

Warm-Up: What Is A Database?

What are some examples of ways to store data?

-

SQL

Announcements

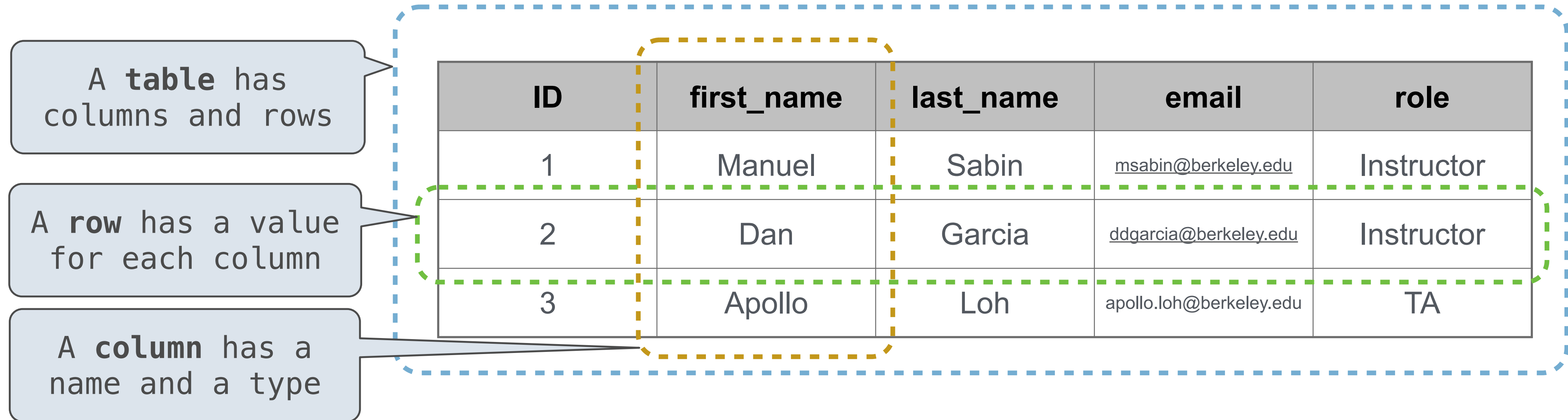
Databases

Database Management Systems

Database management systems (DBMS) are important, heavily used, and interesting!

This class: Relational Databases i.e. tables

A table is a collection of records, which are rows that have a value for each column



The Structured Query Language (SQL) is perhaps the most widely used programming language

The “structure” here is tables (i.e. relational databases)

Structured Query Language (SQL)

SQL: Structured Query Language

This is a Domain Specific Language (DSL)

Not for writing arbitrary computation like Python. It's specifically for writing *queries* to retrieve data from relational (table-based) databases

It is a *Declarative* programming language: You don't write how to algorithmically retrieve the data. You just *declare* what you want as a query, and SQL figures out how to get it for you.

Tables and Queries

Give the result a name

```
CREATE TABLE [name] AS
```

```
SELECT [columns]  
FROM [table]  
WHERE [condition]  
ORDER BY [order];
```


Data represented
as tables



Select statements
transform one
table to another

Output table may
have different
rows and columns
than input

SELECT Statements

```
SELECT [columns] FROM [table] WHERE [condition] ORDER BY [order];
```

ID	first_name	last_name	email	role
1	Manuel	Sabin	msabin@berkeley.edu	Instructor
2	Dan	Garcia	ddgarcia@berkeley.edu	Instructor
3	Apollo	Loh	apollo.loh@berkeley.edu	TA

Which columns

```
SELECT email, first_name FROM users WHERE role = "Instructor";
```

column name

email	first_name
msabin@berkeley.edu	Manuel
ddgarcia@berkeley.edu	Dan

(Demo)

SELECT Statements

```
SELECT [columns] FROM [table] WHERE [condition] ORDER BY [order];
```

ID	first_name	last_name	email	role
1	Manuel	Sabin	msabin@berkeley.edu	Instructor
2	Dan	Garcia	ddgarcia@berkeley.edu	Instructor
3	Apollo	Loh	apollo.loh@berkeley.edu	TA

Make a new table with the TA's emails:

```
CREATE TABLE ta_emails AS SELECT email FROM users WHERE role = "TA";
```

Make a new table **ta_emails** with the result

email
apollo.loh@berkeley.edu

(Demo)

Internet Movie Database

<https://developer.imdb.com/non-commercial-datasets/>

(Demo)

Command overview

SQLite shell commands:

.tables: List all tables

.mode box (or **.mode column**): Make the display pretty

SQL:

SELECT * FROM [table] LIMIT 5; Show the first 5 rows of table

[expression] AS [name], [expression] AS [name], ...

SELECT [columns] FROM [table] WHERE [condition] ORDER BY [order];

Which columns

Which rows

Example titles queries

tconst	title	year	runtime	genres
tt0012349	The Kid	1921	68	Comedy,Drama,Family
...				

Create a new table **oldest** with the 5 oldest movie titles and years in the dataset

```
CREATE TABLE oldest AS SELECT title, year FROM titles ORDER BY year LIMIT 5;
```

Select the **title** and **runtime in hours** of the 20 longest movies

```
SELECT title, runtime / 60.0 FROM titles ORDER BY runtime DESC LIMIT 20;
```

Example names queries

nconst	name	birth	death	profession	knownForTitles
nm0000002	Lauren Bacall	1924	2014	actress,miscellaneous,soundtrack	tt0037382,tt0075213,tt0038355,tt0117057
...					

Select the names and ages of all people:

```
SELECT name, death - birth FROM names;
```

Select the names and ages of all alive people, where birth is not '\N':

```
SELECT name, 2025 - birth FROM names WHERE death = '\N' AND birth != '\N' ;
```

Declarative Programming

In declarative programming:

- A "program" is a description of the desired result
- The interpreter figures out how to generate the result

Database is responsible for determining how to run this quickly

```
SELECT name, 2025 - birth FROM names WHERE death = '\N' AND birth != '\N';
```

nconst	name	birth	death	profession	knownForTitles
nm0000002	Lauren Bacall	1924	2014	actress,miscellaneous,soundtrack	tt0037382,tt0075213,tt0038355,tt0117057
...					
nm0000250	Renée Zellweger	1969	\N	actress,producer,writer	tt0299658,tt0159365,tt0116695,tt0243155
nm0000255	Ben Affleck	1972	\N	producer,actor,writer	tt1024648,tt2267998,tt0840361,tt0119217

Possible process:

Scan all rows for death = '\N' and birth != '\N'

Declarative Programming

```
SELECT name, 2025 - birth FROM names WHERE death = '\N' AND birth != '\N';
```

nconst	name	birth	death	profession	knownForTitles
nm0000002	Lauren Bacall	1924	2014	actress,miscellaneous,soundtrack	tt0037382,tt0075213,tt0038355,tt0117057
...					
nm0000250	Renée Zellweger	1969	\N	actress,producer,writer	tt0299658,tt0159365,tt0116695,tt0243155
nm0000255	Ben Affleck	1972	\N	producer,actor,writer	tt1024648,tt2267998,tt0840361,tt0119217
...					

Index: Separate data structure that keeps track of where certain values are in the table

death index: {2014: •, '\N': •}

Possible process:

Use death index to lookup only entries with death = '\N'