SQL

Algorithms and programming (Paper 2) · revision notes

SQL (Structured Query Language) is used to search a database table and return the records that meet a condition. You need SELECT, FROM and WHERE, with comparison operators, AND, OR and the * wildcard. Expect 'write an SQL statement' and 'state the output of this query' questions based on a table.

Key facts

- SELECT fields (comma-separated), or * for all fields
- FROM the table name
- WHERE the condition a record must meet
- Operators: = > < >= <= ; combine conditions with AND / OR
- Text values in quotation marks; numbers without
- Record = row; field = column
- Pattern: SELECT Title, Year FROM Films WHERE Year > 2020

Notes

The parts of an SQL query

- `SELECT` lists the fields to show, separated by commas. `SELECT *` shows all fields.
- `FROM` names the table to search.
- `WHERE` gives the condition a record must meet to be included.
- Conditions use = , > , < , >= and <=, and can be combined with AND and OR.
- Text values go in quotation marks (`Genre = 'Drama'`); numbers do not (`Year > 2020`).

Worked example

- Table **Films**:

FilmID	Title	Genre	Year	Minutes
F01	Nova Harbour	Sci-fi	2019	128
F02	Quiet Tides	Comedy	2021	94
F03	Deep Signal	Sci-fi	2023	141
F04	Salt Road	Drama	2020	110

- `SELECT Title, Year FROM Films WHERE Genre = 'Sci-fi' AND Minutes > 130` returns one record: Deep Signal, 2023. (Nova Harbour is Sci-fi but only 128 minutes.)
- `SELECT * FROM Films WHERE Minutes >= 128` returns all the fields of F01 and F03.
- The table has 4 records and 5 fields.

Records and fields

- Each row of a table is a **record** (all the data about one film).
- Each column is a **field** (one piece of data, such as Year).

FilmID	Title	Genre	Year	Minutes
F01	Nova Harbour	Sci-fi	2019	128
F02	Quiet Tides	Comedy	2021	94
F03	Deep Signal	Sci-fi	2023	141
F04	Salt Road	Drama	2020	110

one record

one field

A row is a record; a column is a field

- Field names in a query must match the table exactly.

How to answer each type of question

State the output of a query

2 to 3 marks | Grade 5

1. Check each record against the WHERE condition, one at a time.
2. For AND both parts must be true; for OR either part.
3. Give only the fields listed after SELECT, for every matching record.

Example. Using the Films table in the notes above, state the output of:
`SELECT Title FROM Films WHERE Genre = 'Sci-fi' OR Year > 2020` [2]

Model answer. Nova Harbour and Deep Signal (1); Quiet Tides (1), with no other titles. (Salt Road is Drama and from 2020, which is not greater than 2020.)

Write an SQL query

3 marks | Grade 6

1. SELECT the fields asked for (or * for all).
2. FROM the table named in the question.
3. WHERE the condition, using exact field names, quotation marks for text and the right operator.

Example. Using the Films table in the notes above, write an SQL statement to display the title and year of every film that is shorter than 120 minutes. [3]

Model answer.

`SELECT Title, Year` (1) `FROM Films` (1) `WHERE Minutes < 120` (1)

(This returns Quiet Tides, 2021 and Salt Road, 2020.)

Identify records and fields

1 to 2 marks | **Grade 4**

1. Count the rows of data (not the heading row) for records.
2. Count the columns for fields.

Example. Using the Films table in the notes above, state the number of records and the number of fields. [2]

Model answer. 4 records (1); 5 fields (1).

Write a query with two conditions

3 to 4 marks | **Grade 7**

1. Decide whether both conditions must be true (AND) or either one is enough (OR).
2. Write each condition in full, with its own field name.
3. Check your query against the table to make sure it returns the right records.

Example. Using the Films table in the notes above, write an SQL statement to display the FilmID and Title of every Sci-fi film made after 2020. [4]

Model answer.

`SELECT FilmID, Title` (1) `FROM Films` (1) `WHERE Genre = 'Sci-fi'` (1) `AND Year > 2020` (1)

(This returns F03, Deep Signal.)

Don’t lose marks

- Missing the commas between field names after SELECT.
- Leaving the quotation marks off text values, or putting them round numbers.
- Writing `WHERE Genre = 'Comedy' OR 'Drama'` instead of repeating the field name.
- Using AND when a record can only meet one of the conditions (a film cannot be Comedy and Drama at once).
- Getting > and >= wrong at the boundary value.

More tips

Memory tricks

- SFW: Select (what), From (where), Where (which ones).
- * means 'every field', not 'every record': WHERE still filters the records.
- Check each record against the condition one row at a time; tick the matches.

Exam technique

- Copy field and table names exactly as they appear in the question.
- Write SELECT, FROM and WHERE in capitals so each part is clear.
- In output questions, list only the fields in the SELECT line, in that order.

What each grade needs

What you need to be able to do, from the first marks up to the top grade.

Grade 4

Identify records and fields in a table

Each row is a record; each column is a field.

Grade 5

State the output of a simple query

Pick out the fields named after SELECT for every record that meets the WHERE condition.

Grade 5

Write a query using SELECT, FROM and WHERE

Choose the fields, the table and a condition such as Minutes < 120.

Grade 6

Use * to select all fields

SELECT * returns every field of each matching record.

Grade 7

Combine conditions with AND and OR

For example WHERE Genre = 'Sci-fi' AND Minutes > 130.

Free notes, Full packs, practice papers and mock exams at chhetriacademy.com. Written for this site; not taken from past papers. Stuck? 1-to-1 tutoring with a free 30-minute trial lesson.