

Exercise walk-through

8.2

Database Management Systems (DBMS)

A-Level Computer Science

Why not just use files?

Separate files for each program duplicate data and go out of step.

duplication

the same customer stored in sales, accounts and support

inconsistency

one file is updated; the others are not

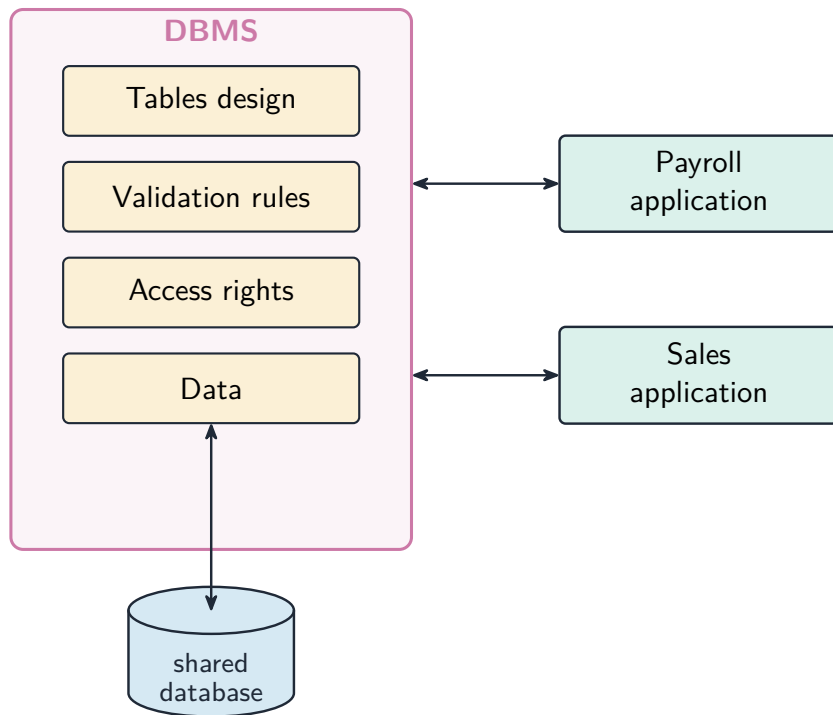
no sharing

each program owns its files – no single source of truth

hard queries

“all orders for this customer”
means writing a new program

The database approach



A **relational database** stores data in **tables**, managed by one **DBMS** that all programs share. Each fact is stored **once**.

The DBMS

A **DBMS** manages the database centrally – **data** management and **data modelling** define the **logical schema**, and it also handles **backup** and recovery, per-user security, **transactions** that wholly succeed or wholly fail, a **query processor** to run queries, and a **developer interface** of tools and APIs:

data dictionary

concurrent access

transactions

views

backup & security

query processor

What a DBMS provides, and the problems it removes

the tools

a data-dictionary editor, a query builder, a forms builder, a report generator, user management, and an SQL editor

data redundancy

file-based storage repeats the same fact in several files; a database stores it once

data inconsistency

which is what happens when one copy is updated and another is not

data integrity

constraints and keys enforced by the DBMS itself, not by each program

data security

one place to set who may read and who may write

data independence

programs are written against the logical view, so the storage can change without rewriting them

Redundancy causes inconsistency, and inconsistency destroys integrity. The three arguments are one chain, and the exam expects it stated as one.

Exercises 2.1-2.2

2.1 State what a **data dictionary** holds.

[1]

2.2 State what a **transaction** is.

[1]

Exercises 2.1-2.2

2.1 State what a **data dictionary** holds.

[1]

2.1

Worked steps

- a description of every table, field, data type and key in the database (its structure) **B1**

2.2 State what a **transaction** is.

[1]

2.2

Worked steps

- a group of operations treated as one unit — they **all succeed or all fail** **B1**

Exercises 2.3-2.4

2.3 State **two** features a DBMS provides that a file-based system does not. [2]

2.4 State what a **view** is. [1]

Exercises 2.3-2.4

2.3 State **two** features a DBMS provides that a file-based system does not. [2]

2.3

Worked steps

- any two of: data dictionary; concurrency control; transactions; central integrity rules; views; per-user security; backup/recovery **B1** **B1**

2.4 State what a **view** is. [1]

2.4

Worked steps

- a **virtual table** that shows a user a chosen slice of the data (selected rows/columns) **B1**

Exercises 3.1-3.2

3.1 Describe **two** features of a DBMS that solve problems of the file-based approach. [4]

3.2 Explain how a DBMS allows many users to work on the data at the same time **safely**. [2]

Exercises 3.1-3.2

3.1 Describe **two** features of a DBMS that solve problems of the file-based approach. [4]

3.1

Worked steps

- any two, described, e.g. **central integrity rules** — keys and constraints enforced for all programs, so data stays valid (B1) (B1)
- **concurrency control** — locks/transactions let many users edit without corrupting data (B1) (B1)

3.2 Explain how a DBMS allows many users to work on the data at the same time safely. [2]

3.2

Worked steps

- the DBMS uses **locks** and **transactions** (B1)
- two users cannot change the same data at once and each transaction completes fully or not at all (B1)

Exercises 3.3-3.4

3.3 State **two** software tools found in a DBMS and what each is used for. [2]

3.4 Describe what is meant by the DBMS feature **data dictionary** and the DBMS feature **logical schema**. [2]

Exercises 3.3-3.4

3.3 State **two** software tools found in a DBMS and what each is used for.

[2]

3.3

Worked steps

- any two with a use, e.g. query builder (build queries without writing SQL); forms designer (build data-entry screens); report generator (produce formatted output) **B1** **B1**

3.4 Describe what is meant by the DBMS feature **data dictionary** and the DBMS feature **logical schema**.

[2]

3.4

Worked steps

- **data dictionary:** data about the data — a description of every table, field, data type and key **B1**
- **logical schema:** the overall logical design of the database — its tables, fields, keys and relationships **B1**