

INTRO TO DATA SCIENCE

DATABASES

I. INTRO TO DATABASES

II. RELATIONAL DATABASES

III.NOSQL DATABASES

EXERCISES:

III. MYSQL AND MONGO TUTORIALS

I. INTRO TO DATABASES

Databases are a structured data source optimized for efficient retrieval and storage.

Databases are a structured data source optimized for efficient retrieval and persistent storage.

structured: we have to pre-define organization strategy

retrieval: the ability to read data out

storage: the ability to write data and save it

II. RELATIONAL DATABASES

Relational databases are traditionally organized in the following manner:

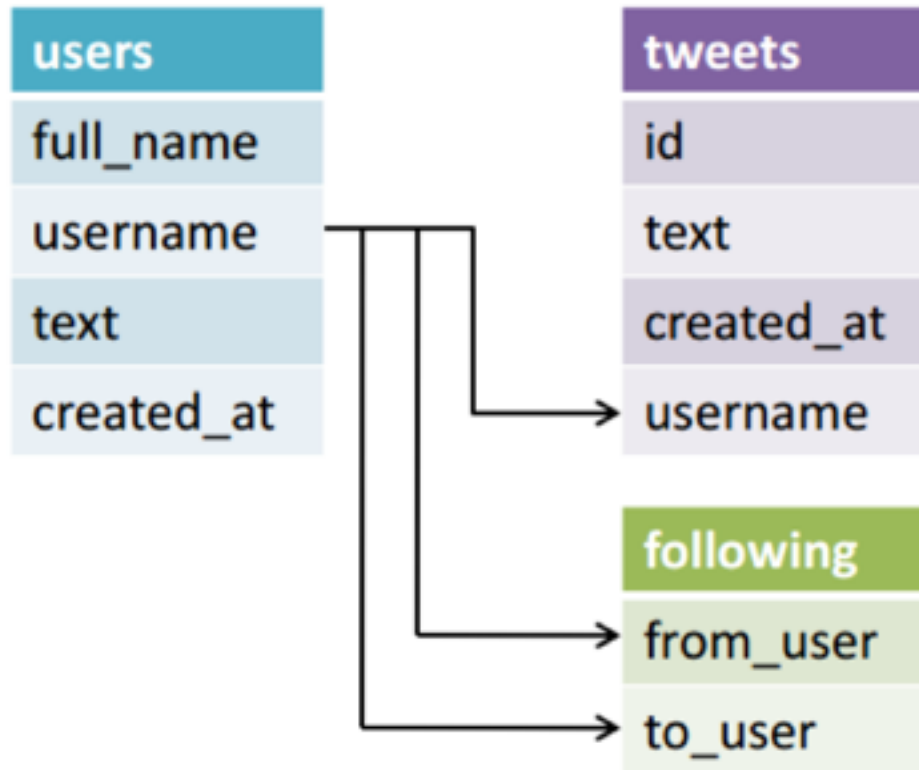
A database has tables which represent individual entities or objects

Tables have predefined schema – rules that tell it what the data will look like

Each table should have a primary key column – a unique identifier for that row

Each table should have a primary key column – a unique identifier for that row

Additionally each table can have a foreign key column – an id that links this to another table.



We could have had a table structure as follows:
Why is this different?

tweets
id
text
created_at
username
full_name
username
text
created_at

We could have had a table structure as follows:
Why is this different?

We would repeat the user information in each row.

This is called
denormalization.

tweets
id
text
created_at
username
full_name
username
text
created_at

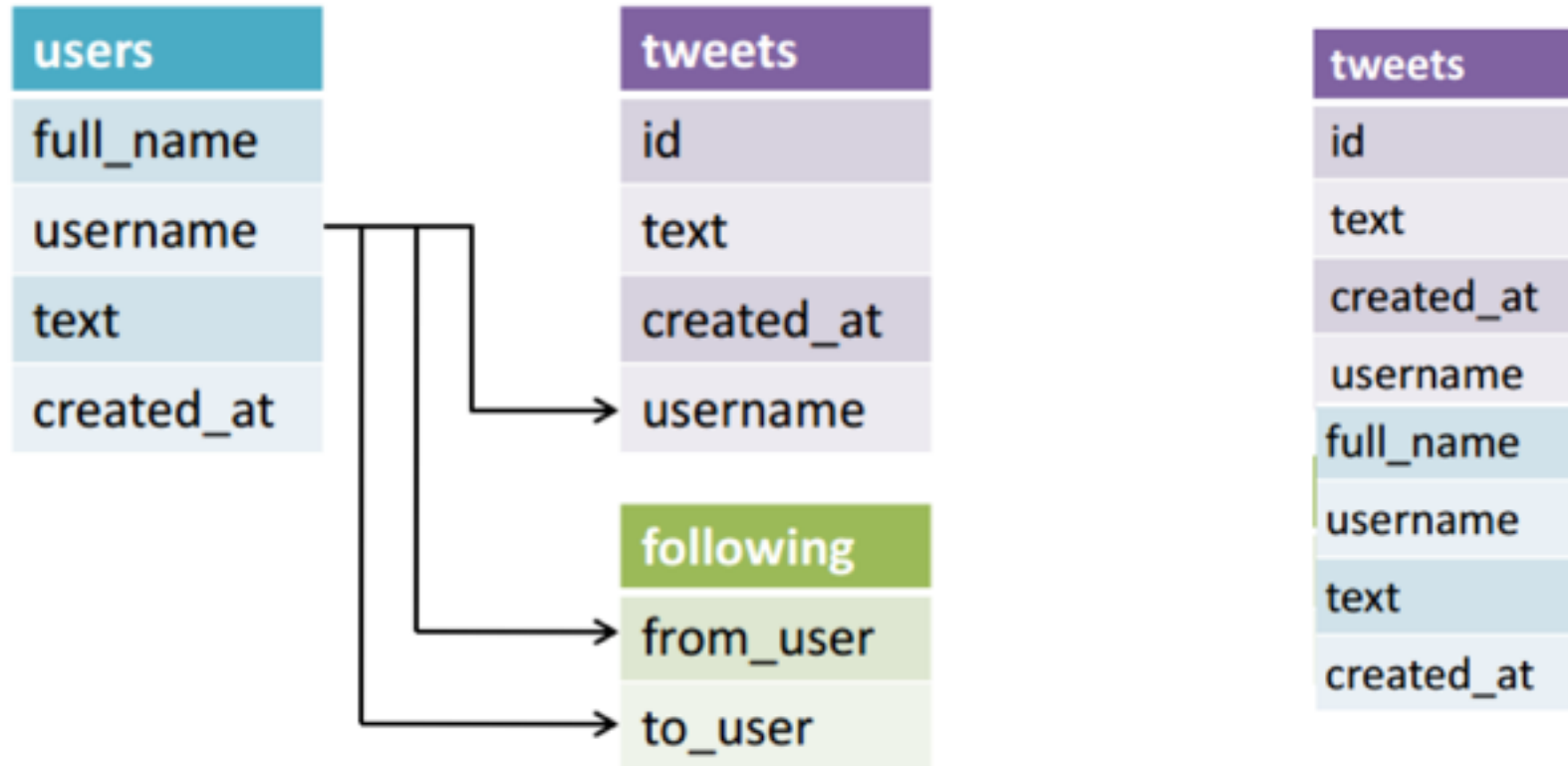
Normalized Data: Many tables to reduce redundant or repeated data in a table

Denormalized Data: Wide data with fields often repeated but removes the need to join together multiple tables

Normalized Data: Many tables to reduce redundant or repeated data in a table

Denormalized Data: Wide data with fields often repeated but removes the need to join together multiple tables

This is a trade off of speed vs storage.

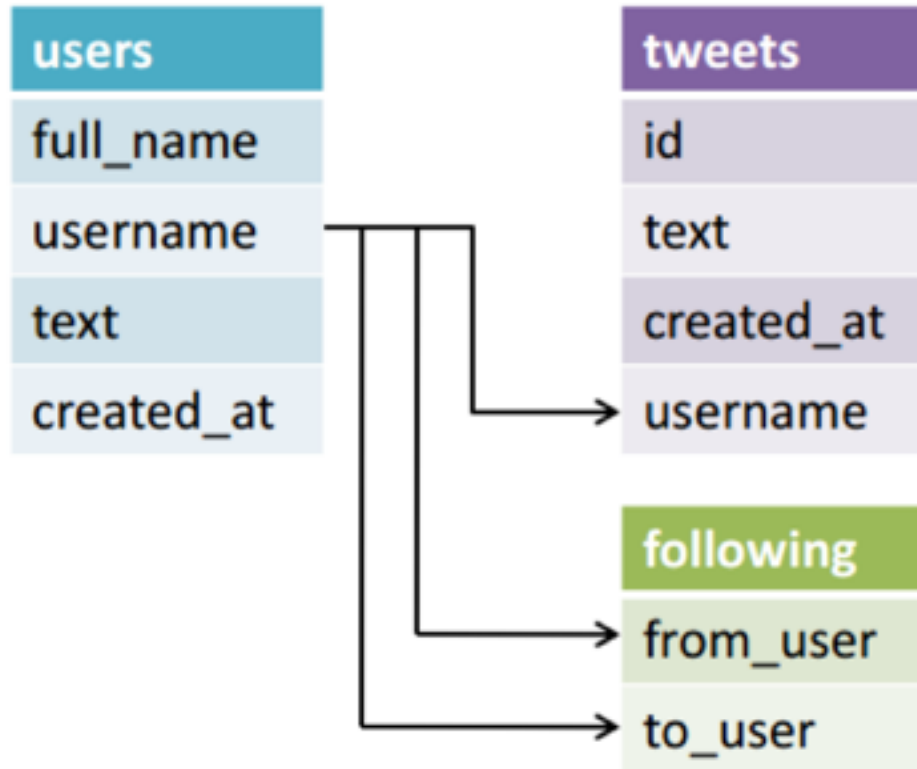


Q: How do we commonly evaluate databases?

read-speed vs write-speed
space considerations
(and many, many other criteria)

Q: Why are normalized tables (possibly) slower to read?

A: We'll have to get data from multiple tables to answer some questions



Q: Why are denormalized tables (possibly) slower to write?

A: We'll have to write more information on each write

tweets
id
text
created_at
username
full_name
username
text
created_at

SQL is a query language to load, retrieve, and update data in relational databases

Most commonly known SQL-like Databases include:

Oracle

MySQL

PostgreSQL

SELECT: Allows you to retrieve information from a table

Syntax:

```
SELECT col1, col2  
FROM table WHERE [some condition]
```

Example:

```
SELECT poll_title, poll_date FROM polls WHERE  
romney_pct > obama_pct
```

GROUP BY: Allows you to aggregate information.

Syntax:

```
SELECT col1, AVG(col2)  
FROM table GROUP BY col1
```

Example:

```
SELECT poll_date, AVG(obama_pct)  
FROM polls GROUP BY poll_date
```

GROUP BY: Allows you to aggregate information.

Syntax:

```
SELECT col1, AVG(col2)  
FROM table GROUP BY col1
```

Example:

```
SELECT poll_date, AVG(obama_pct)  
FROM polls GROUP BY poll_date
```


GROUP BY: Allows you to aggregate information.

Syntax:

```
SELECT col1, AVG(col2)  
FROM table GROUP BY col1
```

There are usually a few common built-in operations:

SUM, AVG, MIN, MAX, COUNT

JOIN: Allows you to combine multiple tables

Syntax:

SELECT t1.c1, t1.c2, t2.c2

FROM t1 JOIN t2 ON t1.c1 = t2.c2

JOIN: Allows you to combine multiple tables

Syntax:

```
SELECT t1.c1, t1.c2, t2.c2  
FROM t1 JOIN t2 ON t1.c1 = t2.c2
```

INSERT: Allows you to add data to tables

Syntax:

```
INSERT INTO table1 (col1, col2)  
VALUES (...)
```

```
INSERT INTO classroom (first_name, last_name)  
VALUES ('John', 'Doe')
```

INTRO TO DATA SCIENCE

LAB: MYSQL QUERYING

III. NO-SQL DATABASES

NoSQL databases are a new trend in databases

NoSQL databases are a new trend in databases

The name NoSQL refers to the lack of a relational structure between stored objects.

Most importantly they attempt to minimize the need for JOIN operations, or solve other data needs

Memcached

Apache HBase

Cassandra

MongoDB

Hadoop

Memcached	::	Livejournal
Apache HBase	::	Google BigTable
Cassandra	::	Amazon Dynamo
MongoDB	::	10Gen
Hadoop	::	Google MapReduce

Memcached was:

- Developed by LiveJournal
- Distributed key-value store (like a Python Dict)
- Supports two very fast operations: get and set

Memcached is best used for storing application configuration settings, and essential caching those settings.

Cassandra was:

- Developed by Facebook
- Messages application and Inbox Search
- Key-Value (ish)
 - Supports query by key or key range
- Very fast writing speeds
- Useful for record keeping, logging

Mongo was:

- Developed by 10Gen (now MongoDB, Inc)
- Document and Collection Structure
- BSON (JSON-like) Storage system
- Aggregation Framework