H2 Console Access & SQL Join Reference

⊘ Access H2 Console

Open your browser and go to:

http://localhost:8080/h2-console

K H2 Console Configuration

Field Value

JDBC URL jdbc:h2:mem:testdb

Username sa

Password (leave blank)

* Ensure the following exists in your application.properties:

spring.datasource.url=jdbc:h2:mem:testdb

spring.datasource.driverClassName=org.h2.Driver

spring.datasource.username=sa

spring.datasource.password=

spring.h2.console.enabled=true

spring.jpa.hibernate.ddl-auto=update

spring.jpa.show-sql=true

Basic SELECT Queries

1. View all students

SELECT * FROM student;

2. View all courses

SELECT * FROM course;

3. View all enrollments

SELECT * FROM enrollment;

SQL Join Queries

4. INNER JOIN – Show only students who are enrolled in courses

SELECT s.id, s.name, c.title

FROM student s

JOIN enrollment e ON s.id = e.student_id

JOIN course c ON c.id = e.course id;

• 5. LEFT JOIN - Show all students even if not enrolled in any course

SELECT s.id, s.name, c.title

FROM student s

LEFT JOIN enrollment e ON s.id = e.student_id

LEFT JOIN course c ON c.id = e.course_id;

• 6. RIGHT JOIN - Show all courses even if no students are enrolled

SELECT s.id, s.name, c.title

FROM student s

RIGHT JOIN enrollment e ON s.id = e.student_id

RIGHT JOIN course c ON c.id = e.course id;

• 7. FULL JOIN (H2 workaround) – Show all students and courses with or without enrollments

⚠ H2 does not support FULL OUTER JOIN natively — simulate it using UNION.

SELECT s.id, s.name, c.title

FROM student s

LEFT JOIN enrollment e ON s.id = e.student_id

LEFT JOIN course c ON c.id = e.course_id

UNION

SELECT s.id, s.name, c.title

FROM student s

RIGHT JOIN enrollment e ON s.id = e.student_id

Usage Tips

- Use **Run** on each query in the H2 Console to test data.
- Use ORDER BY s.id to sort results if needed.
- This setup will help you **understand and explain joins in interviews** with live working examples.