Announcements

Test 1 details:
• When: Friday 02/19 at 4pm
• Duration: 60 minutes
• How: Canvas Quiz
• Format: T/F + MC + SQL coding
• Review: Tuesday 02/16 at 1pm

Exam rules:
• Open-note and open-book
• May not crowd source notes
• May not consult with any human in any form
• Piazza will be disabled

On the horizon:
• Project 3 will be due in 2 weeks
• Begin NoSQL module after Test 1
Practice Problem

Who are the students who take CS329E with Prof. Mitra?

Return their sid, first name, last name and grade

Sort the results by sid.

Student(sid, fname, lname, dob, status)
Class(cno, cname, credits)
Instructor(tid, name, dept)
Takes(sid, cno, grade)
Teaches(tid, cno)
A World without Transactions

<table>
<thead>
<tr>
<th>Time</th>
<th>Client 1</th>
<th>Client 2</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>UPDATE account</td>
<td>SELECT name, balance</td>
</tr>
<tr>
<td></td>
<td>SET balance = balance - 100</td>
<td>FROM account</td>
</tr>
<tr>
<td></td>
<td>WHERE name = 'Alice';</td>
<td>WHERE name IN ('Alice', 'Bob');</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>(t_0)</td>
<td>UPDATE account</td>
<td></td>
</tr>
<tr>
<td></td>
<td>SET balance = balance - 100</td>
<td></td>
</tr>
<tr>
<td></td>
<td>WHERE name = 'Alice';</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>(t_1)</td>
<td></td>
<td>SELECT name, balance</td>
</tr>
<tr>
<td></td>
<td></td>
<td>FROM account</td>
</tr>
<tr>
<td></td>
<td></td>
<td>WHERE name IN ('Alice', 'Bob');</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>(t_2)</td>
<td>UPDATE account</td>
<td></td>
</tr>
<tr>
<td></td>
<td>SET balance = balance + 100</td>
<td></td>
</tr>
<tr>
<td></td>
<td>WHERE name = 'Bob';</td>
<td></td>
</tr>
</tbody>
</table>
# A World without Transactions

<table>
<thead>
<tr>
<th>Time</th>
<th>Client 1</th>
<th>Client 2</th>
</tr>
</thead>
</table>
| \( t_0 \) | UPDATE playlist  
SET count = count + 1  
WHERE user = 'Alice'; | UPDATE playlist  
SET count = count + 1  
WHERE user = 'Alice'; |
| \( t_1 \) | SELECT count  
FROM playlist  
WHERE user = 'Alice'; | SELECT count  
FROM playlist  
WHERE user = 'Alice'; |
Transaction Properties

- Atomicity
- Consistency
- Isolation
- Durability
Transaction Blocks

BEGIN TRANSACTION;
    {some SQL statement 1}
    {some SQL statement 2}
    {some SQL statement n}
COMMIT;

BEGIN TRANSACTION;
    {some SQL statement 1}
    {some SQL statement 2}
    {some SQL statement n}
ROLLBACK;
Database Indexes

- **Critical** to database systems
- At least one index per table
- DBA analyzes workload and chooses which indexes to create (no easy answers)
- Creating indexes can be an expensive operation
- They work “behind the scenes”
- Query optimizer decides which indexes to use during query execution

```
CREATE INDEX empid_idx ON Employee(empid);
CREATE INDEX empid_idx ON Employee(empid, salary);
```
B-Trees

- Standard index implementation in relational databases
- Designed to speed up lookups and range queries
- One tree node maps to one disk page
- Nodes store index entries
- Index entry = (key, ref)
- Branching factor 100+
- Height is $O(\log n)$
- Search speed $\approx$ height of tree
Why Spanner?

- Globally distributed database system
- Regional and multi-regional configurations
- Implements relational model
- Standard SQL (+ table hierarchies)
- Implements ACID transactions
- TrueTime assigns globally consistent time
- Compute and storage are decoupled
- Data splits assigned to Spanner nodes
- Splits based on load and data volume
- Massive scale (PBs, 1000+ nodes)
- Higher latency per QPS
Set up Spanner (Emulator)

Practice Problem 1
Debug this query and then optimize it.

SELECT *, c.title
WHERE c.title = 'Productivity'
FROM categories c JOIN apps_categories
ON c.id = category_id
AND reviews_count >= 50
AND rating >= 4.0
JOIN apps ON id = app_id;
Practice Problem 2

Write a query to find all foreign key violations on these two tables:

- `pricing_plans`
- `key_benefits`
Project 3