Solutions

Each answer shows the steps, then the **result**.

2.1

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

2.2

- a group of changes treated as one unit: they **all succeed or all fail**

2.3

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

2.4

- a **virtual table** that shows a user only some of the data (chosen rows or fields)

3.1

- any two, each described, e.g.:
- data is stored **once** and shared, so there is less redundancy and no inconsistent copies
- **integrity rules** (keys, validation) are checked by the DBMS for every program, so the data stays valid
- **locking** and transactions let many users change the data without losing updates

3.2

- the DBMS uses **locks** and **transactions**
- conflicting updates are controlled so they do not overwrite each other; a transaction completes fully or not at all

3.3

- Developer interface: facilities through which the developer creates or changes database objects, for example an SQL editor used to enter a query.
- Query processor: interprets/checks and plans the query, then executes it to retrieve matching data.

3.4

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

3.5

(a) any two, each described:

- **user IDs and passwords**: only authorised staff can log in
- **access rights**: e.g. sales staff can read customer details but cannot change or delete them
- **views**: staff see only the fields they need, not e.g. payment details
- **encryption**: stolen data cannot be understood without the key
- **backups**: the data can be restored if it is lost or damaged

(b) any three: data types; field sizes; validation rules; relationships between tables; indexes; default values; access rights of users

(c) any three:

- **validation rules** are checked when data is entered, so unreasonable data is rejected
- **referential integrity**: an order cannot refer to a customer who does not exist
- **transactions**: a change either completes fully or not at all
- **record locking**: two users cannot change the same record at the same time, so no update is lost