B6.

Consider the computer games industry scenario described below and answer the questions in part a) and b) below. The following rules describe the scenario:

- A game has a title and an age rating.
- A game has exactly one genre (for example, ‘role play’).
- A genre is described by its name.
- A game is made by a producer.
- Producers can collaborate on making one game.
- A producer can make more than one game (or might not yet have made a game).
- A producer has a name.
- A game runs on one or more platforms.
- Each platform supports one or more games.
- A platform has exactly one company making it.
- A company might create more than one platform.
- A platform has a year of introduction to the market and a name.
- A company has a name and a country for its head quarter.

a) Create an entity relationship diagram for the scenario provided using a suitable notation. Your answer must show entities with their attributes and their relations (including cardinality and optionality).

(13 marks)

b) Using your design from part a) design a set of tables. Clearly identify all primary and foreign keys and show a few (up to 3) rows of data.

(12 marks)

END OF EXAMINATION

BCS THE CHARTERED INSTITUTE FOR IT

BCS HIGHER EDUCATION QUALIFICATIONS
BCS Level 5 Diploma in IT

DATABASE SYSTEMS

Monday 24th April 2023 – Morning

Answer any FOUR questions out of SIX. All questions carry equal marks.

Time: TWO hours

Answer any Section A questions you attempt in Answer Book A
Answer any Section B questions you attempt in Answer Book B

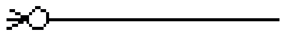
The marks given in brackets are **indicative** of the weight given to each part of the question.

Calculators are **NOT** allowed in this examination.

This page is left intentionally blank

b) The following symbols are used in ERD diagrams (crow's foot notation or Chen's notation) and relate to the cardinality and modality of associations. Explain what they mean:

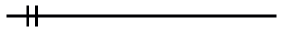
i)



or the equivalent (0,M)

(2 marks)

ii)



or the equivalent (1,1)

(2 marks)

c) Relational algebra provides a mathematical foundation to relation databases. Consider the following relational algebra statements and provide their SQL equivalent.

i) $\pi_{name,\ capital}\ \sigma_{name\ =\ \text{"Spain"}}\ countries$

(2 marks)

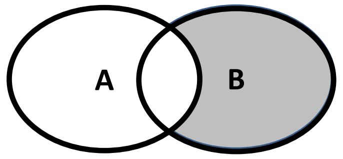
ii) $\pi_{movies.title}\ \sigma_{actor.name=\text{"Brad Pitt"}}\ (\text{movies} \times actor)$

(2 marks)

iii) $\pi_{customers.customername, orders.orderid}\ (\text{customers} \bowtie_{customers.customerid=orders.customerid}\ orders)$

(3 marks)

d) Assume tables A and B, with their respective keys AID and BID. Write the SQL query that returns all values in B that are not in A using a suitable `JOIN` notation (as indicated in the shaded section in the following diagram):



(4 marks)

[Turn Over]

Section B
Answer Section B questions in Answer Book B

B4.

a)

i) Provide the definitions for Boyce Codd Normal Form.

(3 marks)

ii) Explain the concept of ‘update anomaly’ that might occur in a database that is **not** normalised. Provide an example.

(3 marks)

b) Consider the following unnormalised table:

Full Name	Title	Address	Movies
Adam Smith	Mr	15 Park Lane	Pirates of the Caribbean, Robin Hood
Adam Smith	Mr	28 The Mall	Harry Potter, Love Actually
Berta Miller	Ms	12 Princess Street	Harry Potter, Robin Hood

i) Convert the table into 1st Normal Form.

(3 marks)

ii) Convert the table into 2nd Normal Form.

(6 marks)

iii) Convert the table into 3rd Normal Form.

(3 marks)

c) Provide the SQL create table statements required to create the set of 3rd NF tables identified in part b). Suitable data types for the attributes should be chosen. Ensure that suitable referential integrity constraints are included. You do not need to show the insertion of any data in the tables.

(7 marks)

B5.

a) Define the following concepts and provide an example for **each**:

a) Attribute.

(2 marks)

b) Tuple.

(2 marks)

c) One-to-many relationship.

(2 marks)

d) Functional dependency.

(2 marks)

e) Modality (of a relation).

(2 marks)

Section A
Answer Section A questions in Answer Book A

A1.

a) A fundamental feature of a Database Management System (DBMS) is to support data independence.

Using examples, explain how physical and logical data independence is achieved in the three level ANSI SPARC database architecture. Discuss briefly why **ONE** of the two types of data independence is harder to achieve than the other type.

(9 marks)

b) Compare and contrast the following pairs of relational database concepts:

i) Attribute vs. domain. **(4 marks)**
ii) View vs. table. **(4 marks)**
iii) Relational algebra vs. relational calculus. **(4 marks)**
iv) Referential integrity vs. entity integrity. **(4 marks)**

A2.

a) Describe briefly the terms ‘data integrity’ and ‘data recovery’ in the context of maintaining consistency of data in a database system.

(5 marks)

b) What is domain integrity? How does a database enforce domain integrity?

(5 marks)

c) Explain the concept of checkpoints and describe, using a timeline diagram, how checkpoints are used to recover a database following failure of a transaction sequence.

(6 marks)

d) The weather can cause devastating events in which a business can suffer permanent loss of IT infrastructure, particularly if it destroys computers/servers holding a database.

To avoid becoming a victim of catastrophic data loss, you have been recruited to prepare a plan of action that the business must enact before such a disaster.

Write the plan as a series of actions that the business would apply to minimise data loss and preserve data integrity.

(9 marks)

[Turn Over]

A3.

Refer to the `Products`, `Shops` and `ShopProducts` tables supplied below for this question.

Products Table

```
PRODUCTID PRODUCTNAME PRODUCTTYPE
--------- ----------- -----------
        1 Apple       Fruit
        2 Mango       Fruit
        3 Leeks       Vegetable
        4 Cauliflower Vegetable
        5 Roses       Flowers
        6 Milk        Dairy
```

Shops Table

```
    SHOPID SHOPNAME
    ------ -----------
         1 Mandana
         2 Tobeys
         3 Hansels
         4 FritzStores
```

ShopProducts table

```
SHOPID PRODUCTID QUANTITYINSTOCK
------ --------- ---------------
     1         1              10
     1         2               2
     2         2               4
     2         1               6
     2         3               2
     3         3               4
     3         2               9
```

a) Consider the following SQL query:

<u>**SQL query 1:**</u>

```
SELECT SUM(QuantityInStock)
      ,sp.ShopID
      ,s.shopname
FROM shopProducts sp
INNER JOIN shops s
ON s.shopID = sp.shopID
GROUP BY sp.shopID
        ,s.shopname;
```

i) State the overall function of SQL statement query1. Show the output produced from SQL query 1.

(2 marks)

ii) Explain the SQL keywords `SUM`, `INNER JOIN` and `GROUP BY`.

(5 marks)

iii) Explain the effect of the `ORDER BY` clause by incorporating an example `ORDER BY` clause in SQL query 1.

(2 marks)

iv) Briefly explain how a `RIGHT OUTER JOIN` or a `LEFT OUTER JOIN` differs from an `INNER JOIN`.

(2 marks)

v) Show the new result set when the `INNER JOIN` is replaced by a `RIGHT OUTER JOIN` in SQL query 1.

(4 marks)

vi) Explain what any NULL values represent.in the result set obtained in v) above.

(3 marks)

b) Consider the SQL statement in SQL query 2 below:

<u>**SQL query 2:**</u>

```
SELECT products.*
      ,shops.shopname
FROM products
INNER JOIN shopProducts
ON products.productID = shopproducts.productID
INNER JOIN shops
ON shopProducts.shopID = shops.shopID
WHERE shopProducts.QuantityInStock > 7
      AND products.productType  = 'Fruit';
```

i) Show the result set that is returned when SQL query 2 is executed.

(5 marks)

ii) Summarise what the results show about the shops and products data.

(2 marks)

[Turn Over]