****

##### **MSc Business Analytics Examinations 2020/2021**

For internal students of Imperial College London.

This paper also forms part of the examination for the Associateship.

|  |
| --- |
| FUNDAMENTALS OF DATABASETECHNOLOGIES (BUSI97270) |

# 

**Wednesday 9th December 2020; 09:00 – 11:00 (Plus 30 minutes remote exam allowance)**

**If you experience any technical difficulties during this exam please contact the helpdesk asap via email to**[**specialisedonlineexams@imperial.ac.uk**](mailto:specialisedonlineexams@imperial.ac.uk?subject=IHM%20-%20BUSI97098%20Global%20Healthcare%20Marketing%20%20-%20Exam%20Issue)**.**

**CLOSED BOOK**

This time-limited remote assessment has not been designed to be open book. We expect you not to use any additional resources when completing this assessment. The concept of academic integrity is fundamental to your student experience at Imperial so as to ensure that your academic achievements are a true reflection of your abilities. Academic integrity means conducting all aspects of your academic life in a professional manner. The use of the work of another student, past or present, constitutes plagiarism. Giving your work to another student to use (other than in a group assessment) may also constitute an offence. The College will investigate all instances where an examination or assessment offence is reported or suspected, using plagiarism software, vivas and other tools, and apply appropriate penalties to students who are found guilty. Collusion is a form of plagiarism and will be treated in a similar manner.  This is an individual assessment and thus should be solely based on your work.

**Plagiarism: Plagiarism is the presentation of another person’s thoughts, words, images or diagrams as though they were your own. Another form of plagiarism is self-plagiarism, which involves using your own prior work without acknowledging its reuse. Plagiarism may be intentional, by deliberately trying to use another person’s work by disguising it or not citing the source, or unintentional where citation and/or referencing is incorrect. Please ensure that if you are using text memorised word for word from a source that it is either paraphrased or quoted, with the source cited.**

By submitting this assessment, you certify that you have read and understood the above definition of plagiarism and that the work presented is entirely your own, except where indicated. Random plagiarism checks will be undertaken on student work.

# CLOSED BOOK

# Instructions

Answer **ALL** questions from both Part A and Part B.

Write **ALL** of your answers on a Word document.

Assume you are using the Postgres dialect of SQL.

College approved calculators may be used.

The supplied Postgres information sheet may be used.

Questions are not equally weighted; the marks are indicated next to each question and sub-question.

**Part A: Theory (Total: 20 marks)**

**Question 1**

Give an example of a transitive property from everyday life (make no mention of databases, attributes, or normalisation).

**(2 marks)**

London is a city of United Kingdom, and United Kingdom is a country that is located on the British Isles, so London is also located on the British Isles.

**Question 2**

a) Does 1NF prevent the storage of an address in a table cell?

No, we could consider an address as a single entity, so saving the whole address in a table cell does not violate 1NF.

b) Describe another problem which addresses can pose to a database.

If we save the entire address into a table cell, and someday we need to modify the country information, we will need to update every single row of the address separately, which could result in update error. For example, we have a table which stores the addresses of residents in Puerto Rico, but someday Puerto Rico might become an official state of United States. Therefore, if we save the entire address into a slot of table, we need to modify every row of address for residents in Puerto Rico.

**(2 marks)**

**Question 3**

Briefly state the definitions of 2NF and 3NF.

**(3 marks)**

2NF: for a table to be in 2NF, it should first follow all the constraint of 1NF, so there is only 1 single and indivisible entity in every slot of the table. Moreover, all the non-prime attributes in 2NF should have the functional dependency on the whole set of candidate keys. If any of the non-prime attributes have the functional dependency on the proper subsets of candidate keys, instead of the whole set, the table is not in 2NF.

3NF: for a table to be in 3NF, it should first follow all the constraints of 2NF and 1NF. Moreover, all the non-prime attribute should have a direct functional dependency on the whole set of candidate keys. If the dependency is transitive, for example, one non-prime attribute C is depended on the whole set of candidate keys of A and B, and another non-prime attribute D is depended on non-prime attribute C. Although there is functional dependency between non-prime attribute D and the whole set of candidate keys of A and B, this dependency is transitive, and the table is not in 3NF.

**Question 4**

a) Describe two ways in which a database designer could make bad decisions

concerning attribute names.

Way 1: the mixing use of camel and snake case

Way 2: does not explicitly identify the meaning of this attributes, which might confuse the user who tries to export value from the database.

**(2 marks)**

b) Give three examples of attributes with bad datatype choices, explaining why (state the name and description of each attribute).

**(3 marks)**

|  |  |  |  |
| --- | --- | --- | --- |
|  | Name | Description | Datatype |
| Example 1: | address | This attribute stores the address of all supermarket in United States | Varchar(50) |
| Example 2 | Price | This attribute stores the price of commodities in supermarket | decimal |
| Example 3 | Purchase\_time | This attribute stores the time that a commodity is paid with the cashier | text |

Why it is bad:

* Example 1: The address might need the space more than **Varchar(50)** could store, but maximum accepted size for varchar is fixed. Therefore, **text** is preferred.
* Example 2: The type **decimal** requires the exact precision. However, the precision of price of commodities might vary because of discount. Therefore, **float**, which does not require exact precision, is preferred.
* Example 3: It is better to use **timestamp** to store any date or time. Using the datatype **timestamp**, it is easier for user to access or update the time data. For example, if a user wants to change the hours of a time data, it will be much easier that if the data is stored in **timestamp**.

**Question 5**

Imagine that you are training a group of data scientists who are familiar with spreadsheets, but have never heard of databases, SQL, or the relational paradigm. Briefly explain the advantages of an SQL database over a folder full of spreadsheets, using examples to describe what can go wrong if the relational paradigm is not applied.

**(4 marks)**

**Question 6**

a) Describe a business situation in which it is better to avoid using an SQL database. In this situation, explain the negative consequences of using an SQL database on the business. **(4 marks)**

**End of Part A**

**The paper continues on the next page (Part B)**

**Part B: Queries (Total: 30 marks)**

Consider the following entity-relationship (ER) diagram.

**Legend**

1 Up to one entity at this end of the relationship

\* Unlimited entities at this end of the relationship

highlighted Foreign key for another table



This ER diagram shows a database for a dog kennel business. Dogs stay in the kennels while their owners are on holiday. The **dogs** table describes dogs. The **customers** table describes customers. The **kennels** table describes kennels. The **bookings** table links dogs and kennels. The **payments** table describes payments made for bookings.

**Question 1**

Write a query to show the name of each dog along with the name of their owner.

**(2 marks)**

**Question 2**

Write a query to show the number of bookings for each dog.

**(2 marks)**

**Question 3**

Kennel number 7 is being closed for repairs. Write a query to show all the bookings for that kennel which are currently in progress or have not started yet, ordered by customer surname. Include a column with the cumulative total time of these bookings, ordered by start time.

**(4 marks)**

**Question 4**

Write a query to show the total capacity of the kennels.

**(2 marks)**

**Question 5**

Customers may have more than one dog. Write a query to show the total amount of time that each customer's dogs have spent (or will spend in the future) in the kennels.

**(3 marks)**

**Question 6**

Do customers with more dogs spend more money in the kennels? Write a query to show the number of dogs owned by the top 10 biggest-spending customers.

**(5 marks)**

**Question 7**

To find a customer's total spend, we have to join customers to bookings to payments to dogs. This requires doing three JOIN operations. How could you make the total spend query easier to write by denormalising the database? Describe what modifications you would make and how they could allow corruption in return for making total spend queries easier to write.

**(6 marks)**

**Question 8**

The original schema does not deal with cancelled bookings. Propose some modifications to the original schema to track cancelled bookings. In particular, discuss why just deleting bookings from the database is a bad idea.

**(6 marks)**

**End of Paper**