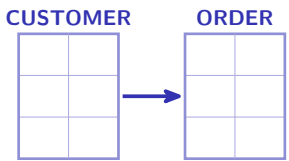
Databases

AS Computer Science ▪ Topic 8

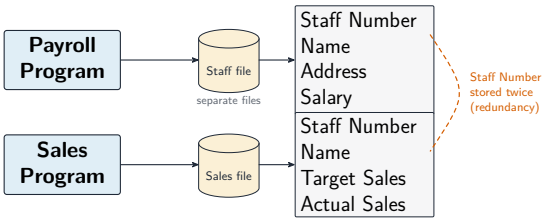
A-Level Computer Science



What we will cover

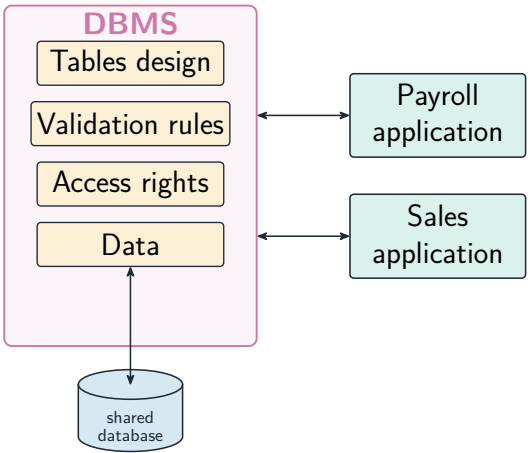
1. Files vs the database approach
2. The relational model
3. E-R diagrams
4. Normalisation
5. DBMS & SQL
6. Recap

Why not just use files?



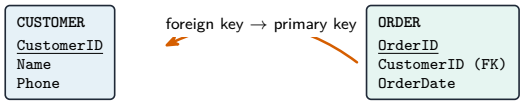
hard drives

The database approach



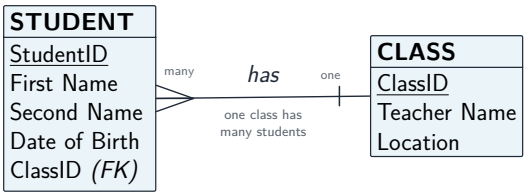
A **relational database** 关系数据库 stores data in tables, managed by one **DBMS** 数据库管理系统 that all programs share. Each fact is stored once.

The relational model



- **table** 表 of **records** 记录 (rows) and **fields** 字段 (columns)
- **primary key** 主键 – uniquely identifies a record
- **foreign key** 外键 – matches a PK in another table
- **referential integrity** 参照完整性 – no orphan FKs

Entity-relationship diagrams



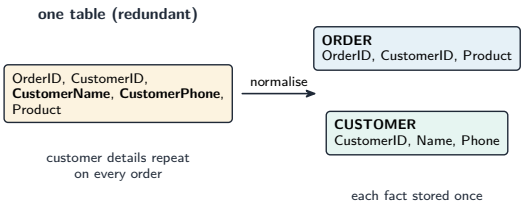
one class, many students

Cardinality 基数 at each end:

- one-to-one (1:1)
- **one-to-many** 一对多 (1:M)
- **many-to-many** 多对多 (M:N)

M:N needs a **link table** 连接表 of two foreign keys.

Normalisation



Normalisation 规范化 cuts redundancy through **normal forms** 范式:

- **1NF**: atomic fields, a key
- **2NF**: depend on the whole key
- **3NF**: no **transitive dependency** 传递依赖

Aim for 3NF – each fact stored once.

Worked example: to third normal form

Not 3NF: `TeacherName` depends on `ClassID`, not on the key –

StudentID	Name	ClassID	TeacherName
S01	Li Wei	C1	Mr Chen
S02	Anna	C1	Mr Chen
S03	Ben	C2	Ms Roy

split off the transitive fact

StudentID	Name	ClassID
S01	Li Wei	C1
S02	Anna	C1
S03	Ben	C2

FK

ClassID	TeacherName
C1	Mr Chen
C2	Ms Roy

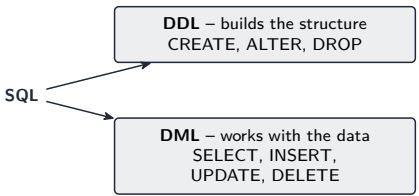
Now “Mr Chen” is stored once – renaming a teacher touches one row.

The DBMS

A **DBMS** 数据库管理系统 manages the database centrally:

- **data dictionary** 数据字典
- **concurrent access** 并发访问
- **transactions** 事务
- **views** 视图
- backup & security
- **query processor** 查询处理器

SQL – DDL and DML



- **DDL** 数据定义语言: structure
- **DML** 数据操纵语言: the data

DDL

```
CREATE TABLE CUSTOMER
(CustomerID INTEGER PRIMARY KEY,
Name VARCHAR(50) NOT NULL);
```

SQL – querying data

SELECT

```
SELECT Name, Phone
FROM CUSTOMER
WHERE City = 'London'
ORDER BY Name ASC;
```

A **join** 连接 combines tables on a foreign key; **aggregate functions** 聚合函数 (`COUNT`, `SUM`) pair with `GROUP BY`.

Always put a `WHERE` on `UPDATE`/`DELETE`, or it hits every row.

Topic 8 – key takeaways

1 Why databases

remove redundancy & inconsistency

2 Keys

primary identifies; foreign links tables

3 Normalisation

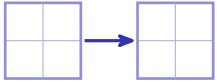
1NF $\rightarrow$ 2NF $\rightarrow$ 3NF; each fact once

4 SQL

DDL builds structure; DML queries data

Check yourself

1. What links one table to another?
2. How do you store a many-to-many relationship?
3. Which SQL keyword filters rows?
4. What does 3NF remove?



Answers **1.** a foreign key **2.** a link table **3.** `WHERE` **4.** transitive dependencies