Databases

Announcements

Aggregation Review

Grouping Rows: Remakes

```
SELECT [columns] FROM [table] GROUP BY [expression] HAVING [expression];
```

- **COUNT**(*): number of rows in a group
- **MAX**([expression]): largest value of [expression] for any row in a group (also **MIN**, **SUM**, & **AVG**)

titles

tconst	title	year	runtime	genres
tt8404614	The Two Popes	2019	125	Biography,Drama
tt0012349	The Kid	1921	68	Comedy,Drama,Family

Create a table of remakes that have the same title

```
title                              first  second
---------------------------------  -----  ------
How to Train Your Dragon           2010   2025
The Girl with the Dragon Tattoo    2009   2011
```

How can we get the runtime of the first movie?

```
SELECT title, MIN(year) AS first, MAX(year) AS second
    FROM titles
    GROUP BY title
    HAVING COUNT(*) > 1;
```

SELECT: Values each output row contains (and column labels)

FROM: Source of input rows

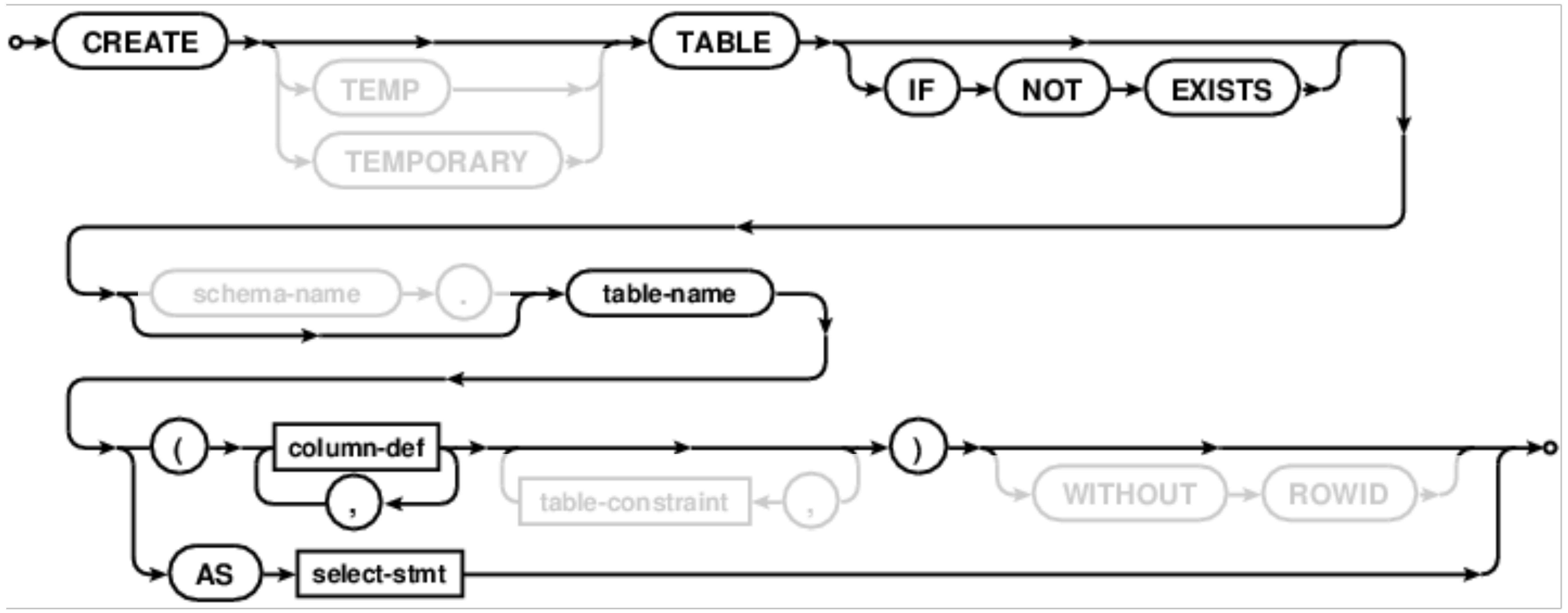
GROUP BY: Form output rows

HAVING: Which output rows

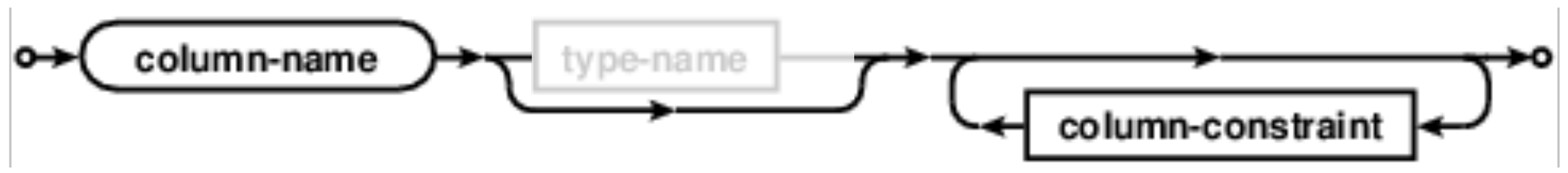
Create Table and Drop Table

Create Table

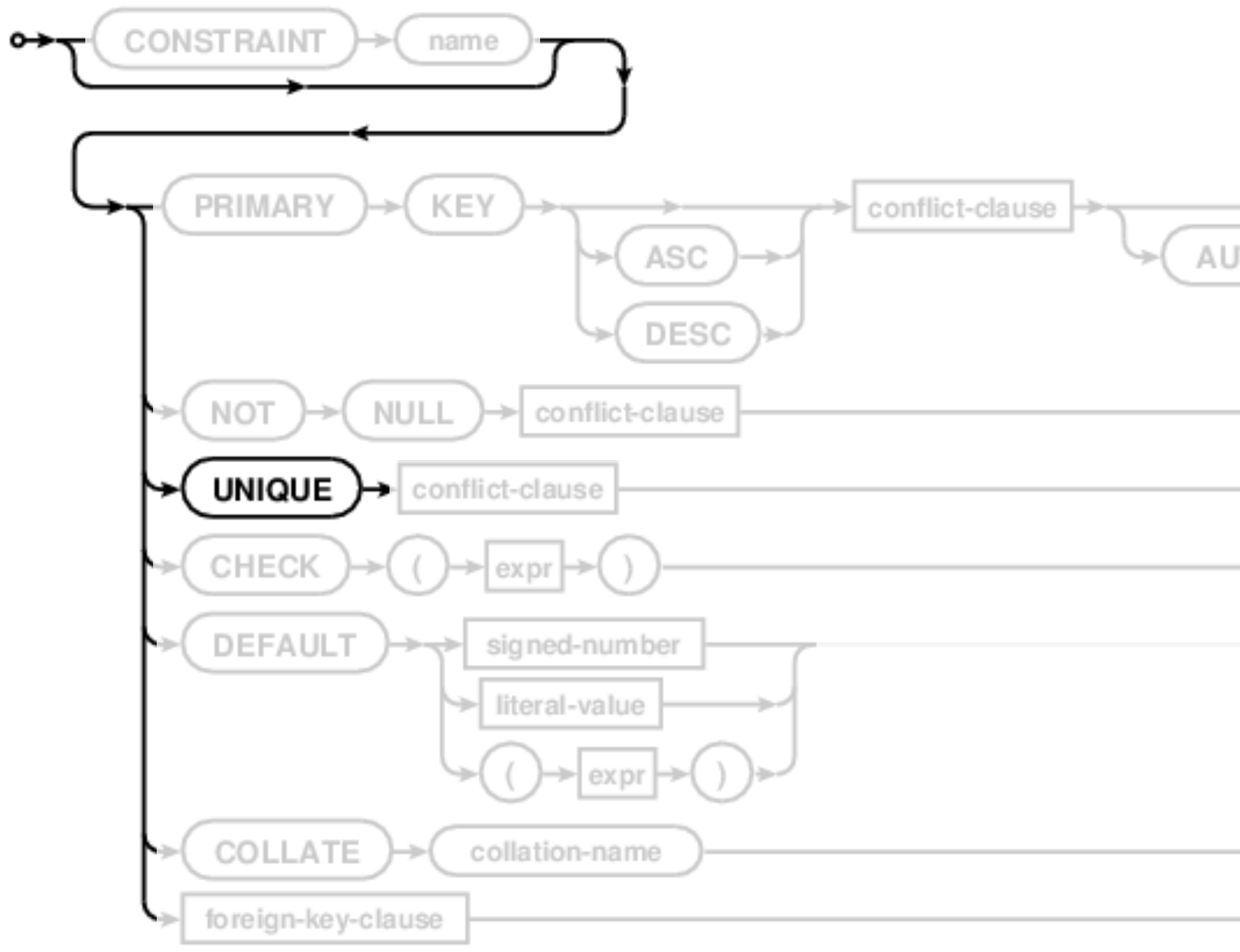
`CREATE TABLE expression syntax:`



`column-def:`



`column-constraint:`

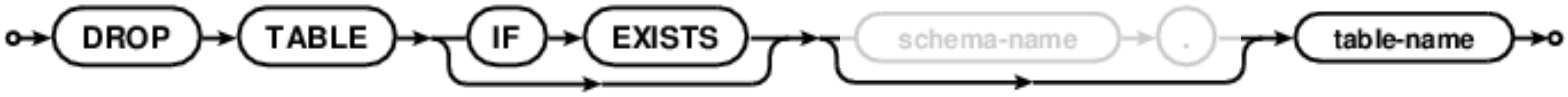


`Examples:`

CREATE TABLE numbers (n, note);

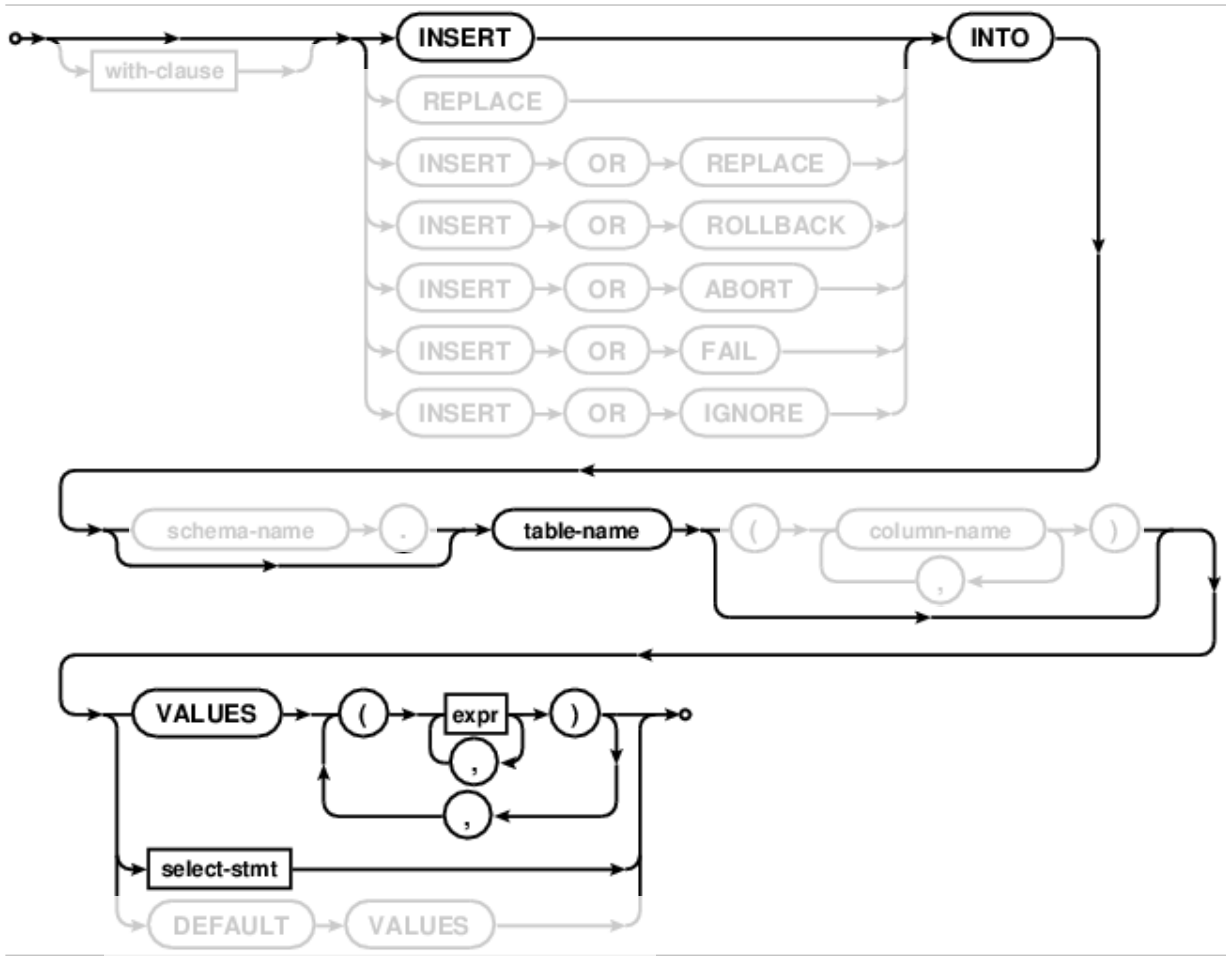
CREATE TABLE numbers (n UNIQUE, note);

Drop Table



Modifying Tables

Insert

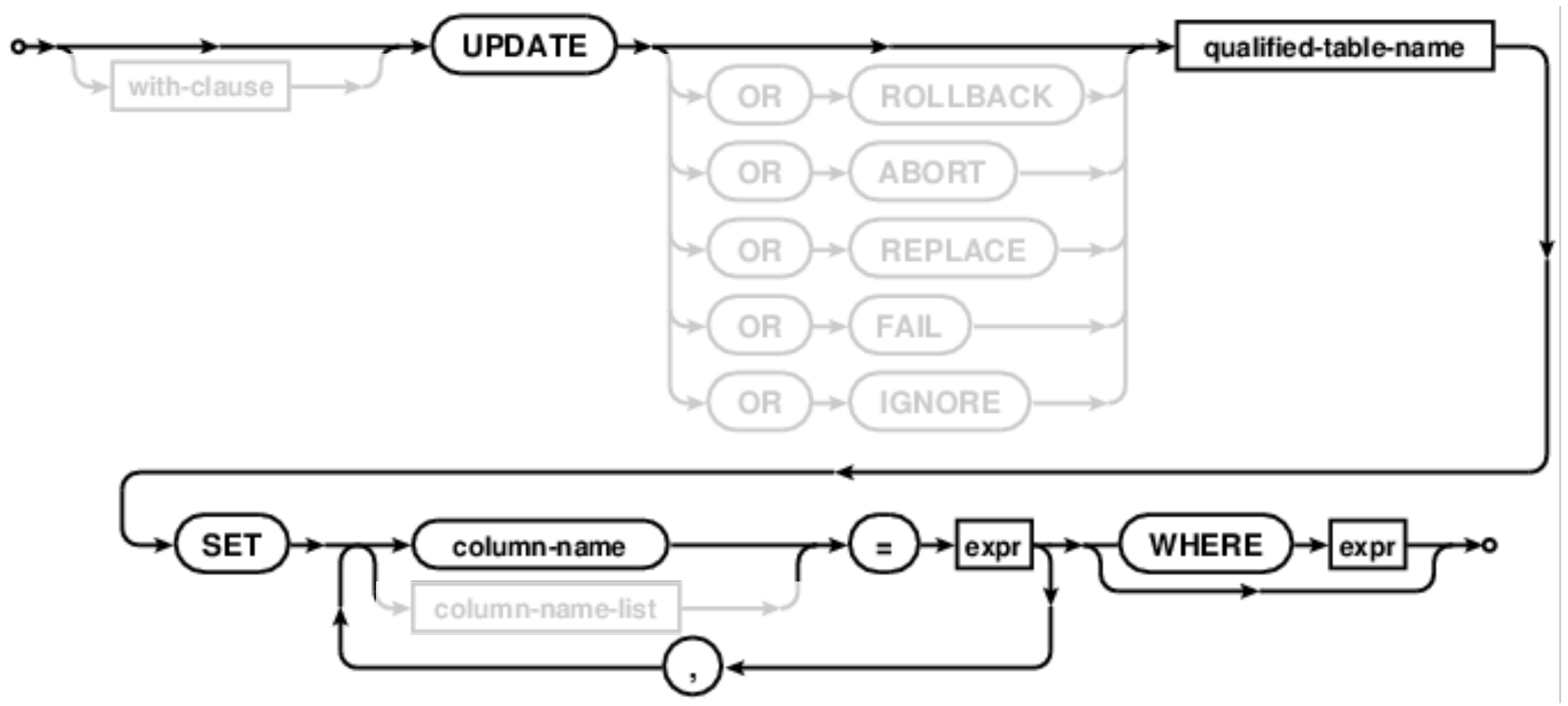


For a table t with two columns...

To insert a row:

INSERT INTO t VALUES (value0, value1);

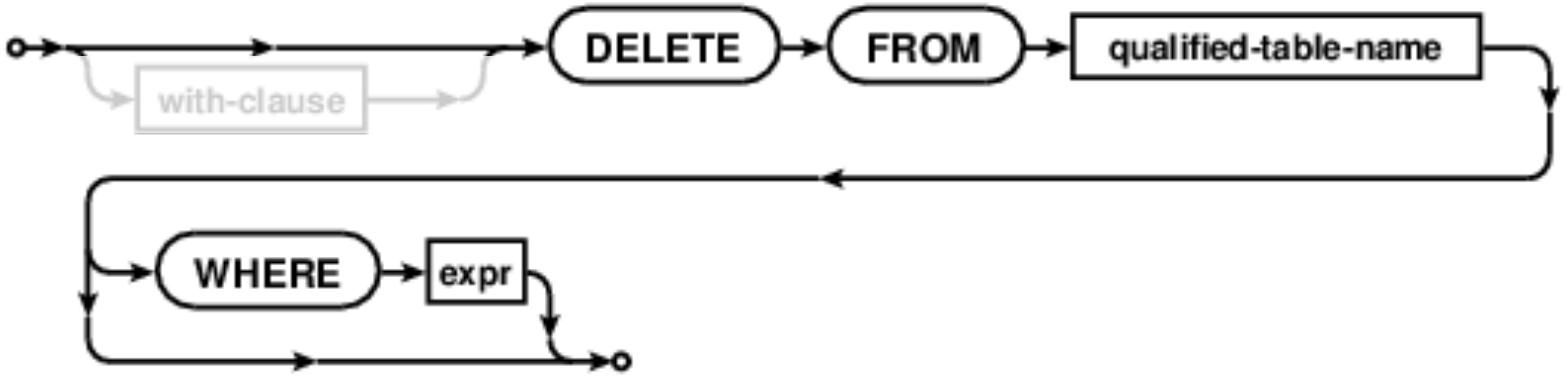
Update



```
Update sets all entries in certain columns to new values, just for some subset of rows.
```

(Demo)

Delete



```
Delete removes some or all rows from a table.
```

```
(Demo)
```

Python and SQL

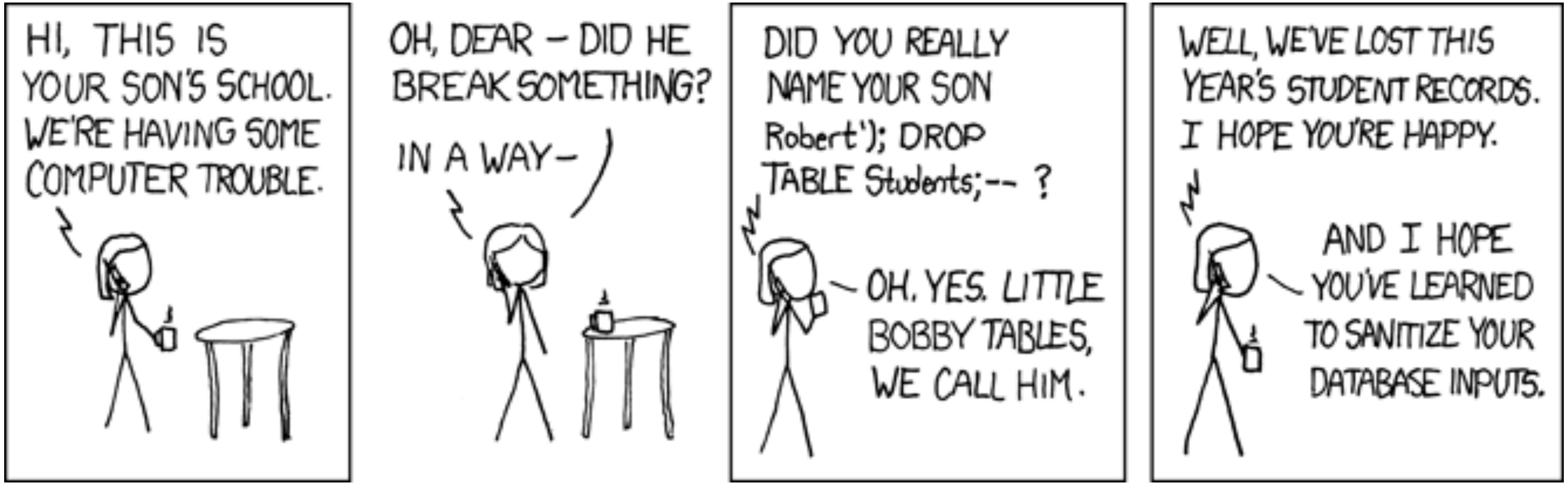
Python Can Access Sqlite Databases

```
import sqlite3

db = sqlite3.Connection('number.db')
db.execute('CREATE TABLE nums (first INT, second INT);’)
db.execute('INSERT INTO nums VALUES (?, ?), (?, ?);', range(6, 10))
print(db.execute('SELECT * FROM nums;').fetchall())
db.commit()
```

SQL Injection Attack

A Program Vulnerable to a SQL Injection Attack


HI, THIS IS YOUR SON'S SCHOOL. WE'RE HAVING SOME COMPUTER TROUBLE.
OH, DEAR – DID HE BREAK SOMETHING?
IN A WAY–
DID YOU REALLY NAME YOUR SON Robert'); DROP TABLE Students;-- ?
OH. YES. LITTLE BOBBY TABLES, WE CALL HIM.
WELL, WE'VE LOST THIS YEAR'S STUDENT RECORDS. I HOPE YOU'RE HAPPY.
AND I HOPE YOU'VE LEARNED TO SANITIZE YOUR DATABASE INPUTS.

```
name = "Robert'); DROP TABLE Students; --"
```

~~cmd = "INSERT INTO Students VALUES ('" + name + "');"~~

~~db.executescript(cmd)~~

```
db.execute("INSERT INTO Students VALUES (?)", [name])
```

SQLite makes a query plan before substitution happens

SQLite gets parameters separately

SQLite gets a string:

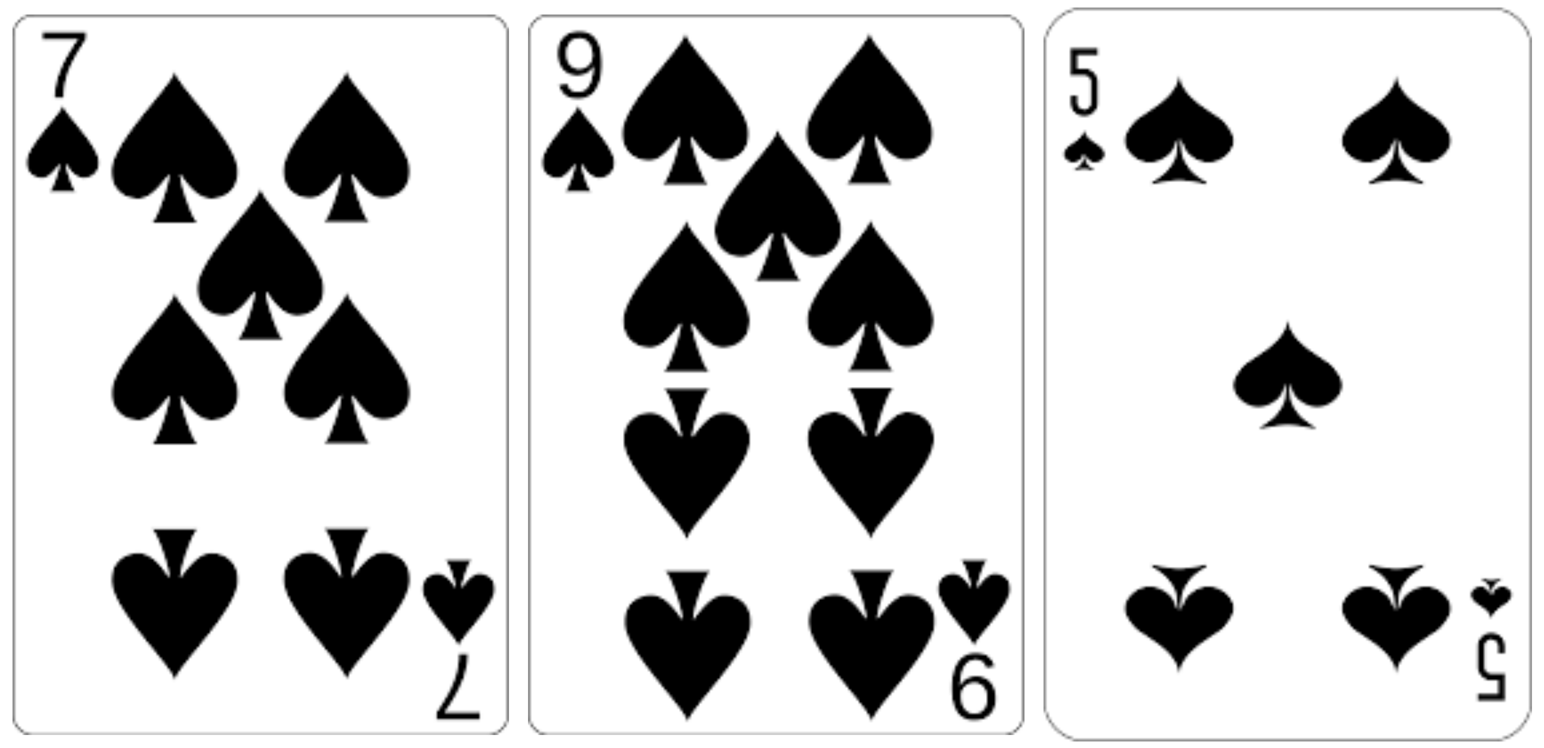
```
INSERT INTO Students VALUES ('Robert'); DROP TABLE Students; --');
```

https://xkcd.com/327/

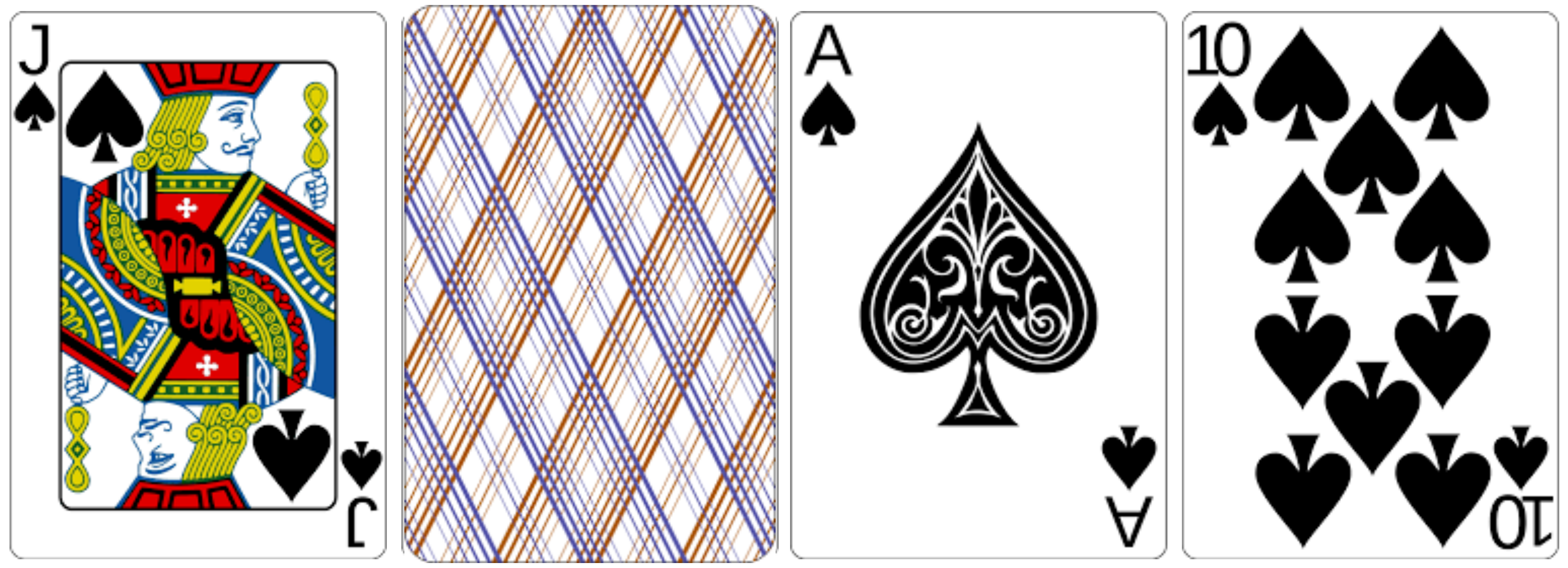
Database Connections

Casino Blackjack

Player:



Dealer:



(Demo)