-- tables

-- Table: Booking

CREATE TABLE Booking (

idBooking int NOT NULL,

Guest\_idGuest int NOT NULL,

Room\_idRoom int NOT NULL,

Clerk\_idClerk int NOT NULL,

CONSTRAINT Booking\_pk PRIMARY KEY (idBooking)

);

INSERT INTO Booking(idBooking, Guest\_idGuest, Room\_idRoom, Clerk\_idClerk)

VALUES(0, 4, 4, 0);

INSERT INTO Booking(idBooking, Guest\_idGuest, Room\_idRoom, Clerk\_idClerk)

VALUES(1, 0, 1, 1);

INSERT INTO Booking(idBooking, Guest\_idGuest, Room\_idRoom, Clerk\_idClerk)

VALUES(2, 1, 2, 3);

INSERT INTO Booking(idBooking, Guest\_idGuest, Room\_idRoom, Clerk\_idClerk)

VALUES(3, 3, 3, 2);

INSERT INTO Booking(idBooking, Guest\_idGuest, Room\_idRoom, Clerk\_idClerk)

VALUES(4, 2, 0, 0);

-- Table: Caretaker

CREATE TABLE Caretaker (

idCaretaker int NOT NULL,

Surname nvarchar2(100) NOT NULL,

Department\_idDepartment int NOT NULL,

CONSTRAINT Caretaker\_pk PRIMARY KEY (idCaretaker)

);

INSERT INTO Caretaker(idCaretaker, Surname, Department\_idDepartment)

VALUES(0, 'BROWN', 10);

INSERT INTO Caretaker(idCaretaker, Surname, Department\_idDepartment)

VALUES(1, 'PINKMAN', 20);

INSERT INTO Caretaker(idCaretaker, Surname, Department\_idDepartment)

VALUES(2, 'PICASSO', 30);

INSERT INTO Caretaker(idCaretaker, Surname, Department\_idDepartment)

VALUES(3, 'PUTIN', 20);

INSERT INTO Caretaker(idCaretaker, Surname, Department\_idDepartment)

VALUES(4, 'LEE', 10);

-- Table: Clerk

CREATE TABLE Clerk (

idClerk int NOT NULL,

Surname nvarchar2(100) NOT NULL,

CONSTRAINT Clerk\_pk PRIMARY KEY (idClerk)

);

INSERT INTO Clerk(idClerk, Surname) VALUES(0, 'BLAKE');

INSERT INTO Clerk(idClerk, Surname) VALUES(1, 'SMITH');

INSERT INTO Clerk(idClerk, Surname) VALUES(2, 'GREEN');

INSERT INTO Clerk(idClerk, Surname) VALUES(3, 'SHELBY');

INSERT INTO Clerk(idClerk, Surname) VALUES(4, 'LEE');

-- Table: Department

CREATE TABLE Department (

idDepartment int NOT NULL,

Mngr\_idMngr int NOT NULL,

CONSTRAINT Department\_pk PRIMARY KEY (idDepartment)

);

INSERT INTO Department(idDepartment, Mngr\_idMngr) VALUES(10, 2);

INSERT INTO Department(idDepartment, Mngr\_idMngr) VALUES(20, 0);

INSERT INTO Department(idDepartment, Mngr\_idMngr) VALUES(30, 1);

-- Table: Guest

CREATE TABLE Guest (

idGuest int NOT NULL,

Name nvarchar2(100) NOT NULL,

Surname nvarchar2(100) NOT NULL,

Address nvarchar2(100) NOT NULL,

CONSTRAINT Guest\_pk PRIMARY KEY (idGuest)

);

INSERT INTO Guest (idGuest, Name, Surname, Address)

VALUES(0, 'WALTER', 'WHITE', 'ZLOTA ST. 3');

INSERT INTO Guest (idGuest, Name, Surname, Address)

VALUES(1, 'BRUCE', 'WAYNE', 'WEST ST. 41');

INSERT INTO Guest (idGuest, Name, Surname, Address)

VALUES(2, 'RICK', 'SANCHEZ', 'AZADLIQ ST. 26');

INSERT INTO Guest (idGuest, Name, Surname, Address)

VALUES(3, 'ABUSAT', 'AGHALI', 'M.ALIYEV ST. 130');

INSERT INTO Guest (idGuest, Name, Surname, Address)

VALUES(4, 'ELON', 'MUSK', 'EAST ST. 113');

-- Table: Manager

CREATE TABLE Mngr (

idMngr int NOT NULL,

Surname nvarchar2(100) NOT NULL,

CONSTRAINT Mngr\_pk PRIMARY KEY (idMngr)

);

INSERT INTO Mngr (idMngr, Surname) VALUES (0, 'JACKSON');

INSERT INTO Mngr (idMngr, Surname) VALUES (1, 'TRUMAN');

INSERT INTO Mngr (idMngr, Surname) VALUES (2, 'FLOYD');

-- Table: Room

CREATE TABLE Room (

idRoom int NOT NULL,

Beds int NOT NULL,

Price int NOT NULL,

Caretaker\_idCaretaker int NOT NULL,

CONSTRAINT Room\_pk PRIMARY KEY (idRoom)

);

INSERT INTO Room (idRoom, Beds, Price, Caretaker\_idCaretaker) VALUES (1, 3, 500, 4);

INSERT INTO Room (idRoom, Beds, Price, Caretaker\_idCaretaker) VALUES (2, 3, 500, 1);

INSERT INTO Room (idRoom, Beds, Price, Caretaker\_idCaretaker) VALUES (3, 2, 350, 2);

INSERT INTO Room (idRoom, Beds, Price, Caretaker\_idCaretaker) VALUES (4, 3, 500, 0);

INSERT INTO Room (idRoom, Beds, Price, Caretaker\_idCaretaker) VALUES (5, 1, 100, 4);

INSERT INTO Room (idRoom, Beds, Price, Caretaker\_idCaretaker) VALUES (6, 1, 90, 3);

INSERT INTO Room (idRoom, Beds, Price, Caretaker\_idCaretaker) VALUES (7, 2, 300, 1);

-- foreign keys

-- Reference: Booking\_Clerk (table: Booking)

ALTER TABLE Booking ADD CONSTRAINT Booking\_Clerk

FOREIGN KEY (Clerk\_idClerk)

REFERENCES Clerk (idClerk);

-- Reference: Booking\_Guest (table: Booking)

ALTER TABLE Booking ADD CONSTRAINT Booking\_Guest

FOREIGN KEY (Guest\_idGuest)

REFERENCES Guest (idGuest);

-- Reference: Booking\_Room (table: Booking)

ALTER TABLE Booking ADD CONSTRAINT Booking\_Room

FOREIGN KEY (Room\_idRoom)

REFERENCES Room (idRoom);

-- Reference: Department\_Manager (table: Department)

ALTER TABLE Department ADD CONSTRAINT Department\_Mngr

FOREIGN KEY (Mngr\_idMngr)

REFERENCES Mngr (idMngr);

-- Reference: Room\_Caretaker (table: Room)

ALTER TABLE Room ADD CONSTRAINT Room\_Caretaker

FOREIGN KEY (Caretaker\_idCaretaker)

REFERENCES Caretaker (idCaretaker);

-- Reference: Staff\_Department (table: Caretaker)

ALTER TABLE Caretaker ADD CONSTRAINT Staff\_Department

FOREIGN KEY (Department\_idDepartment)

REFERENCES Department (idDepartment);

-- Show number of caretakers for each department

Select COUNT(c.idcaretaker), d.iddepartment

From Caretaker c, Department d

where c.department\_iddepartment = d.iddepartment

group by iddepartment;

-- For each caretaker show number of rooms

Select c.idcaretaker, COUNT(r.idRoom)

From Caretaker c, Room r

Where c.idcaretaker = r.caretaker\_idcaretaker

group by idcaretaker;

-- Select bookings managed by clerk whose id is not 0

Select idbooking

from Booking

Where clerk\_idclerk != 0;

-- Show department which its manager named TRUMAN

Select d.iddepartment, m.surname

from Department d, Mngr m

Where d.mngr\_idmngr = m.idmngr

AND surname = 'TRUMAN';

-- Are there any clerk and caretaker who has the same surname?

Select \*

from Clerk

Where surname IN(Select surname from Caretaker);

create or replace trigger TR1 after insert on Guest

for each row

begin

DBMS\_output.put\_line('Welcome, new guest');

end;

INSERT INTO Guest (idGuest, Name, Surname, Address)

VALUES(5, 'MICHAEL', 'JORDAN', 'Salt Lake City');

create or replace trigger TR2 before delete on Booking

for each row

begin

DBMS\_output.put\_line('Row deleted');

end;

INSERT INTO Booking(idBooking, Guest\_idGuest, Room\_idRoom, Clerk\_idClerk)

VALUES(45, 2, 0, 0);

delete from booking

where idbooking = 45;

create table Event(What nvarchar2(100), day date);

create or replace trigger TRG3

after insert on booking

for each row

begin

insert into event values('New Booking', sysdate);

end;

INSERT INTO Booking(idBooking, Guest\_idGuest, Room\_idRoom, Clerk\_idClerk)

VALUES(45, 2, 0, 0);

create or replace trigger TRG4

after delete on booking

for each row

begin

delete from guest where guest.idguest = :old.guest\_idguest;

end;

delete from booking

where idbooking = 0;

create or replace trigger TRG5 before insert on guest

for each row

Declare x integer;

begin

SELECT (COUNT(idguest)+1) INTO X FROM guest;

DBMS\_output.put\_line('Number of guests '||x||' ');

end;

INSERT INTO Guest (idGuest, Name, Surname, Address)

VALUES(6, 'MIKE', 'MIKE', 'EAST ST. 11');

create or replace trigger TRG6

before update on Room

for each row

begin

end;

create or replace trigger TRG7 after delete on department

for each row

begin

delete

from Mngr

where mngr.idmngr = :old.mngr\_idmngr;

end;

delete from Booking

where idbooking = 3;

delete from Room

where idroom = 3;

delete from Caretaker

where idcaretaker = 2;

delete from department

where iddepartment = 30;

select guest\_idguest

from Booking

having COUNT(idbooking) = (Select MAX(COUNT(idbooking)) from Booking)

group by guest\_idguest;

-- End of file.