-- Comments

-- CREATE TABLE Fruit (

-- Name VARCHAR(64) PRIMARY KEY,

-- Season ENUM('Winter', 'Spring', 'Summer', 'Fall') DEFAULT 'Fall',

-- Shape VARCHAR(32) NOT NULL

-- );

-- CREATE TABLE Fruit\_sales (

-- Vendor VARCHAR(64),

-- Fruit\_type VARCHAR(64),

-- Unit ENUM('kg','lb'),

-- Price DECIMAL(5,2),

-- PRIMARY KEY (Vendor, Fruit\_type),

-- FOREIGN KEY (Fruit\_type) REFERENCES Fruit(Name)

-- );

-- INSERT INTO Fruit\_sales VALUES('Sprouts','apple','lb',1.99),

-- ('Ralphs','apple','lb',1.49),

-- ('Sprouts','dragonfruit','lb',7.99)

-- INSERT INTO Fruit\_sales VALUES('Sprouts','passionfruit','lb',5.99)

-- INSERT INTO Fruit(name,shape) VALUES('apple','sphere');

-- INSERT INTO Fruit VALUES('pear','Fall', 'pear-shaped'),

-- ('dragonfruit', NULL, 'Oval');

-- show tables

-- DROP TABLE Fruit\_sales

-- DESCRIBE Fruit\_sales

-- SELECT \* FROM Fruit\_sales;

-- 1 setp file for each dataset and each file will contain all the .csv file from that dataset

CREATE TABLE CSU\_Campuses (

Id INT PRIMARY KEY,

Campus VARCHAR(128),

Location VARCHAR(64),

County VARCHAR(64),

Year INT

-- Primary key(Id)

-- Unique key(Campus, Location)

);

CREATE TABLE CSU\_Fees (

Campus INT,

Year INT,

Campus\_Fee INT,

FOREIGN KEY (Campus) REFERENCES CSU\_Campuses(Id)

);

CREATE TABLE CSU\_Degrees (

Year INT,

Campus INT,

Degrees INT,

FOREIGN KEY (Campus) REFERENCES CSU\_Campuses(Id)

);

CREATE TABLE CSU\_Discipline\_enrollments (

Campus INT,

Discipline INT,

Year INT,

Undergrad INT,

Grad INT,

FOREIGN KEY (Campus) REFERENCES CSU\_Campuses(Id),

FOREIGN KEY (Discipline)

REFERENCES CSU\_Discipline(Id)

);

CREATE TABLE CSU\_Discipline (

Id INT PRIMARY KEY,

Name VARCHAR(64)

);

CREATE TABLE CSU\_Enrollments (

Campus INT,

Year INT,

Total\_enrollment\_AY INT,

FTE\_AY INT,

FOREIGN KEY (Campus) REFERENCES CSU\_Campuses(Id)

);

CREATE TABLE CSU\_Faculty (

Campus INT,

Year INT,

Faculty INT,

FOREIGN KEY (Campus) REFERENCES CSU\_Campuses(Id)

);

SELECT \* FROM CSU\_Degrees

SHOW TABLES

-- =========================CARS=============================================

CREATE TABLE CARS\_Continents (

Cont\_Id INT PRIMARY KEY,

Continent VARCHAR(16)

);

CREATE TABLE CARS\_Countries (

Country\_Id INT UNIQUE, -- Maybe we'll need to make this Candidate key as the dataset has NULL

Country\_name VARCHAR (16),

Continent INT,

FOREIGN KEY (Continent) REFERENCES CARS\_Continents(Cont\_Id)

);

CREATE TABLE CARS\_Car\_makers (

Id INT PRIMARY KEY,

Maker VARCHAR (32),

Full\_Name VARCHAR (64),

Country INT,

FOREIGN KEY (Country) REFERENCES CARS\_Countries(Country\_Id)

);

CREATE TABLE CARS\_Model\_list (

Model\_Id INT PRIMARY KEY,

Maker INT,

Model VARCHAR (32) UNIQUE,

FOREIGN KEY (Maker) REFERENCES CARS\_Car\_makers(Id)

);

CREATE TABLE CARS\_names (

Make\_Id INT PRIMARY KEY,

Model VARCHAR (32),

Make\_description VARCHAR (128),

FOREIGN KEY (Model) REFERENCES CARS\_Model\_list(Model)

);

CREATE TABLE CARS\_data (

Id INT PRIMARY KEY,

MPG INT,

Cylinders INT,

Engine\_displacement INT,

HP INT,

Weight INT,

Acc DECIMAL(3,3),

Year INT,

FOREIGN KEY (Id) REFERENCES CARS\_names(Make\_Id)

);

INSERT INTO CARS\_Continents VALUES(99,NULL);

DROP TABLE CARS\_names

DROP TABLE CARS\_Model\_list

DROP TABLE CARS\_Car\_makers

DROP TABLE CARS\_Countries

DROP TABLE CARS\_Continents

SELECT \* FROM CARS\_data

SHOW TABLES

-- =============================BAKERY==============================

-- CUSTOMER , GOODS, RECIEPTS, ITEMS

CREATE TABLE BAKERY\_customers (

Id INT PRIMARY KEY,

Last\_name VARCHAR(32),

First\_name VARCHAR(32)

);

CREATE TABLE BAKERY\_goods (

Id VARCHAR (32) PRIMARY KEY,

Flavor VARCHAR(32),

Food VARCHAR(32),

Price DECIMAL(5,3)

);

CREATE TABLE BAKERY\_reciepts (

Reciept\_number INT PRIMARY KEY,

Date\_purchase DATE,

Customer\_Id INT,

FOREIGN KEY (Customer\_Id) REFERENCES BAKERY\_customers(Id)

);

CREATE TABLE BAKERY\_items (

Reciept INT,

Ordinal INT,

Item VARCHAR(32),

FOREIGN KEY (Reciept) REFERENCES BAKERY\_reciepts(Reciept\_number),

FOREIGN KEY (Item) REFERENCES BAKERY\_goods(Id)

);

-- ========================STUDENTS================================

CREATE TABLE STUDENTS\_list (

Last\_name VARCHAR(32),

First\_name VARCHAR(32),

Grade INT,

Classroom INT

);

CREATE TABLE STUDENTS\_teachers (

Last\_name VARCHAR(32),

First\_name VARCHAR(32),

Classroom INT

);

SHOW TABLES

-- ===========================KATZENJAMMER============================

-- BAND, ALBUM, SONGS, Tracklist, Vocals, Instruments, performance

CREATE TABLE KATZENJAMMER\_band (

Id INT PRIMARY KEY,

First\_name VARCHAR(32),

Last\_name VARCHAR(32)

);

CREATE TABLE KATZENJAMMER\_album (

Album\_Id INT PRIMARY KEY,

ALbum\_Title VARCHAR(128),

Year INT,

Label VARCHAR(128),

Rec\_Type VARCHAR(16)

);

CREATE TABLE KATZENJAMMER\_song (

Song\_Id INT PRIMARY KEY,

Song\_Title VARCHAR(128)

);

CREATE TABLE KATZENJAMMER\_tracklist (

Album\_Id INT,

Position INT,

Song\_Id INT,

FOREIGN KEY (Album\_Id) REFERENCES KATZENJAMMER\_album(Album\_Id),

FOREIGN KEY (Song\_Id) REFERENCES KATZENJAMMER\_song(Song\_Id)

);

CREATE TABLE KATZENJAMMER\_vocals (

Song\_Id INT,

Bandmate INT,

vocal\_type VARCHAR (16),

FOREIGN KEY (Bandmate) REFERENCES KATZENJAMMER\_band(Id),

FOREIGN KEY (Song\_Id) REFERENCES KATZENJAMMER\_song(Song\_Id)

);

CREATE TABLE KATZENJAMMER\_instruments (

Song\_Id INT,

Bandmate INT,

instrument VARCHAR (32),

FOREIGN KEY (Bandmate) REFERENCES KATZENJAMMER\_band(Id),

FOREIGN KEY (Song\_Id) REFERENCES KATZENJAMMER\_song(Song\_Id)

);

CREATE TABLE KATZENJAMMER\_performance (

Song\_Id INT,

Bandmate INT,

Stage\_position VARCHAR (16),

FOREIGN KEY (Bandmate) REFERENCES KATZENJAMMER\_band(Id),

FOREIGN KEY (Song\_Id) REFERENCES KATZENJAMMER\_song(Song\_Id)

);

-- ================================MARATHON========================

CREATE TABLE MARATHON (

Place INT PRIMARY KEY,

Race\_time TIME, -- Should we use TIME or VARCHAR since it's in quotes, but can't compare string later?

Pace TIME, -- Same, do we use TIME or VARCHAR?

Group\_place INT,

Par\_group VARCHAR (8),

Age INT,

Sex CHAR (1),

BIB\_number INT,

First\_name VARCHAR (64),

Last\_name VARCHAR (64),

Town VARCHAR (64),

State CHAR (2)

);

-- ============================AIRLINES=========================

CREATE TABLE AIRLINES\_airlines (

Id INT PRIMARY KEY,

Airline\_name VARCHAR(128),

Abbreviation VARCHAR(64),

Country VARCHAR(32)

);

CREATE TABLE AIRLINES\_airports (

City VARCHAR(128),

Airline\_code CHAR(3) UNIQUE,

Airport\_name VARCHAR(256),

Country VARCHAR(32),

Country\_abbrev VARCHAR(16)

);

CREATE TABLE AIRLINES\_flights (

Airline INT,

Flight\_no INT,

-- should I make Airline and flight# both UNIQUE? since unique for each airline,

-- may repeat for different airlines

Source\_airport CHAR(3),

Destination\_airport CHAR (3),

FOREIGN KEY (Airline) REFERENCES AIRLINES\_airlines(Id)

);

show tables

-- ==========================WINE=============================

CREATE TABLE WINE\_grapes (

Id INT PRIMARY KEY,

Grape VARCHAR(64) UNIQUE,

Color VARCHAR(16)

);

CREATE TABLE WINE\_appelations (

No INT PRIMARY KEY,

Appelations VARCHAR(128) UNIQUE,

County VARCHAR(32),

State VARCHAR(32) UNIQUE,

Area VARCHAR(64),

isAVA VARCHAR(4)

);

CREATE TABLE WINE\_wine (

No INT PRIMARY KEY,

Grape VARCHAR(128),

Winery VARCHAR(64),

Appelation VARCHAR(64),

State VARCHAR(64),

Name VARCHAR(256),

Year INT,

Price INT,

Score INT,

Cases INT,

Drink VARCHAR (16), -- It has int and strings so not sure what to use?

FOREIGN KEY (Grape) REFERENCES WINE\_grapes(Grape),

FOREIGN KEY (Appelation) REFERENCES WINE\_appelations(Appelations),

FOREIGN KEY (State) REFERENCES WINE\_appelations (State)

);

-- ==========================INN================================

CREATE TABLE INN\_rooms (

Room\_Id CHAR(3) PRIMARY KEY,

Room\_name VARCHAR (128),

Beds INT,

Bed\_type VARCHAR (8),

Max\_occupancy INT,

Base\_price INT,

Decor VARCHAR (16)

);

CREATE TABLE INN\_reservations (

Code INT PRIMARY KEY,

Room CHAR(3),

Check\_in VARCHAR(32),

Check\_out VARCHAR(32),

Rate DECIMAL(5,2),

Last\_name VARCHAR(64),

First\_name VARCHAR(64),

Adults INT,

Kids INT,

FOREIGN KEY (Room) REFERENCES INN\_rooms(Room\_Id)

);