

Class 2 MySQL

Elements of Databases

Jan 28, 2022

Announcements

- Survey results on remote learning
- Recap on how participation questions work
- Checking your GCP credits balance
- Authenticate with your EID before joining class meetings:

<https://utexas.zoom.us/>

Simple Select Statements

```
SELECT c1, c2, c3, cn
FROM T1
WHERE c1 > 100 AND c2 = 'foo'
ORDER BY c3, c4;
```

```
SELECT c1, c2, c3, cn
FROM T1
WHERE c1 IS NOT NULL AND (c2 = 'XY' OR c3 = 'ABC')
ORDER BY c2 DESC;
```

Simple Create, Insert, Update, Delete Statements

```
CREATE TABLE T1 (c1 INT PRIMARY KEY,  
                  c2 VARCHAR(30) NOT NULL,  
                  c3 VARCHAR(30));
```

```
INSERT INTO T1 (c1, c2, c3) VALUES (1, 'Austin', 'TX');
```

```
UPDATE T1 SET c2 = 'ATX', c3 = 'Texas' WHERE c1 = 1;
```

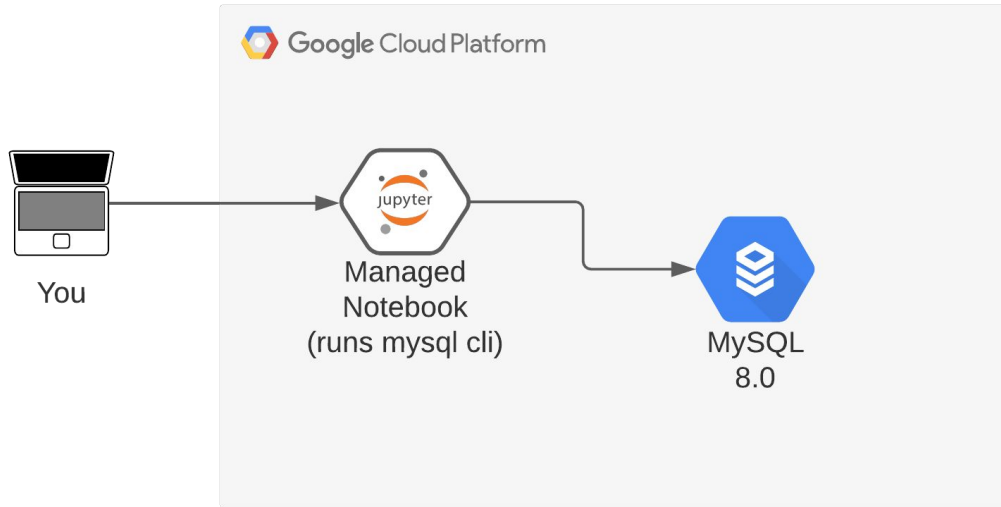
```
DELETE FROM T1 WHERE c3 NOT IN ('Texas' 'TX', 'CA');
```

```
DELETE FROM T1 WHERE c3 IS NULL;
```

MySQL Overview

- MySQL databases are used everywhere
- Simple and easy-to-use
- Open-source software (commercialized by Oracle)
- Implements the relational model
- Designed for storing structured data
- Feature-rich SQL support
- Supports many languages
- Small to medium size data (< TB storage)
- Low to moderate QPS of reads and writes (10K)
- Scale reads through read replicas
- Scale writes through sharding (e.g. [Vitess](#))

MySQL environment on Google Cloud



Environment built by following our [MySQL](#) and [Jupyter](#) setup guides (assigned as homework).

MySQL Code Lab:

- Clone [snippets](#) repo
- Open [mysql notebook](#)
- Create database
- Create tables
- Populate tables
- Sample some records
- Remove header row
- Create Primary Keys
- Create Foreign Keys
- Test Foreign Keys

College Database Schema

Student		
PK	sid	CHAR
	fname	VARCHAR
	lname	VARCHAR
	dob	DATE
	status	CHAR

Class		
PK	sid	CHAR
PK	cno	CHAR
	cname	VARCHAR
	credits	INT
	grade	CHAR

Instructor		
PK	tid	CHAR
	name	VARCHAR
	dept	VARCHAR

Teaches		
PK, FK	tid	CHAR
PK	cno	CHAR



Student(sid, fname, lname, dob, status)

Class(sid, cno, cname, credits, grade)

Instructor(tid, name, dept)

Teaches(tid, cno)

Exercise: SQL

Write the SQL to answer these two questions:

Who takes CS327E or CS329E?

Who takes CS327E and CS329E?

Schema:	Student(<u>sid</u> , fname, lname, dob, status) Class(sid, cno, cname, credits, grade) Instructor(<u>tid</u> , name, dept) Teaches(<u>tid</u> , <u>cno</u>)
----------------	--

Exercise: SQL

Who takes CS327E and CS329E?

Student(sid, fname, lname, dob, status)

Class(cno, cname, credits)

Instructor(tid, name, dept)

Teaches(tid, cno)

Is this query a correct implementation?

```
SELECT sid
FROM Class
WHERE cno = 'CS327E'
      AND cno = 'CS329E'
```

Relational Model

- Database is a collection of relations
- Relation is a table with columns (fields) and rows (tuples)
- Field properties: named, typed, single-valued
- Tuple properties: unique and unordered

Foreign Keys: When one or more fields in a table (child table) refer to one or more fields in another table (parent table).

Notation:	Customer(<u>id</u> , fname, lname, address, ...) Order(<u>orderno</u> , <u>custno</u> , date_ordered, ...)
-----------	---

Data Modeling

- Entity type: A type of object
- Usually a noun
- Common examples: Person, Team, Product, Order, Shipment
- Entity: A real-world object
- Common examples: "Joe Biden", "Texas Longhorns", "Gmail", etc.

Analogies with OOP:

- Entity type: analogous to class
- Entity: analogous to objects
- Attribute: analogous to members of an object

Design Guidelines

1. A table represents a single entity type or a $m:n$ relationship.
2. The fields represent the attributes of an entity type or attributes of a $m:n$ relationship.
3. Each field is assigned a data type that best fits its domain of values.
4. Each table has a Primary Key (PK) constraint which is made up of one or more fields that uniquely represent each entity in that table.
5. 1:1 and 1:m relationships are represented as a Foreign Key (FK) relationship, in which the child table has a FK constraint on the field(s) that reference its parent's PK fields.

Project 1

<http://www.cs.utexas.edu/~scohen/projects/Project1.pdf>