

Databases · supported focus

International Baccalaureate · IB Diploma · Computer Science · HL

Scope and prerequisites

Supported HL focus. First assessment 2027 target; official PDF returns 403; older acquired brief is final assessment 2026. Remaining guide, assessment and practical requirements retain their recorded holds.

Prerequisites: read the stated quantities and units, use arithmetic and the model conditions below. Each lesson develops its own method before independent transfer.

These are original or explicitly fictional teaching examples, not actual measurements or completed assessed learner investigations.

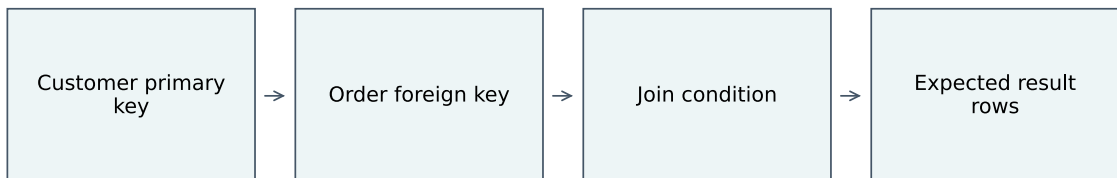
Relational databases: a key and a relationship

■ What would explain this observation?

- Duplicating a user address in every order can produce conflicting records when the address changes.
 - Start with a prediction. State the quantities or features you would compare, then decide what evidence could distinguish two explanations.
-

■ Build the model

- A relational table contains rows and attributes. A **primary key** identifies a row; a **foreign key** links to a referenced key in another table. Normalization can reduce avoidable duplication and update anomalies while preserving meaningful relationships.
- primary key: Attribute set uniquely identifying a table row; foreign key: Attribute set referencing a key in another table.



■ Choose evidence that can test it

- A join combines rows according to a specified condition. The result count depends on relationship cardinality and filters. A foreign key constraint enforces a relationship rule; it does not automatically encrypt personal data.
- Design a small synthetic user-order dataset with no real personal information. Declare keys, test duplicate and missing-reference inserts, then check a query result against a hand-worked expected table.

■ Work from known quantities

- State the known values and their units. Choose the relation because its assumptions fit this case, then rearrange before substitution.
- Known: four customers have respectively 2, 0, 3 and 1 orders. An inner join of customer to order returns $2+0+3+1 = 6$ rows. A left join also retains the customer with no order as one null-matched row, giving 7 here.

Example:

Three customers have 1, 3 and 2 orders. How many rows does the customer-order inner join return? Use the same sequence: known quantities $\rightarrow$ model $\rightarrow$ relation $\rightarrow$ substitution $\rightarrow$ unit and interpretation.

■ Check the conclusion and its limits

- A primary key need not be a person name. A join is not a simple concatenation of tables. Use parameterized queries for untrusted input rather than building SQL from raw strings.
- Return to the original observation. Explain what the result supports, which conditions it assumes, and one way to test a competing explanation.

Warn:

A foreign key automatically encrypts every row. This claim is false: A primary key need not be a person name. A join is not a simple concatenation of tables. Use parameterized queries for untrusted input rather than building SQL from raw strings.

Key:

Relational databases: a key and a relationship: A join combines rows according to a specified condition. The result count depends on relationship cardinality and filters. A foreign key constraint enforces a relationship rule; it does not automatically encrypt personal data.

■ Runnable trace and boundary

```
import sqlite3
db = sqlite3.connect(":memory:")
db.executescript("""CREATE TABLE c(id INTEGER PRIMARY KEY);
CREATE TABLE o(id INTEGER PRIMARY KEY, cid INTEGER);
INSERT INTO c VALUES(1),(2),(3);
INSERT INTO o VALUES(10,1),(11,1),(12,3);""")
for kind in ["INNER", "LEFT"]:
    sql = f"SELECT COUNT(*),COUNT(o.id) FROM c {kind} JOIN o ON
    ↪ c.id=o.cid"
    print(kind, db.execute(sql).fetchone())
db.close()
```

Expected output:

```
INNER (3, 3)
LEFT (4, 3)
```

The left join retains customer 2 with null order fields. `COUNT(*)` includes that row, while `COUNT(o.id)` excludes null. Primary keys distinguish records; customer count is not order count. The interpolated join keyword comes from a fixed internal list, not user input.