SIM GLOBAL EDUCATION

UNIVERSITY OF WOLLONGONG AUSTRALIA

School of Computing and Information Technology

Student to complete:

Family name	
Other names	
Student number	
Table number	

CSCI235
Database Systems

Final Examination Paper
Session 2 2021

Exam duration	3 hours and 40 minutes
Weighting	40% of the subject assessment
Marks available	40 marks
Items permitted by examiner	Text-book, Lecture slides, and Tutorial notes
Directions to students	4 questions to be answered. Marks for each question are shown beside the question. All answers must be written in the answer booklet provided.

This examination is a take-it-home examination to be done on-line on the date of examination.

Version 2.0

Question 3 - (Total 12 marks)
PL/SQL

Time allocated: 45 minutes
Start time: 11:45 am SGT
End time: 12:30 pm SGT
Submission time start: 12:25 pm SGT
Submission time end: 12:40 pm SGT

Consider the relational tables listed below. The database contains information about the employees working for a transportation company, drivers employed by the company, trucks owned by the company, trips made by the drivers using a particular truck, and all legs of each trip.

```
-- Create a relational table EMPLOYEE
CREATE TABLE EMPLOYEE(
 E#                 VARCHAR2(12)    NOT NULL,
 NAME               VARCHAR2(50)    NOT NULL,
 DOB                DATE,
 ADDRESS            VARCHAR2(300)   NOT NULL,
 HIREDATE           DATE            NOT NULL,
 CONSTRAINT EMPLOYEE_PKEY PRIMARY KEY(E#)
);

-- Create a relational table DRIVER
CREATE TABLE DRIVER(
 E#                 VARCHAR2(12)    NOT NULL,
 L#                 VARCHAR2(15)    NOT NULL,
 STATUS             VARCHAR2(25)    NOT NULL,
 totalTripMade      NUMBER(5),
 CONSTRAINT DRIVER_PKEY PRIMARY KEY(E#),
 CONSTRAINT DRIVER_UNIQUE UNIQUE(L#),
 CONSTRAINT DRIVER_FKEY FOREIGN KEY(E#) REFERENCES EMPLOYEE(E#),
 CONSTRAINT DRIVER_STATUS CHECK ( STATUS IN
                        ('AVAILABLE', 'BUSY', 'ON LEAVE'))
);
```

```
-- Create a relational table MECHANIC
CREATE TABLE MECHANIC(
 E#                VARCHAR2(12)     NOT NULL,
 L#                VARCHAR2(15)     NOT NULL,
 STATUS            VARCHAR2(25)     NOT NULL,
 EXPERIENCE        VARCHAR2(25)     NOT NULL,
 CONSTRAINT MECHANIC_PKEY PRIMARY KEY(E#),
 CONSTRAINT MECHANIC_UNIQUE UNIQUE(L#),
 CONSTRAINT MECHANIC_FKEY FOREIGN KEY(E#) REFERENCES EMPLOYEE(E#),
 CONSTRAINT MECHANIC_STATUS CHECK ( STATUS IN
                        ('AVAILABLE', 'BUSY', 'ON_LEAVE')),
 CONSTRAINT MECHANIC_EXPERIENCE CHECK ( EXPERIENCE IN
                        ('BEGINNER', 'STANDARD', 'EXPERT'))
);

-- Create a relational table TRUCK
CREATE TABLE TRUCK(
 REG#            VARCHAR2(10)     NOT NULL,
 CAPACITY        NUMBER(7)        NOT NULL,
 WEIGHT          NUMBER(5)        NOT NULL,
 STATUS          VARCHAR2(25)     NOT NULL,
 CONSTRAINT TRUCK_PKEY PRIMARY KEY(REG#),
 CONSTRAINT TRUCK_STATUS CHECK ( STATUS IN
                        ('AVAILABLE', 'USED', 'MAINTENANCE'))
);

-- Create a relational table TRIP
CREATE TABLE TRIP(
 T#              NUMBER(10)       NOT NULL,
 L#              VARCHAR2(15)     NOT NULL,
 REG#            VARCHAR2(10)     NOT NULL,
 TRIPDATE        DATE             NOT NULL,
 CONSTRAINT TRIP_PKEY PRIMARY KEY (T#),
 CONSTRAINT TRIP_FKEY1 FOREIGN KEY (L#) REFERENCES DRIVER(L#),
 CONSTRAINT TRIP_FKEY2 FOREIGN KEY (REG#) REFERENCES TRUCK(REG#)
);

-- Create a relational table TRIPLEG and insert sample records
CREATE TABLE TRIPLEG(
 T#              NUMBER(10)       NOT NULL,
 LEG#            NUMBER(2)        NOT NULL,
 DEPARTURE       VARCHAR2(30)     NOT NULL,
 DESTINATION     VARCHAR2(30)     NOT NULL,
 CONSTRAINT TRIPLEG_PKEY PRIMARY KEY (T#, LEG#),
 CONSTRAINT TRIPLEG_UNIQUE UNIQUE(T#, DEPARTURE, DESTINATION),
 CONSTRAINT TRIPLEG_FKEY1 FOREIGN KEY (T#) REFERENCES TRIP(T#)
);
```

```
-- Create a relational table MAINTENANCE and insert sample
-- records
CREATE TABLE MAINTENANCE(
 REG#            VARCHAR2(10)       NOT NULL,
 L#              VARCHAR2(15)       NOT NULL,
 TIME            NUMBER(6),
 MAINTENANCEDATE DATE               NOT NULL,
 CONSTRAINT MAINTENANCE_PKEY PRIMARY KEY(REG#, L#,
MAINTENANCEDATE),
 CONSTRAINT MAINTENANCE_FKEY1 FOREIGN KEY (REG#) REFERENCES
TRUCK(REG#),
 CONSTRAINT MAINTENANCE_FKEY2 FOREIGN KEY (L#) REFERENCES
MECHANIC(L#)
);
```

a) Implement **a row trigger** that enforces the following consistency constraint.

Assume, that now, a column totalTripMade in the relational table DRIVER does not contain any values. Create a row trigger that automatically updates the values in the column (totalTripMade) when a new trip made by a driver is inserted into the relational table TRIP. Your trigger, once activated, will compute the total number of trips made by the driver and update the totalTripMade column in the relational table DRIVER.

NOTE: You do not need to consider any other cases that may change the value in the column totalTripMade; that is, NO NEED to consider delete and update cases.

(5.0 marks)

b) Implement a stored PL/SQL function LONGTRIP(DLNUM) that finds the length (the total number of legs) of the longest trip performed by a driver identified by a driving license number (L# attribute in table TRIP) and parameter DLNUM parameter in the function.

Use a stored function LONGTRIP in SELECT statement to list the names of all drivers together with the length of the longest trip performed by each driver.

(7.0 marks)

END OF QUESTION 3