Past-paper walk-through

Database Management Systems (DBMS) ▪ 8.2

eXercise

Questions in this set

- 9618/13 N25 Q5 ▪ 16 marks
- 9618/13 J25 Q6 ▪ 18 marks
- 9618/13 N24 Q4 ▪ 19 marks

Reading the mark scheme

- **B1** a mark that stands on its own – the point is either made or not.
- **M1** a *method* mark: the right approach, even if the number is wrong.
- **A1** an *answer* mark, and it usually needs its M mark first.
- **C1** a working mark on the way to the answer.
- // separates answers the examiner accepts equally.
- **ecf** = error carried forward: a later part is marked on your own earlier answer.
- **owtte** = "or words to that effect" – the wording need not match.

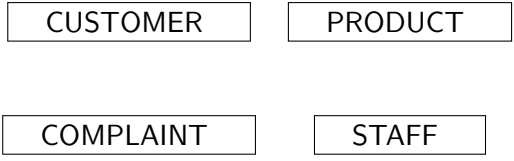
Question 5

9618/13 N25 ▪ Q5 ▪ 16 marks

A company creates a relational database to store data about its customers.
The database, REVIEWS, stores data about the customers, products, complaints and staff.
The database has four tables:
CUSTOMER(CustomerID, CustomerFirstName, CustomerLastName, CustomerEmail)
PRODUCT(ProductID, ProductName, ProductDetail, Price, Rating)
COMPLAINT(ComplaintID, ProductID, CustomerID, ComplaintDetails, StaffID)
STAFF(StaffID, StaffFirstName, StaffLastName, Department, RemoteWorker)

(a) Complete the entity-relationship (E-R) diagram for the database REVIEWS.

Question 5



[3]

Question 5

StaffID	StaffFirstName	StaffLastName	Department	RemoteWorker
1	Ralph	Jura	E	Yes
2	Luca	Emcee	A	Yes
3	Darwin	Acula	F	No

Question 5

9618/13 N25 ▪ Q5 ▪ 16 marks

A company creates a relational database to store data about its customers.
The database, REVIEWS, stores data about the customers, products, complaints and staff.
The database has four tables:

```
CUSTOMER(CustomerID, CustomerFirstName, CustomerLastName, CustomerEmail)
PRODUCT(ProductID, ProductName, ProductDetail, Price, Rating)
COMPLAINT(ComplaintID, ProductID, CustomerID, ComplaintDetails, StaffID)
STAFF(StaffID, StaffFirstName, StaffLastName, Department, RemoteWorker)
```

(b) Some example data from the STAFF table is shown.

Question 5

StaffID	StaffFirstName	StaffLastName	Department	RemoteWorker
1	Ralph	Jura	E	Yes
2	Luca	Emcee	A	Yes
3	Darwin	Acula	F	No

Write a Structured Query Language (SQL) script to define the table `STAFF`. **[4]**

Question 5

9618/13 N25 ▪ Q5 ▪ 16 marks

A company creates a relational database to store data about its customers.
The database, REVIEWS, stores data about the customers, products, complaints and staff.
The database has four tables:

```
CUSTOMER(CustomerID, CustomerFirstName, CustomerLastName, CustomerEmail)
PRODUCT(ProductID, ProductName, ProductDetail, Price, Rating)
COMPLAINT(ComplaintID, ProductID, CustomerID, ComplaintDetails, StaffID)
STAFF(StaffID, StaffFirstName, StaffLastName, Department, RemoteWorker)
```

(c) Products are given a rating between 1 and 10 inclusive.
Write an SQL script to return only the `ProductID`, `ProductName` and `ComplaintDetails` for all products with a rating of 5 or less. The results need to be displayed in descending order of rating. **[5]**

Question 5

9618/13 N25 ▪ Q5 ▪ 16 marks

A company creates a relational database to store data about its customers.
The database, REVIEWS, stores data about the customers, products, complaints and staff.
The database has four tables:

```
CUSTOMER(CustomerID, CustomerFirstName, CustomerLastName, CustomerEmail)
PRODUCT(ProductID, ProductName, ProductDetail, Price, Rating)
COMPLAINT(ComplaintID, ProductID, CustomerID, ComplaintDetails, StaffID)
STAFF(StaffID, StaffFirstName, StaffLastName, Department, RemoteWorker)
```

(d) A Database Management System (DBMS) is used to maintain and manage the database.

Describe **two** ways in which the DBMS can be used to ensure the security of the customer data.

1.

Question 5

2. **[4]**

Solution 5

(a)

Mark scheme

- 1 mark for each correct relationship, max 3 marks

Solution 5

(b) Mark scheme

- 1 mark per bullet point, max 4 marks
- ▪ Create table with opening and closing brackets and commas separating the attributes
- ▪ Appropriate data types for StaffFirstName, StaffLastName and
- Department
- ▪ Appropriate data types for StaffID and RemoteWorker
- ▪ Primary key correctly defined
- Example Answer 1:

Solution 5

- CREATE TABLE STAFF(
- StaffID INTEGER,
- StaffFirstName VARCHAR,
- StaffLastName VARCHAR,
- Department CHAR,
- RemoteWorker BOOLEAN,
- PRIMARY KEY (StaffID)
-);
- Example Answer 2:
- CREATE TABLE STAFF(

Solution 5

- StaffID INTEGER NOT NULL PRIMARY KEY,
- StaffFirstName VARCHAR,
- StaffLastName VARCHAR,
- Department CHAR,
- RemoteWorker BOOLEAN
-);
- 4
- October/November
- 2025

Solution 5

(c) Mark scheme

- 1 mark per bullet point, max 5 marks
- ▪ SELECT and the correct attributes
- ▪ FROM and the correct tables
- ▪ Tables joined correctly
- ▪ Correct condition for the rating
- ▪ Correct ORDER BY clause
- Example Answer 1
- SELECT PRODUCT.ProductID, ProductName, ComplaintDetails

Solution 5

- FROM PRODUCT, COMPLAINT
- WHERE PRODUCT.ProductID = COMPLAINT.ProductID
- AND Rating <=5
- ORDER BY Rating DESC;
- Example Answer 2
- SELECT PRODUCT.ProductID, ProductName, ComplaintDetails
- FROM PRODUCT INNER JOIN COMPLAINT
- ON PRODUCT.ProductID = COMPLAINT.ProductID
- WHERE Rating <=5
- ORDER BY Rating DESC;

Solution 5

(d) Mark scheme

- One mark for identification of method one mark for corresponding description max 4 marks
- ▪ Authentication methods / passwords / biometrics / 2-factor authentication can be implemented
- ▪ ... which prevents unauthorised access to the customer's data
- ▪ Access rights / privileges can be set
- ▪ ... so that only those with correct permissions can read / edit customer's data
- ▪ Regular backups can be scheduled

Solution 5

- ▪ ... so that a second copy of the customer's data is available in case of loss/damage
- ▪ The data can be encrypted
- ▪ ... so that the customer's data cannot be understood by anyone who gains unauthorized access
- ▪ Different views can be created
- ▪ ... so that not everyone can see the customer's data
- 4
- October/November
- 2025

Question 6

9618/13 J25 ▪ Q6 ▪ 18 marks

6 A shop sells pens to customers. Customers place an order with the shop and collect the items the next day. The shop uses a database to store the information about the orders.

The database contains the following tables:

CUSTOMER(CustomerID, CustomerName, Email)

ORDER(OrderID, CustomerID, Date, Collected)

ORDER_PRODUCT(OrderID, ProductID, Quantity)

PRODUCT(ProductID, ProductName, QuantityInBox, Cost, SupplierID)

SUPPLIER(SupplierID, SupplierName, SupplierEmail)

The primary keys are underlined.

(a) Complete the entity-relationship (E-R) diagram for the database.

Question 6

Question 6

Question 6

[4]

(b) A new product needs to be entered into the database. The product has the ID 002323, the product name 'Blue ball point 2mm', there are 50 in a box, the product costs $5.00 and the supplier has the ID SFX223.

Write a Structured Query Language (SQL) script to enter the new product into the table `PRODUCT`.

.....

.....

.....

Question 6

Question 6

.....

.....

Question 6

CustomerName	NotCollected
Jack Wright	2
Lin Cho	1
Santaya Yui	1

Question 6

.....

.....

Question 6

Identify **three** other items stored in a data dictionary.

1 .....

2 .....

3 .....

[3]

(ii) The DBMS provides a developer interface.

Explain how a database designer can make use of the developer interface.

.....

.....

.....

Question 6

Question 6

.....

.....

..... [3]

Solution 6

(a)

Mark scheme

- 1 mark for each correct relationship

Solution 6

(b) Mark scheme

- 1 mark each
- ▪ INSERT INTO PRODUCT
- ▪ VALUES with opening and closing brackets
- ▪ Inserting ProductID, ProductName and SupplierID correctly as strings including quotation marks into correct fields
- ▪ Inserting quantity and cost correctly into correct fields
- Example 1:
- INSERT INTO PRODUCT

Solution 6

- VALUES ("002323", "Blue ball point 2 mm", 50, 5.00, "SFX223");
- Example 2:
- INSERT INTO PRODUCT (ProductID, ProductName, QuantityInBox, Cost,
- SupplierID)
- VALUES ("002323", "Blue ball point 2 mm", 50, 5.00, "SFX223");
- 4
- CUSTOMER
- ORDER
- ORDER_PRODUCT
- SUPPLIER

Solution 6

- PRODUCT

Solution 6

(c) Mark scheme

- 1 mark each
- ▪ Selection of customer name and counting any field from ORDER as an appropriate identifier
- ▪ Joining tables ORDER and CUSTOMER
- ▪ AND (or WHERE) clause: Collected = FALSE
- ▪ Grouping by customer ID or customer name
- Example 1:
- SELECT CustomerName, COUNT(OrderID) AS NotCollected

Solution 6

- FROM ORDER, CUSTOMER
- WHERE ORDER.CustomerID = CUSTOMER.CustomerID
- AND Collected = FALSE
- GROUP BY CUSTOMER.CustomerID
- Example 2:
- SELECT CustomerName, COUNT(OrderID) AS NotCollected
- FROM ORDER INNER JOIN CUSTOMER
- ON ORDER.CustomerID = CUSTOMER.CustomerID
- WHERE Collected = FALSE
- GROUP BY CUSTOMER.CustomerID

Solution 6

(d)(i)

Mark scheme

- 1 mark each to max 3 e.g.
- ▪ Relationships
- ▪ Views
- ▪ Data types
- ▪ Validation rules

Solution 6

(d)(ii) Mark scheme

- 1 mark each to max 3 e.g.
- ▪ To create / modify / delete database objects
- ▪ To create a form for data input
- ▪ To add tools to a form
- ▪ for example, drop-down boxes / buttons etc.
- ▪ To design a report to show the output in an organised manner
- ▪ To add a menu to enable users to choose different actions / run different queries

Question 4

9618/13 N24 ▪ Q4 ▪ 19 marks

4 A company makes ice cream and sells it to shops.

The ice cream is made in batches: a large quantity of one type and flavour of ice cream that is then split into smaller quantities for sale.

The company's owner has designed a relational database, `ICECREAM`, to store data about their ice cream and customers.

Some of the tables in the database are given. The database is not normalised.

`BATCH(BatchID, Type, Flavour, Size, SellingPrice, EndDate)`

`CUSTOMER(CustomerID, CompanyName, EmailAddress, TelephoneNumber)`

`SALE(SaleID, BatchID, CustomerID, Quantity, Date)`

(a) Identify **two** foreign keys in the table `SALE` **and** the table that each foreign key references.

Question 4

[2]

(b) Write an SQL script to return the total quantity of ice cream sold to the customer with the ID of 0034E in the year 2023.

Question 4

(c) The table definition for `BATCH` is repeated here:

`BATCH(BatchID, Type, Flavour, Size, SellingPrice, EndDate)`

Sample data for the table `BATCH` is given:

BatchID	Type	Flavour	Size	SellingPrice	EndDate
KIV12	Plain	Vanilla	1	2.20	12/12/2024
RIC14	Plain	Chocolate	0.5	1.80	12/12/2024
TYL1	Non-dairy	Lemon	0.5	2.10	01/02/2025
FYV2	Non-dairy	Vanilla	0.25	1.50	02/02/2025
BIV13	Plain	Vanilla	1	2.20	02/02/2024

(i) Write an SQL script to define the table `BATCH`.

Question 4

Include constraints (restrictions) on the data that can be entered into each field where appropriate.

Question 4

(ii) The table `BATCH` is not normalised.

Normalise the database table `BATCH`.

Write the table definitions for your new tables.

Identify any primary and foreign keys in your tables.

Do **not** change or include the tables `CUSTOMER` and `SALE`.

Question 4

(d) Complete the following table by defining each database term.

Database term	Definition
Entity	....
Attribute	....

[2]

Question 4

Question 4

(e) A Database Management System (DBMS) supports data integrity.

Explain how a DBMS supports data integrity.

Question 4

Solution 4

(a)

Mark scheme

- 1 mark each:
- ▪
- Foreign key: BatchID, table BATCH
- ▪
- Foreign key: CustomerID, table CUSTOMER
- 2

Solution 4

(b) Worked steps

Three marks, one for each part of the query.

```
kSELECT+w kSUMp(nQuantityp)
kFROM+w   nSALE
kWHERE+w  nCustomerID+w o=+w l+s+ss0034E
kAND+w    n+nbDate+w oo=+w ol+m+mi01o/l+m+mi01o/l+m+mi2023o+w kAND+w n+nbDate+w oo
```

Solution 4

- `SELECT SUM(Quantity)` — the aggregate. The question asks for a total, so `SUM`, not `COUNT`.
- `FROM SALE WHERE` with one correct condition.
- `AND` joining the remaining conditions.

A date **range** needs two comparisons joined by `AND` — one for each end — because there is no single operator for "within this year". `BETWEEN #01/01/2023# AND #31/12/2023#` says the same thing and is equally acceptable.

Solution 4

Note there is no `GROUP BY`: the query wants **one** total for one named customer, so the `WHERE` does the selecting and the aggregate collapses what is left to a single row. `GROUP BY` would be needed only if the question asked for a total *per* customer. **Mark scheme**

- 1 mark each:
- ▪
- SELECT SUM(Quantity)
- ▪
- FROM SALE WHERE and one correct condition
- ▪
- AND with remainder correct conditions
- e.g.
- SELECT SUM(Quantity)
- FROM SALE
- WHERE CustomerID = "0034E"
- AND Date >= #01/01/2023# AND Date <= #31/12/2023#;

Solution 4

(c)(i) Worked steps

Five marks, and they map to the parts of the statement rather than to the number of columns.

```
kCREATE+w kTABLE+w nBATCHp(
+w    nBatchID+w      n+nbVARCHARp(l+m+mi10p)+w   kNOT+w kNULLp,
+w    kType+w         n+nbVARCHARp(l+m+mi20p)p,
+w    nFlavour+w      n+nbVARCHARp(l+m+mi20p)p,
+w    kSize+w         n+nbDECIMALp(l+m+mi6p,l+m+mi2p)p,
+w    nSellingPrice+w nCURRENCYp,
+w    nEndDate+w      n+nbDATEp,
+w    kPRIMARY+w kKEY+w  p(nBatchIDp)
p)p;
```

Solution 4

- `CREATE TABLE BATCH( … )` with **every column inside the brackets**.
- `BatchID`, `Type` and `Flavour` as **VARCHAR** or equivalent …
- … with a **suitable constraint**, such as `NOT NULL` on the key field.
- `Size` as **DECIMAL**, `SellingPrice` as **CURRENCY**, `EndDate` as **DATE**.
- The **primary key identified** as `BatchID`.

Solution 4

The type choices are the substance. A price is CURRENCY, not REAL, because it is fixed to two places; a quantity that can be fractional is DECIMAL; and a date is DATE so that it can be compared and ranged over — as part (b) needed. **Mark scheme**

- 1 mark for each bullet point
- ▪
- Create table BATCH with opening and closing brackets, all statements
- within brackets
- ▪
- BatchID, Type and Flavour as varchar or equivalent
- ▪
- ...with suitable constraint(s)
- ▪
- Size as decimal, SellingPrice as currency, EndDate as date (or
- equivalent)
- ▪

Solution 4

(c)(ii) Mark scheme

- 1 mark for each bullet point each (max 4)
- e.g.
- ▪
- Ice cream table with an appropriate name
- ▪
- ... containing type, flavour, size, selling price
- ▪
- ... with suitable primary key

Solution 4

- ▪
- ... foreign key identified in BATCH that links to the primary key in
- ICECREAM
- Example table definitions – not example answer:
- BATCH(BatchID, IceCreamID, EndDate)
- ICE_CREAM(IceCreamID, Type, Flavour, Size,
- SellingPrice)
- 4

Solution 4

(d) Mark scheme

- 1 mark for each definition
- Entity:
- ▪
- A real-life object that is represented as a table
- Attribute:
- ▪
- An item of data about an entity
- 2

Solution 4

(e) Worked steps

Referential integrity is a rule the DBMS enforces on relationships. Three marks from:

- **Referential integrity is enforced** by the database itself.
- Changes propagate — **cascade update and cascade delete** mean a value changed in one place is changed everywhere it appears.
- Every **foreign key must have a matching primary key**; a row cannot refer to a record that does not exist.

The consequence worth naming: it makes **orphan records impossible**. Without it, deleting a customer leaves their sales pointing at nothing, and the database is quietly inconsistent — which no query will report as an error. **Mark scheme**

Solution 4

- 1 mark for each bullet point each (max 3)
- ▪
- Referential integrity is enforced
- ▪
- ... such as cascade update/delete // if the data is changed in one place it
- is updated in every other place
- ▪
- ... and ensures each foreign key has a corresponding primary key
- 3