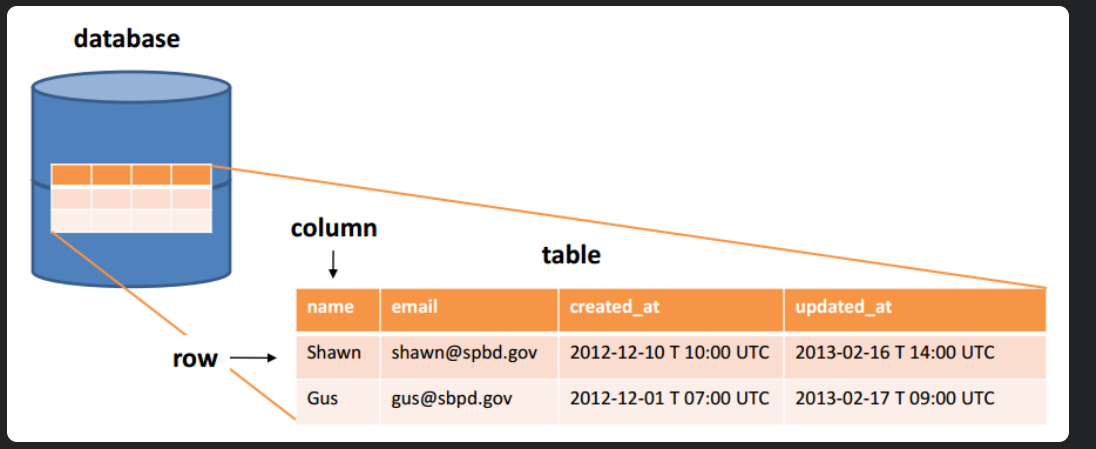
Introduction to relational databases

Web applications can be split into 2 main parts:

1. The front-end: displays and collects information
2. The back-end: stores and provides data for use by the front end

Since the web application back end is concerned with storing and retrieving data, often for manty thousands or even millions of users, it is natural that a database is one of the major back-end components

A database stores data in an organized way so that it can be searched and retrived later. The database is made up of one or more tables. A table is much like a spreadsheet, in that it’s made up of rows and columns. All rows have the same columns and each column contains the data itself. If it helps, think of a table as being like a speadsheet in excel or google sheets



Data can be insterted, retirved, updated, and deleted from a table. The word created is generally used instead of insterted, so , collectively, these four fuinctions are affectionately abbreviated as CRUD.

What is a relational database?

A relational database is a type of database that organizes data into tables and creates links between these tables, based on defined relationships. These relationships enable you to retieve and combine data from one or more tables with a single query.

1.Get some data

Lets start with some twitter data.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Full\_name | username | text | Created\_at | Following\_username |
| "Boris Hadjur" | "\_DreamLead" | "What do you think about #emailing #campaigns #traffic in #USA? Is it a good market nowadays? do you have #databases?" | "Tue, 12 Feb 2013 08:43:09 +0000" | "Scootmedia", "MetiersInternet" |
| "Gunnar Svalander" | "GunnarSvalander" | "Bill Gates Talks Databases, Free Software on Reddit https://t.co/ShX4hZlA #billgates #databases" | "Tue, 12 Feb 2013 07:31:06 +0000" | "klout", "zillow" |
| "GE Software" | "GEsoftware" | "RT @KirkDBorne: Readings in #Databases: excellent reading list, many categories: https://t.co/S6RBUNxq via @rxin Fascinating." | "Tue, 12 Feb 2013 07:30:24 +0000" | "DayJobDoc", "byosko" |
| "Adrian Burch" | "adrianburch" | "RT @tisakovich: @NimbusData at the @Barclays Big Data conference in San Francisco today, talking #virtualization, #databases, and #flash memory." | "Tue, 12 Feb 2013 06:58:22 +0000" | "CindyCrawford", "Arjantim" |
| "Andy Ryder" | "AndyRyder5" | "http://t.co/D3KOJIvF article about Madden 2013 using AI to prodict the super bowl #databases #bus311" | "Tue, 12 Feb 2013 05:29:41 +0000" | "MichaelDell", "Yahoo" |
| "Andy Ryder" | "AndyRyder5" | "http://t.co/rBhBXjma an article about privacy settings and facebook #databases #bus311" | "Tue, 12 Feb 2013 05:24:17 +0000" | "MichaelDell", "Yahoo" |
| "Brett Englebert" | "Brett\_Englebert" | "#BUS311 University of Minnesota's NCFPD is creating #databases to prevent "food fraud." http://t.co/0LsAbKqJ" | "Tue, 12 Feb 2013 01:49:19 +0000" | "RealSkipBayless", "stephenasmith" |
| Brett Englebert | "Brett\_Englebert" | "#BUS311 companies might be protecting their production #databases, but what about their backup files? http://t.co/okJjV3Bm" | "Tue, 12 Feb 2013 01:31:52 +0000" | "RealSkipBayless", "stephenasmith" |
| "Nimbus Data Systems" | "NimbusData" | "@NimbusData CEO @tisakovich @BarclaysOnline Big Data conference in San Francisco today, talking #virtualization, #databases,& #flash memory" | "Mon, 11 Feb 2013 23:15:05 +0000" | "dellock6", "rohitkilam" |
| "SSWUG.ORG" | "SSWUGorg" | "Don't forget to sign up for our FREE expo this Friday: #Databases, #BI, and #Sharepoint: What You Need to Know! http://t.co/Ijrqrz29" | "Mon, 11 Feb 2013 22:15:37 +0000" | "drsql", "steam\_games" |

Here’s what each column name means:

* Full\_name
  + The users full name
* Username
  + The users twitted handle
* Text
  + The tweet itself
* Created\_at
  + The timestamp of the tweet
* Following\_username
  + A list of people this user fiollows, separated by commans. For brevity, I limited the list length to two

This is all real data; you can search twitter and actually find those tweets.

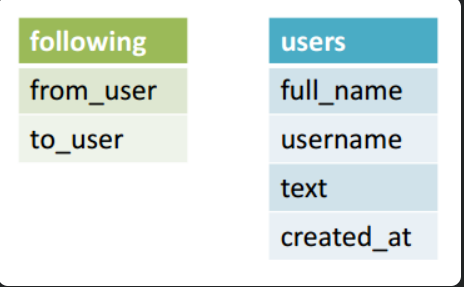
This is good. The data is all in one place, so its easy to find right? Nmot exactly. There are a couple of problems with this table

* First, there is repetitive data across the comlumns. The username and following\_username columns are repetitive, because both contain the same tyopes of data: twitter handles
* Second, there is another form of repetition within the following\_username column. The fields should only contain one value, but each following\_username field contains two.
* Third, there is repetitive data across rows. @andyryder5 and @brett\_englebert each tweeted twice, so the rest of their information has been duplicated.

Duplkicates are problematic because they make the CRUD operations more challenging. For example, it would take longer to retieve data because time would be weasted going through duplicate rows. Also, updating data would be an issue; if a user changes their handle, we would need to fing every duplicate and update it.

2.Remove repetitive data across columns

As noted above the username and following\_username columns in table 1 are repetitive. This repetition occurred because it was trying to express thw follow relationship between users. Let’s improve on table 1’s design by splitting it up into two tables: one just for following relationships and one for the rest of the information



Because @brett\_englebert follows @realskipbayless, the following table will express that relationship by storing @brett\_englebert as the from\_user and @realskipbayless as the to\_user. Let’s go ahead and split table 1 into these two tables:

Table2: the following table

|  |  |
| --- | --- |
| from\_user | to\_user |
| \_DreamLead | Scootmedia |
| \_DreamLead | MetiersInternet |
| GunnarSvalander | klout |
| GunnarSvalander | zillow |
| GEsoftware | DayJobDoc |
| GEsoftware | byosko |
| adrianburch | CindyCrawford |
| adrianburch | Arjantim |
| AndyRyder | MichaelDell |
| AndyRyder | Yahoo |
| Brett\_Englebert | RealSkipBayless |
| Brett\_Englebert | stephenasmith |
| NimbusData | dellock6 |
| NimbusData | rohitkilam |
| SSWUGorg | drsql |
| SSWUGorg | steam\_games |

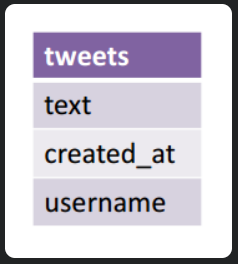
Table 3: the users table

|  |  |  |  |
| --- | --- | --- | --- |
| full\_name | username | text | created\_at |
| "Boris Hadjur" | "\_DreamLead" | "What do you think about #emailing #campaigns #traffic in #USA? Is it a good market nowadays? do you have #databases?" | "Tue, 12 Feb 2013 08:43:09 +0000" |
| "Gunnar Svalander" | "GunnarSvalander" | "Bill Gates Talks Databases, Free Software on Reddit http://t.co/ShX4hZlA #billgates #databases" | "Tue, 12 Feb 2013 07:31:06 +0000" |
| "GE Software" | "GEsoftware" | "RT @KirkDBorne: Readings in #Databases: excellent reading list, many categories: http://t.co/S6RBUNxq via @rxin Fascinating." | "Tue, 12 Feb 2013 07:30:24 +0000" |
| "Adrian Burch" | "adrianburch" | "RT @tisakovich: @NimbusData at the @Barclays Big Data conference in San Francisco today, talking #virtualization, #databases, and #flash memory." | "Tue, 12 Feb 2013 06:58:22 +0000" |
| "Andy Ryder" | "AndyRyder5" | "http://t.co/D3KOJIvF article about Madden 2013 using AI to prodict the super bowl #databases #bus311" | "Tue, 12 Feb 2013 05:29:41 +0000" |
| "Andy Ryder" | "AndyRyder5" | "http://t.co/rBhBXjma an article about privacy settings and facebook #databases #bus311" | "Tue, 12 Feb 2013 05:24:17 +0000" |
| "Brett Englebert" | "Brett\_Englebert" | "#BUS311 University of Minnesota's NCFPD is creating #databases to prevent "food fraud." http://t.co/0LsAbKqJ" | "Tue, 12 Feb 2013 01:49:19 +0000" |
| Brett Englebert | "Brett\_Englebert" | "#BUS311 companies might be protecting their production #databases, but what about their backup files? http://t.co/okJjV3Bm" | "Tue, 12 Feb 2013 01:31:52 +0000" |
| "Nimbus Data Systems" | "NimbusData" | "@NimbusData CEO @tisakovich @BarclaysOnline Big Data conference in San Francisco today, talking #virtualization, #databases,& #flash memory" | "Mon, 11 Feb 2013 23:15:05 +0000" |
| "SSWUG.ORG" | "SSWUGorg" | "Don't forget to sign up for our FREE expo this Friday: #Databases, #BI, and #Sharepoint: What You Need to Know! http://t.co/Ijrqrz29" | "Mon, 11 Feb 2013 22:15:37 +0000 |

This is looking better. Now in the users table there is only one column with twitter handles. In the following tables, there is only one twitter handle per field in the to\_user column.

3.Remove repetitive data across rows

Now that we have fixed repetitions across columns, we need to fix repetitions across rows. Since the users @AndyRyder5 and @Brett\_Englebert each tweeted twice, their information is duplicated in the users table. This indicates that we need to pull out the tweets and place them in their own table.



As before, "text" stores the tweet itself. Since the "created\_at" column stores the timestamp of the tweet, it makes sense to pull it into this table as well. I also include a reference to the "username" column so we know who published the tweet. Here is the result of placing the tweets in their own table:

Table 4 : the tweets table

|  |  |  |
| --- | --- | --- |
| text | created\_at | username |
| "What do you think about #emailing #campaigns #traffic in #USA? Is it a good market nowadays? do you have #databases?" | "Tue, 12 Feb 2013 08:43:09 +0000" | "\_DreamLead" |
| "Bill Gates Talks Databases, Free Software on Reddit http://t.co/ShX4hZlA #billgates #databases" | "Tue, 12 Feb 2013 07:31:06 +0000" | "GunnarSvalander" |
| "RT @KirkDBorne: Readings in #Databases: excellent reading list, many categories: http://t.co/S6RBUNxq via @rxin Fascinating." | "Tue, 12 Feb 2013 07:30:24 +0000" | "GEsoftware" |
| "RT @tisakovich: @NimbusData at the @Barclays Big Data conference in San Francisco today, talking #virtualization, #databases, and #flash memory." | "Tue, 12 Feb 2013 06:58:22 +0000" | "adrianburch" |
| "http://t.co/D3KOJIvF article about Madden 2013 using AI to prodict the super bowl #databases #bus311" | "Tue, 12 Feb 2013 05:29:41 +0000" | "AndyRyder5" |
| "http://t.co/rBhBXjma an article about privacy settings and facebook #databases #bus311" | "Tue, 12 Feb 2013 05:24:17 +0000" | "AndyRyder5" |
| "#BUS311 University of Minnesota's NCFPD is creating #databases to prevent "food fraud." http://t.co/0LsAbKqJ" | "Tue, 12 Feb 2013 01:49:19 +0000" | "Brett\_Englebert" |
| "#BUS311 companies might be protecting their production #databases, but what about their backup files? http://t.co/okJjV3Bm" | "Tue, 12 Feb 2013 01:31:52 +0000" | "Brett\_Englebert" |
| "@NimbusData CEO @tisakovich @BarclaysOnline Big Data conference in San Francisco today, talking #virtualization, #databases,& #flash memory" | "Mon, 11 Feb 2013 23:15:05 +0000" | "NimbusData" |
| "Don't forget to sign up for our FREE expo this Friday: #Databases, #BI, and #Sharepoint: What You Need to Know! http://t.co/Ijrqrz29" | "Mon, 11 Feb 2013 22:15:37 +0000" | "SSWUGorg" |

Table 5: the users table

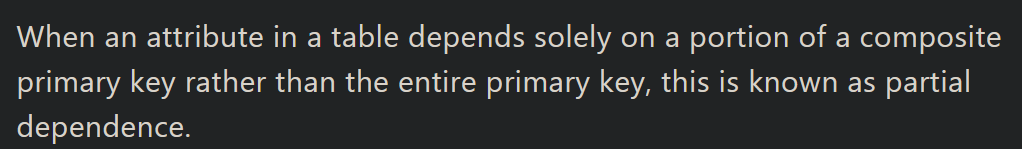
|  |  |
| --- | --- |
| full\_name | username |
| "Boris Hadjur" | "\_DreamLead" |
| "Gunnar Svalander" | "GunnarSvalander" |
| "GE Software" | "GEsoftware" |
| "Adrian Burch" | "adrianburch" |
| "Andy Ryder" | "AndyRyder5" |
| "Brett Englebert" | "Brett\_Englebert" |
| "Nimbus Data Systems" | "NimbusData" |
| "SSWUG.ORG" | "SSWUGorg" |

After the split, the users table has unique rows for users and their Twitter handles.

Edgar F. Codd called this step of removing repetitive data across rows the second normal form (2NF).

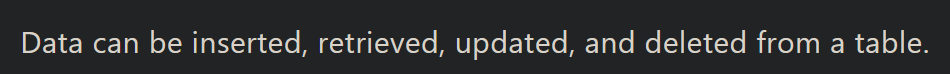
The next stage of normalizing a database is called the second normal form (2NF). 2NF advances the first normal form (1NF). The 2NF makes sure that every piece of information with a many-to-many link is sorted and stored in a separate table.

A table must be in the first normal form and also free of partial dependencies in order to be in the second normal form.



We can separate the table, remove the property that is generating partial dependency (text), and relocate it to another table (tweets) where it will fit in nicely to eliminate partial dependency.

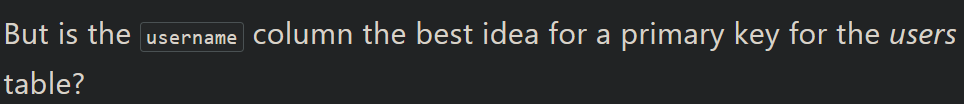
4.Linking tables with keys



So far, table 1 has been split into three new tables: following, tweets, and users. But how is this useful? Repetitive data has been removed, but now the data is spread out across three independent tables. In order to retrieve the data, we need to draw meaningful links between the tables. This way, we can express queries like "what a user has tweeted and who a user is following".

The way to draw links between tables is to first give each row in a table a unique identifier, termed a primary key, and then reference that primary key in the other table to which you want to link.

We've already done this in users and tweets. In the users table, the primary key is the username column because no two users will have the same Twitter handle. In tweets, we reference this key in the username column so we know who tweeted what. Since it is a reference, the username column in tweets is called a foreign key. In this way, the username key links the users and tweets tables together.



On one hand, it's a natural key—it makes sense to search using a Twitter handle instead of assigning each user a numerical ID and searching on that. On the other hand, what if a user wants to change their Twitter handle? That could cause errors if the primary key and all referencing foreign keys aren't updated accurately, errors that could be avoided if a constant numerical ID was used. Ultimately, the choice depends on your system. If you want to give your users the ability to change their username, it's better to add a numerical auto-incrementing id column to users and use that as the primary key. Otherwise, username should do just fine. I'll continue to use username as the primary key for users.

Let's move on to tweets. A primary key should uniquely identify each row, so what should the primary key be here? The created\_at field won't work because if two users tweet at the same time, their tweets would have an identical timestamp. The text has the same problem in that if two users both tweet "Hello world," we couldn't distinguish between the rows. The username column is the foreign key that defines the link with the users so let's not mess with that. Since the other columns are not good candidates, it makes sense here to add a numerical auto-incrementing id column and use that as the primary key.

Table 6: The tweets Table With an id Column

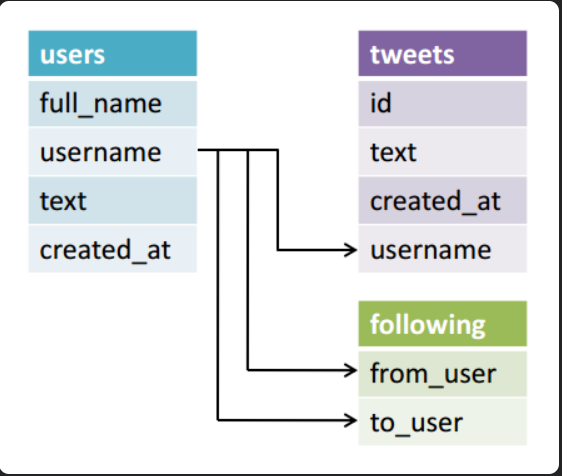
|  |  |  |  |
| --- | --- | --- | --- |
| id | text | created\_at | username |
| 1 | "What do you think about #emailing #campaigns #traffic in #USA? Is it a good market nowadays? do you have #databases?" | "Tue, 12 Feb 2013 08:43:09 +0000" | "\_DreamLead" |
| 2 | "Bill Gates Talks Databases, Free Software on Reddit http://t.co/ShX4hZlA #billgates #databases" | "Tue, 12 Feb 2013 07:31:06 +0000" | "GunnarSvalander" |
| 3 | "RT @KirkDBorne: Readings in #Databases: excellent reading list, many categories: http://t.co/S6RBUNxq via @rxin Fascinating." | "Tue, 12 Feb 2013 07:30:24 +0000" | "GEsoftware" |
| 4 | "RT @tisakovich: @NimbusData at the @Barclays Big Data conference in San Francisco today, talking #virtualization, #databases, and #flash memory." | "Tue, 12 Feb 2013 06:58:22 +0000" | "adrianburch" |
| 5 | "http://t.co/D3KOJIvF article about Madden 2013 using AI to prodict the super bowl #databases #bus311" | "Tue, 12 Feb 2013 05:29:41 +0000" | "AndyRyder5" |
| 6 | "http://t.co/rBhBXjma an article about privacy settings and facebook #databases #bus311" | "Tue, 12 Feb 2013 05:24:17 +0000" | "AndyRyder5" |
| 7 | "#BUS311 University of Minnesota's NCFPD is creating #databases to prevent "food fraud." http://t.co/0LsAbKqJ" | "Tue, 12 Feb 2013 01:49:19 +0000" | "Brett\_Englebert" |
| 8 | "#BUS311 companies might be protecting their production #databases, but what about their backup files? http://t.co/okJjV3Bm" | "Tue, 12 Feb 2013 01:31:52 +0000" | "Brett\_Englebert" |
| 9 | "@NimbusData CEO @tisakovich @BarclaysOnline Big Data conference in San Francisco today, talking #virtualization, #databases,& #flash memory" | "Mon, 11 Feb 2013 23:15:05 +0000" | "NimbusData" |
| 10 | "Don't forget to sign up for our FREE expo this Friday: #Databases, #BI, and #Sharepoint: What You Need to Know! http://t.co/Ijrqrz29" | "Mon, 11 Feb 2013 22:15:37 +0000" | "SSWUGorg" |

Finally, let's add a primary key to the following table. In this table, neither the from\_user column nor the to\_user column uniquely identifies each row on its own. However, from\_user and to\_user together do, since they represent a single relationship. A primary key can be defined on more than one column, so we'll use both these columns as the primary key for the following table.

As for the foreign key, from\_user and to\_user are each foreign keys since they can be used to define a link with the users table. If we query for a Twitter handle on the from\_user column, we'll get all the users they follow. Correspondingly, if we query for a Twitter handle on the to\_user column, we'll get all the users following them.

We've accomplished a lot so far. We removed repetitions across columns and rows by separating data into three different tables, and then we chose meaningful primary keys to link the tables together. This process is called normalization, and its output consists of data that is clean and organized according to the relational model. The consequence of this organization is that rows will appear in the database only once moving forward, which in turn will make the CRUD operations easier.

The figure below shows the finalized database schema. The three tables are linked, and the primary keys are highlighted.



Relational Database Management Systems

Now that we know how to design a relational database, how do we actually implement one? Relational database management systems (RDBMS) are software that let you create and use relational databases. There are several commercial and open-source vendors to choose from. On the commercial side, Oracle Database, IBM DB2, and Microsoft SQL Server are three well-known solutions. On the free and open-source side, MySQL, SQLite, and PostgreSQL are three widely used solutions.

MySQL is used at just about every Internet company you have heard of. In the context of this article, Twitter uses MySQL to store its users' tweets.

SQLite is common in embedded systems. iOS and Android let developers use SQLite to manage their app's private database. Google Chrome uses SQLite to store your browsing history, cookies, and your thumbnails on the "Most visited" page.

PostgreSQL is also a widely used RDBMS. Its PostGIS extension supplements PostgreSQL with geospatial functions that make it useful for mapping applications. A notable user of PostgreSQL is OpenStreetMap.

Structured Query Language (SQL)

Once you've downloaded and set up an RDBMS on your system, the next step is to create a database and tables inside it in order to insert and manage your relational data. The way you do this is with Structured Query Language (SQL), which is the standard language for working with RDBMSs.

There are small variations in SQL between each RDBMS vendor, termed SQL dialects.

Here's a brief overview of common SQL statements that are relevant to the example Twitter data above.

Create a Database Named development

|  |
| --- |
|  |
| CREATE DATABASE development; | |

Create a Table Named users

|  |
| --- |
|  |
| CREATE TABLE users ( | |
|  | |

|  |
| --- |
| full\_name VARCHAR(100), |
|  |

|  |
| --- |
| username VARCHAR(100) |
|  |

|  |
| --- |
| ); |

RDBMSs require that each column in a table is given a data type. Here I have assigned the "full\_name" and "username" columns the data type VARCHAR, which is a string that can vary in width. I've arbitrarily set a max length of 100. A full list of the various data types can be found on Wikipedia.

Insert a Record

|  |
| --- |
|  |
| INSERT INTO users (full\_name, username) | |
|  | |

|  |
| --- |
| VALUES ("Boris Hadjur", "\_DreamLead"); |

Retrieve All Tweets Belonging to a User

|  |
| --- |
|  |
| SELECT text, created\_at | |
|  | |

|  |
| --- |
| FROM tweets |
|  |

|  |
| --- |
| WHERE username="\_DreamLead"; |

Update a User's Name

|  |
| --- |
|  |
| UPDATE users | |
|  | |

|  |
| --- |
| SET full\_name="Boris H" |
|  |

|  |
| --- |
| WHERE username="\_DreamLead"; |

Delete a User

|  |
| --- |
|  |
| DELETE FROM users | |
|  | |

|  |
| --- |
| WHERE username="\_DreamLead"; |

SQL is pretty similar to regular English sentences. There are small variations in SQL between each RDBMS vendor, termed SQL dialects, but the differences are not so dramatic that you can't easily transfer your SQL knowledge from one to the other.