TECH TALENT SOUTH

SQL Part 1: Databases

Data is everything.

<Image Source: http://www.hotelogix.com/blog/wp-content/uploads/2017/09/Data-collection.jpg >

- Web and Mobile apps are (for the most part) interfaces for data retrieval and interaction.
- Many companies make a major portion of their income off of data they can collect around user preferences
- User data can drive personalized and, therefore, more favorable user experiences

TECH TALENT SOUTH

Companies

Rely

On

Web Applications

Databases

Which in turn rely on...

TECH TALENT SOUTH

What is a Database?

A database is a collection of information that is organized so that it can be easily accessed, managed and updated.

Computer databases typically contain aggregations of data records or files, such as sales transactions, product catalogs and inventories, as well as customer profiles.

TECH TALENT SOUTH

What is a Database?

Databases can be held on large mainframe systems, but are also present in smaller distributed workstations, midrange systems, and on the cloud.

TECH TALENT SOUTH

The History of Databases

The necessity to store, update, and retrieve data has been around since ancient times.

However, as technological advances sped up other aspects of society, they also necessitated more modern approaches to data management.

TECH TALENT SOUTH

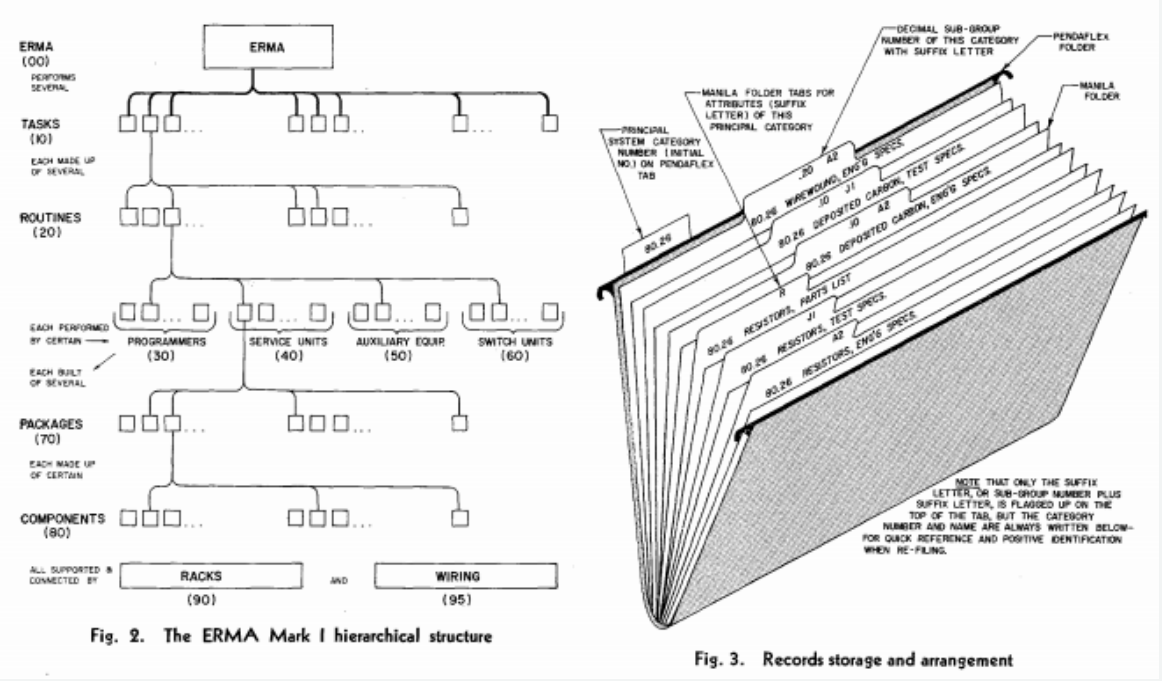
The History of Databases

The use of electronic databases began in the early 1960s and correlated directly with the evolution of contemporary computing as well as the rise of the internet.

TECH TALENT SOUTH

The History of Databases

Many models of electronic data management evolved from earlier, paper based systems.



TECH TALENT SOUTH

The History of Databases

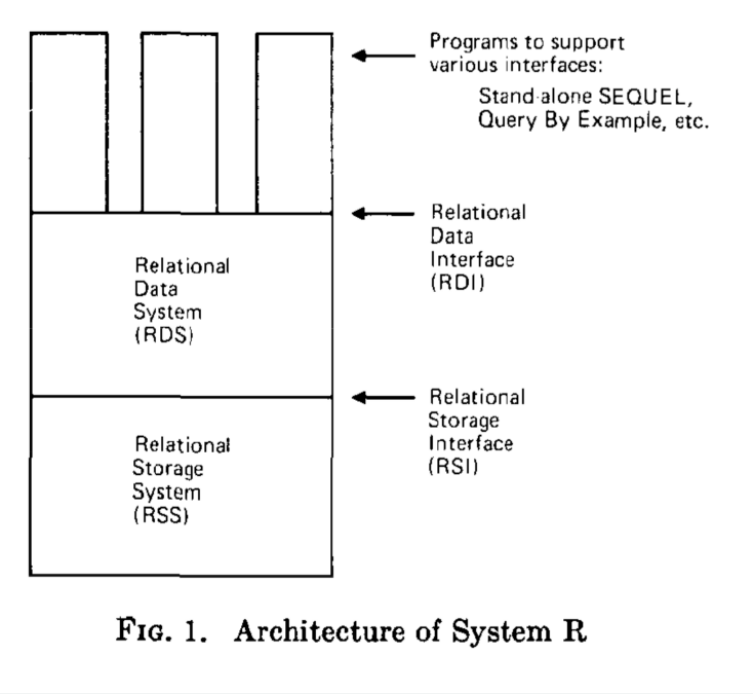


FIG. 1. Architecture of System R

The prototypal database system developed by IBM in 1974, System R, ultimately became SQL -- which has dominated the database field since the 1980s.

TECH TALENT SOUTH

What is a DBMS

DBMS (Database Management System) refers to a group of programs which allows the clients to obtain and modify data.

The management system is also used to regulate the accessibility of any form of data, taking into consideration that some types of data are confidential.

TECH TALENT SOUTH

Basic Components of a DBMS

Each DBMS contains some basic elements which enhances the functionality. These elements are described below:

1. **Table** - Any set of like items in a DBMS is stored in a table. Tables are made up of rows and columns.
2. **Row** - A row represents each distinct data entry (**record)** found in the table.
3. **Column** - This represents the upright data entry found in a table which is related to a particular field.
4. **Field** - After storing data in a table, the tables are then sub divided into fields. The field is where specific data is kept.

TECH TALENT SOUTH

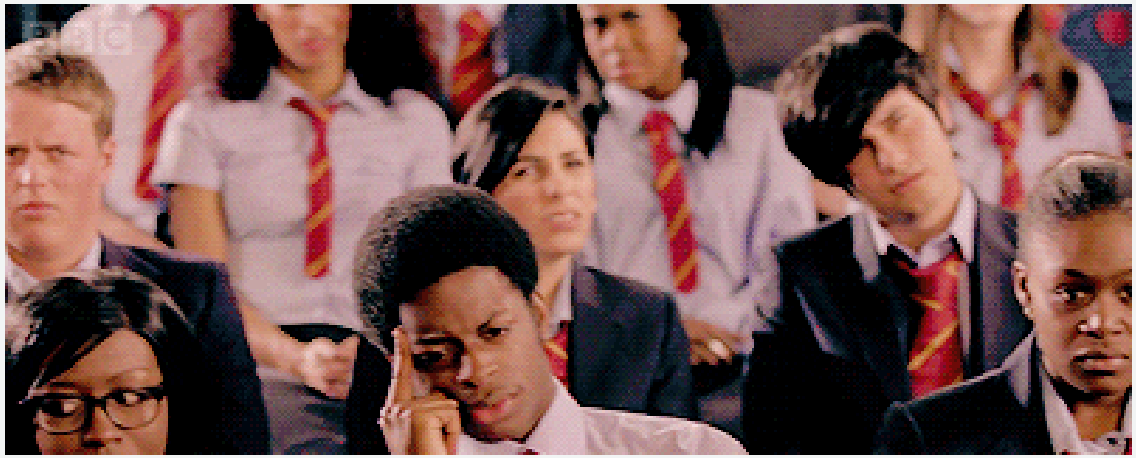
Data Integrity

DBMSs help to maintain data consistency in a database.

Data integrity comprises of the following:

1. **Domain integrity** – used to verify the relevant data entries made in a column by limiting the scope and format of values.
2. **Entity integrity** – it ensures no repeated rows are put on a table.
3. **User-defined integrity** – it implements particular operational regulations that are not found in the other types of data integrity.
4. **Referential integrity** – it safeguards the rows to ensure they cannot be erased.

Data Integrity



Database Normalization refers to the process of ensuring that data is stored effectively in a database. The aim of this process is to eradicate unnecessary data as well as making sure that the data is logically structured.

TECH TALENT SOUTH

Data Types

Every column will have a data type.

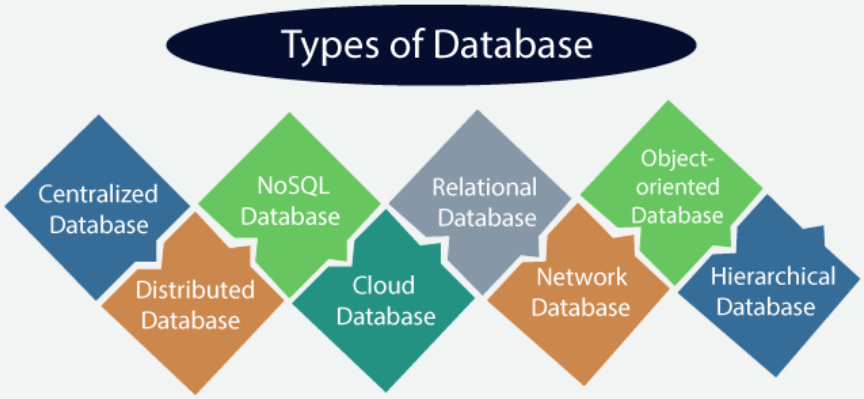
By assigning data types we keep a column's attributes standardized.

Just about every ID column (especially key columns) will be of data type INTEGER, whereas columns with names or descriptions will usually be Strings or char-based datatypes columns.

When you create a table in a relational database, you must assign a data type to each column in the table.

TECH TALENT SOUTH

Types of Database Management Systems (DBMS)



DBMS is categorized into four types:

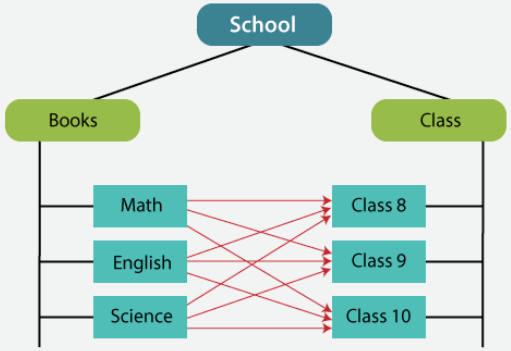
- Network DBMS
- Hierarchal DBMS
- Object Oriented Relational DBMS *
- Relational DBMS

TECH TALENT SOUTH

Types of Database Management Systems (DBMS)

Network DBMS

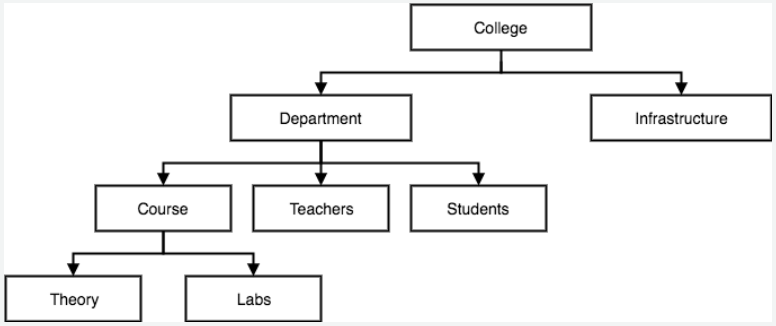
As the name suggests, this type of DBMS comprises of diverse relations which form complex database programs.



Network Database Management System

Hierarchical DBMS

It utilizes hierarchies of data storage. The main data is subdivided into smaller bits which are stored in compartments.



TECH TALENT SOUTH

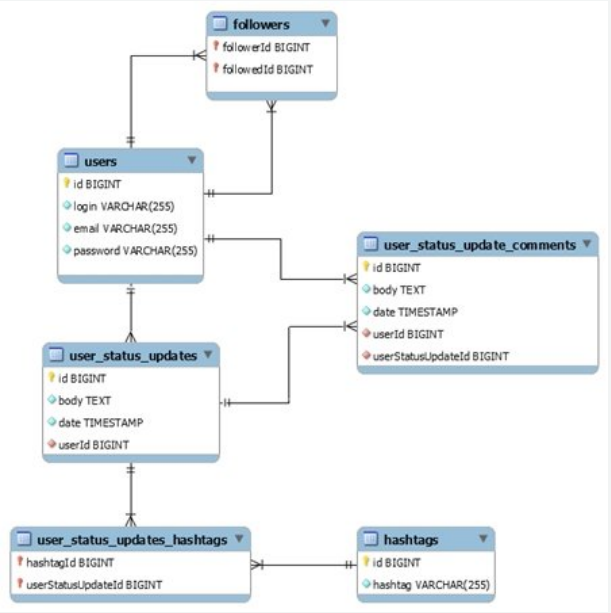
Types of Database Management Systems (DBMS)

Object oriented relational DBMS

It enables the data to be stored as objects as opposed to values. The objects are characterized distinctly.

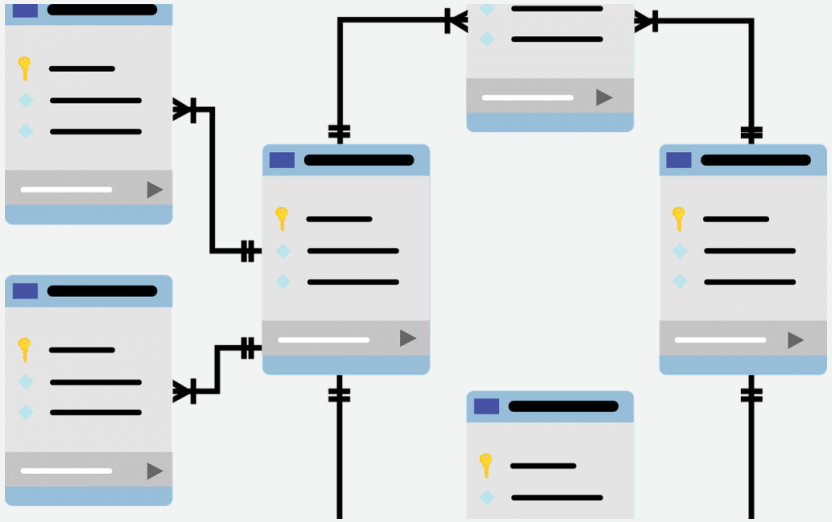
Relational DBMS

This is the most prominent DBMS available. The database integration is defined as tables. It also contains predefined data that is supported by the relations.



TECH TALENT SOUTH

Relational Database



- Stores data in tables (similar to Excel Spreadsheets)
- The tables consist of columns and rows
- Tables are related to each other by way of primary and foreign keys
- A record is a row in a table
- The relationships between tables must be created and defined before any data (rows) can be added

TECH TALENT SOUTH

Relational Databases

Real World Example

Suppose that you own a construction company and you need a way to keep track of your equipment items and their service histories.

TECH TALENT SOUTH

Real World Example

Your database might have a table to hold all of your equipment assets.

- each **record** (row) refers to a real-world item

Equipment Id	Name	Purchase Price	Purchase Date

In the Equipment table, we might have four columns:

1. EquipmentID (the primary key, which we will get to shortly)
2. Name
3. PurchasePrice
4. PurchaseDate.

TECH TALENT SOUTH

Real World Example

You might also have a Service table, each **record** of which would represent a particular service performed.

Service ID	Equipment ID	Service Type	Service Date

In the service table, we also have four columns:

1. ServiceID (primary key)
2. EquipmentID (foreign key to the Equipment table)
3. ServiceType
4. PurchaseDate.

TECH TALENT SOUTH

Linked Tables

Tables can be related or linked to one another by way of two kinds of keys:

1. Primary Key
2. Foreign Key

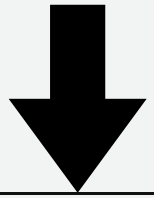
Generally, primary and foreign keys must be assigned to tables upon creation, although some tables do not need them.

TECH TALENT SOUTH

Primary Key

A **primary key** is a **column** in a relational database that holds a unique identifier for each record.

Each table in a relational database can have only one **primary key.**



Equipment ID	Name	Purchase Price	Purchase Date

TECH TALENT SOUTH

Primary Key

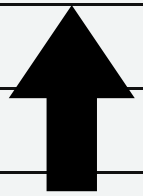
Primary key columns are often set as identity columns, meaning that the database will auto-generate the value when a new row is added.

For example, if I added a new row in the Equipment table, I would insert all of the relevant information—Name, Purchase Price, etc.—except for the EquipmentID. The database will determine that value.

TECH TALENT SOUTH

Foreign Key

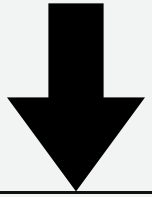
Service ID	Equipment ID	Service Type	Service Date



Notice the EquipmentID column in the Service table.

It is a **foreign key**.

Each Equipment ID in the Service table refers to an equipment record in the Equipment table.



Equipment ID	Name	Purchase Price	Purchase Date

TECH TALENT SOUTH

One-to-Many Association

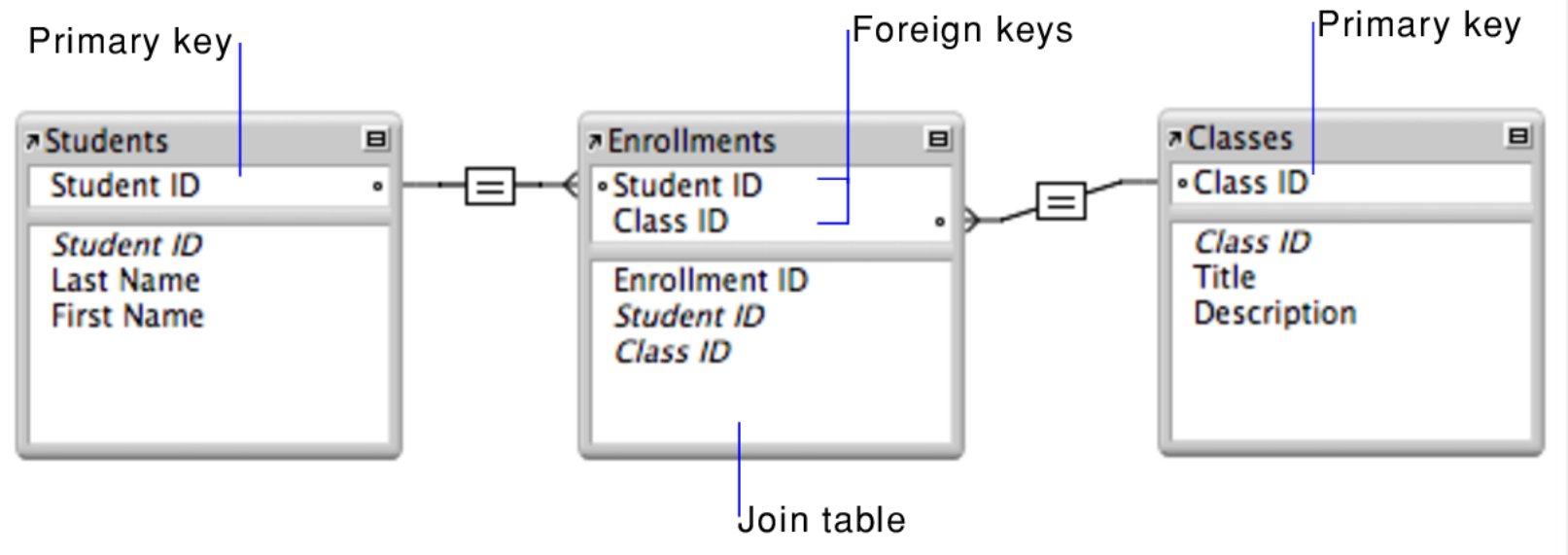
We call the relationship between the Equipment and it's Services a **one-to-many** relationship.

A **one-to-many** relationship refers to the relationship between two entities, A and B, in which an element of A may be linked to many elements of B, but a member of B is linked to only one element of A.

TECH TALENT SOUTH

Many-to-Many Association

Many-to-Many relationships occur when item A can have many elements of item B, but item B can also be associated with many elements of item A.



TECH TALENT SOUTH

SQL

How exactly do we interact with a relational database?

SQL, short for “Structured Query Language” is a set of commands and procedures that allow us store, manipulate, and retrieve data in a relational database.

TECH TALENT SOUTH

SQL and NoSQL

Not all databases are SQL databases though....

NoSQL is defined as Not only Structured Query Language. This type of database management system is non-relational and, therefore, provides storage and retrieval of data not stored in tables.

Benefits

- It has simple interface and design
- It has flexible data structures

Challenges

On the other hand, NoSQL has faced several barriers to entry which have made SQL more popular.

TECH TALENT SOUTH

So many SQLs

Structured Query Language (**SQL**) is considered the standard language for relational database management systems.

MySQL

SQLite3

Postgres

SQL Server

Oracle

TECH TALENT SOUTH

So Many SQLs...

While all of these SQL Databases are relational and all use the same basic SQL syntax, some have differences in dialect

The concepts we'll learn here will be transferable no matter the SQL dialect you use. However, we will focus on SQLite3 as it's easy to setup and it will allow us to learn in a playground environment.

TECH TALENT SOUTH

MySQL

MySQL is a decentralized SQL database attributed to a Swedish company. This language is applicable in operation systems such as MS Windows, Linux, Mac OS X as well as UNIX and has the following features:

- It has ease of use.
- It is efficient.
- It has the capability to increase its output.
- It is operational on a 24/7 basis.
- It is easily manageable.

TECH TALENT SOUTH

MS SQL Server

Developed by Microsoft Incorporation, its main features are:

- It is efficient.
- It is easily accessible.
- It can create and maintain redundant database copies.
- It can create Data Definition Language (DDL).
- It can combine the Common Language Runtime (CLR).
- It can also utilize the Extensible Markup Language (XML).

TECH TALENT SOUTH

Oracle

Developed by Oracle Corporation, this RDMS can serve multiple users, works well for both the clients and server database input, and can accommodate diverse operating systems:

- It can display commands concurrently.
- It is consistent.
- The database can be inactivated.
- It can accommodate bitmap indexes.
- It has a component for managing the database resources.
- It has the ability to evaluate large databases to establish patterns.
- It has data warehousing function. This system allows data to be reported and analyzed.

TECH TALENT SOUTH

SQLite

We will use an SQL managing client called sqliteonline.com

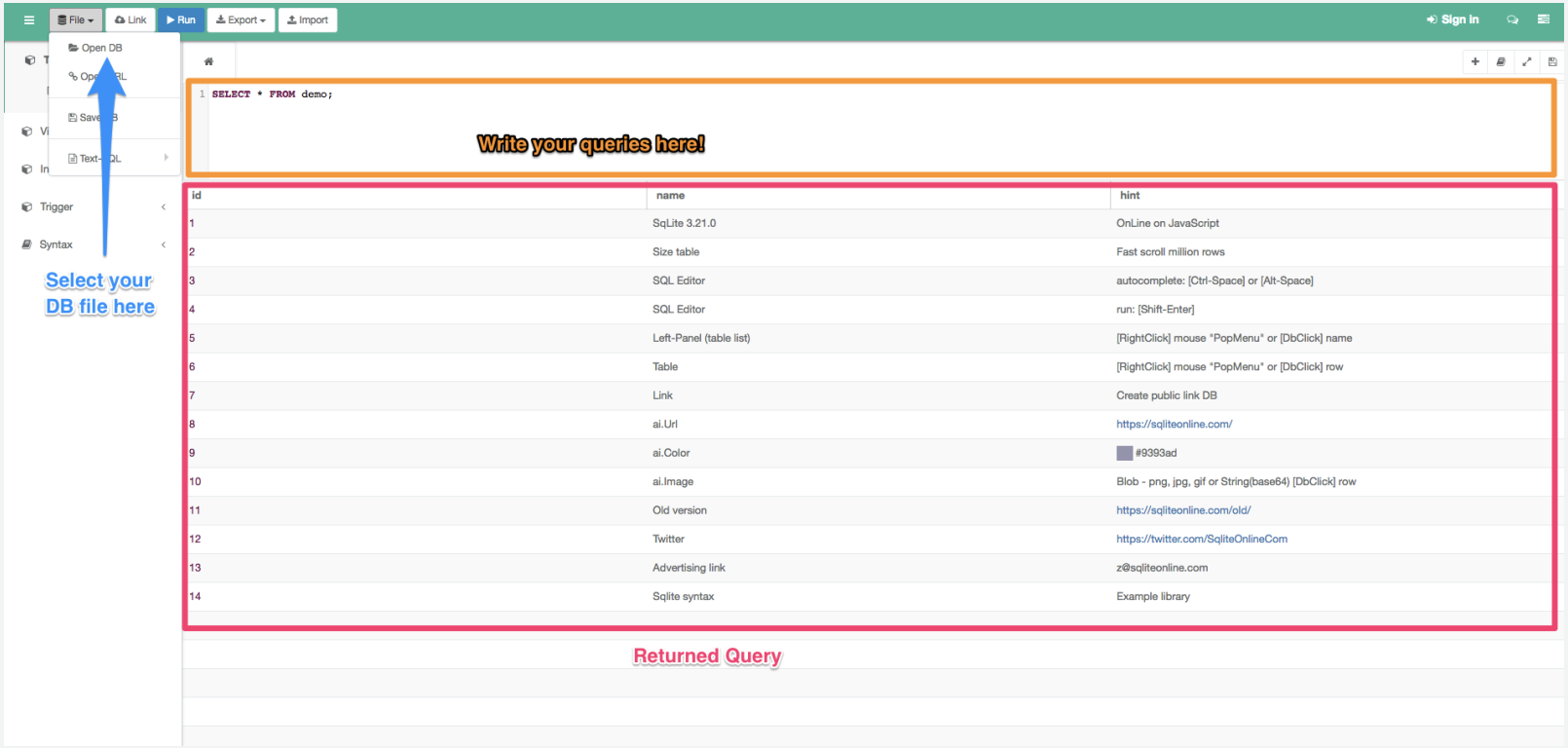
This is a lightweight SQLite Manager, and we can easily upload databases, and work with them directly in the browser.

SQLite is most like MySQL as they are virtually the same dialect, but the behind the scenes setup and use is different.

TECH TALENT SOUTH

Open your DB

Let's look at the SQL managing client we'll be using, sqliteonline.com



id	name	hint
1	SqLite 3.21.0	OnLine on JavaScript
2	Size table	Fast scroll million rows
3	SQL Editor	autocomplete: [Ctrl-Space] or [Alt-Space]
4	SQL Editor	run: [Shift-Enter]
5	Left-Panel (table list)	[RightClick] mouse "PopMenu" or [DbClick] name
6	Table	[RightClick] mouse "PopMenu" or [DbClick] row
7	Link	Create public link DB
8	ai.Url	https://sqliteonline.com/
9	ai.Color	#9393ad
10	ai.Image	Blob - png, jpg, gif or String(base64) [DbClick] row
11	Old version	https://sqliteonline.com/old/
12	Twitter	https://twitter.com/SqliteOnlineCom
13	Advertising link	z@sqliteonline.com
14	Sqlite syntax	Example library

TECH TALENT SOUTH

Practice with Example DBs

There are example databases that we will work with in this course.

You can download these example databases and their corresponding schema at this GitHub repository.

Go ahead and download the database "my_ipod.sqlite3"

Let's also take a look at its schema, so we understand how it is organized.

TECH TALENT SOUTH

Exercise 1

Based on the schema for my_ipod:

1. What are the tables in this database?
2. What are the columns (properties associated) with each of the tables?
3. Can you identify the data types that each column will hold?
4. Are there any tables that include foreign keys? What foreign keys do they include?
5. Are there any one-to-many relationships? Many-to-Many?

TECH TALENT SOUTH