TP: Basic SQL Queries and Relationships

Date: 09/10/2024

Objectives:

- Understand and practice SQL queries involving one-to-many and many-to-many relationships.
- Learn how to use JOINs, ORDER BY, and GROUP BY in SQL queries.
- Apply basic SQL operations on relational databases.

Exercise 1: One-to-Many Relationship

Consider the following two tables:

Table 1: students

student_id	name	age	class_id
1	Alice	15	1
2	Bob	16	1
3	Charlie	17	2
4	David	15	3

Table 2: classes

class_id	class_name
1	Mathematics
2	History
3	Science

1. Write a query to display each student's name along with the name of their class (use a JOIN between the students and classes tables).

2. Display the number of students in each class (use GROUP BY on class_id).

3. List all the students ordered by their age in descending order (ORDER BY on age).

Exercise 2: Many-to-Many Relationship

You are managing a database for a library, and you have the following two tables that represent books and authors, along with a junction table that connects them in a many-to-many relationship:

Table 1: books

book_id	title	published_year
1	SQL for Beginners	2020
2	Advanced SQL Queries	2021
3	Database Design	2019

Table 2: authors

author_id	name
1	John Smith
2	Jane Doe
3	Alice Johnson

Junction Table: book_authors

book_id	author_id
1	1
2	1
2	2
3	3

1. Write a query to display the titles of all books along with the names of their authors (use a JOIN between the books, authors, and book_authors tables).

2. List all the authors who have written more than one book (use GROUP BY and HAVING).

3. Write a query to display the books and their authors, ordered by the published year (ORDER BY on published_year).

Exercise 3: JOINs and Aggregation

Consider the following two tables representing a sales database:

Table 1: orders

order_id	customer_id	total_amount	order_date
1	1	500	2023-01-01
2	2	300	2023-01-03
3	1	700	2023-01-04

Table 2: customers

customer_id	name	city
1	Alice	Paris
2	Bob	Lyon
3	Charlie	Marseille

1. Write a query to display the name of each customer along with their total spending (JOIN between orders and customers).

2. Show the total amount spent by all customers in each city (GROUP BY on city).

3. Display the top 2 customers with the highest total spending (ORDER BY on total spending in descending order, and limit the result to 2).

Exercise 4: Combining Queries with Filtering

Consider the following table for employees:

employee_id	name	department_id	salary
1	Alice	1	70000
2	Bob	2	65000
3	Charlie	1	50000
4	David	2	80000

1. Write a query to display all employees who earn more than the average salary in their department (use JOIN and GROUP BY with a subquery).

2. Write a query to find the department with the highest average salary (GROUP BY on department_id and use ORDER BY to get the top result).