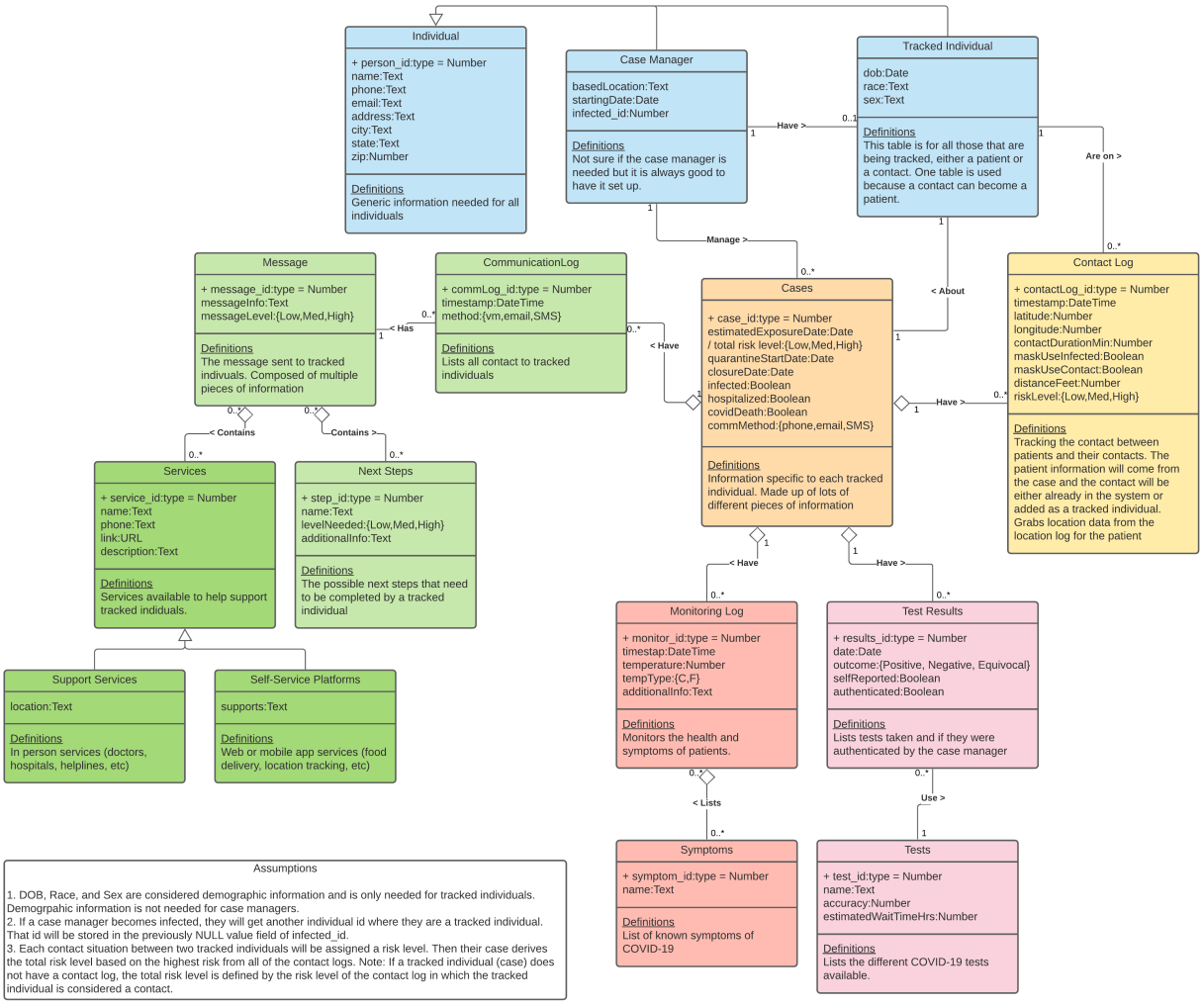
CS5200 Fall 2020: Practicum 1

Chandra Davis, Evan Douglass

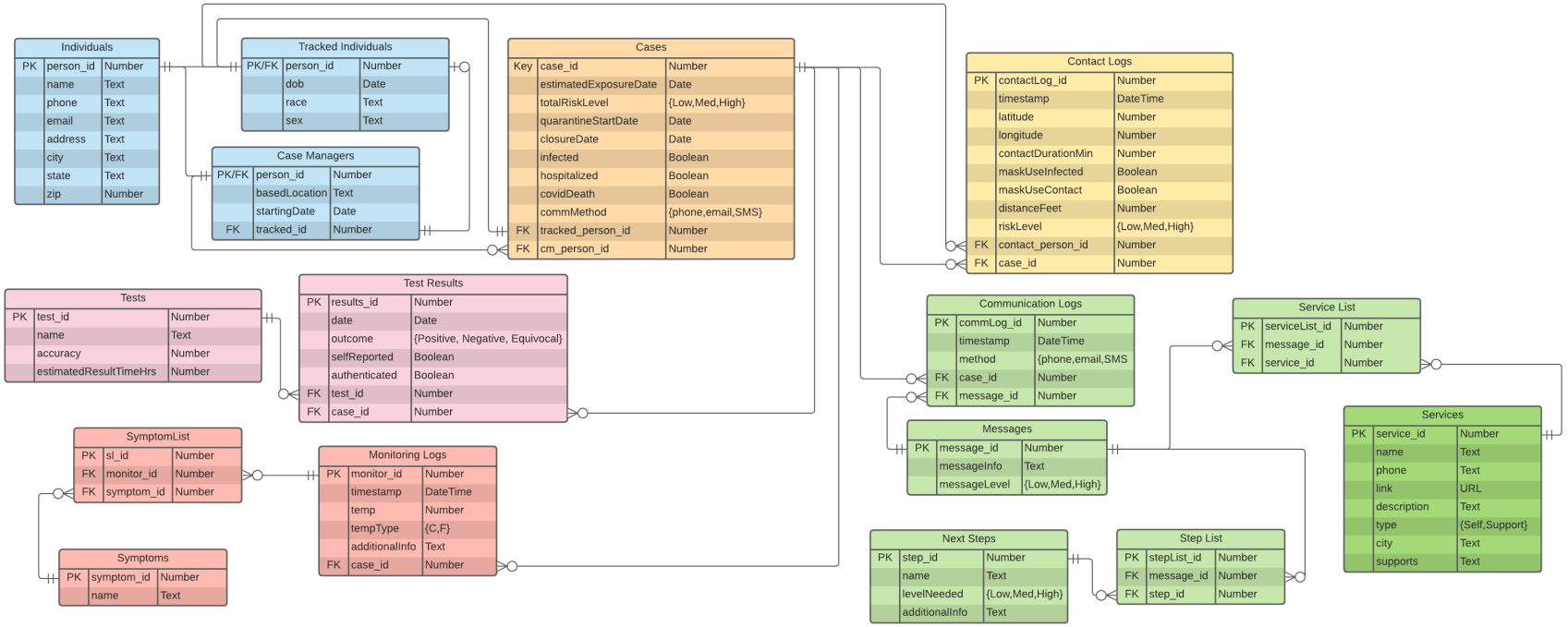
Conceptual Model: UML

View the conceptual model in Lucid Chart here: https://app.lucidchart.com/invitations/accept/5602312e-dfc3-4423-975c-47190ce6022e



Logical Model: ERD

View the logical model in Lucid Chart here: https://app.lucidchart.com/invitations/accept/7b497cbf-268d-4a03-b1a7-822b5a844fea


Individuals
PK person_id Number
name Text
phone Text
email Text
address Text
city Text
state Text
zip Number
Tracked Individuals
PK/FK person_id Number
dob Date
race Text
sex Text
Case Managers
PK/FK person_id Number
basedLocation Text
startingDate Date
FK tracked_id Number
Cases
Key case_id Number
estimatedExposureDate Date
totalRiskLevel {Low,Med,High}
quarantineStartDate Date
closureDate Date
infected Boolean
hospitalized Boolean
covidDeath Boolean
commMethod {phone,email,SMS}
FK tracked_person_id Number
FK cm_person_id Number
Contact Logs
PK contactLog_id Number
timestamp DateTime
latitude Number
longitude Number
contactDurationMin Number
maskUseInfected Boolean
maskUseContact Boolean
distanceFeet Number
riskLevel {Low,Med,High}
FK contact_person_id Number
FK case_id Number
Tests
PK test_id Number
name Text
accuracy Number
estimatedResultTimeHrs Number
Test Results
PK results_id Number
date Date
outcome {Positive, Negative, Equivocal}
selfReported Boolean
authenticated Boolean
FK test_id Number
FK case_id Number
Communication Logs
PK commLog_id Number
timestamp DateTime
method {phone,email,SMS
FK case_id Number
FK message_id Number
Service List
PK serviceList_id Number
FK message_id Number
FK service_id Number
Messages
PK message_id Number
messageInfo Text
messageLevel {Low,Med,High}
Services
PK service_id Number
name Text
phone Text
link URL
description Text
type {Self,Support}
city Text
supports Text
SymptomList
PK sl_id Number
FK monitor_id Number
FK symptom_id Number
Monitoring Logs
PK monitor_id Number
timestamp DateTime
temp Number
tempType {C,F}
additionalInfo Text
FK case_id Number
Symptoms
PK symptom_id Number
name Text
Next Steps
PK step_id Number
name Text
levelNeeded {Low,Med,High}
additionalInfo Text
Step List
PK stepList_id Number
FK message_id Number
FK step_id Number

Schema

View the schema in Google Docs here: https://docs.google.com/document/d/1o8pk51aed3BJSaBcwO2EMbT8I3ru_wpGTcTN4W-DbIM/edit?usp=sharing

Individuals(person_id: Number, name: Text, phone: Text, email: Text, address: Text, city: Text, state: Text, zip: Number)

TrackedIndividuals(*person_id: Number*, dob: Date, race: Text, sex: Text)

CaseManagers(*person_id: Number*, basedCity: Text, basedState: Text, startingDate: Date, *infected_id: Number*)

Tests(test_id: Number, name: Text, accuracy: Number, estimatedResultTimeHrs: Number)

TestResults(results_id: Number, date: Date, outcome: {Positive, Negative, Equivocal}, selfReported: Boolean, authenticated: Boolean, *test_id: Number*, *case_id: Number*)

MonitoringLogs(monitor_id: Number, timestamp: DateTime, temp: Number, tempType: {C, F}, additionalInfo: Text, *case_id: Number*)

SymptomList(sl_id: Number, *monitor_id: Number, symptom_id: Number*)

Symptoms(symptom_id: Number, name: Text)

Cases(case_id: Number, estimatedExposureDate: Date, totalRiskLevel: {Low, Med, High}, quarantineStartDate: Date, closureDate: Date, infected: Boolean, hospitalized: Boolean, covidDeath: Boolean, commMethod: {phone, email, SMS}, *cm_person_id: Number, infected_person_id: Number*)

ContactLogs(contactLog_id: Number, timestamp: DateTime, latitude: Number, longitude: Number, contactDurationMin: Number, maskUseInfected: Boolean, maskUseContact: Boolean, distanceFeet: Number, riskLevel: {Low, Med, High}, *contact_person_id: Number, case_id: Number*)

CommunicationLogs(commLog_id: Number, timestamp: DateTime, method: {phone, email, SMS}, *case_id: Number, message_id: Number*)

Messages(message_id: Number, messageLevel: {Low, Med, High}, messageInfo: Text)

StepList(stepList_id: Number, *message_id: Number, step_id: Number*)

NextSteps(step_id: Number, name: Text, levelNeeded: {Low, Med, High}, additionalInfo: Text)

ServiceList(serviceList_id: Number, *message_id: Number, service_id: Number*)

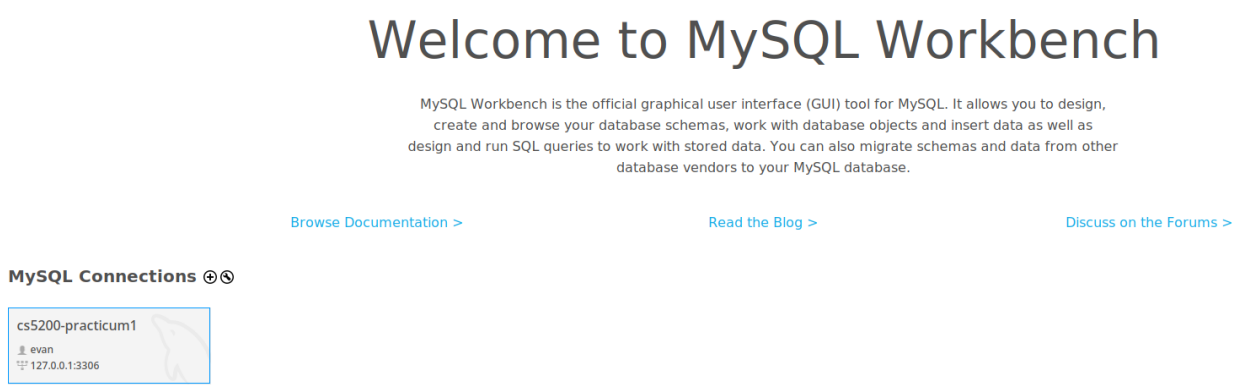
Services(service_id: Number, name: Text, phone: Text, link: Text, description: Text, type: {Self, Support}, city: Text, supports: Text)

Creating Database Tables

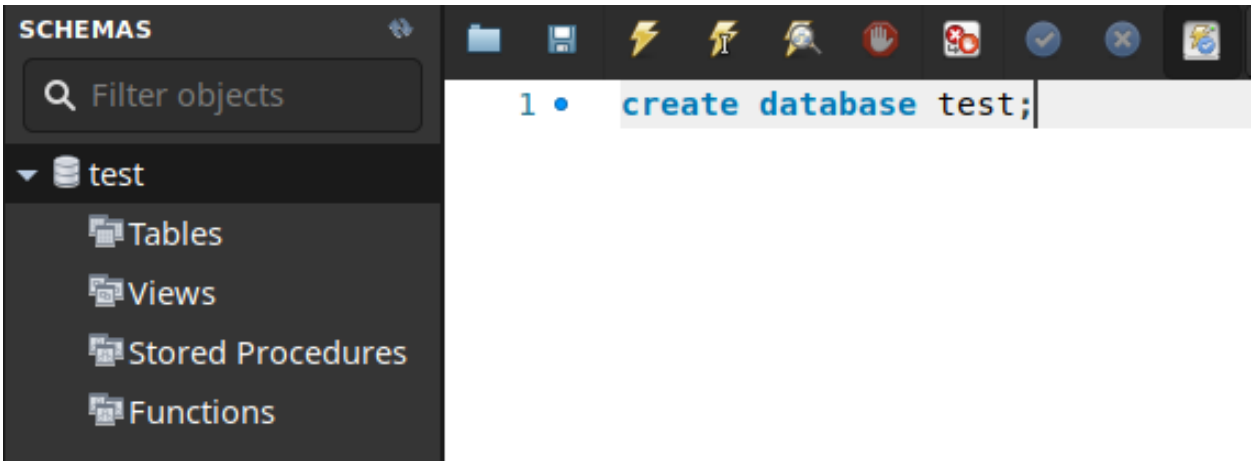
Should you wish to inspect the scripts that create the database and populate data, they can be found at https://github.com/eldss-classwork/databases-practicum1-scripts along with other Practicum 1 related files, including this one.

The following images will show a progression from an empty database through table creation in MySQL Workbench.

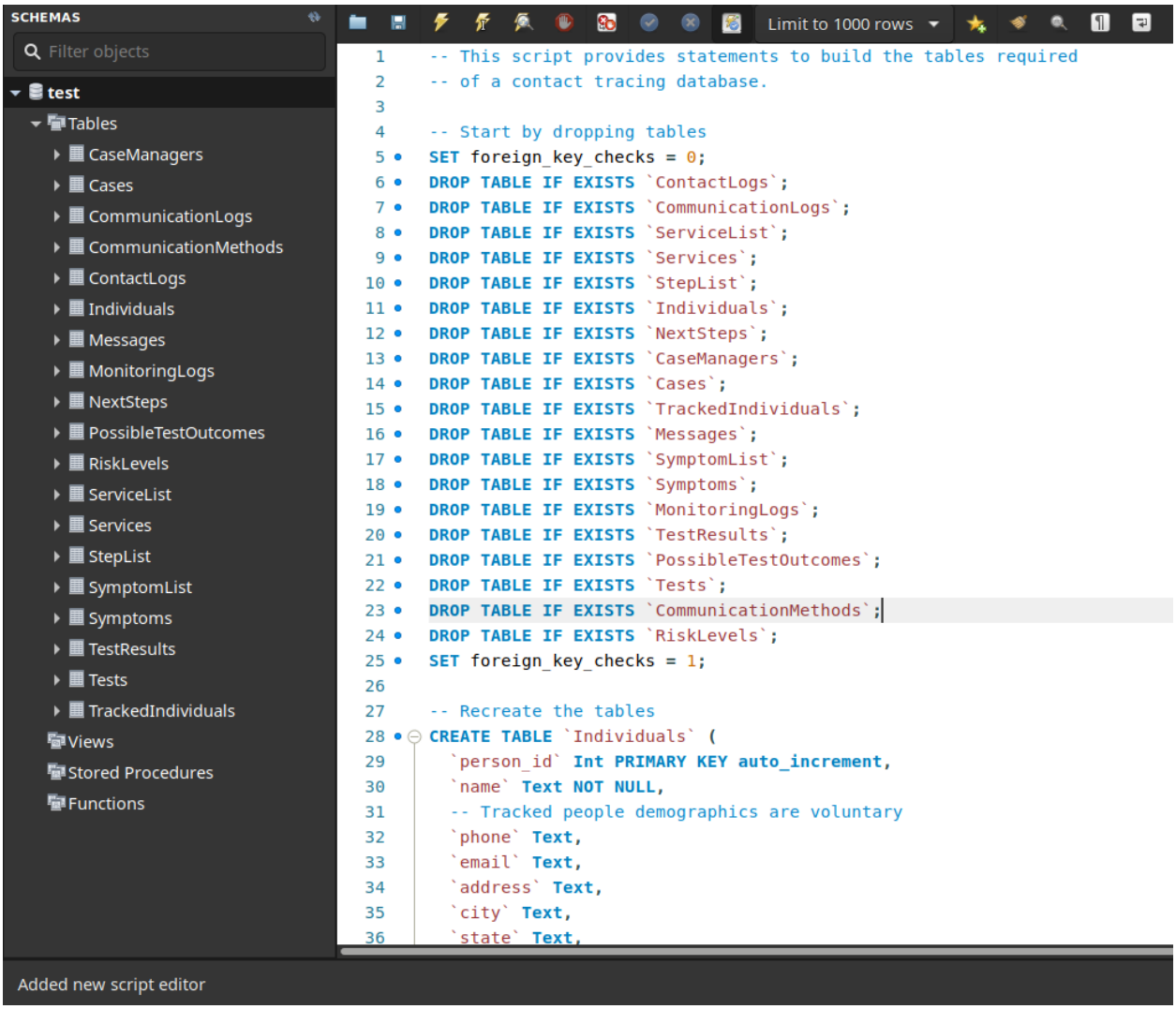
The MySQL Workbench start screen.



The newly created, empty `test` database.



The `test` database after table creation.



Populating The Database

Queries