

CS 166 Database Management



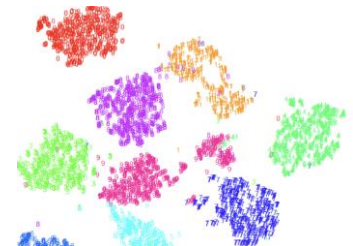
Introductions ...

Prof. Mariam Salloum

Office: Bourns Hall A (Room 159B)

Email: msalloum@cs.ucr.edu

- Current research work on Data Integration and Big Data Visualization.
 - Given millions of relevant sources/sites to a particular query, how to identify relevant sources, query sources in optimized fashion then perform record linkage and data fusion.
 - How to visualize large datasets in high dimensions.



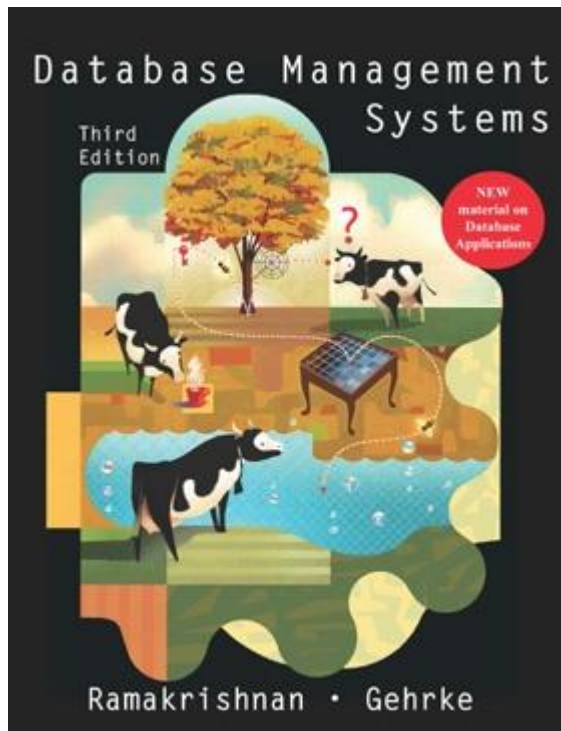
Textbook

<http://pages.cs.wisc.edu/~dbbook/>

Contains many solved exercises:

<http://pages.cs.wisc.edu/~dbbook/openAccess/thirdEdition/solutions/ans3ed-odonly.pdf>

3rd Edition



2nd Edition



Course Logistics

- **Website / iLearn :**
 - Will be used to post grades
- **Lectures / Labs**
 - **Lecture:** MWF 2:00PM - 2:50PM WCH 142
 - **Lab:** F 8:00 AM - 10:50 AM , 11:00 AM - 01:50 PM
- **My Info:**
 - **Office:** Bourns Hall A (Room 159B)
 - **Email:** msalloum@cs.ucr.edu
 - **Website:** www.cs.ucr.edu/~msalloum
 - **Office Hours:** MW 1-2 and by appointment

Requirements & Grading

Item	Percentage	Notes
Midterm	20%	In-class midterm. Closed book, notes, etc.
Final Exam	20%	Cumulative final exam
Quizzes	10%	Mini in-class quizzes (usually on Fridays)
Project	25%	A 3-stage project to create a database for a given application from scratch.
Labs	25%	<p>Each week we will have a lab assignment that will enforce the concepts covered in lecture.</p> <p>Labs can be turned in up-to 1 week late and must be demoed to the TA.</p>

Course Goals

- DBMS architecture
- ER-design
- Relational model
- Relational query languages
- Implementation issues (indexing, hashing)
- Query optimization
- Advanced topics

Course Syllabus

- Ch.1: Overview of Database Systems
- Ch.2: Introduction to Database Design
- Ch.3: The Relational Model
- Ch.4: Relational Algebra
- Ch.5: SQL
- Ch.8: Storage and Indexing
- Ch.9: Storing Data: Disks and Files
- Ch.10: Tree-Structured Indexing
- Ch.11: Hash-Based Indexing
- Ch.12: Overview of Query Evaluation
- Ch.13: External Sorting
- Ch.14: Evaluation of Relational Operators
- Ch.15: A Typical Relational Query Optimizer

Cheating Policy

Students must read and understand UCR policy on academic honesty. Look at:

<https://conduct.ucr.edu/policies/academic-integrity-policies-and-procedures>

Read the Guidelines & Definitions, and download the *Academic Integrity Student Brochure*

Anyone caught cheating will get a final grade of **F** and may have a letter placed in his/her permanent record. Students are expected to take care that others cannot “cheat off them”. For example, if your homework is left on a shared hard drive and someone else hands it in, you are liable and will have your grade adjusted downward.

Classroom Attendance

- Attendance to class is expected.
 - It is hard for someone to do well in this class without regular attendance.
 - We will do in-class worksheets and activities that reenforce lecture concepts, which will prepare you for the midterm(s).
- **Lab attendance is mandatory**
 - Let the TA and myself know if you miss lab due to an illness or another excused reason.

Labs and projects must contain a coversheet **in** the format shown here.

Any text, URL, or person consulted must be referenced.

Lab 1	CS 166 Database Management Systems
Jane Smith	Instructor: M. Salloum
ID 860-01-1234	Section 21
jane@cs.ucr.edu	TA: xxxx
January 1, 2020	

In completing this project I consulted...

- Nilsson, N.J.: Principles of Artificial Intelligence. Tioga Publishing Co. 1980 Pages 72-84 (for clarification on the A* algorithm).
- Fellow student Bingo Little, who is my study partner. He showed me how to test for infinite sets in Matlab.
- <http://yoda.cis.temple.edu:8080/UGAIWWW/lectures95/search/alpha-beta.html> (for a more detailed explanation of the Alpha Beta Pruning algorithm).

Motivation

- The world is drowning in data
 - Affects almost every app / service
- Need professionals to help manage it
 - Help domain scientists achieve new discoveries
 - Help companies provide better services
 - Help governments become more efficient
- CS 166 – Data Management
 - Covers both principles and tools

How much data?

- Google
 - 24PB data processed daily
- Walmart
 - 2.5 petabytes every hour
- Twitter
 - 500 million tweets per day
 - 1.6 billion search queries
 - 7 TB of data generated per day
- Facebook
 - 500TB of data generated per day
- Weather Satellites
 - Estimated to generate 10TB per day (Ex. NPOESS)

Data Management is Universal

- Managing data is at the core of most apps / services
 - Whether they store small or large amounts of data
 - Whether they are modern systems or older ones
- Hard problem even with small amount of data
 - We'll see and discuss examples later in the course
- Doing it right typically makes everything else much easier

Intro to Databases

- What is a database ?
 - Is an Excel / CSV file a database?
 - A collection of files storing related data
- Examples of databases
 - Bank accounts database
 - Payroll database
 - Amazon products database
 - Airline reservation database
 - Browsing history
 - Student scheduling system

Can I just use files to store data???

Student File

Jane Smith , 123 boxwood 91823
John Smith , 451 lemonsirl 91709
Kate Aron , 925 buttonwood 91703

Winter 2020 Schedule

Jane Smith , CS166 , Database Management , TH
John Smith , CS105 , Intro to DS , MW
John Smith , CS166 , Database Management , TH
Kate Aron , CS172 , Information Retrieval, MWF

What happens if Kate drops CS 172? Do we lose all information about CS172??

Can I just use files to store data???

Winter 2020 Schedule

Student File

Jane Smith , 123 boxwood 91823
John Smith , 451 lemonsirl 91709
Kate Aron , 925 buttonwood 91703

Jane Smith , CS166 , Database Management , TH
John Smith , CS105 , Intro to DS , MW
John Smith , CS166 , Database Management , TH
Kate Aron , CS172 , Information Retrieval, MWF

Course File

CS166 , Database management, TH
CS 105 , CS105 , Intro to DS , MW
CS 172 , CS172 , Information Retrieval, MWF

What happens if we want to rename the course ?

Do we only do it in one place, or do we have to do it in many places?

What is an Anomaly?

- Definition
 - Problems that can occur in poorly planned, un-normalized databases where all the data is stored in one table
 - Ex. One big file
- Types of anomalies
 - Insert
 - Delete
 - Update

Insert Anomaly

- An **Insert Anomaly** occurs when certain attributes cannot be inserted into the database without the presence of other attributes.

Course_no	Tutor	Room	Room_size	En_limit
353	Smith	A532	45	40
351	Smith	C320	100	60
355	Clark	H940	400	300
456	Turner	H940	400	45

e.g. we have built a new room (e.g. B123) but it has not yet been timetabled for any courses or members of staff.

Delete Anomaly

- A **Delete Anomaly** exists when certain attributes are lost because of the deletion of other attributes.

Course_no	Tutor	Room	Room_size	En_limit
353	Smith	A532	45	40
351	Smith	C320	100	60
355	Clark	H940	400	300
456	Turner	H940	400	45

e.g. if we remove the entity, course_no:351 from the above table, the details of room C320 get deleted. Which implies the corresponding course will also get deleted.

Update Anomaly

- An Update Anomaly exists when one or more instances of duplicated data is updated, but not all.

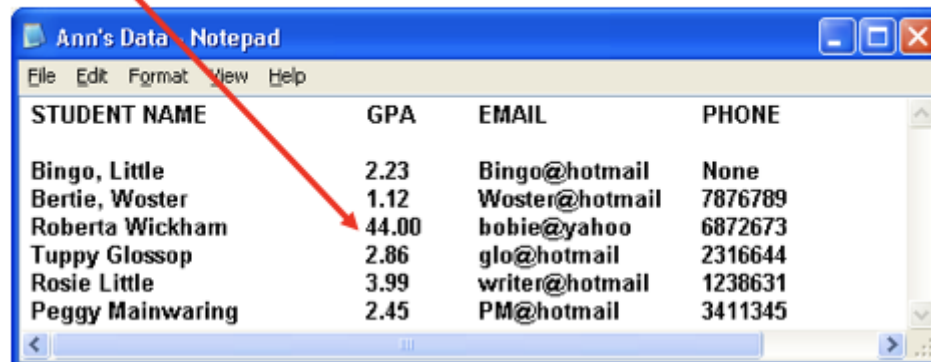
Course_no	Tutor	Room	Room_size	En_limit
353	Smith	A532	45	40
351	Smith	C320	100	60
355	Clark	H940	400	300
456	Turner	H940	400	45

e.g. Room H940 has been improved, it is now of RSize = 500. For updating a single entity, we have to update all other columns where room=H940.

Enforcing Constraints

- With the simple file solution there is no way to enforce integrity constraints on the data. In other words people can put bad data into the text file.
- In contrast, a DBMS allows us to enforce all kinds of constraints. This really helps (but does not guarantee) that our data is correct.

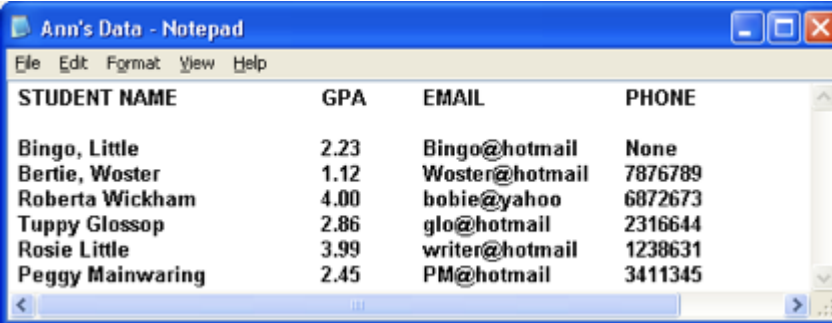
A typo gives Roberta Wickham a GPA of 44.00



STUDENT NAME	GPA	EMAIL	PHONE
Bingo, Little	2.23	Bingo@hotmail	None
Bertie, Woster	1.12	Woster@hotmail	7876789
Roberta Wickham	44.00	bobie@yahoo	6872673
Tuppy Glossop	2.86	glo@hotmail	2316644
Rosie Little	3.99	writer@hotmail	1238631
Peggy Mainwaring	2.45	PM@hotmail	3411345

Scalability

- The simple file solution might work for small datasets. What happens when we have big datasets?
- Most real world datasets are so large that we can only have a small fraction of them in main memory at any time, the rest has to stay on disk.
- Even if we had lots of main memory, with 32 bit addressing we can only refer to 4GB of data!

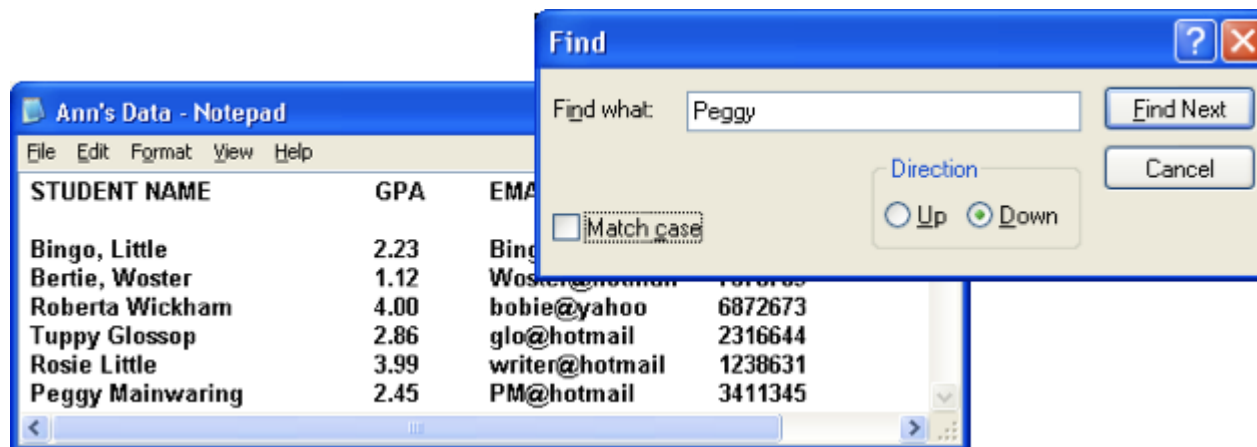


A screenshot of a Windows Notepad window titled "Ann's Data - Notepad". The window contains a table with four columns: STUDENT NAME, GPA, EMAIL, and PHONE. The table lists six students: Bingo, Little; Bertie, Woster; Roberta Wickham; Tuppy Glossop; Rosie Little; and Peggy Mainwaring. Each student has a corresponding GPA, email address, and phone number. The window has a standard menu bar with File, Edit, Format, View, and Help. The table is displayed in a simple text format with no borders.

STUDENT NAME	GPA	EMAIL	PHONE
Bingo, Little	2.23	Bingo@hotmail	None
Bertie, Woster	1.12	Woster@hotmail	7876789
Roberta Wickham	4.00	bobie@yahoo	6872673
Tuppy Glossop	2.86	glo@hotmail	2316644
Rosie Little	3.99	writer@hotmail	1238631
Peggy Mainwaring	2.45	PM@hotmail	3411345

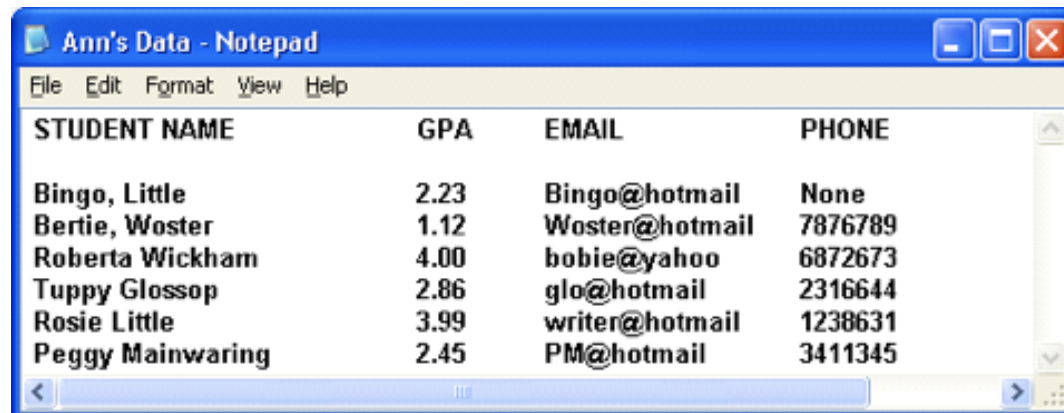
Query Expressiveness

- The simple file solution would allow me to search for keywords or certain numbers (slowly).
- With a DBMS I can search with much more expressive queries. For example I can ask.. *“Find all students whose GPA is greater than 2.5, and who don’t own a phone”* or *“what is the average GPA of the students”*



Query Expressiveness II

- I could write some program that might allow more expressive queries on my text file, but it would be tied into the structure of my data and the operating system etc..
- With a DBMS we are completely isolated from the physical structure of our data. If we change the structure of our data (by adding a field, for example) or moving from a PC to a Mac, nothing changes at the front end!

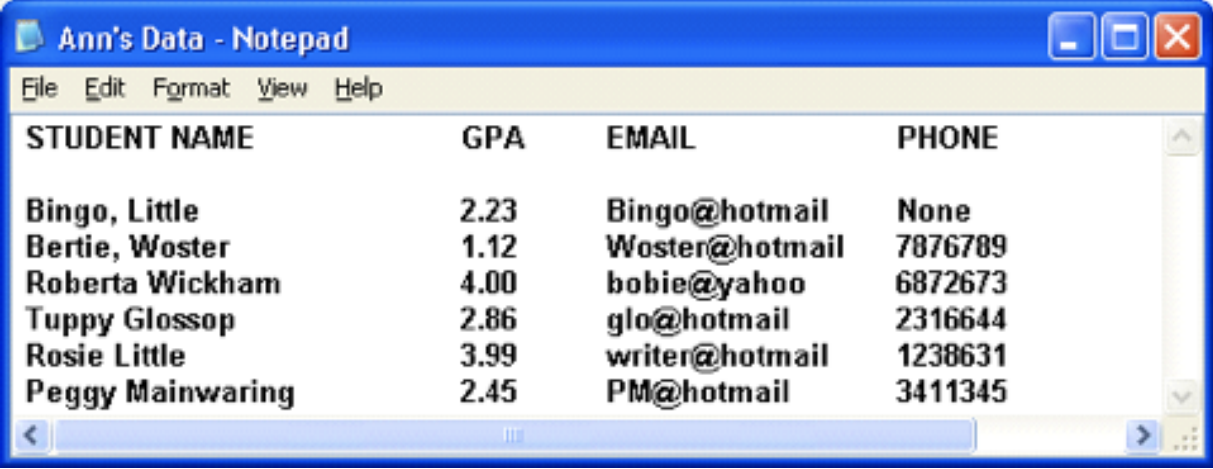


A screenshot of a Notepad window titled "Ann's Data - Notepad". The window displays a table with four columns: STUDENT NAME, GPA, EMAIL, and PHONE. The table contains six rows of student data. The window has a standard menu bar with File, Edit, Format, View, and Help. The table is displayed in a monospaced font, and the window has a blue title bar and standard Windows window controls.

STUDENT NAME	GPA	EMAIL	PHONE
Bingo, Little	2.23	Bingo@hotmail	None
Bertie, Woster	1.12	Woster@hotmail	7876789
Roberta Wickham	4.00	bobie@yahoo	6872673
Tuppy Glossop	2.86	glo@hotmail	2316644
Rosie Little	3.99	writer@hotmail	1238631
Peggy Mainwaring	2.45	PM@hotmail	3411345

Different Views

- With a DBMS I can arrange for different people to have different views of the data. For example, I can see everything, a student can see only his/her data, the TA can see...

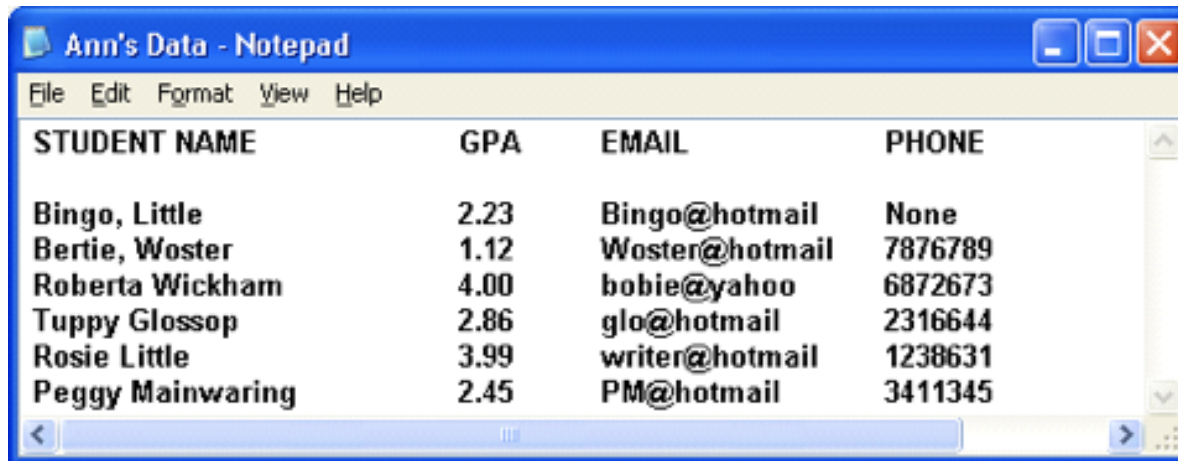


A screenshot of a Notepad window titled "Ann's Data - Notepad". The window contains a table with four columns: STUDENT NAME, GPA, EMAIL, and PHONE. The table lists six students: Bingo, Little; Bertie, Woster; Roberta Wickham; Tuppy Glossop; Rosie Little; and Peggy Mainwaring. The window has a menu bar with File, Edit, Format, View, and Help. The table is displayed in a simple text format with no borders.

STUDENT NAME	GPA	EMAIL	PHONE
Bingo, Little	2.23	Bingo@hotmail	None
Bertie, Woster	1.12	Woster@hotmail	7876789
Roberta Wickham	4.00	bobie@yahoo	6872673
Tuppy Glossop	2.86	glo@hotmail	2316644
Rosie Little	3.99	writer@hotmail	1238631
Peggy Mainwaring	2.45	PM@hotmail	3411345

Concurrency

- Suppose I leave my text file on UNIX account, and I log in and begin to modify it at the same time my TA is modifying it!
- A DBMS will automatically make sure that this kind of thing cannot happen.

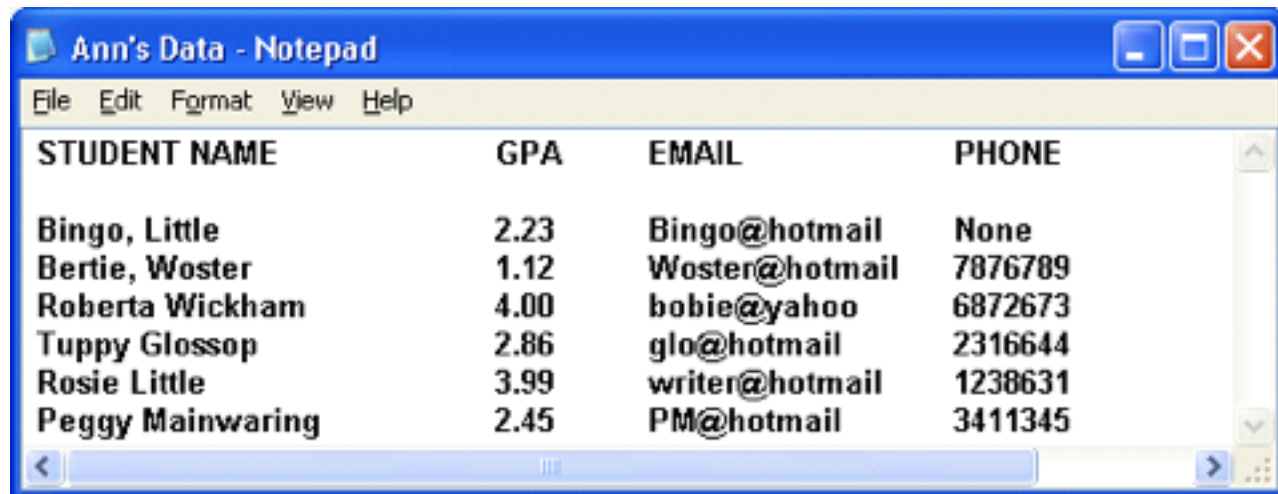


A screenshot of a Windows Notepad window titled "Ann's Data - Notepad". The window has a menu bar with "File", "Edit", "Format", "View", and "Help". The text content is a table with four columns: "STUDENT NAME", "GPA", "EMAIL", and "PHONE". The table contains six rows of student data. The window has standard Windows window controls (minimize, maximize, close) in the top right corner and a scrollbar on the right side.

STUDENT NAME	GPA	EMAIL	PHONE
Bingo, Little	2.23	Bingo@hotmail	None
Bertie, Woster	1.12	Woster@hotmail	7876789
Roberta Wickham	4.00	bobie@yahoo	6872673
Tuppy Glossop	2.86	glo@hotmail	2316644
Rosie Little	3.99	writer@hotmail	1238631
Peggy Mainwaring	2.45	PM@hotmail	3411345

Security

- Suppose I leave my text file on UNIX account, and a student hacks in and changes their grades...
- A DBMS will allow multiple levels of security.

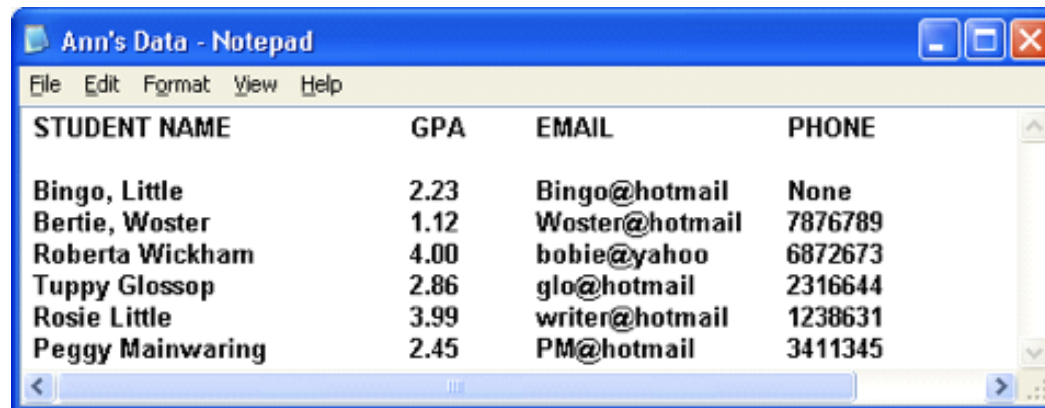


A screenshot of a Notepad window titled "Ann's Data - Notepad". The window has a menu bar with "File", "Edit", "Format", "View", and "Help". The text content is a table with four columns: "STUDENT NAME", "GPA", "EMAIL", and "PHONE". The data rows are as follows:

STUDENT NAME	GPA	EMAIL	PHONE
Bingo, Little	2.23	Bingo@hotmail	None
Bertie, Woster	1.12	Woster@hotmail	7876789
Roberta Wickham	4.00	bobie@yahoo	6872673
Tuppy Glossop	2.86	glo@hotmail	2316644
Rosie Little	3.99	writer@hotmail	1238631
Peggy Mainwaring	2.45	PM@hotmail	3411345

Crash Recovery

- Suppose I am editing my text file and the system crashes!
- A DBMS is able to guarantee 100% recovery from system crashes.



A screenshot of a Windows Notepad window titled "Ann's Data - Notepad". The window has a menu bar with "File", "Edit", "Format", "View", and "Help". The text content is a table with four columns: "STUDENT NAME", "GPA", "EMAIL", and "PHONE". The table contains six rows of student data. The window has standard Windows window controls (minimize, maximize, close) in the top right corner and a scrollbar on the right side.

STUDENT NAME	GPA	EMAIL	PHONE
Bingo, Little	2.23	Bingo@hotmail	None
Bertie, Woster	1.12	Woster@hotmail	7876789
Roberta Wickham	4.00	bobie@yahoo	6872673
Tuppy Glossop	2.86	glo@hotmail	2316644
Rosie Little	3.99	writer@hotmail	1238631
Peggy Mainwaring	2.45	PM@hotmail	3411345

Database Management Systems

- 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.
 - Built to eliminate insert, delete, update anomalies
- Examples of DBMS
 - Oracle
 - Microsoft SQL Server
 - MySQL (opensource)
 - PostgreSQL (opensource) ** Used for this course **
 - SQLite
 - BigTable
 - Teradata
 - AsterixDB

An Example: Online Bookseller

- What data do we need?
 - Maybe data about books, customers pending orders, order histories, trends, references, 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 and bugs.

Multi-User Issues

- Jane and John both share an account with a gift certificate (credit) of \$200
 - 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 the server crashes ?
 - What if the data center goes offline?

Required Functionality for Data Management

- Describe real-world entities in terms of stored data
- Persistently store large datasets
- Efficiently query and update
 - Must handle complex questions about data
 - Must handle sophisticated updates
 - Performance matters
- Easily change structure
- Enable simultaneous updates
- Crash recovery
- Security and integrity

Different Database Related Roles

- **DB application developer** – writes programs that query and modify data
- **DB Designer** – establishes schema
- **DB administrator** – loads data, tunes system, keeps whole thing running
- **Data Analyst** – data mining, data cleaning, data integration, ETL
- **DBMS implementer** – builds the DBMS

Timeline of Databases

- **1960s** – hierarchical databases which provided support for concurrency, recover, and fast access.
- **1970-1972** - Edgar Codd who was working at IBM proposed the 'relational database model'. Provided support for more reliability, less redundancy, more flexibility, etc.
- **1970s** – two major RDBMS prototypes were proposed: Ingres and System R
- **Mid 1970s** – A DB model called Entity –Relationship(ER) was proposed
- **1980s** – Structured Query Language (SQL) became standard querying language.
- **Late 1980s - 1990s** – Parallel and distributed databases