FIT3171 Databases

Creating, Populating and Manipulating Database - Run Monash (RM)

Purpose	Students will be asked to implement, via SQL, a small database in the Oracle RDBMS from a provided logical model case study, followed by the insert of appropriate data to the created tables. Once populated the database will be used to: carry out specified DML commands and make specified changes to the database structure via SQL. This task covers learning outcomes: 1. Apply the theories of the relational database model. 3. Implement a relational database based on a sound database design. 4. Manage data that meets user requirements, including queries and transactions.
Your task	This is an open book, **individual task**. The final output for this task will be a set of tables and data implemented in the Oracle RDBMS
Value	**25%** of your total marks for the unit
Due Date	Thursday, **26 May 2022**, 4:30 PM (AEST) / 2:30 PM (MYT)
Submission	● Via Moodle Assignment Submission. ● FIT GitLab check ins will be used to assess history of development
Assessment Criteria	● Application of relational database principles. ● Handling of transactions and the setting of appropriate transaction boundaries. ● 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).
Late Penalties	● 10% deduction per calendar day or part thereof for up to one week ● Submissions more than 7 calendar days after the due date will receive a mark of zero (0) and no assessment feedback will be provided.
Support Resources	See Moodle Assessment page
Feedback	Feedback will be provided on student work via: ● general cohort performance ● specific student feedback ten working days post submission ● a sample solution

Marking Guide

Submitted code will be assessed against an optimal solution for this task - this optimal solution will be available as a sample solution after Assignment 2 has been graded. Given that this is SQL there are often several alternative approaches possible, such alternatives will be graded based on the code successfully meeting the requirements, If it does the answer will be accepted and graded appropriately.

Marking Criteria	**Items Assessed**
TASK 1 DDL 20 marks	
DDL Creation of tables	**Maximum 9.5 marks - Create table:** ● Marks awarded for correct table DDL ● Marks awarded for correct attributes/data types ● Marks awarded for correct PK definition ● Marks awarded for correct use of column comments ● Mark penalty applied if generated schema used
DDL implementation of non PK database constraints	**Maximum 10.5 marks - Non PK Constraints:** ● Marks awarded for correct implementation of non PK constraints: ○ CHECK ○ UNIQUE ○ FK
TASK 2 Populate Sample Data 20 marks	
Insert of required items test data	**Maximum 10 marks - Insert of data:** ● Marks awarded for correct insert of required data ○ 5 EMERCONTACT entries ○ 15 COMPETITOR entries ○ 30 ENTRY entries ○ 5 TEAM entries ● Marks awarded for correct management of transactions
Insert of valid test data	**Maximum 10 marks - Valid data inserted:** ● Marks awarded for validity of data inserted ○ meets the requirements expressed in the assignment brief ● Marks awarded for correct management of dates when inserting

Task 3 DML 25 marks	
	Maximum 25 marks - Satisfy brief requirements: ● Marks awarded (a) - (e) for SQL code which meets the expressed requirement ● Mark penalty applied if commit not used appropriately ● Mark penalty applied if date handling and string database lookups not managed correctly
Task 4 Database Modifications 15 marks	
	Maximum 15 marks - Satisfy brief requirements: ● Marks awarded (a) - (c) for SQL code which meets the expressed requirement (including appropriate use of constraints). In making these modifications there must be no loss of existing data or data integrity within the database. ● Mark penalty applied if commit not used appropriately ● Mark penalty applied if column comments not used where required
Task 5 PL/SQL 20 marks	
	Maximum 20 marks - Satisfy brief requirements: ● Marks awarded (a) - (c) for PL/SQL code which meets the expressed requirement. ● Marks awarded for writing a test harness for each question (a) - (c) which includes both successful and failed tests (all errors your code raises must be tested) .
Penalty Criteria	**Penalty Applied**
Limited/No push of model to FITGitLab server resulting in lack of development history.	If less than five pushes showing a clear development history a ***grade deduction of 10 marks applied***. Note that the expectation is that you would push significantly more times than this.
Use of ● PL/SQL in Task 1 - Task 4, ● VIEWs, and/or ● SET ECHO or SPOOL commands	Use of PL/SQL in Task 1 - Task 4, VIEWS, inclusion of SET ECHO/SPOOL commands in submitted scripts will result in a ***grade deduction of 10 marks being applied.*** ***** PL/SQL may ONLY be used for Task 5 *****
Incorrect application of relational database principles	Marks will be deducted, based on the particular question, where basic relational model principles have been violated. For example, creation of a table which is not in 3NF