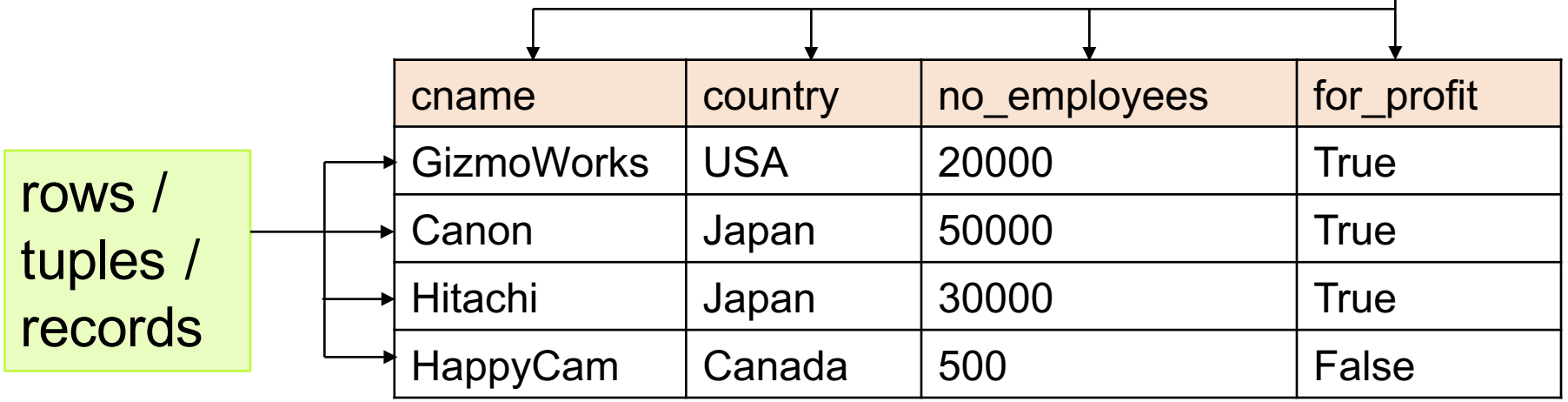
3 Elements of Data Models

- Instance
  - The actual data
- Schema
  - Describe what data is being stored
- Query language
  - How to retrieve and manipulate data

Relational Model

columns / attributes / fields

- Data is a collection of relations / tables:



cname	country	no_employees	for_profit
GizmoWorks	USA	20000	True
Canon	Japan	50000	True
Hitachi	Japan	30000	True
HappyCam	Canada	500	False

rows / tuples / records

- mathematically, relation is a set of tuples
  - each tuple appears 0 or 1 times in the table
  - order of the rows is unspecified

The Relational Data Model

- Degree (arity) of a relation = #attributes
- Each attribute has a type.
  - Examples types:
    - Strings: CHAR(20), VARCHAR(50), TEXT
    - Numbers: INT, SMALLINT, FLOAT
    - MONEY, DATETIME, …
    - Few more that are vendor specific
  - Statically and strictly enforced

Keys

- Key = one (or multiple) attributes that uniquely identify a record

Keys

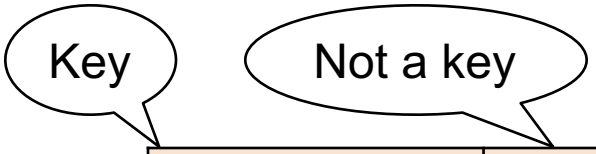
- Key = one (or multiple) attributes that uniquely identify a record

Key

cname	country	no_employees	for_profit
GizmoWorks	USA	20000	True
Canon	Japan	50000	True
Hitachi	Japan	30000	True
HappyCam	Canada	500	False

Keys

- Key = one (or multiple) attributes that uniquely identify a record



cname	country	no_employees	for_profit
GizmoWorks	USA	20000	True
Canon	Japan	50000	True
Hitachi	Japan	30000	True
HappyCam	Canada	500	False

Keys

- Key = one (or multiple) attributes that uniquely identify a record

cname	country	no_employees	for_profit
GizmoWorks	USA	20000	True
Canon	Japan	50000	True
Hitachi	Japan	30000	True
HappyCam	Canada	500	False

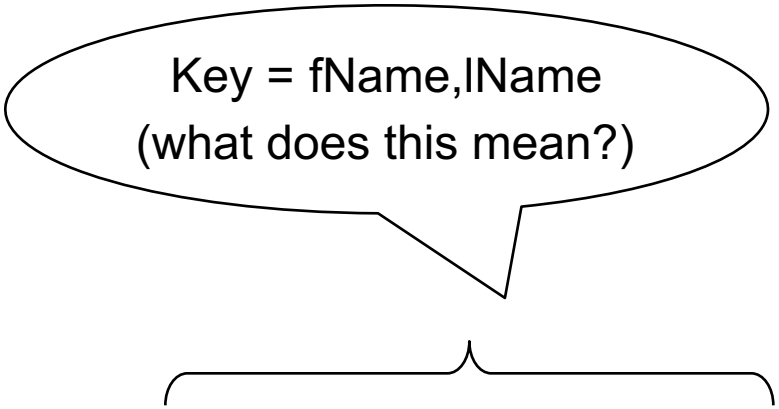
Keys

- Key = one (or multiple) attributes that uniquely identify a record

No: future updates to the database may create duplicate no_employees

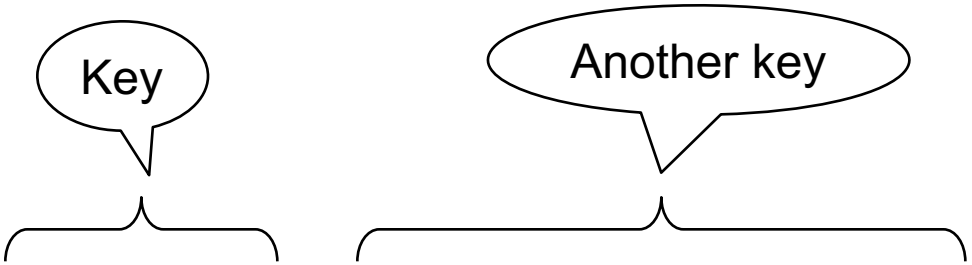
cname	country	no_employees	for_profit
GizmoWorks	USA	20000	True
Canon	Japan	50000	True
Hitachi	Japan	30000	True
HappyCam	Canada	500	False

Multi-attribute Key



fName	lName	Income	Department
Alice	Smith	20000	Testing
Alice	Thompson	50000	Testing
Bob	Thompson	30000	SW
Carol	Smith	50000	Testing

Multiple Keys



SSN	fName	lName	Income	Department
111-22-3333	Alice	Smith	20000	Testing
222-33-4444	Alice	Thompson	50000	Testing
333-44-5555	Bob	Thompson	30000	SW
444-55-6666	Carol	Smith	50000	Testing

We can choose one key and designate it as *primary key*

E.g.: primary key = SSN

Foreign Key

```
Company(cname, country, no_employees, for_profit)
Country(name, population)
```

Company

cname	country	no_employees	for_profit
Canon	Japan	50000	Y
Hitachi	Japan	30000	Y

Country

name	population
USA	320M
Japan	127M

Keys: Summary

- Key = columns that uniquely identify tuple
  - Usually we underline
  - A relation can have many keys, but only one can be chosen as *primary key*
- Foreign key:
  - Attribute(s) whose value is a key of a record in some other relation
  - Foreign keys are sometimes called *semantic pointer*

Query Language

- SQL
  - **S**tructured **Q**uery **L**anguage
  - Developed by IBM in the 70s
  - Most widely used language to query relational data
- Other relational query languages
  - Datalog, relational algebra

Our First DBMS

- SQL Lite
- Will switch to SQL Server later in the quarter

Demo

What to Do Now

- https://courses.cs.washington.edu/courses/csep514/17wi/
- Webquiz 1 is open
  - Create account at http://www.newgradiance.com/services/servlet/COTC
  - Sign up for class online
  - Webquiz due next Sunday, 11 pm
- Homework 1 is posted
  - Simple queries in SQL Lite
  - Homework due on Tuesday, 11 pm
- Post message on discussion board (say 'hi')