Data Modeling

🎯 Goal: populate your `northwind` database with the data from northwind `csv`'s.

What is *data modeling*?

- Data model specifies the relationships between the data in a database and structures information into related tables.
- In our case, it will mean creating the right kinds of tables for the data we want to import.

📝 Workflow

1. Pick a file from your data dump.
2. Look at the file contents.
3.

```
CREATE TABLE mytable (
  column1 datatype [constraint],
  column2 datatype [constraint],
...
);
```

4.

```
COPY mytable FROM '/path/to/file/mytable.csv' DELIMITER ',' CSV HEADER;
```

5. Repeat until you've imported all files. 🙌

🔢 Data Types

PostgreSQL has a wide range of data types it supports (some personal favorites: `POLYGON`, `CIRCLE`). Data types actually likely to be of use to you in your day-to-day work:

data type	description
INT	integer number
NUMERIC	floating point number
TEXT	long text
VARCHAR(N)	text with a maximum length of N characters
CHAR(N)	text with exact length of N characters
DATE	year/month/day
TIMESTAMP	year/month/day hour:min:sec
SERIAL	integer that counts up automatically
BOOL	boolean
JSON	Json document
UUID	Universally Unique Identifiers

🚫 Constraints

`NOT NULL` constraint: column can't contain missing / null values

`UNIQUE` constraint: column can't contain duplicates

`PRIMARY KEY` constraint: column which uniquely identifies each row; has to have unique values and can't contain null values. A table can contain only one primary key (but it can consist of more than one column). Behind the scenes primary key constraint creates the index for a table -> more on that on Thursday.

📜 Summary of Commands

action	psql
create database	`CREATE DATABASE mydb;`
delete database	`DROP DATABASE mydb;`
list databases	`\l`
list tables	`\dt`
list users	`\du`

Aa action	psql
show connection (connect to a database)	`\c` , `\c mydb`
describe table	`\d mytable`

Extras

1. To run commands from a `.sql` file: `psql -f myfile.sql`
2. For Windows users, here's how you specify your path inside `\copy`:

   ```
   'C:\Users\Username\northwind_data_clean\data\order_details.csv'
   ```

   (You need to state your partition, and slashes are reversed from how it's done on mac/linux)
3. You can see what kind of data you have in your table with `\d mytable` (column names, data types, constraints, primary keys). This will **not** show you any data. To have a look at the data you can use `SELECT`, like this for example:

   ```
   SELECT * FROM mytable LIMIT 20;
   ```
4. `ALTER TABLE` is how you modify table definitions (remove columns, change data types, etc).

   https://www.postgresql.org/docs/9.1/sql-altertable.html

   Specifically, to change/remove primary key:

```
ALTER TABLE products DROP CONSTRAINT products_pkey;
ALTER TABLE products ADD PRIMARY KEY (productid);
```

5. If you want to skip / ignore some columns when copying the data from the csv into the database, you can:
   a. first copy all data into a temporary table, and then drop the column you are not interested in keeping, e.g: `ALTER TABLE categories_temp DROP COLUMN picture;`
   b. use `FROM PROGRAM` with `cut`

   ```
   COPY categories
   FROM PROGRAM 'cut -f 1-3 -d "," /Users/Username/northwind_data_clean/data/categories.csv'
   CSV HEADER;
   ```