INFO6002: Database Management 2

Trimester 2, 2021 – Callaghan & Online

Assignment 2 – Database Design & Implementation

Blackboard Submission Due: *10pm, Friday 2nd July 2021*

Demonstration Sessions:
Callaghan students (Lab Session): *4 - 6pm, Wednesday 7th July 2021*
Online students (via Zoom): *5 - 6pm, Tuesday 6th July 2021*

WORTH 25% of final course mark.
This is an INDIVIDUAL Assignment.

Assignment Requirements

This assignment contains 5 parts. You will submit your work to Blackboard as well as demonstrate your working scripts. For Callaghan enrolled students, you will demonstrate during the Week 9 lab session. For Online enrolled students, you will demonstrate on Zoom (5 - 6pm, Tuesday 6th July 2021).

<u>Part 1: Revised Data Requirements, EER Model & Data Dictionary (2 marks)</u>

In this section, you will revise your work from Assignment 1 based on any feedback given by your lecturer. You will re-submit your:

A. Data Requirements
B. EER Model
C. Data Dictionary

<u>Part 2: Relational Mapping & Normalisation (3 marks)</u>

Next, the EER Model needs to be mapped to a relational schema and normalised.

The relational model needs to be documented in DBDL format. Sample DBDL format is given in the below:

ISBN (id, number, itemNo)
Primary Key id
Alternate Key number
Foreign Key itemNo **references** Book(itemNo)
ON UPDATE CASCADE, ON DELETE CASCADE

DBDL format is provided in your text. (Databases Systems – A Practical Approach to Design, Implementation, and Management – 6th Edition by Connolly and Begg 2015).

Secondly, identify any relations that are not normalised and show the steps to transform them into a normalised relation.

Part 3: Implementation – Database Script (5 marks)

Create a T-SQL script for the database design in Part 2. You will create a database with all the necessary tables and constraints: primary key, foreign key, not null, unique and check constraints. The database must be populated with sufficient and meaningful records for evaluation.

Part 4: Stored Procedure (10 marks)

Implement the following stored procedures. Ensure that each stored procedure is tested with appropriate sample data. Test cases should be saved in a separate test script.

(1) Create an order	
Procedure name	usp_createCustomerOrder
Description	This stored procedure creates a new customer order. The sales tax is 10% of order amount.
Input Parameters	Customer id – Id of customer Items – A Table-valued Parameter (TVP) of items (item number, quantity, discountPromotionCode). Note that the discountPromotionCode is null for items where a discount does not apply or items are not part of promotion FulfilmentType – Type of order fulfilment (delivery or pickup) OrderType – Type of order (phone, walk-in, app, website) Employee id – Employee id of employee taking the order. This will be null for an online order OrderDateTime – Date and time of order is placed DeliveryAddress – Delivery address if it is a delivery order ExpectedOrderFulfilmentDateTime – Date and time when the order needs to be fulfilled.
Output Parameter	Order number of the newly created order
Functionality	Creates a new order with the provided input parameters. After each order, the ingredients used for the order are deducted from the current stock levels of the ingredients. Returns the newly created order number. If there is any error an appropriate error message is raised.
SQL script	create_usp_createCustomerOrder.sql
Test script	test_usp_createCustomerOrder.sql

Section 5: Business Rule (5 marks)

Business Rule: Order Satisfiability

Before an order can be taken, it is important to verify that the order can be satisfied with the available ingredients in the store. If the ingredients available are insufficient to fulfill the order an appropriate error message needs to be generated and the order cancelled.

Ensure that the above business rule is enforced in the database. You need to generate appropriate error messages if an attempt to violate the constraint is attempted.

Blackboard Submission Requirements

The following items need to be submitted to Blackboard at:

Assessment / ASSIGNMENT 2 / Assignment 2 Submission.

Submit a single .zip folder named as:

A2, your first name, your surname and your student number
e.g. *A2SimonLee1234567.zip*

The zip folder will contain the following files:

	Description	**Format**
Part 1A: Requirements Document	Revised Requirements Document including Data Requirements, Transaction Requirements & Business Rules	MS Word or PDF format
Part 1B: EER Model	Revised EER Model	Visio or PDF format
Part 1C: Data Dictionary	Revised Data Dictionary	MS Word or PDF format
Part 2: Relational Mapping & Normalisation	Document containing: • Relational Database Schema in DBDL Format • Normalisation discussion including identifying the normal form of each relation and clear documentation of normalisation steps for any relations not already normalised	MS Word or PDF format

The following T-SQL Script files must also be included in the .zip file:

	SQL Script name	**Description**
Part 3: Implementation – Database Script	createDB.sql	Contains the script that creates the database along with all constraints. Also, inserts sample data into the tables.
Part 4: Stored Procedure	create_usp_createCustomerOrder.sql	Contains T-SQL Script to create the stored procedure.
	test_usp_createCustomerOrder.sql	Contains the test scripts to test the stored procedure.
Part 5: Business Rule	create_enforceBusinessRule.sql	Contains T-SQL Script to create the business rule.
	test_enforceBusinessRule.sql	Contains the test scripts to test the business rule.

Demonstration Requirements

Each student must demonstrate their working SQL Scripts during the week 9 lab session on Wednesday 7th July 2021 from 4-6pm (Callaghan students) or Zoom session on Tuesday 6th July from 5-6pm (Online students). Failure to attend the demonstration can result in a zero grade for the assignment.

Marking Rubric

The assessment RUBRIC is given below:

	Excellent	Satisfactory	Fail
Part 1: Revised Data Requirements, EER Model & Data Dictionary. (2)	(2)	(1)	(0)
	All requirements documented in clear and complete manner. The document includes data requirements, transaction requirements and business rules. All requirements are accurately captured and modelled in EER. Data Dictionary without errors.	Many requirements outlined. Some requirements missing/incorrect. Most requirements are accurately captured and modelled in EER. Data Dictionary mostly without errors.	No requirements document or EER model or Data Dictionary submitted.
	(3)	(1-2)	0
Part 2: Relational Mapping & Normalisation (3)	Conceptual model is correctly mapped to relational model without any omissions.. Normalisation discussed in detail.	The relational model is mostly mapped accurately. Normalisation has omissions/errors.	Relational schema is missing and/or poorly constructed. No normalisation performed

	Excellent	Good	Satisfactory	Poor	Fail
Part 3: Implementation – Database Script (5)	(5)	(4)	(3)	(1-2)	(0-1)
	The T-SQL script maps to the database design accurately. The script executes without any errors. The code is well-documented. All necessary tables and constraints are clearly shown. The database is populated with sufficient and meaningful records for evaluation.	The T-SQL script maps to the database design accurately. The script executes without any errors. The code is documented. Necessary tables and constraints are shown. The database is partially populated.	T-SQL script maps to database design for most cases. The script executes correctly with some constraints. Partial data inserted to database.	The T-SQL script has missing content/ partially maps to design. Many critical objects missing/errors in script. Partial or no documentation of scripts. Missing or little data inserted.	Missing or poorly written script with errors, missing content – objects, constraints and data.

Part 4: Stored Procedure (10)	(9-10)	(7-8)	(4-6)	(2-3)	(0-1)
	The functionality is implemented without errors. The code is well documented and presented. All appropriate warning and error messages are raised. All appropriate test cases are implemented to verify the correctness.	The functionality is implemented correctly. The code is partially documented and/or tested. Major test cases are implemented.	The code has minor errors. The code is partially documented and tested. The code executes and has at least 1 test case.	The code has major errors. It is poorly documented and tested.	No code and/or basic outline of functionality presented. No functionality executed or tested.
Part 5: Business Rule (5)	(5)	(4)	(3)	(1-2)	(0-1)
	The business rule is correctly implemented. The code is well documented. Error messages and warnings are appropriately raised. The code is tested with all appropriate test cases.	The business rule is correctly implemented. The code is partially documented. Appropriate error messages and warnings are raised. Major cases are tested.	The business rule is implemented with minor errors/omissions. The code is partially commented. Error messages are raised. Partially tested.	The code has major errors and/or partially executes. Poor documentation and testing.	The code does not compile or run. Has major errors and/or partial logic is shown. No testing and/or documentation