MONASH University
Information Technology

FIT2094-FIT3171 Databases

2022 Summer Semester B

Assignment 2A - Olympic Logistics Services (OLS)

FIT2094 Learning Outcomes: 2, 3, 4, 7 (see Unit Preview)
FIT3171 Learning Outcomes: 2, 4, 5, 8 (see Unit Preview)

Assignment weighting 10%

Assignment marked out of 100 and released as a grade out of 10

Olympic Logistics Services (OLS) is a private company subcontracted by the Olympic Federation to transport officials around during the Olympic competition.

OLS owns a fleet of vehicles, which vary from sedans to vans. Some vehicles have special features such as full blackout, armour plating, ability to transport long items such as javelins and pole vault poles, integrated GPS tracking etc.

OLS records the Vehicle Identification Number (VIN), the registration number, make, model, colour, and the number of seats available for each vehicle. OLS only purchases vehicles of a single colour.

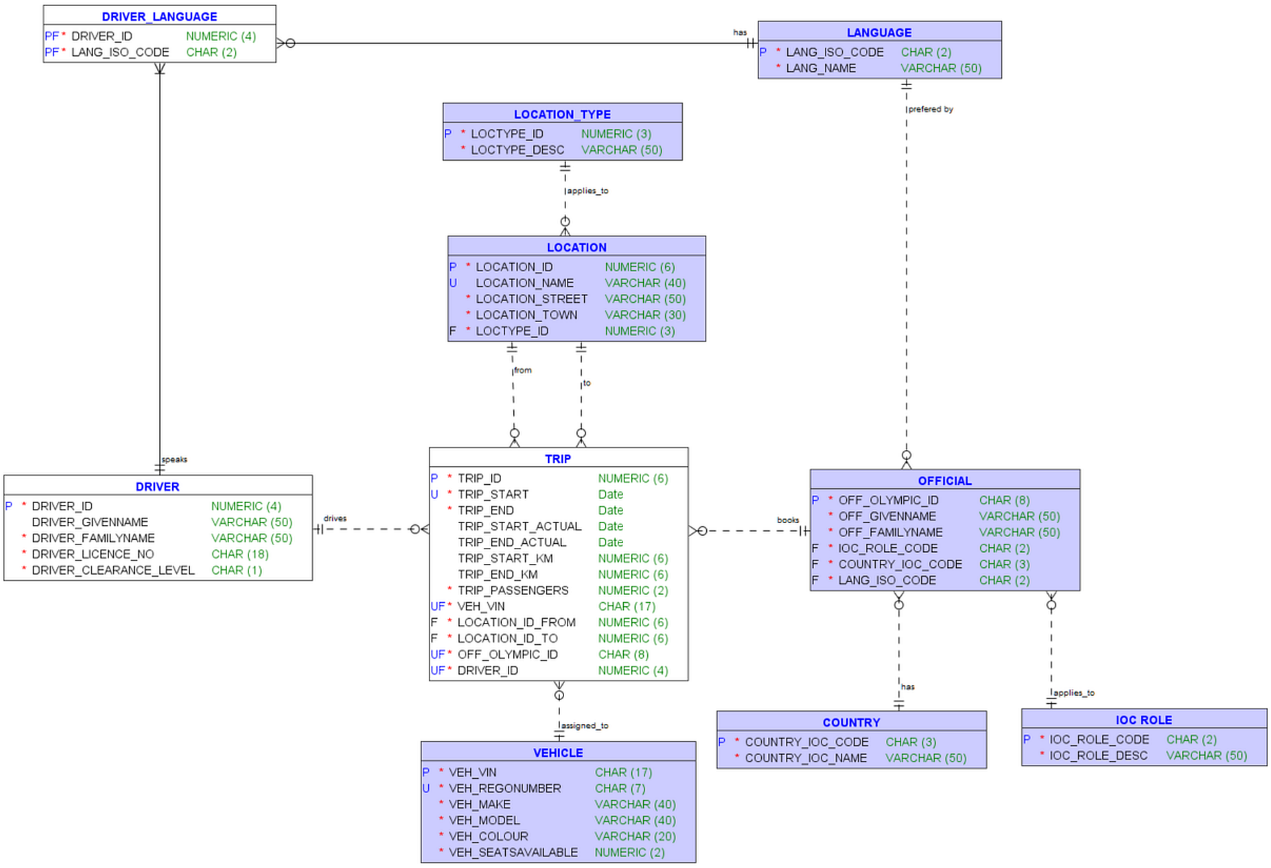
Olympic officials from all participating countries use OLS's services. OLS records the name of the country that an official is representing, the official's Olympic ID (8 characters), their name, their role at the Olympics (eg head coach, judge, physician etc) and the official's preferred language. To record languages, OLS makes use of the ISO6391 two character language codes, for example English (EN) and Chinese (ZH) – OLS records the ISO6391 code and name of the language. An official may use OLS's services multiple times (even during a single day); the only limiting factor is whether a suitable vehicle is available at the time they wish to travel. Where multiple officials travel in a single vehicle, the vehicle is booked in the name of the official who made the booking and will travel in the vehicle for this trip, the other passengers do not need to be recorded.

OLS vehicles are driven by the company's drivers. Each driver is assigned a unique driver’s ID. The driver's name, licence number (18 characters in length), date of birth and the level of clearance granted to the driver are recorded. The clearance levels are digits from 1 to 4 representing the security clearance of the driver (4 is the highest level of clearance). OLS employees are familiar with the meaning of these codes (1 to 4) and thus they do not require any explanation of the codes in the system, just the actual codes.

OLS records (using ISO6391 language codes) the languages that a driver speaks – some drivers speak several languages. The employee organising a booking will try to match an official's preferred language with that which a driver speaks.

Transport bookings require the assignment of a suitable driver and a suitable vehicle matching the needs of the official. This matching will take place via a computer program to be created; you are not concerned with this program, only the back end database to support such decisions. Each trip is assigned a unique trip ID. The intended start date and time and projected end date and time are recorded when a booking is first placed. The pickup location and drop off location are recorded (all locations added to the system are assigned a unique location ID). When the trip has been completed the actual start date and time and actual end date and time are recorded. OLS also records the start odometer reading and the end odometer reading.

A model to represent this system has been developed:


DRIVER_LANGUAGE
PF * DRIVER_ID NUMERIC (4)
PF * LANG_ISO_CODE CHAR (2)
has
LANGUAGE
P * LANG_ISO_CODE CHAR (2)
* LANG_NAME VARCHAR (50)
prefered by
LOCATION_TYPE
P * LOCTYPE_ID NUMERIC (3)
* LOCTYPE_DESC VARCHAR (50)
applies_to
LOCATION
P * LOCATION_ID NUMERIC (6)
U LOCATION_NAME VARCHAR (40)
* LOCATION_STREET VARCHAR (50)
* LOCATION_TOWN VARCHAR (30)
F * LOCTYPE_ID NUMERIC (3)
from
to
speaks
DRIVER
P * DRIVER_ID NUMERIC (4)
DRIVER_GIVENNAME VARCHAR (50)
* DRIVER_FAMILYNAME VARCHAR (50)
* DRIVER_LICENCE_NO CHAR (18)
* DRIVER_CLEARANCE_LEVEL CHAR (1)
drives
TRIP
P * TRIP_ID NUMERIC (6)
U * TRIP_START Date
* TRIP_END Date
TRIP_START_ACTUAL Date
TRIP_END_ACTUAL Date
TRIP_START_KM NUMERIC (6)
TRIP_END_KM NUMERIC (6)
* TRIP_PASSENGERS NUMERIC (2)
UF * VEH_VIN CHAR (17)
F * LOCATION_ID_FROM NUMERIC (6)
F * LOCATION_ID_TO NUMERIC (6)
UF * OFF_OLYMPIC_ID CHAR (8)
UF * DRIVER_ID NUMERIC (4)
books
OFFICIAL
P * OFF_OLYMPIC_ID CHAR (8)
* OFF_GIVENNAME VARCHAR (50)
* OFF_FAMILYNAME VARCHAR (50)
F * IOC_ROLE_CODE CHAR (2)
F * COUNTRY_IOC_CODE CHAR (3)
F * LANG_ISO_CODE CHAR (2)
assigned_to
has
applies_to
VEHICLE
P * VEH_VIN CHAR (17)
U * VEH_REGONUMBER CHAR (7)
* VEH_MAKE VARCHAR (40)
* VEH_MODEL VARCHAR (40)
* VEH_COLOUR VARCHAR (20)
* VEH_SEATSAVAILABLE NUMERIC (2)
COUNTRY
P * COUNTRY_IOC_CODE CHAR (3)
* COUNTRY_IOC_NAME VARCHAR (50)
IOC ROLE
P * IOC_ROLE_CODE CHAR (2)
* IOC_ROLE_DESC VARCHAR (50)

You have been supplied with an SQL script **ols_initialSchemaInsert.sql** which partially implements the OLS model (it creates the tables coloured in purple on the above diagram and inserts some initial *sample* data). This file **may NOT be altered (edited) in any way**.

Steps for working on Assignment 2A

1. Download the Assignment 2A Required Files zip archive from Moodle

2. Place the zip archive in your local (MoVE or local HDD) repository in the folder
   /Assignments/Ass2A

3. Extract the zip archive and place the contained files in your local (MoVE or local HDD) repository in the folder /Assignments/Ass2A. Do not add the zip archive to your local repo. Then add, commit and push them to the FITGitLab server.

4. Run ols_initialSchemaInsert.sql

5. Write your answer for each task in its respective file (e.g. write your answer for Task 1 in T1-ols-schm.sql and so on).

6. Save, add, commit and push the file/s **regularly** while you are working on the assignment

7. Finally, when you have completed all tasks, upload all required files from your local repository to Moodle (if you are using MoVE you will need to download them to your local HDD first - do not attempt to upload from MoVE). Check that the files you have uploaded are the correct files (download them from Moodle into a temporary folder and check they are correct). After you are sure they are correct, submit your assignment.

Note that the final SQL scripts you submit MUST NOT contain SPOOL or ECHO commands (you may include them as you work but **must** remove them before submission). Please carefully read the "CRITERIA FOR MARKING" on page 10 of this document.

TASK 1: DDL (25 marks)

For this task you are required to add to **T1-ols-schm.sql**, the CREATE TABLE and CONSTRAINT definitions which are missing from the supplied partial schema script in the positions indicated by the comments in the script.

The table below provides details of the meaning of the attributes in the missing three tables. You **MUST** use exactly the same relation and attribute names as shown in the data model above to name the tables and attributes which you add. The attributes must be in the same order as shown in the model. These new DDL commands *must be hand-coded, not generated in any manner (generated code will not be marked)*.

Table name	**Attribute name**	**Meaning**
DRIVER_LANGUAGE		
	driver_id	Unique identifier for a driver
	lang_iso_code	ISO code for a language (e.g., EN for English language)
DRIVER		
	driver_id	Unique identifier for a driver
	driver_givenname	Driver's given name
	driver_familyname	Driver's family name
	driver_license_no	Driver's license number
	driver_clearance_level	The level of security clearance granted to the driver (1 to 4).
TRIP		
	trip_id	Unique identifier for a trip
	trip_start	Date and time the trip is booked to start
	trip_end	Date and time the trip is booked to end
	trip_start_actual	Date and time the booked trip actually starts
	trip_end_actual	Date and time the booked trip actually ends
	trip_start_km	The odometer reading at the start of the trip
	trip_end_km	The odometer reading at the end of the trip
	trip_passengers	The number of booked passengers for the trip

Note that:

- During the design phase, it was decided that the three natural keys for the TRIP table listed below should be enforced:
    - trip_start, veh_vin
    - trip_start, off_olympic_id
    - trip_start, driver_id

To test your code you will need to first run the provided script **ols_initialSchemaInsert.sql** to create the other required tables. **ols_initialSchemaInsert.sql** contains at the head of the file the drop commands for ***all*** tables in this model. If you have problems with Task 1 simply rerun **ols_initialSchemaInsert.sql** which will cause all tables to be dropped and correct the issues in your script.

TASK 2:-Populate Sample Data (25 marks)

Before proceeding with Task 2, you must ensure you have run the file ols_initialSchemaInsert.sql (which **must not be edited in any way**) followed by the extra definitions that you added in Task 1 above (T1-ols-schm.sql).

Load the DRIVER, DRIVER_LANGUAGE, and TRIP tables with **your own test data** using the supplied **T2-ols-insert.sql file** script file, and SQL commands which will insert *as a minimum*, the following sample data:

(i) 8 DRIVER entries

(ii) 10 DRIVER_LANGUAGE entries

- Included at least 2 drivers being able to speak at least 2 languages

(iii) 10 TRIP entries

- Have at least 3 different trip_start dates
- Booked by at least 5 different officials
- Driven by at least 5 different drivers
- Have at least 3 completed trips
- Have at least 3 incompleted trips (booking only)

In adding this data you must ensure that the test data thoroughly tests the model as supplied, so as to ensure your schema is correct.

Your inserted data must conform to the following rules:

(i) You may treat all of the data that you add as a single transaction since you are setting up the initial test state for the database.

(ii) The primary key values for this data should be hardcoded values (i.e. **NOT** make use of sequences) and must consist of values below 100.

(iii) Dates used must be chosen between the 26th Jan 2022 and 9th Feb 2022 (inclusive). The trip start times of a booked trip must be between 7 AM and 10 PM (inclusive). The data added must be sensible eg. trip end dates/times should be after trip start dates/times.

For this task **ONLY,** Task 2, you may look up and include values for the loaded tables/data directly where required. However, if you wish, you can still use SQL to get any non-key values.

In carrying out this task you must not modify any data or add any further data to the tables which were populated by the *ols_initialSchemaInsert.sql* script.

For all subsequent questions (Task 3 onwards) **you are NOT permitted to**:

- manually lookup a value in the database, obtain its primary key or the highest/lowest value in a column,
- manually calculate values (including dates/times) external to the database, e.g. on a calculator and then use such values in your answers. ***Any necessary calculations must be carried out as part of your SQL code***, or
- assume any particular contents in the database - rows in a table are potentially in a constant state of change

Your answers must recognise the fact that you have been given, with the supplied insert file, only a very small sample snapshot of a multiuser database, as such you must operate on the basis that there will be ***more data in all of the tables of the database than you have been given. Your answers must work regardless of the extra quantity of this extra "real" data and the fact that multiple users will be operating in the tables at the same time. You must take this aspect into consideration when writing SQL statements.***

You must ONLY use the data as provided in the text of the questions. **Failure to adhere to this requirement will result in a mark of 0 for the relevant question**.

Your SQL must correctly manage transactions and use sequences to generate new primary keys for numeric primary key values (under no circumstances may a new primary key value be hardcoded as a number or value).

TASK 3: DML (25 marks)

Your answers for this task (Task 3) must be placed in the supplied SQL Script **T3-ols-dm.sql**

For this task you are required to complete the following sub-tasks in the same order they have mentioned. Where you have been supplied with a string contained in quotes, such as 'X33445566778899MCH' you may search in the database using that exact string, and when a name is supplied you may break the name into given name and family name, for example 'Michael Chu' can be split into 'Michael' and 'Chu' (you may assume the string will match the case in the database).

(a) An Oracle sequence is going to be implemented in the database for the subsequent insertion of records into the database for the DRIVER and TRIP tables.

Provide the CREATE SEQUENCE statement to create a sequence which could be used to provide primary key values for the DRIVER and TRIP tables. Both sequences should start at 100 and increment by 1. Immediately prior to the create sequence commands place appropriate DROP SEQUENCE commands so they will cause the sequences to be dropped before being created if they exist. *Please note that there can only be these two sequences introduced and used in Task 3.*

[2 marks]

(b) Record a new driver named ‘Michael Chu’ who has a driver license number of ‘X33445566778899MCH’, clearance level ‘3’ and can speak English (‘EN’) and Spanish (‘ES’). Then, record a new booking for a trip to be made for this new driver and the vehicle with the registration number ‘**OLSID22’** on 7th February 2022 with the trip start time at 7:30AM and trip end time at 9:00 AM. This trip is booked by the Olympic official ‘Simon Nathan’ from Australia (‘AUS’) to transport 3 Olympic officials from ‘Beijing National Stadium’ to 'National Alpine Ski Centre’.

You may assume ‘Simon Nathan’ is the only official of that name in the Australian contingent.

All of the above actions should be treated as a single transaction.

[11 marks]

(c) After the above trip is finished, the following information is recorded in the OLS system to complete the trip's details. The odometer reading at the start of the trip is 5678 and at the end of the trip is 5780. The actual start time is late at 7:35 AM and the actual end time is at 9:00 AM as planned.

Write necessary SQL statement(s) to record the above data in the OLS system.

[6 marks]

(d) After completing the above trip booked on 7th February 2022, the Australian ('AUS') official ‘Simon Nathan’ had his COVID-19 test returned as positive, hence unfortunately, has been asked to be isolated and prevented from attending all remaining events. As a result, all future trips starting from 8th February 2022 (inclusive) that he has booked must be cancelled. OLS asks you to remove all his future booked trips.

Write the necessary SQL statement(s) to remove all future trips that ‘Simon Nathan’ has booked.

[6 marks]

TASK 4: DATABASE MODIFICATIONS (25 marks)

Your answers for these tasks (Task 4) must be placed in the supplied SQL script **T4-ols-alter.sql**

The required changes must be made to the "live" database (the database *after* you have completed tasks 1, 2 and 3) **not** by editing and executing your schema file again. Before carrying out the work below, please ensure that you have completed tasks 1, 2 and 3 above.

If in answering these questions you need to create a table, please place a drop table statement prior to your create table statement.

(a) OLS wishes to record the actual trip duration for each completed trip (the duration of each completed trip must be rounded to whole minutes).

Modify the database structure and update the appropriate trip data to implement this requirement.

[5 marks]

(b) In general, Olympic officials are asked to book trips at least 24 hours in advance, however, emergency bookings on a particular day are accepted if a driver is available. On any given day usually most drivers have been booked for trips, however, some trips may be canceled with short notice due to unexpected circumstances (e.g., officials are asked to isolate themselves due to a COVID positive test). OLS requires that the system indicates whether a driver is *currently* available so they may be assigned to emergency booking requests. Please note the trip start times must be between 7 AM and 10 PM. This availability will be updated first thing each morning.

Modify the database structure to meet OLS's requirements and update the current availability status (e.g., Y or N) for each driver for today.

When working on this task, you may assume that a driver is not available if they are already booked for at least 1 trip today.

[8 marks]

(c) For each trip, instead of recording only the official who books the trip, OLS wishes to record the details of all Olympics officials who take part for each trip for contact tracing purposes. OLS still requires the system to record the Olympic official who made the booking (ie. contacted them).

Modify the database structure to meet OLS's above requirements.

[12 marks]

SUBMISSION REQUIREMENTS

Due Date: **Friday 4th February 2022 at 2 PM (AEDT)**

*Please note, if you need to resubmit, you **cannot** depend on your tutors' availability, for this reason, please be VERY CAREFUL with your submission*. **It is strongly recommended that you submit several hours before this time to avoid such issues.**

For this assignment there are four files you are **required** to submit:

- T1-ols-schm.sql
- T2-ols-insert.sql
- T3-ols-dm.sql
- T4-ols-alter.sql

If you need to make any comments to your marker/tutor please place them at the head of each of your solution scripts in the "Comments for your marker:" section.

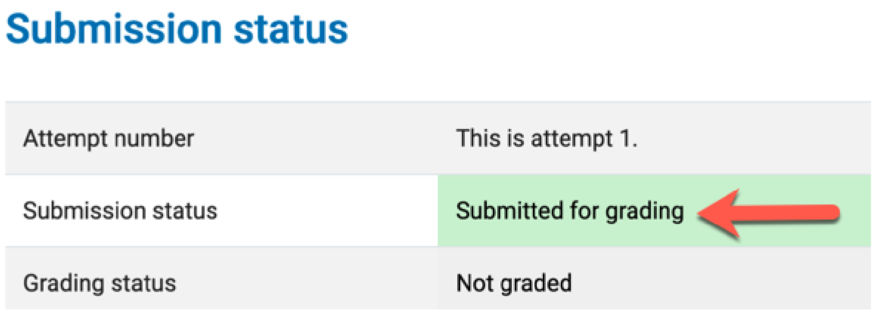
Do not zip these files into one zip archive, submit four independent SQL scripts. The individual files must also have been pushed to the FIT GitLab server with an appropriate history as you developed your solutions (a minimum of four pushes - 1 per file, however we would *strongly recommend more than this*). **Please ensure your commit comments are meaningful**.

Late submission will incur penalties at the rate of -5 mark for every 12 hours the submission is late.

Please note we **cannot mark any work on the GitLab Server**, you need to ensure that you submit correctly via Moodle since it is only in this process that you complete the required student declaration without which work **cannot be assessed**.

It is your responsibility to ENSURE that the files you submit are the correct files - we strongly recommend after uploading a submission, and prior to actually submitting, that you download the submission and double-check its contents.

Your assignment **MUST** show a status of "Submitted for grading" before it will be marked.



Submission status

Attempt number	This is attempt 1.
Submission status	Submitted for grading
Grading status	Not graded

If your submission shows a status of "Draft (not submitted)" it will not be assessed and **will incur late penalties after the due date/time**.

Please ***carefully*** read the documentation under the "Assignment Submission" on the Moodle Assessments page which covers things such as extensions and resubmission.

CRITERIA FOR MARKING

Submissions **will be graded on**:

- the correct application of relational database principles,
- the correct handling of transactions and the setting of appropriate transaction boundaries i.e. correct placement of commits, and
- the correct application of SQL statements and constructs to:
    - create and alter tables including the required constraints and column comments,
    - populate tables,
    - modify existing data in tables, and
    - modify the "live" database structure to meet the expressed requirements (including appropriate use of constraints). In making these modifications there must be no loss of existing data or data integrity within the database.

Submissions **will be penalised** as follows:

- if SQL scripts contain contain SET ECHO … or SPOOL commands [-10 marks]
- if SQL scripts makes use of views [-10 marks]
- if SQL scripts do not use to_char/to_date where appropriate in handling dates [-10 marks],
- if SQL scripts do not have an appropriate development history on the FIT GitLab server for all source files (**at least four pushes required, but more expected**) [-10 marks].