-- THESE ARE THE BASICS OF SQL

-- create database chicken;

-- show databses;

use chicken;

select database ();

CREATE TABLE birds (

name varchar(50),

age INT);

CREATE TABLE cats (

name VARCHAR(50),

age INT

);

CREATE TABLE dogs (

name VARCHAR(50),

breed VARCHAR(50),

age INT

);

SHOW TABLES;

SHOW COLUMNS FROM cats;

SHOW COLUMNS FROM dogs;

desc birds;

desc dogs;

drop table cats;

-- creating the cats table again to perform the insert function

CREATE TABLE cats (

name VARCHAR(50),

age INT

);

-- checking the cats table

desc cats;

-- the information is inserted

INSERT INTO cats(name, age)

VALUES('Jetson', 7);

-- checking the cats table

desc cats;

-- inserting next cat

INSERT INTO cats (name, age)

VALUES ('Jenkins', 7);

INSERT INTO cats (name, age)

VALUES ('Blue Steele', 5);

-- select from chickden database the value of cats table.

SELECT \* FROM chicken.cats;

-- alternate order of inserts.

INSERT INTO cats (age, name)

VALUES

(2, 'Beth');

-- multiple insert.

INSERT INTO cats (name, age)

VALUES

('Meatball', 5),

('Turkey', 1),

('Potato Face', 15);

-- check the cats table.

select \* from cats;

-- This exercise is to create a people table insert 1 set then reverse order then multiple insert

CREATE TABLE people(

first\_name VARCHAR(20),

last\_name VARCHAR(20),

age INT

);

-- Check the info from people table

desc people;

-- first person

INSERT INTO people(first\_name, last\_name, age)

VALUES ('Tina', 'Belcher', 13);

-- second person

INSERT INTO people(age, last\_name, first\_name)

VALUES (42, 'Belcher', 'Bob');

-- multiple insert

INSERT INTO people(first\_name, last\_name, age)

VALUES

('Linda', 'Belcher', 45),

('Phillip', 'Frond', 38),

('Calvin', 'Fischoeder', 70);

SELECT \* FROM people;

SHOW TABLES;

DROP TABLE people;

-- New table cats 2 not NULL

CREATE TABLE cats2 (

name VARCHAR(100) NOT NULL,

age INT NOT NULL

);

-- THIS EXAMPLE IS SHOWING HOW TO PUT ' IN SQL.

CREATE TABLE shops (name VARCHAR(100) NOT NULL);

INSERT INTO shops(name) VALUES ('shoe emporium');

INSERT INTO shops(name) VALUES ('mario\'s pizza');

SELECT \* FROM shops;

-- DEFAULT VALUES

CREATE TABLE cats3 (

name VARCHAR(20) DEFAULT 'no name provided',

age INT DEFAULT 99

);

DESC cats3;

INSERT INTO cats3(age) VALUES(13);

INSERT INTO cats3() VALUES();

-- THIS PREVENTS A NOT NULL CASE

CREATE TABLE cats4 (

name VARCHAR(20) NOT NULL DEFAULT 'unnamed',

age INT NOT NULL DEFAULT 99

);

DESC cats4;

-- PRIMARY KEY

CREATE TABLE unique\_cats (

cat\_id INT PRIMARY KEY,

name VARCHAR(100) NOT NULL,

age INT NOT NULL

);

CREATE TABLE unique\_cats2 (

cat\_id INT,

name VARCHAR(100) NOT NULL,

age INT NOT NULL,

PRIMARY KEY (cat\_id)

);

-- Auto increment primary key

CREATE TABLE unique\_cats3 (

cat\_id INT AUTO\_INCREMENT,

name VARCHAR(100) NOT NULL,

age INT NOT NULL,

PRIMARY KEY (cat\_id)

);

-- eample of employee id

CREATE TABLE employees (

id INT AUTO\_INCREMENT PRIMARY KEY,

first\_name VARCHAR(255) NOT NULL,

last\_name VARCHAR(255) NOT NULL,

middle\_name VARCHAR(255),

age INT NOT NULL,

current\_status VARCHAR(255) NOT NULL DEFAULT 'employed'

);

-- test

INSERT INTO employees(first\_name, last\_name, age) VALUES

('Dora', 'Smith', 58);

select \* from chicken.employees;