Welcome

Tuesday, January 18, 2022 8:44 PM

Welcome to CS-GY 6083:
Principles of Database Systems

Sections INET

Spring 2024

Prof Phyllis Frankl
pfrankl@nyu.edu

Pronouns: she/her/hers

Overview of Week 1

Tuesday, January 18, 2022 8:49 PM

Today's class

- 1A (these slides):
  - Motivation for Database Systems
  - Course Logistics
  - Overview of DB system and its users
- 1B (coming soon)
  - Entity Relationship Model

Motivation 1

Tuesday, January 18, 2022 8:52 PM

How would you approach this programming exercise in a conventional language (e.g. Java)?

- Given:
  - A file containing
    - users' IDs
    - Title and release year of movies they've watched
    - Ratings they've given these movies (0 to 5 stars)
  - Title and year of a movie input interactively
- Find
  - The average rating of the movie

Pause the recording and take a few minutes to jot down:

- High level outline of how you would approach this in your favorite imperative language
- Some info you would need to know in order to fully implement this
- Issues you'd have to deal with

Report back on average ratings of a movie problem

Tuesday, January 18, 2022 9:01 PM

My outline

Tuesday, January 18, 2022 9:01 PM

Assume the file is organized into lines, each with userID, title, year, number of stars, separated by some delimitter

- Initialize:
  - Open file, initialize counters, …
- For each line
  - Read line
  - Parse line

    lots of details
  - If title == 'Little Women' and year == 2019 and this line includes a rating and other stuff about the line is valid …
    - Update total_rating and update count
- If everything's OK
    - Compute average
    - Format output and write it out
- Handle exceptional cases

Issues

Tuesday, January 18, 2022 9:08 PM

- Lots of low-level parsing details:
  - Types
  - Delimiters
  - Missing or malformed data
  - Case sensitivity?
- Duplicates, mis-spellings, …

- Additional issues if this is important or sensitive data:
  - Robustness in case of system failure
  - Handling concurrent updates of the file
  - Security and access control

Relational data example

Tuesday, January 18, 2022 9:13 PM

Saw

ID	title	year	stars
12345	Little Women	2019	5
12345	Little Women	1994	4
54321	Little Women	2019	4
54321	Finding Dory	2016	3

SQL query

Tuesday, January 18, 2022 9:17 PM

```
SELECT AVG(stars)
FROM Saw
WHERE title = 'Little Women' AND year = 2019
```

Returns

AVG(stars)
4.5

We're Assuming here that the user entered the title and year 'Little Women', 2019;

We'll also need a little interface code (e.g. web interface?) to get the input and plug it into the query

A harder sample problem

Tuesday, January 18, 2022 9:21 PM

How would you approach this programming exercise in a conventional language (e.g. Java)?

- Given:
  - A file containing
    - users' IDs
    - Title and release year of movies they've watched
    - Ratings they've given these movies (0 to 5 stars)
  - Another file containing data about users, including their dates of birth
  - Title and year of a movie input interactively
- Find
  - The average rating people born after 2000 gave the movie

Take a few minutes to jot down:

- High level outline of how you would approach this in your favorite imperative language
- Issues you'd have to deal with beyond those in the first problem and possible approaches

Report back

Tuesday, January 18, 2022 9:25 PM

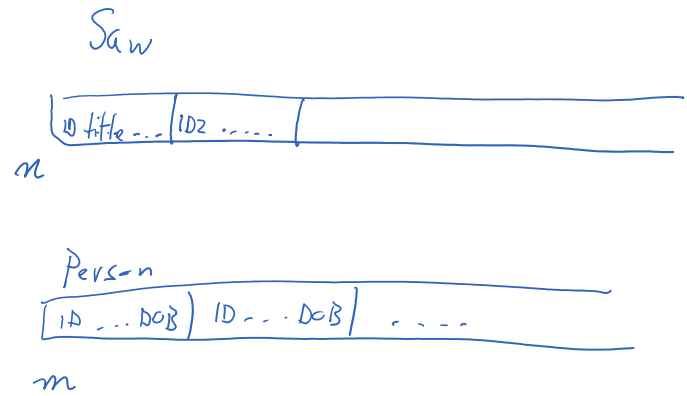
My answer

Tuesday, January 18, 2022 9:25 PM

Issues:

- We need to store and access the data from one of the files
- We need to match IDs from the two files so we can see whether a given line from the file with the ratings is relevant

- Which data structure should be used?
  - Can we avoid *O(nm)* where n is the number of viewings/ratings and m is the number of users?
  - Is it worth pre-processing the data (e.g. sorting or building a binary search tree or hash table) in order to access it more efficiently?



Answer:

It depends

Further issue: memory management

- What if the files are both very large and we can't fit either of them into main memory?

Sample data in the relational model

Tuesday, January 18, 2022 9:33 PM

Saw

ID	title	year	stars
12345	Little Women	2019	5
12345	Little Women	1994	4
54321	Little Women	2019	4
54321	Finding Dory	2016	3
67890	Little Women	2019	5
67890	Finding Dory	2015	3

person

ID	fname	lname	DOB
12345	Sally	Li	2005-03-24
54321	Joe	Smith	1995-01-07
67890	Ravi	Khan	1998-04-28

SQL query

Tuesday, January 18, 2022 9:40 PM

```
SELECT AVG(stars)
FROM Saw NATURAL JOIN Person
WHERE title = 'Little Women'
    AND yeare = 2019
    AND DOB >= 2000-01-01
```

How can the SQL queries be so simple?

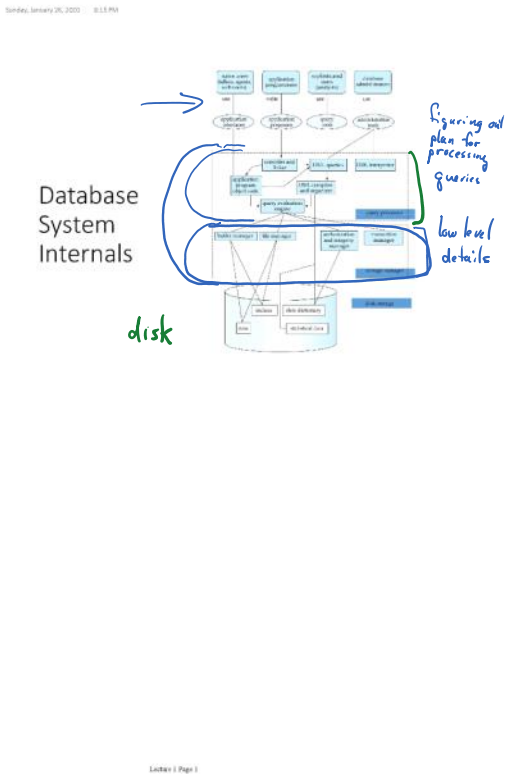
Tuesday, January 18, 2022 9:40 PM

- High level model of the data
- Very high level Query Language
- System support for
  - Some aspects of data integrity
  - Choice of algorithms for matching and searching
  - Memory management
  - Transaction management
    - Concurrency
    - System failure
  - Indexing (data structures to facilitate search)
  - Evaluation and selection of "best" algorithm

Database System Internals

Tuesday, January 18, 2022 9:43 PM

db-system-fig



Overview of Topics

Tuesday, January 18, 2022 9:51 PM

- Data Modeling
  - Entity Relationship Model — more expressive model
  - Relational Model — data organized into tables
- Query Languages
  - Relational algebra
  - Overview of relational calculus

  } simpler "pure" languages
  - SQL — more user-friendly, practical language based on relational algebra
- Application Programming
- How Database Management systems work
  - Storage
  - Indexing
  - Query Processing
  - Transaction Management
  - …

Requirements

Tuesday, January 18, 2022 9:46 PM

- Weekly reading assignments
- About 5 or 6 homework assignments
- Engagement (seeking help when needed, …)
- Project
  - Web based database application program, to be specified
  - Python or Java or … + SQL and a little HTML
  - Three parts
  - Work alone or in small group
    - Group projects will have additional requirements
- Midterm Exam and cumulative final exam

Prerequisites

Tuesday, September 6, 2022 11:57 AM

Familiarity with some undergraduate discrete math and algorithms topics :

- sets
- relations
- big-Oh notation
- Search trees
- Hashing

Ability to program in Python, Java, or some other similar language and to learn new APIs and a little HTML through self-study

No prior experience with databases is expected

Other important info

Monday, September 5, 2022 2:35 PM

- Textbook web page, including authors' slides:
  - https://db-book.com/
- Check for announcements every day
- See syllabus for procedure if you are missing a required exam or deadline due to illness
- I will record the class on zoom and post recording; I will usually announce the time when I'll be recording and allow students to attend synchronously and participate in real time. (Generally on Mondays from 11:00 a.m. to 1:30 p.m)
- I may occasionally record the lectures at other times
- Unless otherwise noted homework and project work will be accepted up to 24 hours late, but with a 20% penalty
- See syllabus re academic accommodations
- See syllabus re academic honesty
  - You may discuss general concepts of how to do homework problems with classmates, but should write up and hand in your own work

Overview of Tentative Schedule

Monday, September 5, 2022 1:22 PM

https://docs.google.com/spreadsheets/d/1C3uVAnA_FV-zwCPPyhc4Q8fDStng5S4_mLtPGRBjtzM/edit?usp=sharing

Communication / Getting Help

Monday, September 5, 2022 1:38 PM

Please ask questions!

Office Hours:

Prof Frankl: TBA

other times by appointment

Teaching Assistants: time and location to be announced

Electronic communication:

BrightSpace: announcements, assignments, lecture notes, …

EdStem: link posted on BrightSpace

Questions about course material and logistics

- General questions should be public
- Specific questions revealing partial solutions to problems should be to instructor and Tas
- e-mail pfrankl@nyu.edu: only for communication that cannot be shared w/ Tas; include 6083 in subject line and include your name

GradeScope: link to be posted on BrightSpace

- hand in homework and get feedback

To