--Created By: Calvin Zheng and Vamsi Garghi

Create table Driving\_School (

drivingschool\_id int NOT NULL,

city varchar2(64) NOT NULL,

state varchar2(64) NOT NULL,

manager\_name varchar2(64),

PRIMARY KEY (drivingschool\_id)

);

Insert into Driving\_School Values(1, 'Duluth', 'MN', 'William');

Insert into Driving\_School Values(2, 'Duluth', 'GA', 'John');

Insert into Driving\_School Values(3, 'Dallas', 'TX', 'Keith');

Insert into Driving\_School Values(4, 'Shakopee', 'KS', 'Roy');

Insert into Driving\_School Values(5, 'Bloomington', 'CA', 'Vamsi');

Insert into Driving\_School Values(6, 'Eden Prairie', 'AL', 'Jack');

Insert into Driving\_School Values(7, 'Egaan', 'MI', 'Tahir');

Insert into Driving\_School Values(8, 'Apple Valley', 'CT', 'Grant');

Insert into Driving\_School Values(9, 'Blain', 'DE', 'Danny');

Insert into Driving\_School Values(10, 'LakeVille', 'FL', 'Teddy');

Insert into Driving\_School Values(11, 'Mankato', 'GA', 'Taylor');

Insert into Driving\_School Values(12, 'St.Cloud', 'NC', 'Dinesh');

Insert into Driving\_School Values(13, 'Edina', 'ND', 'Ben');

Insert into Driving\_School Values(14, 'BurnsVille', 'TX', 'Daniella');

Insert into Driving\_School Values(15, 'MapleGrove', 'UT', 'Jess');

Create Table Instructor (

instructor\_ssn int NOT NULL,

instructor\_name varchar2(64),

city varchar2(64),

state varchar2(64),

license\_number int,

license\_state varchar2(64),

Expeired\_license date,

drivingschool\_id int,

PRIMARY KEY (instructor\_ssn),

FOREIGN KEY(drivingschool\_id) REFERENCES Driving\_School(drivingschool\_id) ON DELETE CASCADE,

CHECK(license\_state = state)

);

Insert into Instructor Values(711, 'William', 'Duluth', 'MN', 1234, 'MN',TO\_DATE('01/01/2019', 'MM/DD/YYYY'),1);

Insert into Instructor Values(911, 'Jasmine', 'Duluth', 'MN', 0987, 'MN',TO\_DATE('02/02/2019', 'MM/DD/YYYY'),1);

Insert into Instructor Values(696, 'Lawrence', 'Duluth', 'MN', 4450, 'MN',TO\_DATE('01/01/2019', 'MM/DD/YYYY'),1);

Insert into Instructor Values(420, 'Jack', 'Duluth', 'MN', 7790, 'MN',TO\_DATE('09/06/2018', 'MM/DD/YYYY'),1);

Insert into Instructor Values(342, 'Mike', 'Duluth', 'MN', 1800, 'MN',TO\_DATE('01/01/2019', 'MM/DD/YYYY'),1);

Insert into Instructor Values(560, 'Ben', 'Duluth', 'MN', 1590, 'MN',TO\_DATE('07/20/2018', 'MM/DD/YYYY'),1);

Insert into Instructor Values(889, 'Madelyn', 'Duluth', 'MN', 1670, 'MN',TO\_DATE('05/20/2017', 'MM/DD/YYYY'),1);

Insert into Instructor Values(912, 'Alex', 'Duluth', 'GA', 1499, 'GA',TO\_DATE('02/02/2017', 'MM/DD/YYYY'),2);

Insert into Instructor Values(480, 'Kim', 'Duluth', 'GA', 0111, 'GA',TO\_DATE('05/03/2019', 'MM/DD/YYYY'),2);

Insert into Instructor Values(160, 'Johnson', 'Duluth', 'GA', 1919, 'GA',TO\_DATE('05/11/2018', 'MM/DD/YYYY'),2);

Insert into Instructor Values(680, 'Melvin', 'Duluth', 'GA', 1818, 'GA',TO\_DATE('10/10/2015', 'MM/DD/YYYY'),2);

Insert into Instructor Values(777, 'Zack', 'Duluth', 'GA', 1717, 'GA',TO\_DATE('01/26/2018', 'MM/DD/YYYY'),2);

Insert into Instructor Values(888, 'Zeke', 'Duluth', 'GA', 1717, 'GA',TO\_DATE('01/29/2017', 'MM/DD/YYYY'),2);

Create Table Phone (

phone\_number int NOT NULL,

instructor\_ssn int NOT NULL,

PRIMARY KEY (phone\_number, instructor\_ssn),

FOREIGN KEY(instructor\_ssn) REFERENCES Instructor(instructor\_ssn) ON DELETE CASCADE

);

Insert into Phone Values (123456789, 711);

Insert into Phone Values (278390182, 911);

Insert into Phone Values (578902837, 696);

Insert into Phone Values (201928375, 420);

Insert into Phone Values (901827364, 342);

Insert into Phone Values (887162549, 560);

Insert into Phone Values (904328756, 889);

Insert into Phone Values (189203847, 912);

Insert into Phone Values (348920176, 480);

Insert into Phone Values (198726453, 160);

Insert into Phone Values (982345674, 680);

Insert into Phone Values (463789820, 777);

Insert into Phone Values (478390123, 888);

Create Table Student (

Stein\_ID varchar2(64) NOT NULL,

Student\_name varchar2(64),

Birth\_Day date ,

phone int,

city varchar2(64),

state varchar2(64),

drivingschool\_id int NOT NULL,

instructor\_ssn int NOT NULL,

hasGuardian varchar2(64),

PRIMARY KEY (Stein\_ID),

FOREIGN KEY(instructor\_ssn) REFERENCES Instructor(instructor\_ssn) ON DELETE CASCADE,

FOREIGN KEY(drivingschool\_id) REFERENCES Driving\_school(drivingschool\_id) ON DELETE CASCADE,

CHECK(hasGuardian = 'Yes' OR hasGuardian = 'No')

);

Insert into Student Values('S711','Kevin', TO\_DATE('03/03/2003', 'MM/DD/YYYY'), 882256743, 'Duluth','MN', 1, 711, 'Yes');

Insert into Student Values('S911','Calvin', TO\_DATE('04/04/2003', 'MM/DD/YYYY'), 220192857, 'Duluth','MN', 1, 711, 'Yes');

Insert into Student Values('S140','Sai', TO\_DATE('02/28/2003', 'MM/DD/YYYY'), 901748562, 'Duluth','MN',1, 911, 'Yes');

Insert into Student Values('S690','Jose', TO\_DATE('05/18/2002', 'MM/DD/YYYY'), 808025483, 'Duluth','MN',1, 711, 'Yes');

Insert into Student Values('S555','Ryan', TO\_DATE('05/22/2004', 'MM/DD/YYYY'), 440821954, 'Duluth','MN',1, 711, 'Yes');

Insert into Student Values('S890','Jack', TO\_DATE('03/03/2004', 'MM/DD/YYYY'), 889956743, 'Duluth','GA', 2, 911, 'Yes');

Insert into Student Values('S190','Ben', TO\_DATE('04/10/2004', 'MM/DD/YYYY'), 220199257, 'Duluth','GA', 2, 911, 'Yes');

Insert into Student Values('S081','John', TO\_DATE('02/20/2003', 'MM/DD/YYYY'), 901748062, 'Duluth','GA',2, 911, 'Yes');

Insert into Student Values('S881','Walter', TO\_DATE('05/18/2003', 'MM/DD/YYYY'), 801105483, 'Duluth','GA',2, 911, 'Yes');

Insert into Student Values('S678','Heather', TO\_DATE('01/02/1998', 'MM/DD/YYYY'), 440991654, 'Duluth','GA',2, 911, 'No');

Create Table Vehicle(

vin varchar2(64) NOT NULL,

make varchar2(64),

vehicle\_model varchar2(64),

color varchar2(64),

license\_plate varchar2(64),

date\_lastrepair date,

drivingschool\_id int NOT NULL,

PRIMARY KEY (vin),

FOREIGN KEY(drivingschool\_id) REFERENCES Driving\_School(drivingschool\_id) ON DELETE CASCADE

);

INSERT INTO Vehicle VALUES('S543XD', 'Toyota', 'Corolla', 'Brown', 'XD69X', TO\_DATE('02/20/2016', 'MM/DD/YYYY'), 1);

INSERT INTO Vehicle VALUES('F94kIS', 'Mazda', '3', 'Black', 'FlP10', TO\_DATE('06/10/2008', 'MM/DD/YYYY'), 1);

INSERT INTO Vehicle VALUES('FQ904R', 'Honda', 'CRV', 'Grey', 'LO19D', TO\_DATE('10/20/2009', 'MM/DD/YYYY'), 1);

INSERT INTO Vehicle VALUES('TT45UI', 'BMW', '3 Series', 'Black', 'FP10S', TO\_DATE('08/15/2019', 'MM/DD/YYYY'), 1);

INSERT INTO Vehicle VALUES('XAQ32I', 'Benz', 'C-Class', 'Red', 'SD7SD', TO\_DATE('11/25/2018', 'MM/DD/YYYY'), 1);

INSERT INTO Vehicle VALUES('S543XX', 'Honda', 'Accord', 'Brown', 'HS98Y', TO\_DATE('02/22/2016', 'MM/DD/YYYY'), 1);

INSERT INTO Vehicle VALUES('F94kIH', 'Toyota', '3', 'Black', 'LP00P', TO\_DATE('06/10/2009', 'MM/DD/YYYY'), 1);

INSERT INTO Vehicle VALUES('FQ904L', 'Subaru', 'Series 3', 'GREY', 'XH092', TO\_DATE('10/20/2010', 'MM/DD/YYYY'), 1);

INSERT INTO Vehicle VALUES('TT45UU', 'Ferrari', 'Aventador', 'Black', 'HH091', TO\_DATE('08/15/2017', 'MM/DD/YYYY'), 2);

INSERT INTO Vehicle VALUES('XAQ32Y', 'Ford', '', 'Fiesta', '09HS6', TO\_DATE('11/25/2018', 'MM/DD/YYYY'), 2);

Create Table Lesson (

lesson\_id int NOT NULL,

date\_lesson date,

stein\_id varchar2(64) NOT NULL,

instructor\_ssn int NOT NULL,

PRIMARY KEY (lesson\_id),

FOREIGN KEY(instructor\_ssn) REFERENCES Instructor(instructor\_ssn) ON DELETE CASCADE,

FOREIGN KEY(stein\_id) REFERENCES Student(Stein\_ID) ON DELETE CASCADE

);

Insert into Lesson Values(10001, TO\_DATE('02/15/2019', 'MM/DD/YYYY'), 'S711', 711);

Insert into Lesson Values(10002, TO\_DATE('03/15/2019', 'MM/DD/YYYY'), 'S711', 711);

Insert into Lesson Values(10003, TO\_DATE('04/15/2019', 'MM/DD/YYYY'), 'S711', 711);

Insert into Lesson Values(10004, TO\_DATE('05/15/2019', 'MM/DD/YYYY'), 'S711', 711);

Insert into Lesson Values(10005, TO\_DATE('06/15/2019', 'MM/DD/YYYY'), 'S711', 711);

Insert into Lesson Values(10021, TO\_DATE('06/16/2018', 'MM/DD/YYYY'), 'S911', 711);

Insert into Lesson Values(10006, TO\_DATE('07/16/2018', 'MM/DD/YYYY'), 'S911', 711);

Insert into Lesson Values(10022, TO\_DATE('08/16/2018', 'MM/DD/YYYY'), 'S911', 711);

Insert into Lesson Values(10007, TO\_DATE('09/16/2018', 'MM/DD/YYYY'), 'S911', 711);

Insert into Lesson Values(10008, TO\_DATE('10/16/2018', 'MM/DD/YYYY'), 'S911', 711);

Insert into Lesson Values(10009, TO\_DATE('09/15/2019', 'MM/DD/YYYY'), 'S140', 911);

Insert into Lesson Values(10010, TO\_DATE('09/15/2019', 'MM/DD/YYYY'), 'S140', 911);

Insert into Lesson Values(10011, TO\_DATE('09/15/2019', 'MM/DD/YYYY'), 'S140', 911);

Insert into Lesson Values(10012, TO\_DATE('09/15/2019', 'MM/DD/YYYY'), 'S140', 911);

Insert into Lesson Values(10013, TO\_DATE('09/15/2019', 'MM/DD/YYYY'), 'S140', 911);

Insert into Lesson Values(10014, TO\_DATE('10/15/2019', 'MM/DD/YYYY'), 'S690', 711);

Insert into Lesson Values(10015, TO\_DATE('11/15/2019', 'MM/DD/YYYY'), 'S555', 711);

Insert into Lesson Values(10016, TO\_DATE('08/15/2019', 'MM/DD/YYYY'), 'S890', 911);

Insert into Lesson Values(10017, TO\_DATE('09/21/2019', 'MM/DD/YYYY'), 'S190', 911);

Insert into Lesson Values(10018, TO\_DATE('10/22/2019', 'MM/DD/YYYY'), 'S678', 911);

Insert into Lesson Values(10019, TO\_DATE('11/23/2019', 'MM/DD/YYYY'), 'S690', 711);

Insert into Lesson Values(10020, TO\_DATE('12/20/2019', 'MM/DD/YYYY'), 'S555', 711);

Insert into Lesson Values(10023, TO\_DATE('12/20/2019', 'MM/DD/YYYY'), 'S555', 711);

Insert into Lesson Values(10024, TO\_DATE('12/20/2019', 'MM/DD/YYYY'), 'S555', 711);

Insert into Lesson Values(10025, TO\_DATE('12/20/2019', 'MM/DD/YYYY'), 'S555', 711);

Insert into Lesson Values(10026, TO\_DATE('12/20/2019', 'MM/DD/YYYY'), 'S678', 911);

Insert into Lesson Values(10027, TO\_DATE('12/22/2019', 'MM/DD/YYYY'), 'S678', 911);

Insert into Lesson Values(10028, TO\_DATE('12/24/2019', 'MM/DD/YYYY'), 'S678', 911);

Insert into Lesson Values(10029, TO\_DATE('12/25/2019', 'MM/DD/YYYY'), 'S678', 911);

Insert into Lesson Values(10030, TO\_DATE('10/17/2019', 'MM/DD/YYYY'), 'S690', 711);

Insert into Lesson Values(10031, TO\_DATE('10/19/2019', 'MM/DD/YYYY'), 'S690', 711);

Insert into Lesson Values(10032, TO\_DATE('10/20/2019', 'MM/DD/YYYY'), 'S690', 711);

Insert into Lesson Values(10033, TO\_DATE('10/28/2019', 'MM/DD/YYYY'), 'S690', 711);

Create Table Guardian(

Stein\_id varchar2(64) NOT NULL,

guardian\_name varchar2(64),

guardian\_city varchar2(64),

guardian\_state varchar2(64),

guardian\_phone int,

PRIMARY KEY (Stein\_id),

FOREIGN KEY (Stein\_id) REFERENCES Student(Stein\_ID) ON DELETE CASCADE

);

INSERT INTO Guardian VALUES ('S711','Karen', 'Duluth', 'MN', 657392019);

INSERT INTO Guardian VALUES ('S911', 'Mike', 'Duluth', 'MN', 8910567);

INSERT INTO Guardian VALUES ('S140', 'Jack', 'Duluth', 'MN', 182937468);

INSERT INTO Guardian VALUES ('S690', 'Ben' , 'Duluth', 'Mn', 109283746);

INSERT INTO Guardian VALUES ('S555', 'William,', 'Duluth', 'MN', 201928376);

INSERT INTO Guardian VALUES ('S890', 'Nick', 'Duluth', 'GA', 1099872635);

INSERT INTO Guardian VALUES ('S190','Schmidt', 'Duluth', 'GA', 65289182);

INSERT INTO Guardian VALUES ('S081', 'Winston', 'Duluth', 'GA', 672839012);

INSERT INTO Guardian VALUES ('S678', 'Terry', 'Duluth', 'GA', 650192836);

Create Table Road\_Test(

RoadTest\_ID int NOT NULL,

Test\_Date date,

Score varchar2(64),

StateExaminer\_ID int,

StateExaminer\_name varchar2(64),

Stein\_id varchar2(64) NOT NULL,

vin varchar2(64) NOT NULL,

PRIMARY KEY (RoadTest\_ID),

FOREIGN KEY(Stein\_id) REFERENCES Student(Stein\_ID) ON DELETE CASCADE,

FOREIGN KEY(vin) REFERENCES Vehicle(vin) ON DELETE CASCADE,

CHECK (Score = 'Pass' OR Score = 'Fail')

);

INSERT INTO Road\_Test VALUES (901, TO\_DATE('07/13/2018', 'MM/DD/YYYY'), 'Fail', 9821, 'Charlie', 'S711', 'S543XD');

INSERT INTO Road\_Test VALUES (902, TO\_DATE('07/05/2018', 'MM/DD/YYYY'), 'Fail', 9821, 'Charlie', 'S711', 'S543XD');

INSERT INTO Road\_Test VALUES (903, TO\_DATE('08/10/2019', 'MM/DD/YYYY'), 'Pass', 9821, 'Charlie', 'S711', 'S543XD');

INSERT INTO Road\_Test VALUES (882, TO\_DATE('08/22/2018', 'MM/DD/YYYY'), 'Fail', 2902, 'Walter', 'S911', 'F94kIS');

INSERT INTO Road\_Test VALUES (883, TO\_DATE('08/24/2018', 'MM/DD/YYYY'), 'Fail', 2905, 'Walter', 'S911', 'F94kIS');

INSERT INTO Road\_Test VALUES (884, TO\_DATE('09/10/2019', 'MM/DD/YYYY'), 'Pass', 2906, 'Walter', 'S911', 'F94kIS');

INSERT INTO Road\_Test VALUES (885, TO\_DATE('07/12/2019', 'MM/DD/YYYY'), 'Pass', 9821, 'Charlie', 'S140', 'S543XD');

INSERT INTO Road\_Test VALUES (886, TO\_DATE('08/12/2019', 'MM/DD/YYYY'), 'Fail', 9821, 'Charlie', 'S555', 'S543XD');

INSERT INTO Road\_Test VALUES (887, TO\_DATE('09/12/2019', 'MM/DD/YYYY'), 'Fail', 9821, 'Charlie', 'S678', 'S543XD');

INSERT INTO Road\_Test VALUES (888, TO\_DATE('12/12/2019', 'MM/DD/YYYY'), 'Fail', 9821, 'Charlie', 'S690', 'S543XD');

Create Table Has\_Vehicle(

lessons\_id int NOT NULL,

vin varchar2(64) NOT NULL,

PRIMARY KEY(lessons\_id, vin),

FOREIGN KEY(lessons\_id) REFERENCES Lesson(lesson\_id) ON DELETE CASCADE,

FOREIGN KEY(vin) REFERENCES Vehicle(vin) ON DELETE CASCADE

);

INSERT INTO Has\_Vehicle VALUES (10001,'S543XD');

INSERT INTO Has\_Vehicle VALUES (10002,'F94kIS');

INSERT INTO Has\_Vehicle VALUES (10003,'FQ904R');

INSERT INTO Has\_Vehicle VALUES (10004,'TT45UI');

INSERT INTO Has\_Vehicle VALUES (10005,'XAQ32I');

INSERT INTO Has\_Vehicle VALUES (10006,'S543XX');

INSERT INTO Has\_Vehicle VALUES (10007,'F94kIH');

INSERT INTO Has\_Vehicle VALUES (10008,'XAQ32I');

INSERT INTO Has\_Vehicle VALUES (10009,'S543XD');

INSERT INTO Has\_Vehicle VALUES (10010,'FQ904R');

INSERT INTO Has\_Vehicle VALUES (10011,'TT45UI');

INSERT INTO Has\_Vehicle VALUES (10012,'F94kIH');

INSERT INTO Has\_Vehicle VALUES (10013,'FQ904L');

INSERT INTO Has\_Vehicle VALUES (10014,'FQ904R');

INSERT INTO Has\_Vehicle VALUES (10015,'TT45UI');

INSERT INTO Has\_Vehicle VALUES (10016,'FQ904R');

INSERT INTO Has\_Vehicle VALUES (10017,'XAQ32I');

INSERT INTO Has\_Vehicle VALUES (10018,'TT45UI');

INSERT INTO Has\_Vehicle VALUES (10019,'FQ904R');

INSERT INTO Has\_Vehicle VALUES (10020,'FQ904R');

INSERT INTO Has\_Vehicle VALUES (10021,'FQ904R');

INSERT INTO Has\_Vehicle VALUES (10022,'XAQ32I');

INSERT INTO Has\_Vehicle VALUES (10023,'FQ904R');

INSERT INTO Has\_Vehicle VALUES (10024,'XAQ32I');

INSERT INTO Has\_Vehicle VALUES (10025,'FQ904R');

INSERT INTO Has\_Vehicle VALUES (10026,'S543XD');

INSERT INTO Has\_Vehicle VALUES (10027,'S543XD');

INSERT INTO Has\_Vehicle VALUES (10028,'S543XD');

INSERT INTO Has\_Vehicle VALUES (10029,'S543XD');

INSERT INTO Has\_Vehicle VALUES (10030,'XAQ32I');

INSERT INTO Has\_Vehicle VALUES (10031,'XAQ32I');

INSERT INTO Has\_Vehicle VALUES (10032,'XAQ32I');

INSERT INTO Has\_Vehicle VALUES (10033,'XAQ32I');

Create Table Has\_Instructor(

Instructor\_ssn int NOT NULL,

Stein\_id varchar2(64) NOT NULL,

PRIMARY KEY (Instructor\_ssn, Stein\_id),

FOREIGN KEY(instructor\_ssn) REFERENCES Instructor(instructor\_ssn) ON DELETE CASCADE,

FOREIGN KEY(Stein\_id) REFERENCES Student(Stein\_ID) ON DELETE CASCADE

);

INSERT INTO Has\_Instructor VALUES(711, 'S711');

INSERT INTO Has\_Instructor VALUES(711, 'S911');

INSERT INTO Has\_Instructor VALUES(911, 'S140');

INSERT INTO Has\_Instructor VALUES(711, 'S690');

INSERT INTO Has\_Instructor VALUES(711, 'S555');

INSERT INTO Has\_Instructor VALUES(911, 'S890');

INSERT INTO Has\_Instructor VALUES(911, 'S190');

INSERT INTO Has\_Instructor VALUES(911, 'S081');

INSERT INTO Has\_Instructor VALUES(911, 'S881');

INSERT INTO Has\_Instructor VALUES(911, 'S678');

--Student has to be over the age of 15

--Student has Guardian if under the age of 18

CREATE TRIGGER Age\_Guardian

BEFORE INSERT

ON Student

FOR EACH ROW

BEGIN

IF( months\_between(TRUNC(sysdate),:NEW.Birth\_Day)/12 < 15)

THEN

raise\_application\_error(-20001, 'Student needs to be over age of 15');

END IF;

IF( months\_between(TRUNC(sysdate),:NEW.Birth\_Day)/12 < 18 AND :NEW.hasGuardian = 'No')

THEN

raise\_application\_error(-20001, 'Student under 18 needs a Guardian');

END IF;

END;

--Drops Trigger Age\_Guardian

DROP TRIGGER Age\_Guardian;

--Student can only schedule Road Test if Lesson is greater or equal to 5

CREATE TRIGGER Lesson5

BEFORE INSERT

ON Road\_Test

FOR EACH ROW

DECLARE countlesson int:=0;

BEGIN

SELECT COUNT(Lesson.lesson\_id) INTO countlesson FROM Lesson

WHERE :NEW.Stein\_id = Lesson.stein\_id;

IF( countlesson < 5)

THEN

raise\_application\_error(-20001, 'Student needs to take more than 4 Lessons');

END IF;

END;

--Drops Trigger Lesson5

Drop Trigger Lesson5;

--Student can't schedule more than 2 road test per calendar year

CREATE TRIGGER RoadTest2

BEFORE INSERT

ON Road\_Test

FOR EACH ROW

Declare

countroadtestdate int;

BEGIN

Select COUNT(Test\_Date) INTO countroadtestdate from Road\_Test

Where :NEW.Stein\_id = Stein\_id AND EXTRACT(YEAR FROM :NEW.Test\_Date) = EXTRACT(YEAR FROM Test\_Date);

IF( countroadtestdate >=2)

THEN

raise\_application\_error(-20001, 'Student cant scedule more than 2 road test per calendar year');

END IF;

END;

--Drops Trigger for RoadTest2

Drop Trigger RoadTest2;

--4 Queries

--Insert 1

--Schedules a Road Test

Insert Into Road\_Test Values(0000,TO\_DATE('01/01/2019', 'MM/DD/YYYY'), NULL, 0000, 'ExaminerName', 'Student\_Stein\_ID', 'Vehicle\_Vin');

--Insert2

--Schedules a Driving Lesson

Insert into Lesson Values(0000, TO\_DATE('01/01/2019', 'MM/DD/YYYY'), 'Student\_ID', 0000);

--Select 3

--This select will find the fail scores of different schools

Select DISTINCT( Select COUNT(Road\_Test.Score)

FROM Instructor,Driving\_School,Student,Road\_Test

where Instructor.drivingschool\_id = Driving\_School.drivingschool\_id

AND Instructor.instructor\_ssn= Student.instructor\_ssn

AND Student.Stein\_ID = Road\_Test.Stein\_id

--user can enter a specific driving school id e.g. 1

AND Driving\_School.drivingschool\_id = '1'

AND Road\_Test.Score = 'Fail')/(Select COUNT(Road\_Test.Score)

FROM Instructor,Driving\_School,Student,Road\_Test

where Instructor.drivingschool\_id = Driving\_School.drivingschool\_id

AND Instructor.instructor\_ssn= Student.instructor\_ssn

--user can enter a specific driving school id e.g. 1

AND Driving\_School.drivingschool\_id = '1'

AND Student.Stein\_ID = Road\_Test.Stein\_id)\*100 as Score From Road\_Test;

--Select 4

--This select will find students that have not yet passed their Road Test for Instructors

Select Student\_name FROM (

Select Student.Stein\_ID,Student\_name,MAX(Road\_Test.Test\_Date)

FROM Student, Instructor,Road\_Test

Where Student.instructor\_ssn = Instructor.instructor\_ssn

AND Student.Stein\_ID = Road\_Test.Stein\_id

--user can enter instructor's ssn to retrieve the list of student that has not yet passed e.g. 711

AND Instructor.instructor\_ssn = '711'

AND NOT EXISTS (SELECT Score FROM Road\_Test WHERE Score = 'Pass' AND Student.Stein\_ID = Road\_Test.Stein\_id)

Group BY Student.Stein\_ID,Student\_name);

--Indexes

--Indexes are created because it will speed up the select queries.

--Since insert statement is the only operation that cannot directly benefit from indexing (it has no where clause),

--indexes for insertion are not needed.

--creates index for instructor

--This index is created because it will speed up the process when trying to connect

--instructor and driving school in the WHERE clauses

Create Index Instructor\_FK\_INDEX

On Instructor(drivingschool\_id);

--drops index

Drop Index Instructor\_FK\_INDEX;

--creates index for student

--This index is created because it will speed up the process when trying to connect

--Student to instructor and Student to driving school in the WHERE clauses

Create Index Student\_FK\_INDEX

On Student(instructor\_ssn, drivingschool\_id);

--Drops index

Drop Index Student\_FK\_INDEX;

--creates index for Road Test

--This index is created because it will speed up the process when trying to connect

--Road Test to Vehicle and Road Test to Student in the WHERE clauses

--Also Score is in the index because Score is also used in the WHERE clauses

Create Index RoadTest\_FK\_Score\_INDEX

On Road\_Test(Stein\_id,vin,Score);

--Drops index

Drop Index RoadTest\_FK\_Score\_INDEX;

--(Bonus)Query Plan

EXPLAIN PLAN FOR

--select statement

Select DISTINCT( Select COUNT(Road\_Test.Score)

FROM Instructor,Driving\_School,Student,Road\_Test

where Instructor.drivingschool\_id = Driving\_School.drivingschool\_id

AND Instructor.instructor\_ssn= Student.instructor\_ssn

AND Student.Stein\_ID = Road\_Test.Stein\_id

AND Driving\_School.drivingschool\_id = '1'

AND Road\_Test.Score = 'Fail')/(Select COUNT(Road\_Test.Score)

FROM Instructor,Driving\_School,Student,Road\_Test

where Instructor.drivingschool\_id = Driving\_School.drivingschool\_id

AND Instructor.instructor\_ssn= Student.instructor\_ssn

AND Driving\_School.drivingschool\_id = '1'

AND Student.Stein\_ID = Road\_Test.Stein\_id)\*100 as Score From Road\_Test;

--Prints out the plan table

select plan\_table\_output from table(dbms\_xplan.display('plan\_table',null,'basic'));

EXPLAIN PLAN FOR

--select statement

Select Student\_name FROM (

Select Student.Stein\_ID,Student\_name,MAX(Road\_Test.Test\_Date)

FROM Student, Instructor,Road\_Test

Where Student.instructor\_ssn = Instructor.instructor\_ssn

AND Student.Stein\_ID = Road\_Test.Stein\_id

--user can enter instructor's ssn to retrieve the list of student that has not yet passed e.g. 711

AND Instructor.instructor\_ssn = '711'

AND NOT EXISTS (SELECT Score FROM Road\_Test WHERE Score = 'Pass' AND Student.Stein\_ID = Road\_Test.Stein\_id)

Group BY Student.Stein\_ID,Student\_name);

--Prints out the plan table

select plan\_table\_output from table(dbms\_xplan.display('plan\_table',null,'basic'));