

ICT285 Databases

Semester 2, 2020

Assignment 1

Individual Assignment (20% of the final mark for the unit)

Submission Instructions:

Please submit a single Word document using the link in Moodle. The due date is 20th September, 1130pm. You do not need to include a separate cover sheet, but you should include your name and student number as part of your document filename. Your name and student number should also be included within the assignment document.

From the Unit Information Guide:

- It is your responsibility to **keep a copy of any assignment handed in for assessment**. Make sure you keep a copy of the final (submitted) version as well as the backups you make as you create the Assignment. It is recommended that you also keep a hard copy of any written assignments.
- It is especially **important to maintain regular backups of all your practical work**. You should ensure you keep a separate backup on a thumb drive, cloud or similar. It is recommended that you keep both a hard copy and an electronic copy.
- Late submission will be penalised at a rate of five per cent of the total marks available for the Assignment per day (or part thereof). **The deadline for submission will be specified in LMS and on the Assignment**. Each Assignment will have a final date by which the Assignment will be accepted for marking. Assignments not received by the closing date **WILL NOT BE MARKED** and will be recorded as a 0.
- Under exceptional circumstances, an extension can be provided. Extensions for assignment submissions can only be granted if requested in advance of the due date for submission, and with a good reason. Applications for extensions should be made to the Unit Coordinator, not to the student's lecturer/tutor who is unable to grant extensions. If an extension is granted, the Unit Coordinator will tell you at the time of allowing it whether any penalty in marks will apply to the submitted work.
- This unit uses software called Urkund when viewing work that you submit. Urkund is a pattern-matching system designed to compare work submitted by students with other sources from the internet, journals/periodicals, and previous submissions. Its primary purpose is to detect any submitted work that is not original and provide a thorough comparison between the submitted document and the original sources.
- More information about how to avoid plagiarism is contained within the Murdoch Academic Passport (MAP) unit <https://moodleprod.murdoch.edu.au/course/view.php?id=2684>
University policies on academic integrity can be accessed here: <http://our.murdoch.edu.au/Educational-technologies/What-you-need-to-know/>

This Assignment requires you to answer several questions on relational database principles and SQL and to design a database based on a case study.

The Assignment addresses the following learning outcomes for the unit:

- LO 1. Apply your knowledge of relational database principles and theory to create effective and efficient database designs
- LO 2. Define, create and manage relational database systems using SQL
- LO 3. Design and implement a database to a given specification using a DBMS

Marks are distributed as follows:

Question 1: Relational algebra	20
Question 2: SQL Select queries	20
Question 3: Further SQL	15
Question 4: Normalisation	20
Question 5: Conceptual design	25
Total	100

Question 1: Relational algebra (20 marks)

The database used for this question is a very simple one with the following schema: (**Primary keys are bold**, foreign keys are underlined)

CUSTOMER (**CustID**, FirstName, LastName, City, Phone, Email)

INVOICE (**InvoiceNumber**, CustID, Date)

INVOICE_ITEM(**InvoiceNumber**, **ItemNumber**, Quantity)

ITEM (**ItemNumber**, ItemName, UnitPrice)

When a customer makes a purchase, an invoice is created. The invoice may be for many items. For example, in a single purchase, a customer might buy 10 Back Scratchers, 4 Hair Removers and a Dog Lead.

Provide **relational algebra** (NOT SQL) queries to find the following information. Each question is worth 2 marks.

NOTE:

- You can use the symbols σ , Π , etc., or the words 'PROJECT', 'RESTRICT' etc. as you prefer.
 - You do not need to try to make efficient queries – just correct ones.
 - Where you use a join, **always show the join condition**.
- a. List the first and last names of Customers who come from the City named Perth.
 - b. List the first and last names of customers who had transactions on 1st August 2020.
 - c. List the price of the item called "Back Scratcher"
 - d. List the first and last names of any customer who has purchased more than 10 "Back Scratchers" in a single transaction.
 - e. List the names and quantities of items purchased on 1st August 2020 by the customer Peter Simpson.
 - f. List the dates on which Homer Griffin made purchases.
 - g. List the first and last names of customers who have bought "Back Scratcher" or "Hair Remover"
 - h. List the first and last names of customers who have bought "Back Scratcher" but have not bought "Hair Remover"
 - i. List the first and last names of customers who have bought "Back Scratcher" and "Hair Remover"
 - j. List the first and last names of any customers who have bought all of the items listed in the Item relation. (This does not need to be as part of a single purchase).

Question 2: SQL – SELECT queries (20 marks)

This question is based on the View Ridge Gallery database you have been using in the labs. See the textbook for background to the case and the table structures.

The tables are:

ARTIST

CUSTOMER

WORK

TRANS

CUSTOMER_ARTIST_INT

You can use the dtoohey tables that we have been using. If you prefer, you can create your own copies of these tables under your own account to work with. If you do so, you should ensure you copy the same sample data and constraints as in the dtoohey tables.

Provide SQL AND result tables for the following queries. Paste the queries and the result tables from either your SSH client or SQL Developer into your assignment document. You can use a screen capture for the result tables, but NOT for the SQL.

Each question is worth 2 marks.

- a. List the full details of any work of art, including the name of the artist who created the work, that have been described as Surrealist.
- b. List the details of any work of art (including the name of the artist who created the work and the acquisition and asking price details) currently held in the gallery (*i.e. works of art that have not been sold*) with an asking price of > \$400.
- c. List the title of any work of art that has two (and only two) copies recorded in the database.
- d. List the names of all deceased artists and the age they were when they died.
- e. List the name of each artist represented in the database and the number of works of art by that artist, ordered from lowest to highest number.
- f. List the work ID, title and artist name of all the works of art that sold for more than the average price of all sales, and the price they sold for.
- g. List the total amount of sales of works of art by the gallery for each year, and the number of works sold that year.
- h. Which artist has had the most works of art sold, and how many of the artist's works have been sold?
- i. Calculate the total profit made on all the works of art that have been sold (*the profit/loss on a work of art is the difference between the acquisition price and the sales price*)
- j. List the name of any customers who have an interest in all artists.

Question 3: Further SQL (15 marks)

You have been given the following specifications of a simple database for keeping track of lectures and those who presented the lecture at a national Biological Sciences conference (note that primary keys are shown underlined, **foreign keys in bold**).

You should run your SQL to demonstrate that it works correctly, and paste in the statements used plus the output from Oracle.

PRESENTER (PresenterNo, PresenterName, Biography, InstitutionName)

LECTURE (LectureNo, LectureName, Description, Theme, Capacity, DateAndTime, **PresenterNo**)

Based on the table specifications provided, answer the following questions. Each question is worth 3 marks.

- a. Give the SQL to create the PRESENTER table. Choose appropriate data types. None of the attributes should be allowed to be null. Include the primary key constraint.
- b. Give the SQL to create the LECTURE table. Use appropriate data types, and include the primary key and foreign key constraints. Referential integrity should be set such that if a presenter is deleted from the database, any lectures that he or she is running will also be deleted.
- c. Give the SQL to add your own record to the PRESENTER table. Include your name, 'Murdoch University' as your institution, and make up a short (10-20 words) biography.
- d. Give the SQL to add the attribute VenueName to LECTURE. There are three possible venues: Building A, Building B and Building C, which should be enforced by the SQL definition.
- e. Give the SQL to record the fact that all lectures have had their capacity increased by 10.

Question 4: Normalisation (20 marks)

The following question is based upon the Patient-Treatment relation which records the details of transactions occurring in a medical surgery. You may assume the data are representative.

PATIENT ID	PATIENT NAME	PATIENT DOB	CONSULT DATE	ITEM NUMBER	ITEM DESCRIPTION	FEE	PROVIDER NUMBER	DOCTOR NAME
437	REARDON	4/08/1989	19/7/2020	A012	Standard	56	S55768	Scott
437	REARDON	4/08/1989	20/7/2020	A012	Standard	56	D42433	Mueller
562	BILSTEIN	12/05/2001	20/7/2020	A013	Short	35	S55768	Scott
361	WILSON	13/08/1997	20/7/2020	A014	Long	75	S55768	Scott
895	LE MOS	13/10/1952	20/7/2020	A013	Short	35	D42433	Mueller
895	LE MOS	13/10/1952	20/7/2020	A014	Long	75	D42433	Mueller
678	MORT	27/02/1969	20/7/2020	A014	Long	75	S55768	Scott
432	SOAPIER	1/08/1976	21/7/2020	A012	Standard	56	S55768	Scott

You have been asked to design a relational database based on this design. You know that there are problems with the current design and that it will need to be modified in order to work effectively.

You need to write a report that addresses the following:

- What are the specific problems associated with the current design and why do they arise (about ½-1 page)?
- How would you change the current design and how does your new design address the problems you have identified with the current design.

In order to receive high marks for this question, you will need to demonstrate an understanding of the theories discussed in Topics 1, 2, 3 and 4, and how they apply to this problem. **Simply providing the amended design (even if it is correct) will only attract a small percentage of the marks for this question.**

Question 5: Conceptual Design (25 marks)

You have been asked to develop a data model for the Drip Drip Water Company (DDWC). DDWC is the sole supplier of water to the citizens and businesses of the town of Drip Drip.

Each of DDWC's customers is classified as being either residential or non-residential. Each customer has at least one water meter that measures the water consumed by the customer at a particular address. Water meters must be replaced when they have been in use for 5 years.

All customers are billed monthly. Meter readers, who are employed by DDWC, are paid per meter they read. They are each allocated up to 1000 meters to read each month.

Once the reader has read the meter, the reading is recorded and the bill for that customer is calculated based on the consumption since the last reading. For example, if the last meter reading was 123580 kilolitres (kl) and the next reading was 123683kl, the consumption for that period would be 103kl.

Residential customers' bills are calculated as follows: for each residence, the customer is billed a flat rate of \$25 supply charge. They are then billed a further 24c for each kl consumed since the last reading. There are no GST or other taxes imposed.

Non-residential customers' bills are calculated as follows: there is no supply charge. They are charged only for the consumption since the last reading. They are charged 24c for the first 100 kl, 30c for the second 100 kl and 50c for each kl in excess of 200kl.

All customers are given 30 days from the billing date to pay their account. They can pay by cash, paypal, or EFT. Customers who do not pay within the 30 days will have an accounting charge of \$5 added to the bill. After 45 days a further \$10 accounting charge is added. At 60 days, the customer's water supply is cut off. Once a customer has had their supply cut off, the person or business responsible for the bill will not be able to create a new account at a different address until their arrears have been cleared.

Each customer has the right to request a report that details the historical consumption of water at the address of their account, whether they were the customer at the time or not.

What you have to do:

1. Create an entity-relationship diagram showing the data requirements of the system. Your ERD should be able to be implemented in a relational DBMS. You should use the ERD notation we have been using in the lectures, and should include a legend to explain the notation. You should include attributes in the ERD. The use of a drawing tool such as Visio will make this task easier. However, whichever tool you use, you must copy and paste the ERD into a word-processed document. This is because your tutor might not have access to the tools you have used. **Please note that hand-drawn ERDs are not acceptable.**
2. List and explain any assumptions you have made in creating the data model.

Some important things to note:

1. **Part of understanding a system at sufficient enough detail to model well, involves asking appropriate questions. If you are not sure about some detail of the case study, you should ask on the Discussion Forum in LMS.**
2. The University email server strips out any Visio (.vsd) files that are sent; even if they are included in a zip archive. So, if you want to send a draft of your design to your lecturer by email, you will need to change the extension to something other than .vsd (.blah works well) or paste the diagram into a word document.
3. Marks will be allocated to each of the following functional areas:

- Legend
- Entities
- Have you included all required entities in the design?
- Have you included specialisation hierarchies where appropriate?
- Relationships
 - Have you designed relationships between the entities that will support the functional requirements?
 - Are your relationships correctly annotated?
- Will the overall design support the enterprise requirements?

As Assignment 2 will require you to implement the DDWC database, you will need to take into account the feedback you receive on your conceptual design when commencing your logical and physical designs.