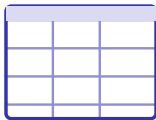


Databases

IGCSE Computer Science ■ Topic 9

IGCSE Computer Science



What we will cover

1 Tables, records & fields

2 Data types for fields

3 The primary key

4 Validation

5 SQL

6 Recap

The diagram shows a table with five columns: StudentID, FirstName, DateOfBirth, FormClass, and FeesPaid. The first two rows are highlighted in light blue. An arrow labeled 'field (a column)' points to the 'StudentID' header. An arrow labeled 'record (a row)' points to the first row. A blue arrow labeled 'primary key (unique)' points to the '1' in the 'StudentID' cell of the first row.

StudentID	FirstName	DateOfBirth	FormClass	FeesPaid
1	Amy	14/03/2009	10A	TRUE
2	Ben	02/11/2008	10B	FALSE

Records and fields

field (a column)

record (a row)

StudentID	FirstName	DateOfBirth	FormClass	FeesPaid
1	Amy	14/03/2009	10A	TRUE
2	Ben	02/11/2008	10B	FALSE

primary key
(unique)



card catalogue

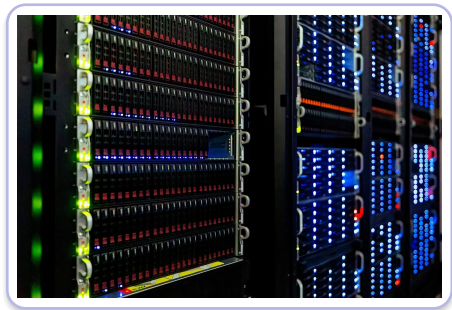
Field data types

text a name

number an age

date/time a birthday

boolean yes / no



server racks

The primary key

A **primary key** 主键 is the field that **uniquely** identifies each record – no two rows share it.

- a StudentID, an ISBN, an order number
- never blank, never repeated

A name is a **poor** key – two people share “Li Wei”. The key must be something guaranteed unique, which is why databases invent an ID field.

Validation in a database

age field: range check 0–120

enter 16 → accepted

enter 200 → rejected

The database can **reject** bad data as it is entered:

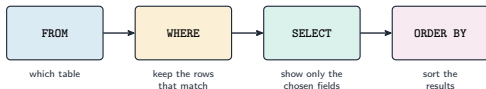
- **range** 范围 – a mark 0–100
- **type** 类型 – a number, not letters
- **presence** 存在性 – not left blank
- **format** 格式 – a date as dd/mm/yyyy

Validation stops **impossible** data at the door – but it cannot tell whether a *valid* value is the *right* one.

Structured Query Language

SQL 结构化查询语言 asks a database questions:

```
SELECT ... FROM ... WHERE ...
```



- **SELECT** the fields (columns)
- **FROM** the table
- **WHERE** a condition is met
- **ORDER BY** to sort

Read it in that order: pick columns, from a table, keeping only rows the WHERE allows.

Worked example – a query

List the name and mark of every student who scored above 80, best first.

```
SELECT Name, Mark  
FROM Students  
WHERE Mark > 80  
ORDER BY Mark DESC
```

WHERE filters **rows**; SELECT picks **columns**. DESC sorts high-to-low – leave it off and you get low-to-high.

Topic 9 – key takeaways

1 Structure

record = row (item); field = column (attribute)

2 Key

primary key uniquely identifies each record

3 Validation

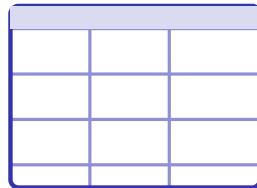
range, type, presence, format at entry

4 SQL

SELECT columns FROM table
WHERE rows

Check yourself

- 1 Is a record a row or a column?
- 2 Why is a name a poor primary key?
- 3 Which SQL clause filters rows?
- 4 Best data type for a phone number?



Answers 1. a row 2. names repeat 3. WHERE 4. TEXT