UNIT 3 - Sprint 2- Module 1

Intro to SQL and Databases

Prof. Aleksandra Singer

Useful Links

DB Browser:	https://sqlitebrowser.org/ OR https://sqlitebrowser.org/dl/
Public Github/Schema Diagram:	https://github.com/LambdaSchool/DS-Unit-3-Sprint-2-SQL-and-Databases/blob/master/module1-introduction-to-sql/schema.png https://github.com/LambdaSchool/DS-Unit-3-Sprint-2-SQL-and-Databases
For tomorrow: Get Elephantsql:	https://www.elephantsql.com/docs/
For Homework:	https://pandas.pydata.org/pandas-docs/stable/reference/api/pandas.DataFrame.to_sql.html

Useful Links for the Rest of the Week

Documentation for psycopg	http://initd.org/psycopg/docs/
Mode SQL Lessons	https://sqlbolt.com/
Postgresql tutorials	http://www.postgresqltutorial.com/
	http://www.postgresqltutorial.com/postgresql-cheat-sheet/
Mongodb	https://www.mongodb.com/
Thinking about functions:	http://mathonweb.com/help_ebook/html/functions_6.htm

Objectives:

1. Define data, data lakes, warehouses, databases, datasets
2. Write basic SQL queries to get specific subsets of data from a database and answer basic "business questions" (with/without python)
3. Understand the purpose of SQL join, and perform joins to access data from multiple tables

Learning Terms

What is data?

What is a dataset?

What is a database?

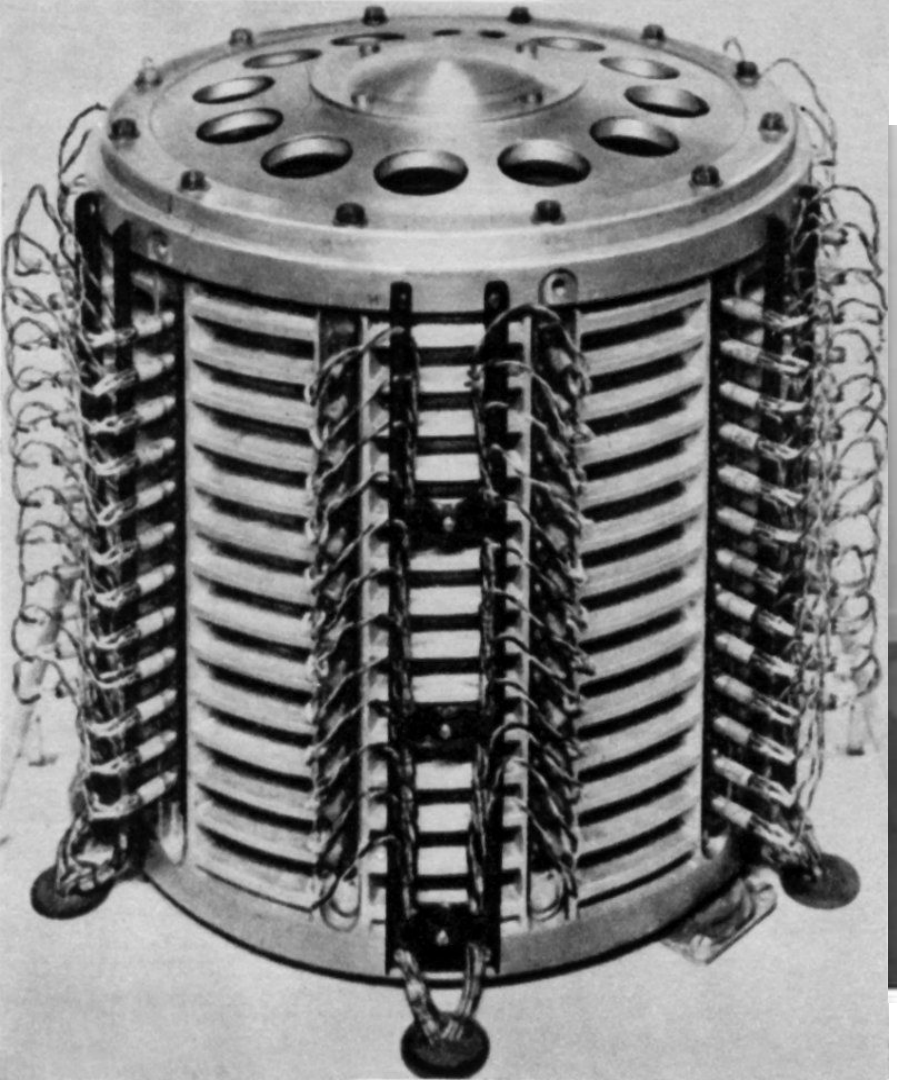
What is a data warehouse?

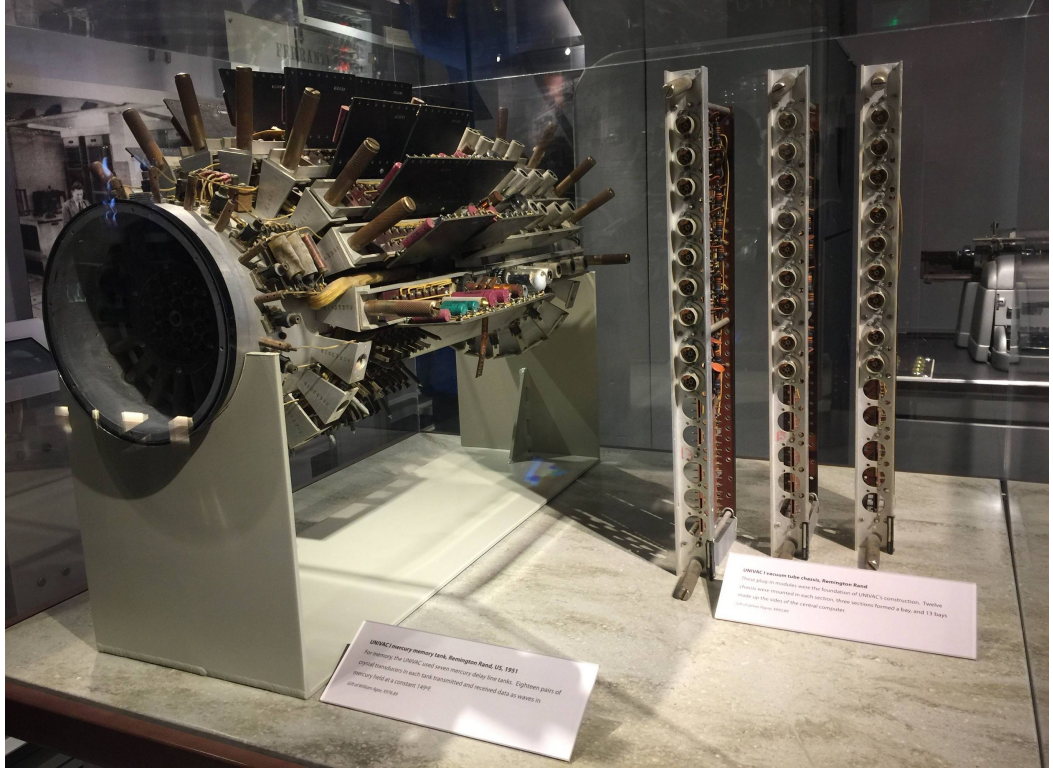
What is a data lake?

Big Data

class	size	manage with	how it fits	examples
small	< 10 GB	Excel, R	fits in one machine's memory	thousands of sales figures
medium	10GB-1TB	indexed files, monolothic DB	fits on one machine's disk	millions of web pages
Big	> 1TB	Hadoop, distributed DBs	stored across many machines	billions of web clicks

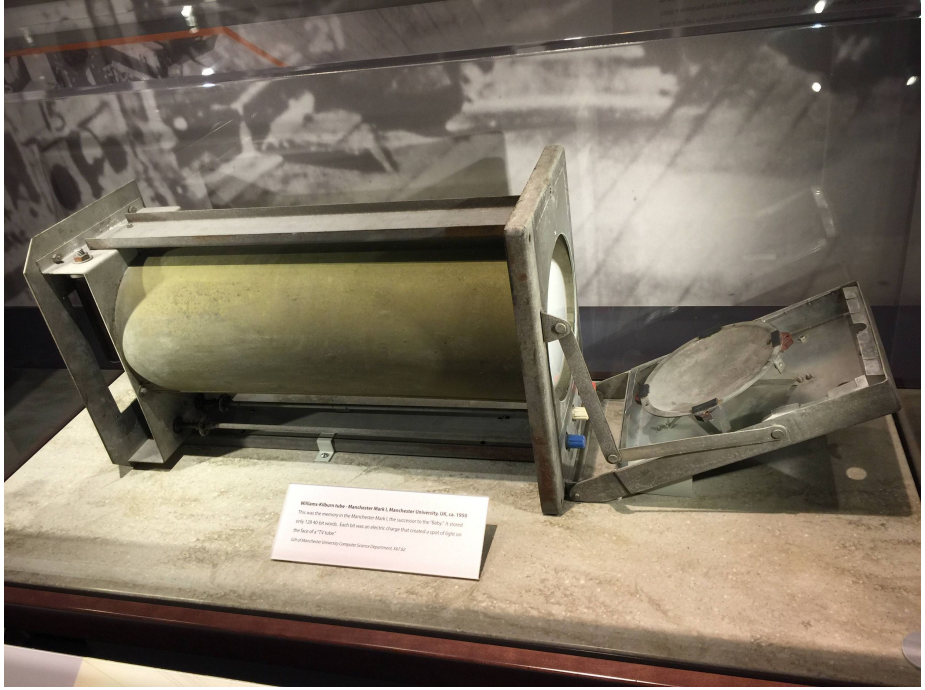
source:
https://www.quora.com/How-much-data-is-Big-Data-Is-there-classification-for-various-levels-of-Big-Data-by-amount-of-data-processed-or-other-constraints-like-for-example-throughput-What%E2%80%99s-the-minimum-data-size-which-still-qualifies-as-a-Big-Data%E2%80%9D



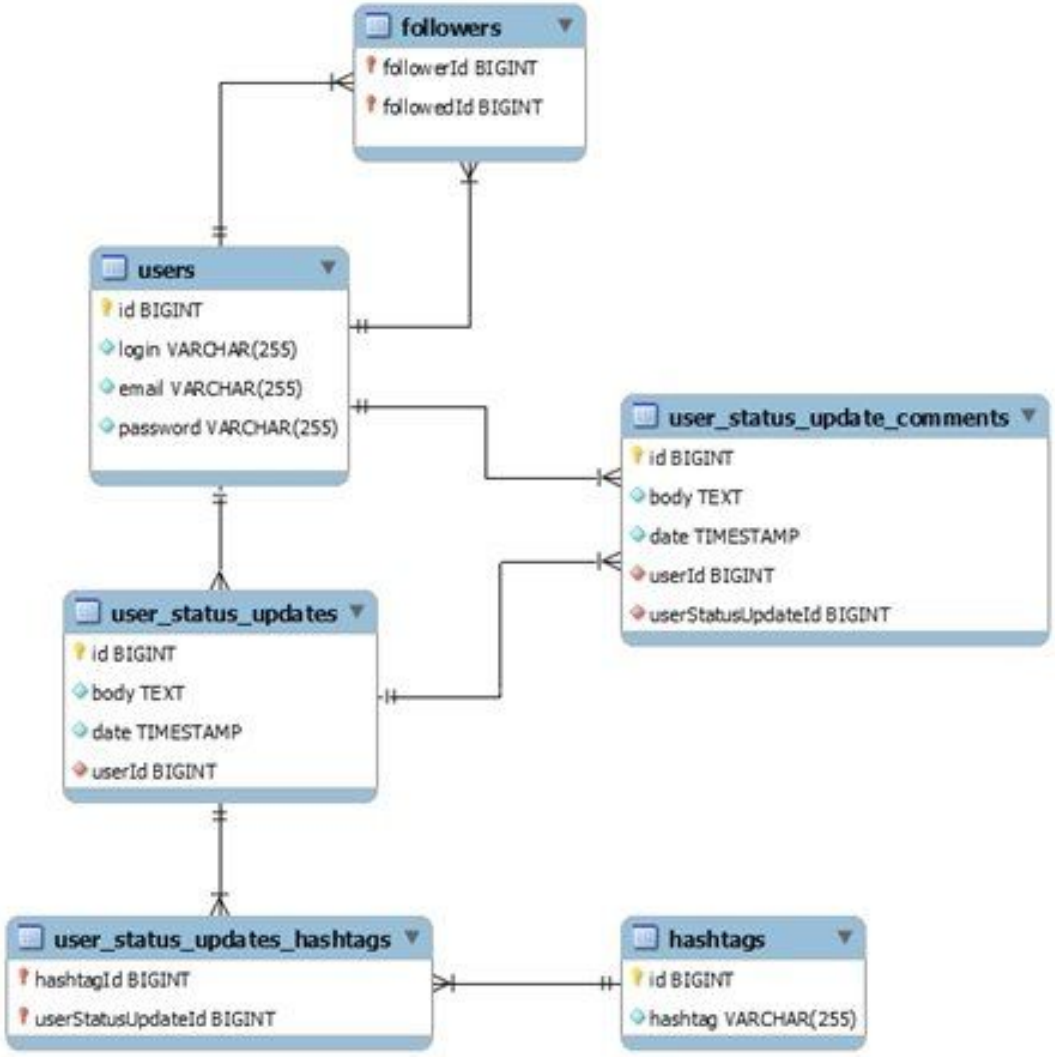


UNIVAC I mercury memory tank, Remington Rand, US, 1951. For memory, the UNIVAC used seven mercury delay line tanks. Eighteen pairs of crystal transducers in each tank transmitted and received data as waves in mercury held at a constant 149°F”

source: https://stackoverflow.com/questions/2822650/why-is-a-database-always-represented-with-a-cylinder



same source. “Williams-Kilburn tube - Manchester Mark I, Manchester University, UK, ca 1950. This was the memory in the Manchester Mark I, the successor to the "Baby." It stored only 128 40-bit words. Each bit was an electric charge that created a spot of light on the face of a "TV tube."”


followers
followerId BIGINT
followedId BIGINT
users
id BIGINT
login VARCHAR(255)
email VARCHAR(255)
password VARCHAR(255)
user_status_update_comments
id BIGINT
body TEXT
date TIMESTAMP
userId BIGINT
userStatusUpdateId BIGINT
user_status_updates
id BIGINT
body TEXT
date TIMESTAMP
userId BIGINT
user_status_updates_hashtags
hashtagId BIGINT
userStatusUpdateId BIGINT
hashtags
id BIGINT
hashtag VARCHAR(255)

SQL keywords:

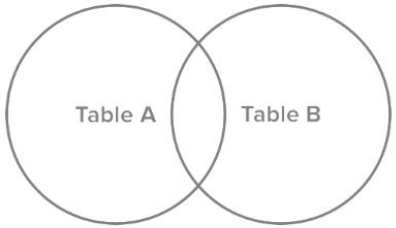
SELECT - how we choose which columns to get
WHERE - how we set conditions on the rows to be returned
LIMIT - when we only want a certain number of rows
ORDER - when we want to sort the output
JOIN - when we need data from multiple tables combined

SQL JOINS. source for image:

http://www.sql-join.com/sql-join-types

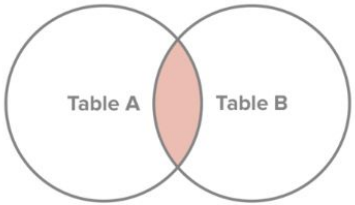
Let’s go back to high school math class! relational algebra. which ones create NAs?

http://mathonweb.com/help_ebook/html/functions_6.htm



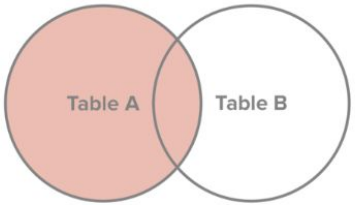
The extent of the overlap, if any, is determined by how many records in Table A match the records in Table B. Depending on what subset of data we would like to select from the two tables, the four join types can be visualized by highlighting the corresponding sections of the Venn diagram:

Inner Join



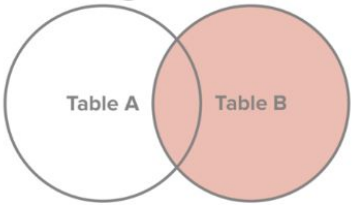
Select all records from Table A and Table B, where the join condition is met.

Left Join



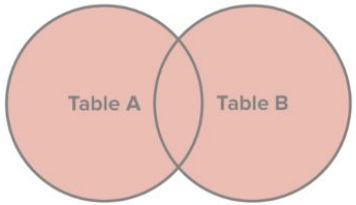
Select all records from Table A, along with records from Table B for which the join condition is met (if at all).

Right Join



Select all records from Table B, along with records from Table A for which the join condition is met (if at all).

Full Join



Select all records from Table A and Table B, regardless of whether the join condition is met or not.

A Visual Guide to Joins

https://blog.codinghorror.com/a-visual-explanation-of-sql-joins/

SQL cheat sheet: http://www.sqltutorial.org/sql-cheat-sheet/

SQL CHEAT SHEET http://www.sqltutorial.org

MANAGING VIEWS

```
CREATE VIEW v(c1,c2)
AS
SELECT c1, c2
FROM t;
```
Create a new view that consists of c1 and c2

```
CREATE VIEW v(c1,c2)
AS
SELECT c1, c2
FROM t;
WITH [CASCADED | LOCAL] CHECK OPTION;
```
Create a new view with check option

```
CREATE RECURSIVE VIEW v
AS
select-statement -- anchor part
UNION [ALL]
select-statement; -- recursive part
```
Create a recursive view

```
CREATE TEMPORARY VIEW v
AS
SELECT c1, c2
FROM t;
```
Create a temporary view

```
DROP VIEW view_name;
```
Delete a view

MANAGING INDEXES

```
CREATE INDEX idx_name
ON t(c1,c2);
```
Create an index on c1 and c2 of the table t

```
CREATE UNIQUE INDEX idx_name
ON t(c3,c4);
```
Create a unique index on c3, c4 of the table t

```
DROP INDEX idx_name;
```
Drop an index

SQL AGGREGATE FUNCTIONS

AVG returns the average of a list

COUNT returns the number of elements of a list

SUM returns the total of a list

MAX returns the maximum value in a list

MIN returns the minimum value in a list

MANAGING TRIGGERS

```
CREATE OR MODIFY TRIGGER trigger_name
WHEN EVENT
ON table_name TRIGGER_TYPE
EXECUTE stored_procedure;
```
Create or modify a trigger

WHEN
- **BEFORE** – invoke before the event occurs
- **AFTER** – invoke after the event occurs

EVENT
- **INSERT** – invoke for INSERT
- **UPDATE** – invoke for UPDATE
- **DELETE** – invoke for DELETE

TRIGGER_TYPE
- **FOR EACH ROW**
- **FOR EACH STATEMENT**

```
CREATE TRIGGER before_insert_person
BEFORE INSERT
ON person FOR EACH ROW
EXECUTE stored_procedure;
```
Create a trigger invoked before a new row is inserted into the person table

```
DROP TRIGGER trigger_name;
```
Delete a specific trigger

Objectives:

1. Define data lakes, warehouses, databases, datasets
2. Write basic SQL queries to get specific subsets of data from a database and answer basic "business questions" (with/without python)
3. Understand the purpose of SQL join, and perform joins to access data from multiple tables