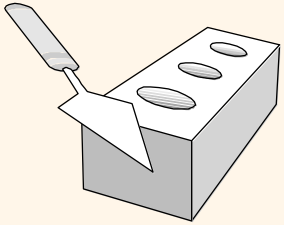
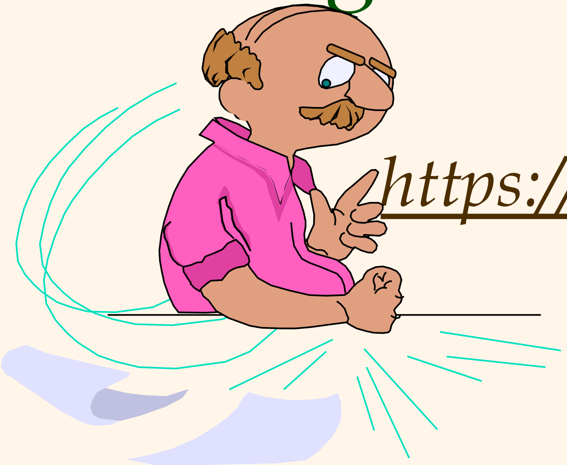


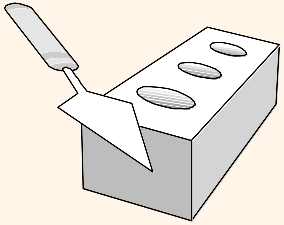
COMM7360
Big Data Management and Analytics



https://github.com/shary777/comm7360bigdata

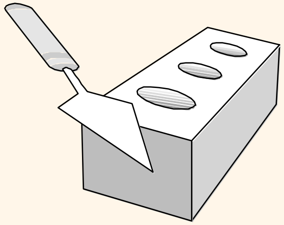
Instructors:

Dr. Paolo Mengoni and Ms. Xiaoyi Fu



Contact Information

- Ms. Xiaoyi Fu (Section 1)
  - Office: RRS 726
  - E-mail: xiaoyifu@comp.hkbu.edu.hk
- Dr. Paolo Mengoni (Section 2)
  - E-mail: paolo.mengoni@gmail.com
- Time
  - WED 18:30 – 21:20



Course Objectives

- To present an introduction of big data analysis and data management, emphasize on
  - How to manage large volume data data (data warehousing)
  - How to design a database (application)

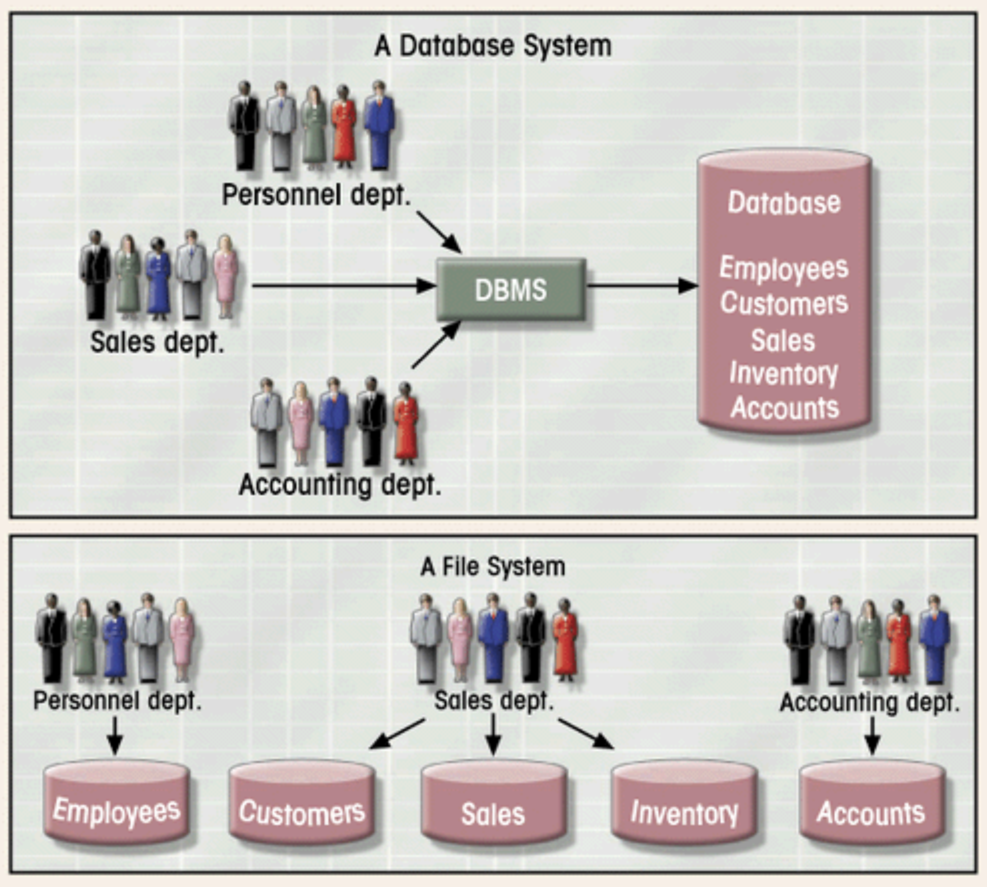
Big Data

What is a DBMS?

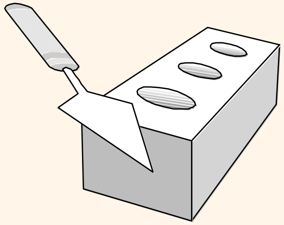
- Database
  - Maintains a *very large*, integrated collection of data…
  - *that is* ***organized*** *so that its contents can easily be accessed and updated*

- A Database Management System (DBMS) is a software package designed to store and manage databases.

Database vs. File Systems

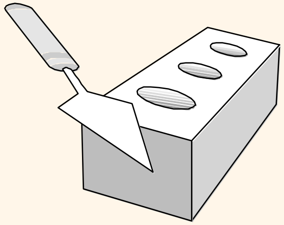


Database vs. File Systems (1)



- Efficient access
  - Write a new program for each new task
- Data consistency
  - Duplication of information in different files
- Data integrity
  - Constrains “buried” in program code

Database vs. File Systems (2)



- Concurrent access
  - Multiple users modify the files at the same time
- Recovery from crashes
  - System crash
- Security
  - Hard to provide user access to some, but not all data

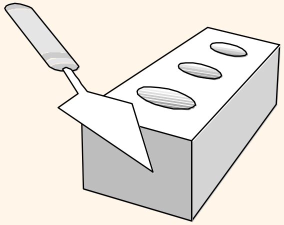
Databases offers solutions to all the above problems

Database Applications

- Banking: all transactions
- Airlines: reservations, schedules
- Universities: registration, grades
- Sales: customers, products, purchases
- Manufacturing: production, inventory, orders, supply chain
- Human Resources: employee records, salaries, tax deductions

Databases touch many aspects of our lives!

Big Names in Database Systems



Company	*Product*	*Remarks*
Oracle	Oracle 10g, 11g, etc.	2nd largest software maker by revenue
IBM	DB2, Universal Server	Largest research organization
Microsoft	Access, SQL Server	Access comes with MS Office

Similar Products:

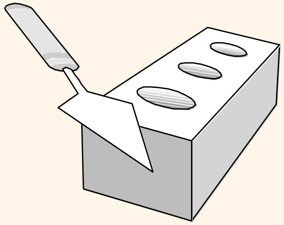
Google: Cloud Database

Amazon: SimpleDB

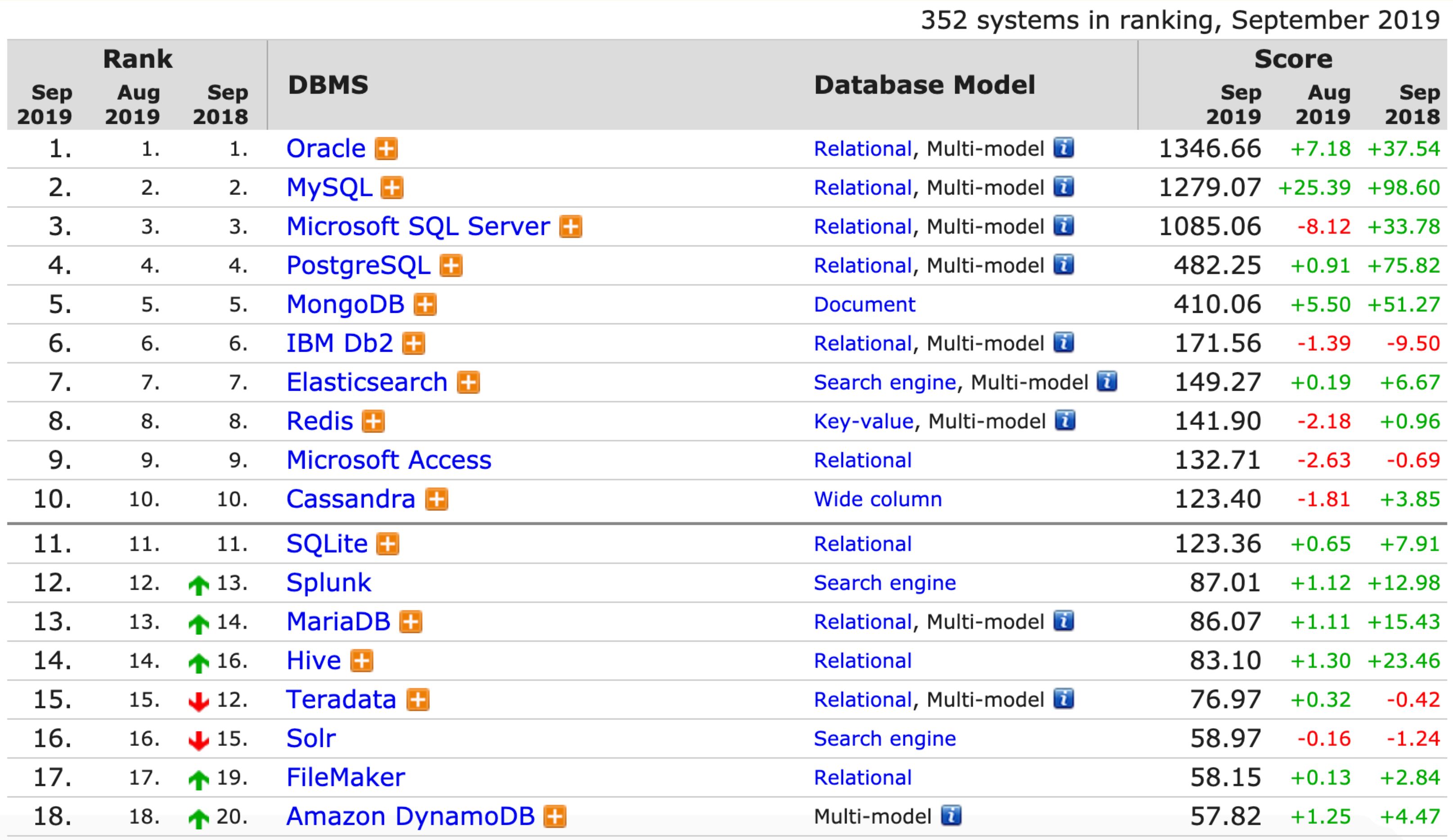
Free & Open Source DB:

PostgreSQL

CUBRID



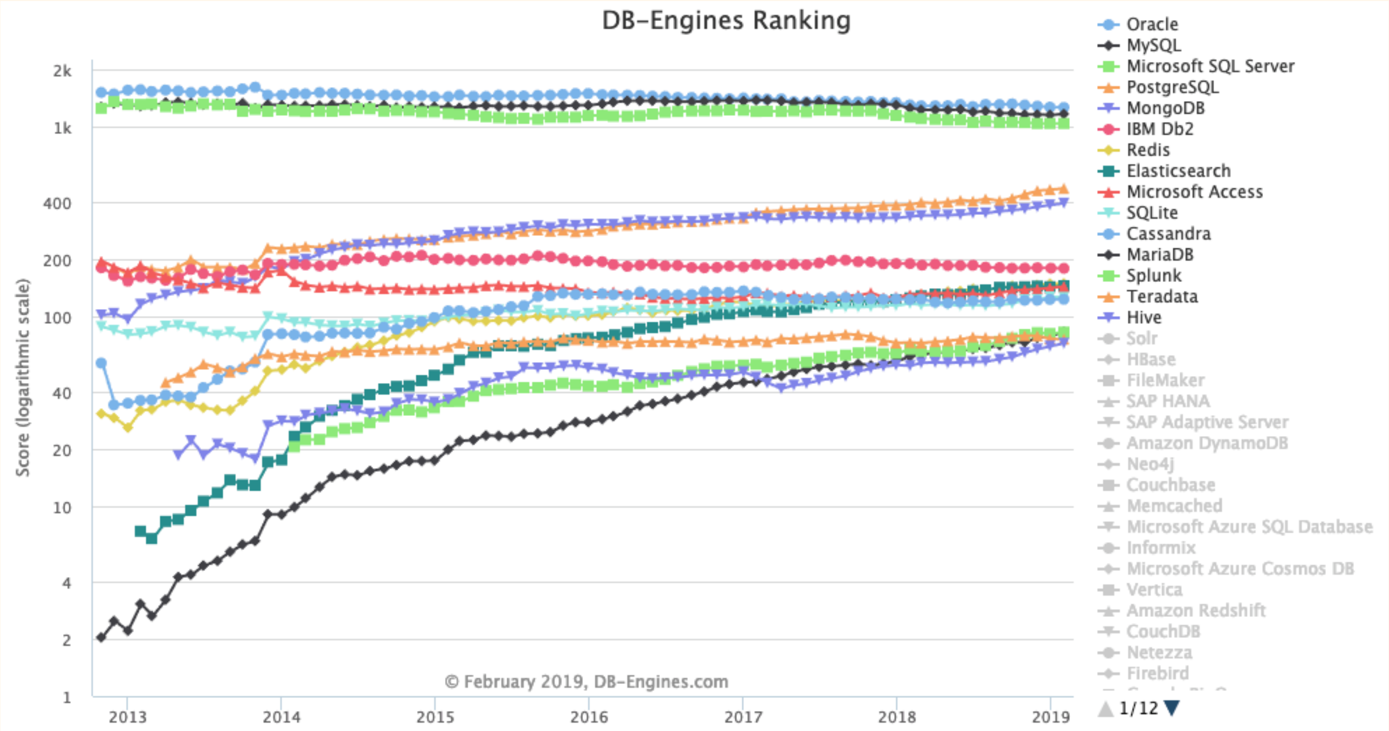
Database Popularity Ranking



352 systems in ranking, September 2019

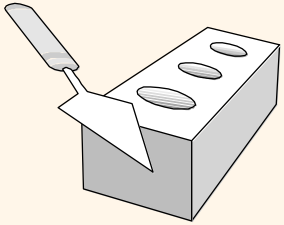
Rank			DBMS	Database Model	Score		
Sep 2019	Aug 2019	Sep 2018			Sep 2019	Aug 2019	Sep 2018
1.	1.	1.	Oracle	Relational, Multi-model	1346.66	+7.18	+37.54
2.	2.	2.	MySQL	Relational, Multi-model	1279.07	+25.39	+98.60
3.	3.	3.	Microsoft SQL Server	Relational, Multi-model	1085.06	-8.12	+33.78
4.	4.	4.	PostgreSQL	Relational, Multi-model	482.25	+0.91	+75.82
5.	5.	5.	MongoDB	Document	410.06	+5.50	+51.27
6.	6.	6.	IBM Db2	Relational, Multi-model	171.56	-1.39	-9.50
7.	7.	7.	Elasticsearch	Search engine, Multi-model	149.27	+0.19	+6.67
8.	8.	8.	Redis	Key-value, Multi-model	141.90	-2.18	+0.96
9.	9.	9.	Microsoft Access	Relational	132.71	-2.63	-0.69
10.	10.	10.	Cassandra	Wide column	123.40	-1.81	+3.85
11.	11.	11.	SQLite	Relational	123.36	+0.65	+7.91
12.	12.	↑ 13.	Splunk	Search engine	87.01	+1.12	+12.98
13.	13.	↑ 14.	MariaDB	Relational, Multi-model	86.07	+1.11	+15.43
14.	14.	↑ 16.	Hive	Relational	83.10	+1.30	+23.46
15.	15.	↓ 12.	Teradata	Relational, Multi-model	76.97	+0.32	-0.42
16.	16.	↓ 15.	Solr	Search engine	58.97	-0.16	-1.24
17.	17.	↑ 19.	FileMaker	Relational	58.15	+0.13	+2.84
18.	18.	↑ 20.	Amazon DynamoDB	Multi-model	57.82	+1.25	+4.47

Database Popularity Ranking



© February 2019, DB-Engines.com

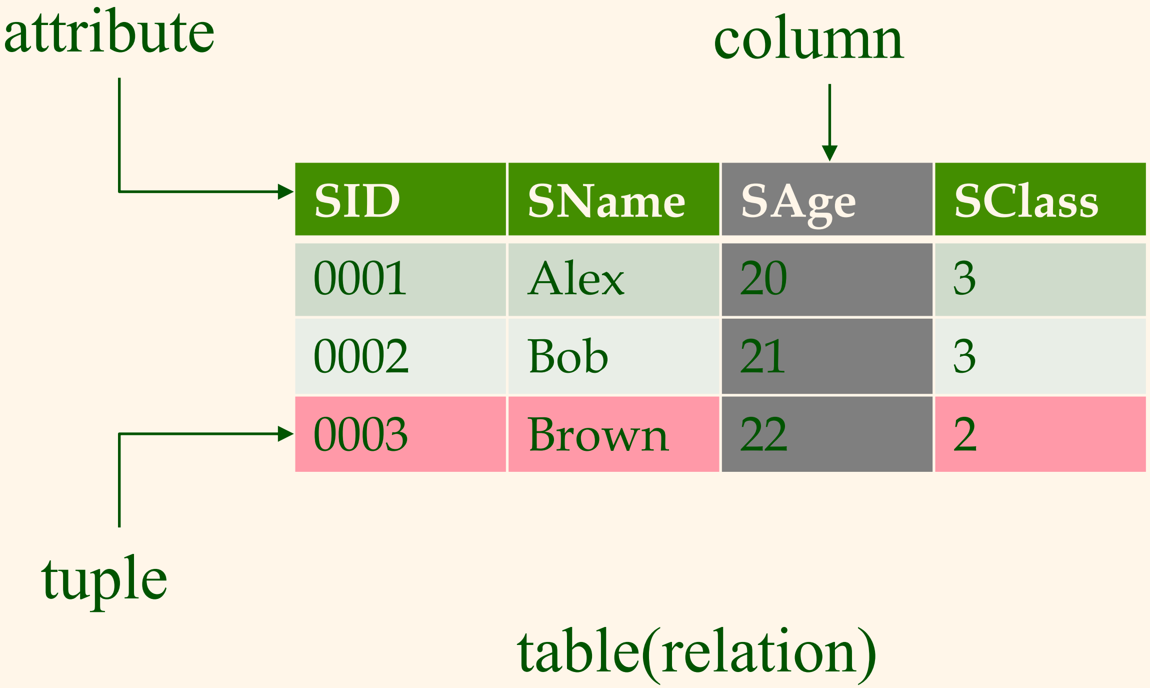
Data Models



- A *data model* is a collection of conceptual tools for describing stored data at physical, logical and view levels.
  - Data semantics, consistency constrains, data relationships

- The *relational data model* is the most widely used data model today.
  - Main concept: *relation*, basically a *set* of distinct *rows* or *tuples*
  - Record-based model.

Data Models



attribute

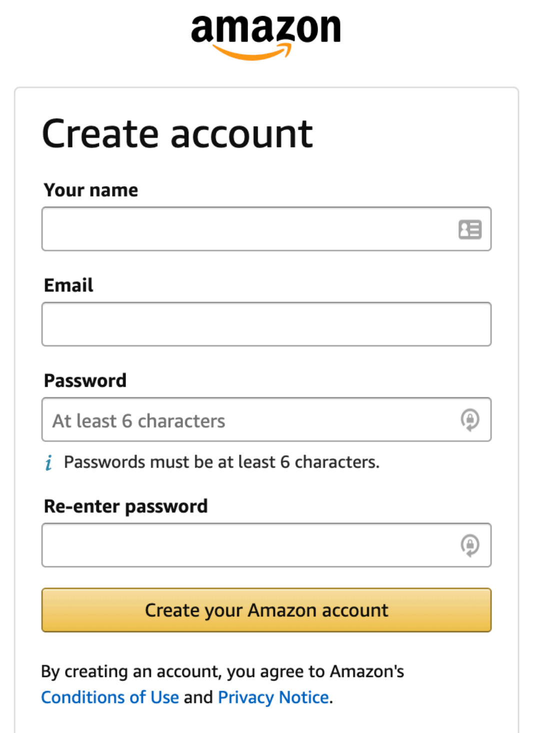
column

SID	SName	SAge	SClass
0001	Alex	20	3
0002	Bob	21	3
0003	Brown	22	2

tuple

table(relation)

Data Models



amazon

Create account

Your name

Email

Password

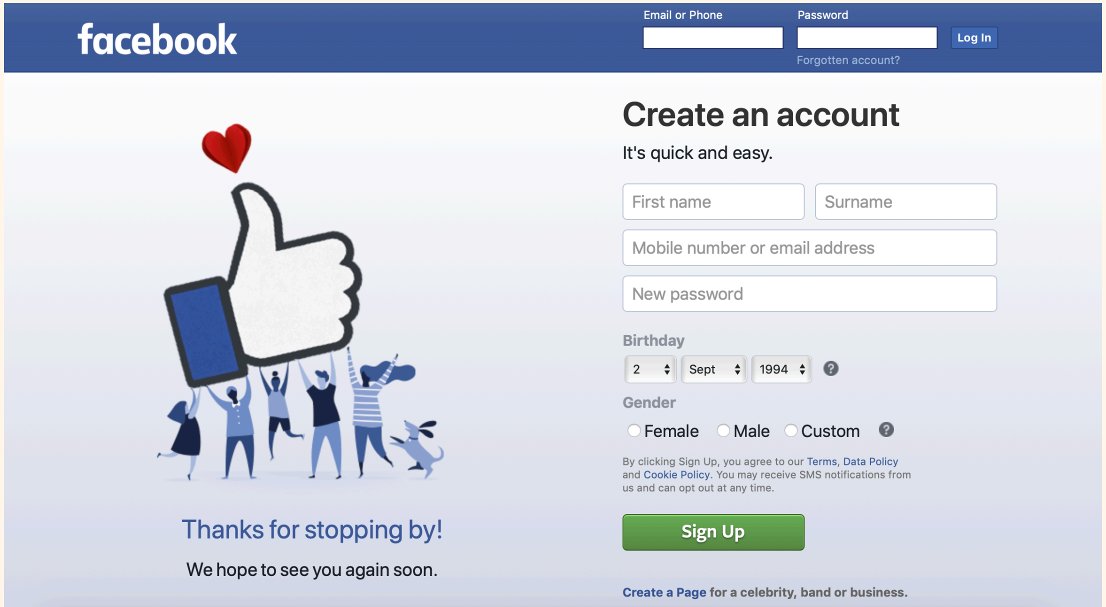
At least 6 characters

i Passwords must be at least 6 characters.

Re-enter password

Create your Amazon account

By creating an account, you agree to Amazon's Conditions of Use and Privacy Notice.



facebook

Email or Phone

Password

Log In

Forgotten account?

Create an account

It's quick and easy.

First name

Surname

Mobile number or email address

New password

Birthday

2 | Sept | 1994

Gender

○ Female ○ Male ○ Custom

By clicking Sign Up, you agree to our Terms, Data Policy and Cookie Policy. You may receive SMS notifications from us and can opt out at any time.

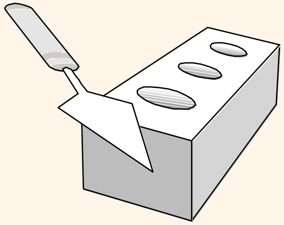
Sign Up

Create a Page for a celebrity, band or business.

Thanks for stopping by!

We hope to see you again soon.

Data Models



- Metadata and Relational Data

sid	name	dept	gpa	Course enrolled
00001	Jones	comm	3.4	COMM7330 Basic Programming, COMM7360 Big Data, COMM7340 AI
00002	Joe	comm	3.2	COMM7340 AI, COMM7360 Big Data, COMM7330 Basic Programming
00003	Smith	math	3.8	COMM7360 Big Data

Instances and Schemas

- *Relation:* made up of 2 parts
  - *Schema*: specifies name of relation, plus name and type of each column
    - The logical structure of the database
    - E.g., Students(*sid*: string, *name*: string, *dept*: string, *gpa*: real)
  - *Instance*: a table with rows and columns
    - The actual content of the database at a particular point in time

Example University Database

Students

sid	name	dept	gpa
00001	Jones	comm	3.4
00002	Joe	comm	3.2
00003	Smith	math	3.8

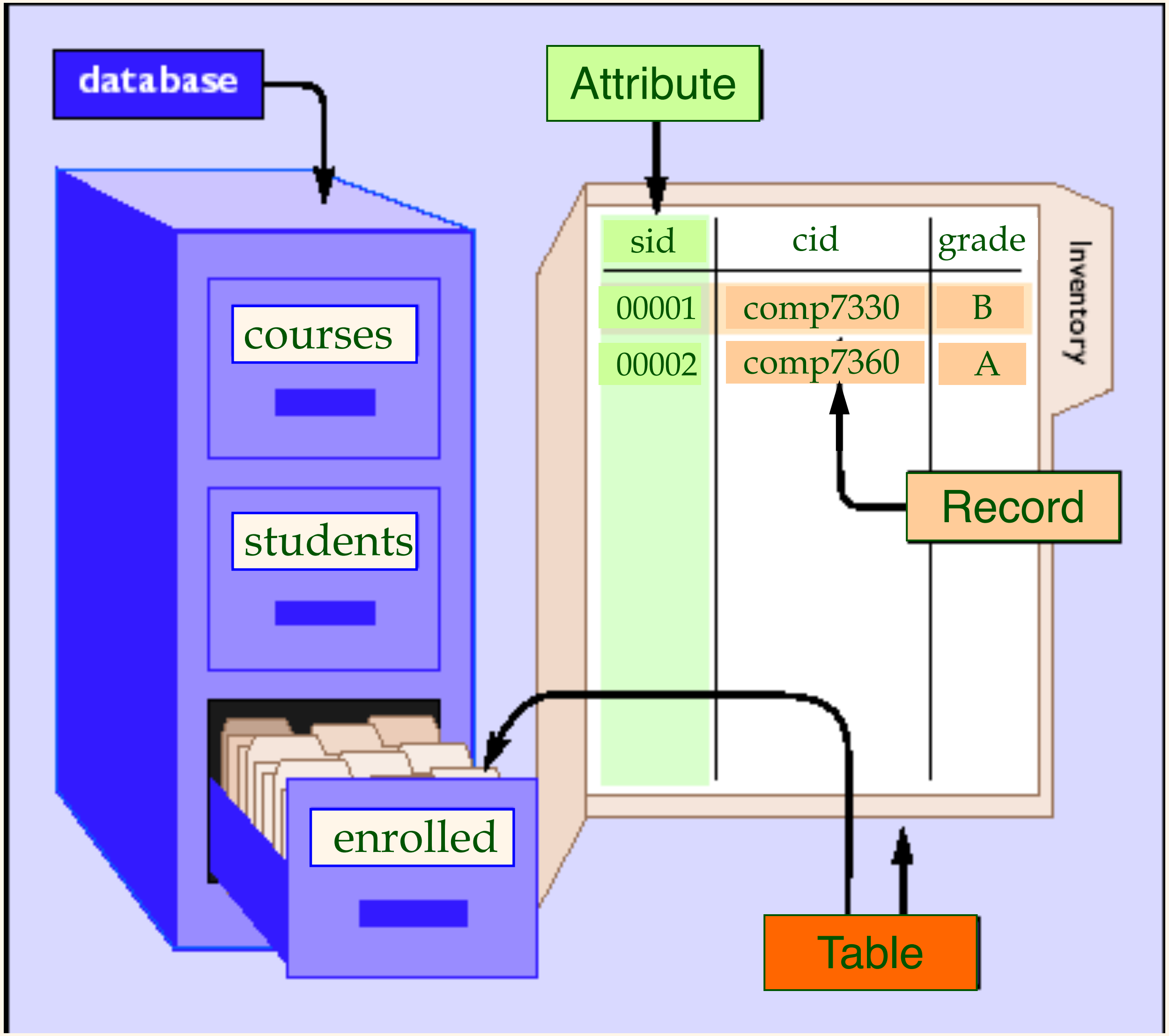
Courses

cid	cname	credit
COMM7330	Program	3
COMM7360	Big data	3
COMM7340	AI	3

Enrolled

sid	cid	grade
00001	COMM7330	B
00002	COMM7360	A

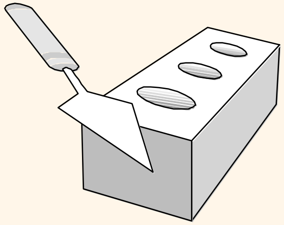
Database concepts



sid	cid	grade
00001	comp7330	B
00002	comp7360	A

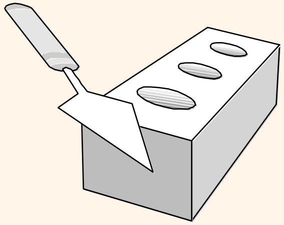
- ❖ A database consists of one or more tables
- ❖ Each table is made up of a number of records (a.k.a tuples)
- ❖ Each record contains several attributes

Levels of Abstraction



- Physical Level:
  - Relations stored as unordered files.
  - The second column of Students is indexed by a *B-tree*.
- Conceptual Level:
  - *Students(sid: string, name: string, dept: string, gpa:real)*
  - *Courses(cid: string, cname:string, credits:integer)*
  - *Enrolled(sid:string, cid:string, grade:string)*
- View:
  - *Course_info(cname:string,enrollment:integer)*

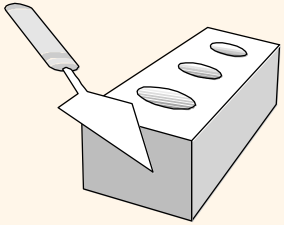
Data Independence



- Applications are insulated from how data is structured and stored.
- *Physical data independence*: Protection from changes in *physical* structure of data.
  - Logical structures (e.g., tables) are supported by different physical storage structures.
  - Applications designed on conceptual level.

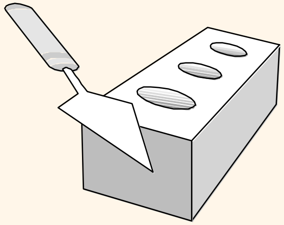
This is one of the most important benefits of using a DBMS!

Queries in A DBMS



- Some questions a user might ask:
  - What is the student id of John?
  - How many students are enrolled in COMM7360?
- Questions involving the data stored in a DBMS are called *queries*.
- A *query language* is used to pose queries.
- *Structural Query Language (SQL),* which supports a rich class of queries, has contributed greatly to the success of relational DBMS.

Example SQL Query



- List the names of all students enrolled in COMM7360.

```
SELECT name
FROM   Students, Enrolled
WHERE  Students.sid = Enrolled.sid
       AND cid = ‘COMM7360’;
```

Example University Database

Students

sid	name	dept	gpa
00001	Jones	comm	3.4
00002	Joe	comm	3.2
00003	Smith	math	3.8

Courses

cid	cname	credit
COMM7330	Program	3
COMM7360	Big data	3
COMM7340	AI	3

Enrolled

sid	cid	grade
00001	COMM7330	B
00002	COMM7360	A

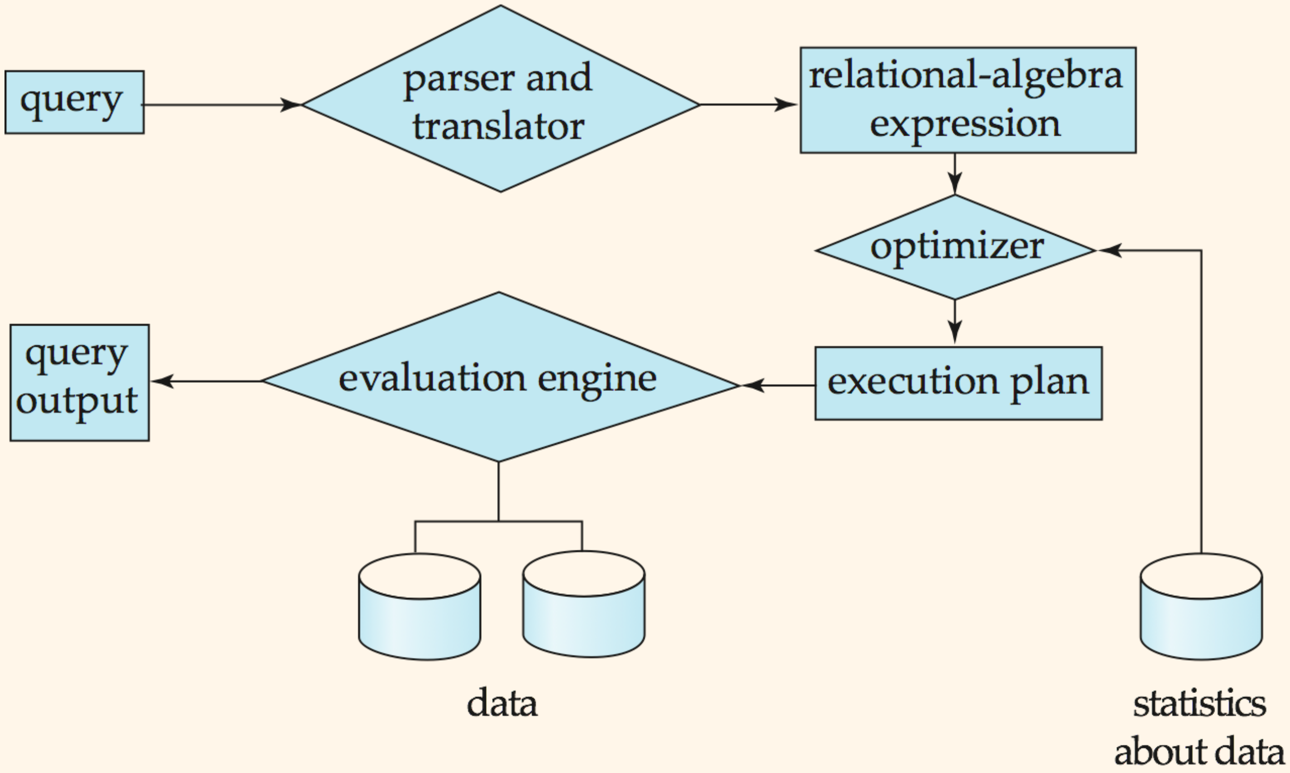
Query Optimization

- Evaluation of the SQL query:
  - Compute the product
  - Perform selection

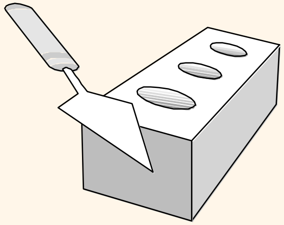
sid	name	dept	gpa	(sid)	cid	grade
00001	Jones	comm	3.4	00001	COMM7330	B
00002	Joe	comm	3.2	00001	COMM7330	B
00003	Smith	math	3.8	00001	COMM7330	B
00001	Jones	comm	3.4	00002	COMM7360	A
00002	Joe	comm	3.2	00002	COMM7360	A
00003	Smith	math	3.8	00002	COMM7360	A

Improvement?

Query Evaluation



Transaction



- A *transaction* is a set of actions that access the database (in response to real-world event)
- E.g., transfer *amount* from *acct1* to *acct2*
- Relation:

  accounts(*account_number*: string, *balance*: real)

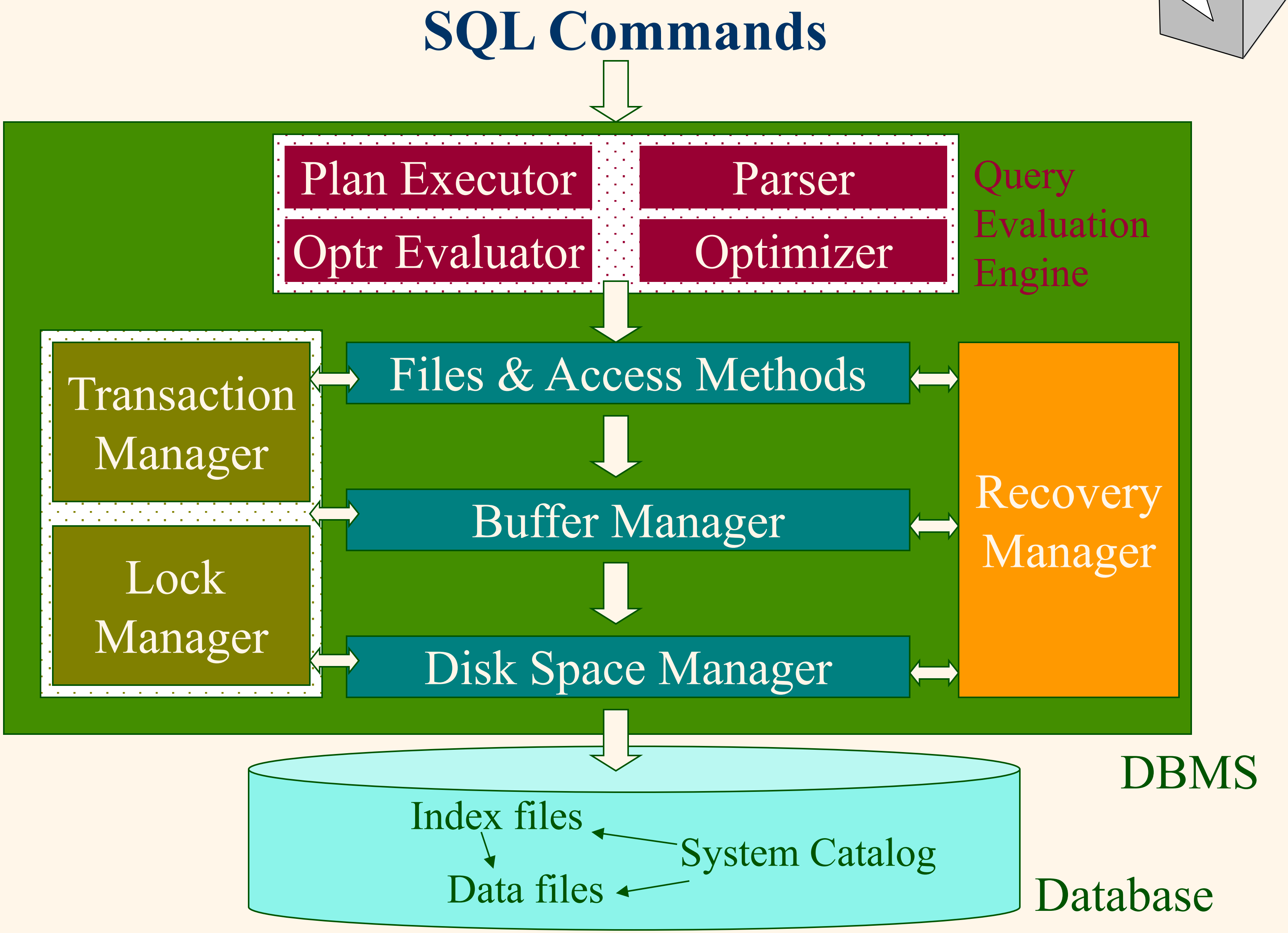
```
begin transaction;

update accounts
set balance = balance - :amount ;
where account_number = :acct1;

update accounts
set balance = balance + :amount;
where account_number = :acct2;
```

Example Transaction

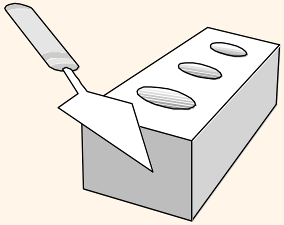
```
begin transaction;

update accounts
set balance = balance - :amount ;
where account_number = :acct1;

update accounts
set balance = balance + :amount;
where account_number = :acct2;
```

- Issues: concurrent access, system crash?

Architecture of a DBMS

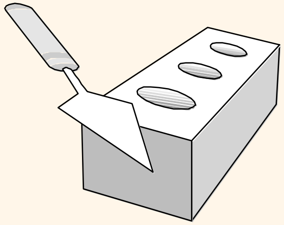


Folks interact with Databases



- End users
  - You and me…
- DB application programmers
  - E.g., webmasters
- *Database administrator (DBA)*
  - Designs logical/physical schemas
  - Handles security and authorization
  - Data availability, crash recovery
  - Database tuning as needs evolve
- DBMS vendors
  - IBM, Oracle, Microsoft…

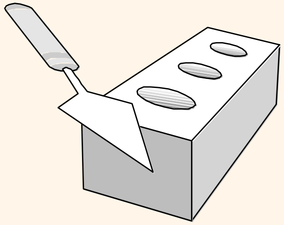
Must understand how a DBMS works!



Summary

- DBMS used to maintain, query large datasets.
- Benefits include recovery from system crashes, concurrent access, quick application development, data integrity and security.
- Levels of abstraction give data independence.
- A DBMS typically has a layered architecture.
- DBAs hold responsible jobs and are well-paid!
- Database area is one of the broadest, most exciting areas in CS.

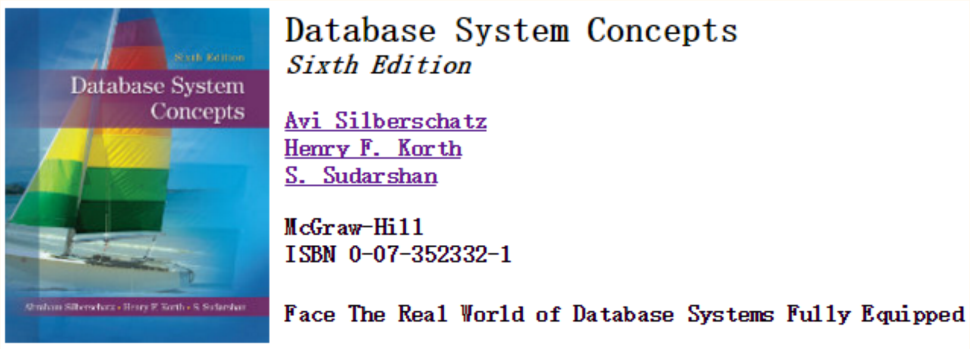
Course Outline



- Week 1, 2
  - Relational Model
- Week 3
  - Relational algebra
- Week 4, 5
  - Basic SQL
- Week 6
  - Advanced SQL
- Week 7
  - SQLite & Python

Textbook

Textbook



Database System Concepts
Sixth Edition

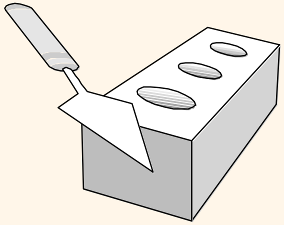
Avi Silberschatz
Henry F. Korth
S. Sudarshan

McGraw-Hill
ISBN 0-07-352332-1

Face The Real World of Database Systems Fully Equipped

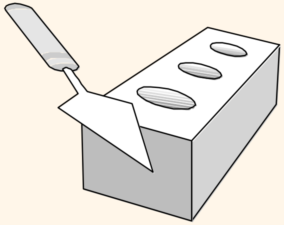
References

- H. Garcia-Molina, J. D. Ullman, J. Widom, Database System Implementation, Prentice Hall.
- R. Elmasri & S. Navathe, Fundamentals of Database Systems, Addison Wesley.
- R. Ramakrishnan and J. Gehrke, Database Management Systems, McGraw-Hill



Assessment

- Continual Assessment: 40%
  - Class participation: 25%
  - Assignments: 35%
- Final Exam: 40%



Guidelines

Avoiding Plagiarism

Unless otherwise stated, all work submitted by you should be your own. Copying or sharing of assignments would constitute cheating. If there is any doubt about the appropriateness of your actions, please contact the instructor for explicit clarification. Cheating is an offense and will result in appropriate disciplinary action against those involved.