http://bundler.io/

Create Main.rb /App.rb file

require “bundler”

Bundler.require

Create new Gemfile (#touch Gemfile)

source <https://rubygems.org>

gem ‘sinatra’

gem ‘sinatra/reloader’ #--> ‘sinatra-reloader’

gem ‘pry’

gem httparty’

gem ‘json’

In bash folder:

bundle install

# 🡪 creates Gemfile.lock

Store data

🡪 storing in database is permanent

🡪 in hash or arrays is temporary

Manage data efficiently

Database -- Big collection of data

Relational database

- large collection of tables

- Name and data type

Columns - fixed amount of columnsin a table.

-You can edit the number of columns but may mess up data

- Number of rows is variable and depends on how much information you have

POSTGRES SQL UNINSTRALL

brew uninstall postgres

Quit TERMINAL

brew install postgresql

initdb /usr/local/var/postgres -E utf8

createdb ${USER}

# Ensure that Postgres launches whenever we login

mkdir -p ~/Library/LaunchAgents

cp /usr/local/Cellar/postgresql/9.\*/homebrew.mxcl.postgresql.plist ~/Library/LaunchAgents/

# Start Postgres now

launchctl load ~/Library/LaunchAgents/homebrew.mxcl.postgresql.plist

\? 🡪 access full index of psql commands

\l List all databases

\c Change db

\d Display all tables in database

\d tablename Describe the details of given table

\h Get a help on syntax of SQL commands

\? Lists all psql slash commands.

\q Quit psql

SQL static language vs. Ruby (dynamic)

<http://stackoverflow.com/questions/125367/dynamic-type-languages-versus-static-type-languages>

<http://en.wikipedia.org/wiki/Type_system>

**varchar** refers to a data type of a field (or [column](http://en.wikipedia.org/wiki/Column_(database))) in a[database management system](http://en.wikipedia.org/wiki/Database_management_system)

char(2)

varchar(255)

- Always 255

text()

<http://stackoverflow.com/questions/125367/dynamic-type-languages-versus-static-type-languages>

Serial – Creates Ids 🡪 auto incrementing #s

Timestap -- Creates record of time 🡪 transactions 🡪 error checking

Sql statements always end in ;

USING COMMANDS SHOULD BE IN ALL CAPS (out of convention 🡪 user readability)

Power Architect: SQL Data Modeling & Profiling Tool

<https://code.google.com/p/power-architect/>

SELECT Statements -🡪 Getting information out.

SELECT \*

FROM tbl-name;

WHERE col-name = ‘something’

If ‘something’=#, do not use single quotes, otherwise need single quotes

Single = 🡪 equals

INSERT INTO demo\_lab

(id, first\_name, last\_name, lab\_name

VALUES (2028, ‘Stephen’, ‘Hawking,’ ‘WDI’);

UPDATE scientists SET id=2038 WHERE id=2028 AND last\_name=’Hawking’;

Tables should have one column that is a unique identifier

projected --

studentID --

session

location

title

date

char(12) 🡪 reserves 12 characters of space

varchar(255) 🡪 255 characters is upper limit but will only use however many characters of space you put in it

INSERT INTO

INSERT INTO projects

(projectID, studentID, session, location, title, date)

VALUES (124, 'April', 0101, 'New York', 'Musicz', 043014);

When you set projectID to variable key 🡪 auto generate, do not input a value

CRUD operations on database

CREATE TABLE students

(

id serial NOT NULL primary key,

first\_name varchar(255),

last\_name varchar(255),

birthdate date

);

require 'pg'

@conn = PG.connect(dbname: 'students\_db')

# INSERT INTO students (first\_name, last\_name, birthdate) VALUES ();

def add\_student(first\_name, last\_name, birthdate)

@conn.exec("INSERT INTO students (first\_name, last\_name, birthdate) VALUES ('#{first\_name}', '#{last\_name}', '#{birthdate}');")

end

🡪 Conn.exec 🡪 IS IT WORKING SQL INSIDE OF DOUBLE QUOTATIONS?

In pry load “./main.rb”

Add\_student(“Neel”, “Patel”, “July 22, 1985”)