SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

if not exists (select \* from sys.databases where name = 'project')

create database project

go

use project

go

-- DOWN

if exists (select \* from INFORMATION\_SCHEMA.TABLE\_CONSTRAINTS

where CONSTRAINT\_NAME = 'fk\_doctor\_animal\_animal\_type')

alter table doctor\_animal drop constraint fk\_doctor\_animal\_animal\_type

if exists (select \* from INFORMATION\_SCHEMA.TABLE\_CONSTRAINTS

where CONSTRAINT\_NAME = 'fk\_doctor\_animal\_doctor\_id')

alter table doctor\_animal drop constraint fk\_doctor\_animal\_doctor\_id

drop table if exists doctor\_animal

if exists (select \* from INFORMATION\_SCHEMA.TABLE\_CONSTRAINTS

where CONSTRAINT\_NAME = 'fk\_patients\_patient\_type')

alter table patients drop constraint fk\_patients\_patient\_type

drop table if exists animal\_types\_lookup

if exists (select \* from INFORMATION\_SCHEMA.TABLE\_CONSTRAINTS

where CONSTRAINT\_NAME = 'fk\_appointments\_appointment\_room')

alter table appointments drop constraint fk\_appointments\_appointment\_room

drop table if exists rooms\_lookup

if exists (select \* from INFORMATION\_SCHEMA.TABLE\_CONSTRAINTS

where CONSTRAINT\_NAME = 'fk\_bills\_bill\_for\_owner')

alter table bills drop constraint fk\_bills\_bill\_for\_owner

if exists (select \* from INFORMATION\_SCHEMA.TABLE\_CONSTRAINTS

where CONSTRAINT\_NAME = 'fk\_bills\_bill\_for\_appointment')

alter table bills drop constraint fk\_bills\_bill\_for\_appointment

drop table if exists bills

if exists (select \* from INFORMATION\_SCHEMA.TABLE\_CONSTRAINTS

where CONSTRAINT\_NAME = 'appointment\_assign\_doctor')

alter table appointments drop constraint appointment\_assign\_doctor

if exists (select \* from INFORMATION\_SCHEMA.TABLE\_CONSTRAINTS

where CONSTRAINT\_NAME = 'fk\_appointments\_appointment\_by\_owner')

alter table appointments drop constraint fk\_appointments\_appointment\_by\_owner

if exists (select \* from INFORMATION\_SCHEMA.TABLE\_CONSTRAINTS

where CONSTRAINT\_NAME = 'fk\_appointments\_appointment\_by\_patient')

alter table appointments drop constraint fk\_appointments\_appointment\_by\_patient

drop table if exists appointments

drop table if exists doctors

if exists (select \* from INFORMATION\_SCHEMA.TABLE\_CONSTRAINTS

where CONSTRAINT\_NAME = 'fk\_patients\_patient\_owned\_by')

alter table patients drop constraint fk\_patients\_patient\_owned\_by

drop table if exists patients

drop table if exists owners

GO

-- UP Metadata

create table owners (

owner\_id int identity not null,

owner\_firstname VARCHAR(50) not null,

owner\_lastname VARCHAR(50) not null,

owner\_email VARCHAR(50) not null,

owner\_phone\_number varchar(50) not null,

owner\_insurance VARCHAR(50) null --not sure

constraint pk\_owners\_owner\_id primary key (owner\_id),

constraint u\_owners\_owner\_email unique (owner\_email),

constraint u\_owners\_owner\_phone\_number unique (owner\_phone\_number)

)

create table patients (

patient\_id int identity not null,

patient\_name VARCHAR(50) not null,

patient\_date DATE not null,

patient\_species VARCHAR(50) not null,

patient\_type VARCHAR(50) not null,

patient\_owned\_by int not null,

constraint pk\_patients\_patient\_id primary key (patient\_id)

)

-- 1 foreign key

alter table patients --who owned patients, patients are many side.

add constraint fk\_patients\_patient\_owned\_by foreign key (patient\_owned\_by)

references owners(owner\_id)

-- end

create table doctors (

doctor\_id int identity not null,

doctor\_firstname VARCHAR(50) not null,

doctor\_lastname VARCHAR(50) not null,

doctor\_email VARCHAR(50) not null,

doctor\_phone\_number varchar(50) not null,

doctor\_schedule DATETIME not null,

constraint pk\_doctors\_doctor\_id primary key (doctor\_id),

constraint u\_doctors\_doctor\_email unique (doctor\_email),

constraint u\_doctors\_doctor\_phone\_number unique (doctor\_phone\_number)

)

create table appointments (

appointment\_id int identity not null,

appointment\_time time not null,

appointment\_date date not null,

appointment\_room varchar(30) not null,

appointment\_reason varchar(300) not null,

appointment\_diagnosis VARCHAR(300) NULL,

appointment\_treatment VARCHAR(300) NULL,

appointment\_by\_patient int not null,

appointment\_by\_owner int not null,

appointment\_assign\_doctor int not null,

constraint pk\_appointments\_appointment\_id primary key (appointment\_id)

)

-- 3 foreign key

alter table appointments --the appointment for patient, appointments are many side.

add constraint fk\_appointments\_appointment\_by\_patient foreign key (appointment\_by\_patient)

references patients(patient\_id)

alter table appointments --the appointment attended by owner, appointments are many side.

add constraint fk\_appointments\_appointment\_by\_owner foreign key (appointment\_by\_owner)

references owners(owner\_id)

alter table appointments --the appointment assigned to doctor, appointments are many side.

add constraint fk\_appointments\_appointment\_assign\_doctor foreign key (appointment\_assign\_doctor)

references doctors(doctor\_id)

-- end

create table bills (

bill\_id int identity not null,

bill\_treatment\_cost decimal(8,2) not null,

bill\_medication\_cost decimal(8,2) not null,

bill\_insurance\_covering decimal(8,2) NULL,

bill\_for\_appointment int not null,

bill\_for\_owner int not null,

bill\_status varchar(40) not null,

constraint pk\_bills\_bill\_id primary key (bill\_id)

)

-- 2 foreign key

alter table bills --bill for which appointment, this is one to one.

add constraint fk\_bills\_bill\_for\_appointment foreign key (bill\_for\_appointment)

references appointments(appointment\_id)

alter table bills --bill paid by owner, this is one to one.

add constraint fk\_bills\_bill\_for\_owner foreign key (bill\_for\_owner)

references owners(owner\_id)

-- end

--adding lookup tables/FK's

create table rooms\_lookup (

room\_name varchar(30)

constraint pk\_rooms\_lookup\_room\_name primary key (room\_name)

)

alter table appointments --set lookup table FK for room name

add constraint fk\_appointments\_appointment\_room foreign key (appointment\_room)

references rooms\_lookup(room\_name)

create table animal\_types\_lookup(

animal\_type varchar(50)

constraint pk\_animal\_types\_lookup\_animal\_type primary key (animal\_type)

)

alter table patients --set lookup table FK for room name

add constraint fk\_patients\_patient\_type foreign key (patient\_type)

references animal\_types\_lookup(animal\_type)

-- this naming isn't the best so we can change it if necessary!

create table doctor\_animal(

doctor\_id int not null,

animal\_type varchar(50),

constraint u\_doctor\_animal\_combination unique (doctor\_id, animal\_type)

)

alter table doctor\_animal --set lookup table FK for dr\_id

add constraint fk\_doctor\_animal\_doctor\_id foreign key (doctor\_id)

references doctors(doctor\_id)

alter table doctor\_animal --set lookup table FK for animal\_type

add constraint fk\_doctor\_animal\_animal\_type foreign key (animal\_type)

references animal\_types\_lookup(animal\_type)

GO

-- UP Data

-- add in room names:

-- I named the rooms after the different kingdoms....

insert into rooms\_lookup (room\_name)

values ('Plantae Room'), ('Animalia Room'), ('Protista Room'), ('Fungi Room')

-- add in animal types

insert into animal\_types\_lookup (animal\_type)

values ('Dog'), ('Cat'), ('Hamster'), ('Snake'), ('Lizard'), ('Bird')

-- Insert Data

delete from appointments

delete from bills

delete from doctors

delete from owners

delete from patients

insert into owners (owner\_firstname, owner\_lastname, owner\_email, owner\_phone\_number, owner\_insurance)

values

('Jianghui','Li','jli159@syr.edu','3159910192','Aetna'), --1

('Leah', 'Mitchell', 'lmmitch07@syr.edu', '123456789', 'BCBS'), --2

('Bill', 'Gates', 'boss@microsoft.com', '978456123', 'Aetna'), --3

('Steve', 'Jobs', 'steve@appl.com', '111222333', 'BCBS'), -- 4

('Alan', 'Turning', 'computerhuman@gmail.com', '312541785', 'Aetna'), --5

('Elon', 'Musk', 'elon@tesla.com', '999888777', 'Travelers'), -- 6

('Thomas', 'Bayes', 'stats@gmail.com', '123654789', 'Travelers'), --7

('Larry', 'Page', 'pagerank@gmail.com', '456127891', 'Liberty Mutual'), -- 8

('Tony', 'Stark', 'ironman@stark.com', '789456123', 'Liberty Mutual') -- 9

insert into patients (patient\_name, patient\_date, patient\_species, patient\_type, patient\_owned\_by)

VALUES

('Hamstrong','2021-1-1','Roborovski dwarf hamster','Hamster','1'),

('Sneaky','2021-1-5','Common garter snake','Snake','1'),

('Beaux','2014-3-24', 'Mixed Pitbull Husky', 'Dog', '2' ),

('Guiness', '2015-5-14', 'Munchkin', 'Cat', '2'),

('Oreo', '2020-3-9', 'Mix', 'Dog', '3'),

('Nilla', '2017-6-12', 'King Charles', 'Dog', '3'),

('iHamster', '2021-2-14', 'Winter White Dwarf', 'Hamster', '4'),

('iCat', '2018-6-12', 'British Shorthair', 'Cat', '4'),

('Cypher', '2016-9-22', 'Gecko', 'Lizard', '5'),

('Planet', '2019-4-21', 'Bulldog', 'Dog', '6'),

('Prob', '2018-8-4', 'Parrot', 'Bird', '7'),

('Searcher', '2017-8-9', 'Bengal', 'Cat', '8'),

('Avenger', '2016-7-16', 'German Sheppard', 'Dog', '9')

insert into doctors (doctor\_firstname, doctor\_lastname, doctor\_email, doctor\_phone\_number, doctor\_schedule)

VALUES

('Gillian','Orlando','gorlando@gmail.com','6461234321','2021-1-1 13:00:00'), --1

('Bob','Zimack','bobzimak@gmail.com','874512698','2021-1-1 13:00:00'), --2

('Taylor','Murphy','taylor.murphy@gmail.com','147845964','2021-1-1 13:00:00'), --3

('Steven','Ross','stevemross@gmail.com','741452365','2021-1-1 13:00:00') --4

insert into doctor\_animal(doctor\_id, animal\_type)

VALUES

('1', 'Hamster'),

('1', 'Snake'),

('1', 'Lizard'),

('2', 'Dog'),

('2', 'Cat'),

('2', 'Bird'),

('3', 'Bird'),

('3', 'Lizard'),

('3', 'Snake'),

('4', 'Dog'),

('4', 'Cat'),

('4', 'Hamster')

-- ('Dog'), ('Cat'), ('Hamster'), ('Snake'), ('Lizard'), ('Bird')

insert into appointments (appointment\_time, appointment\_date, appointment\_room, appointment\_reason, appointment\_diagnosis, appointment\_treatment, appointment\_by\_patient, appointment\_by\_owner, appointment\_assign\_doctor)

values

('13:00:00','2021-1-1','Plantae Room','My hamster will not stop eating meat.','', 'Forcing the hamster to eat vegetable','1','1','1'),

('13:00:00','2021-1-1','Animalia Room','General Checkup','', 'Undecided','3','2','4'),

('13:00:00','2021-1-1','Protista Room','Need a renewal of medication','', 'Undecided','4','2','2'),

('14:00:00','2021-1-3','Fungi Room','Oreo hasnt be acting normal','', 'Observation','5','3','4'),

('11:00:00','2021-1-2','Plantae Room','Nilla isnt eating anything','', 'Mixing pet food with human food','6','3','2'),

('10:00:00','2021-1-4','Animalia Room','General Checkup','', 'Undecided','7','4','1'),

('9:00:00','2021-1-5','Protista Room','My cat keeps getting overheated','', 'Undecided','8','4','2'),

('15:00:00','2021-1-6','Fungi Room','My lizard refuses to eat flys','', 'Try mealworms','9','5','1'),

('8:00:00','2021-1-3','Plantae Room','He keeps spinning in circles','', 'Distract him','10','6','2'),

('12:00:00','2021-1-7','Animalia Room','My bird wont stop chirping','', 'Undecided','11','7','2'),

('14:00:00','2021-1-5','Protista Room','My cat started barking','', 'Undecided','12','8','4'),

('12:00:00','2021-1-6','Fungi Room','I think he is seeing things','', 'Give him a physics test','13','9','4'),

('14:00:00','2021-1-2','Fungi Room','My snake wants to be a vegetarian','', 'Undecided','2','1','3')

insert into bills (bill\_treatment\_cost, bill\_medication\_cost, bill\_insurance\_covering, bill\_for\_appointment, bill\_for\_owner, bill\_status

)

VALUES

(1299.99,199.99,-600,'1','1','FULFILL'),

(1000.99,179.99,-700,'2','2','PROCESS PAYMENT'),

(1100.99,100.99,-500,'3','2','FULFILL'),

(900.99,199.99,-800,'4','3','PROCESS PAYMENT'),

(1050.99,60.99,-450,'5','3','UNFULFILL'),

(1110.99,190.99,-900,'6','4','UNFULFILL'),

(1000.99,99.99,-750,'7','4','FULFILL'),

(900.99,89.99,-800,'8','5','FULFILL'),

(600.99,129.99,-650,'9','6','PROCESS PAYMENT'),

(950.99,59.99,-950,'10','7','UNFULFILL'),

(1399.99,170.99,-1000,'11','8','FULFILL'),

(1000.99,75.99,-975,'12','9','UNFULFILL'),

(500.99,80.99,-400,'13','1','PROCESS PAYMENT')

-- Verify

select \* from rooms\_lookup

select \* from animal\_types\_lookup

select \* from appointments

select \* from bills

select \* from doctors

select \* from owners

select \* from patients