

Class 4 Postgres

Elements of Databases

Feb 18, 2022

Announcements

Preparing for Midterm 1:

- End-of-chapter exercises (requires Sakila sample database)
- Practice SQL on [Hacker Rank](#)
- Practice SQL on [Leetcode](#)

On the horizon:

- BigQuery starting next week (no setup needed)
- Review session for Midterm 1 (week of March 7th)

Homework: remodeled college tables

1. Connect to your Postgres database.
2. Run this query: `select count(*) from college.takes;`
3. Answer the instapoll.

Review Exercise: SQL Joins

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

*For each student, return their
sid, first and last names, and
grade sorted by their sid.*

Schema:

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

Class(cno, cname, credits)

Instructor(tid, name, dept)

Takes(sid, cno, grade)

Teaches(tid, cno)

Exercise 1: SQL Joins

*Who are the students who
take both CS329E and CS327E?*

*For each student, return their
sid and first and last names.*

Sort the results by sid.

Schema:

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

Class(cno, cname, credits)

Instructor(tid, name, dept)

Takes(sid, cno, grade)

Teaches(tid, cno)

Exercise 2: SQL Joins

Which classes have no students taking them?

For each class with zero enrollment, return its cno, cname and credits.

Sort the results by cno.

Schema:

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

Time



	Client 1	Client 2
t_0	<pre>UPDATE account SET balance = balance - 100 WHERE name = 'Alice';</pre>	
t_1		<pre>SELECT name, balance FROM account WHERE name IN ('Alice', 'Bob');</pre>
t_2	<pre>UPDATE account SET balance = balance + 100 WHERE name = 'Bob';</pre>	

A World without Transactions

Time



	Client 1	Client 2
t_0	<pre>UPDATE playlist SET count = count + 1 WHERE user = 'Alice';</pre>	<pre>UPDATE playlist SET count = count + 1 WHERE user = 'Alice';</pre>
t_1	<pre>SELECT count FROM playlist WHERE user = 'Alice';</pre>	<pre>SELECT count FROM playlist WHERE user = 'Alice';</pre>

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;
```

Transaction Guarantees

- Atomicity
- Consistency
- Isolation
- Durability

Postgres Code Lab, Part 2

- Clone [snippets](#) repo
- Open [postgres tx notebook](#)
- Create the Shopify tables and load them
- Sample the tables
- Create the Foreign Keys
- Walk through an insert transaction
- Write an update transaction
- Write a delete transaction

Project 3

<http://www.cs.utexas.edu/~scohen/projects/Project3.pdf>