

Semana 3: SQL intermedio (JOINS y agregaciones)

Objetivo: Dominar JOINS (INNER, LEFT, RIGHT), funciones de agregación (COUNT, SUM, AVG) y GROUP BY, aplicándolos a una base de datos de biblioteca.

Total horas: 15 horas.

Entregable final: 5 consultas SQL con JOINS y agregaciones, implementadas en una base de datos de biblioteca, subidas a GitHub con documentación en el README.

Recursos: PostgreSQL, DBeaver, VS Code (con SQLTools), GitHub, Coursera ("SQL for Data Science"), HackerRank.

Distribución diaria (Lunes a Domingo)

Lunes: Introducción a JOINS (2.5 horas)

- Temas: Tipos de JOINS (INNER, LEFT, RIGHT), relaciones entre tablas.
- Tareas:
 - Ver video: "SQL JOINS Explained" (YouTube, FreeCodeCamp, ~30 min, sección sobre

JOINS, minutos 0:30-1:00) (~30 min).

- Continuar curso: "SQL for Data Science" (Coursera, ~1.5 horas, semana 2, módulo sobre

JOINS). Concéntrate en ejemplos de INNER y LEFT JOIN (~90 min).

- Práctica:
 - Usando la base de datos de la tienda online de la Semana 2 (clientes, productos, pedidos),

escribe 2 consultas con JOINS:

■ Ej. 1: `SELECT c.nombre, p.nombre FROM clientes c INNER JOIN pedidos pe ON c.id = pe.cliente_id INNER JOIN productos p ON pe.producto_id = p.id;` (lista clientes y productos pedidos).

■ Ej. 2: `SELECT c.nombre, pe.fecha FROM clientes c LEFT JOIN pedidos pe ON c.id = pe.cliente_id;` (clientes con o sin pedidos) (~60 min).

- Entregable: 2 consultas con JOINS guardadas en un archivo .sql (ej. semana3_consultas.sql).

- Horas: 2.5 horas.

Martes: Creación de la base de datos de biblioteca (2.5 horas)

- Temas: Diseño de base de datos con relaciones, creación de tablas.
- Tareas:
 - Ver tutorial: "PostgreSQL Tutorial for Beginners" (YouTube, FreeCodeCamp, ~20 min,

sección sobre diseño de tablas con relaciones) (~20 min).

- Diseña una base de datos para una biblioteca con 3 tablas:

■ libros (id: serial PRIMARY KEY, titulo: varchar, autor: varchar, precio: numeric).

■ usuarios (id: serial PRIMARY KEY, nombre: varchar, email: varchar).

■ prestamos (id: serial PRIMARY KEY, usuario_id: integer FOREIGN KEY, libro_id: integer FOREIGN KEY, fecha_prestamo: date) (~40 min).

- Práctica:
 - Crea la base de datos en PostgreSQL: `CREATE DATABASE biblioteca_db;` (~10 min).
 - Implementa las tablas y añade 5 registros de prueba por tabla:

■ Ej. `INSERT INTO libros (titulo, autor, precio) VALUES ('Cien años de soledad', 'Gabriel García Márquez', 20.99), ...;` (~90 min).

- Entregable: Script .sql con creación de biblioteca_db y datos de prueba, añadido a semana3_consultas.sql.

- Horas: 2.5 horas.

Miércoles: Funciones de agregación (2.5 horas)

- Temas: COUNT, SUM, AVG, GROUP BY.
- Tareas:

- Continuar curso: “SQL for Data Science” (Coursera, ~1 hora, semana 2, módulo sobre funciones de agregación) (~60 min).
- Leer: Documentación de PostgreSQL sobre funciones de agregación (postgresql.org/docs, ~15 min).
- Práctica:
 - Escribe 2 consultas con agregaciones usando biblioteca_db:
 - Ej. 1: `SELECT COUNT(*) AS total_prestamos FROM prestamos;` (número total de préstamos).
 - Ej. 2: `SELECT usuario_id, COUNT(*) AS libros_prestados FROM prestamos GROUP BY usuario_id;` (préstamos por usuario) (~75 min).
 - Entregable: 2 consultas con agregaciones añadidas a semana3_consultas.sql.
 - Horas: 2.5 horas.
- Jueves:** Combinar JOINS y agregaciones (2.5 horas)
 - Temas: Uso de JOINS con GROUP BY para reportes.
 - Tareas:
 - Ver video: “SQL GROUP BY and Aggregations” (YouTube, canal Tech With Tim, ~20 min) (~20 min).
 - Continuar curso: “SQL for Data Science” (Coursera, ~1 hora, semana 2, ejemplos de JOINS con agregaciones) (~60 min).
 - Práctica:
 - Escribe 1 consulta combinada para biblioteca_db:
 - Ej.: `SELECT u.nombre, COUNT(p.id) AS prestamos FROM usuarios u LEFT JOIN prestamos p ON u.id = p.usuario_id GROUP BY u.nombre;` (número de préstamos por usuario, incluyendo usuarios sin préstamos) (~70 min).
 - Entregable: 1 consulta combinada añadida a semana3_consultas.sql (total: 5 consultas).
 - Horas: 2.5 horas.
- Viernes:** Introducción a HackerRank (2 horas)
 - Temas: Resolución de problemas SQL intermedios.
 - Tareas:
 - Crea una cuenta en hackerrank.com (~10 min).
 - Resuelve 3 problemas SQL de nivel medio en la sección SQL (ej. “Weather Observation Station”, “Employee Salaries”) (~110 min).
 - Práctica:
 - Aplica conceptos de JOINS y agregaciones en los problemas.
 - Guarda las soluciones en un archivo .sql separado (ej. hackerrank_semana3.sql).
 - Entregable: 3 soluciones de HackerRank en hackerrank_semana3.sql.
 - Horas: 2 horas.
- Sábado:** Práctica intensiva en HackerRank (3 horas)
 - Temas: Consolidar JOINS y agregaciones con más problemas.
 - Tareas:
 - Resuelve 7 problemas SQL adicionales en HackerRank (nivel medio, ej. “Top Earners”, “Aggregation Queries”) (~150 min).
 - Práctica:
 - Revisa tus soluciones en HackerRank y compáralas con las discusiones para optimizar tu código (~30 min).
 - Entregable: 7 soluciones adicionales en hackerrank_semana3.sql, totalizando 10 problemas resueltos.

- Horas: 3 horas.

Domingo: Consolidación y GitHub (2.5 horas)

- Temas: Revisión de consultas, documentación en GitHub.
- Tareas:
 - Revisa todas las consultas de la semana (semana3_consultas.sql y

hackerrank_semana3.sql) en DBeaver para verificar su funcionamiento (~30 min).

- Actualiza tu repositorio GitHub:

■ Añade los archivos semana3_consultas.sql (5 consultas) y hackerrank_semana3.sql (10 soluciones) (~30 min).

- Actualiza el README.md con una sección para la Semana 3, incluyendo:

- Resumen de lo aprendido (JOINS, COUNT, SUM, AVG, GROUP BY).

- Estructura de biblioteca_db (tablas y relaciones).

- Enlace a los problemas de HackerRank resueltos (~60 min).

• Práctica: Escribe una consulta adicional para biblioteca_db que combine JOIN y GROUP BY (ej. SELECT l.titulo, COUNT(p.id) AS veces_prestado FROM libros l LEFT JOIN prestamos p ON l.id = p.libro_id GROUP BY l.titulo;) (~30 min).

- Entregable: Repositorio GitHub actualizado con scripts SQL y README.
- Horas: 2.5 horas.

Resumen de la Semana 3

- Total horas: 15 horas (Lunes: 2.5, Martes: 2.5, Miércoles: 2.5, Jueves: 2.5, Viernes: 2, Sábado: 3, Domingo: 2.5).
- Entregables:
 - Script semana3_consultas.sql con 5 consultas SQL (2 JOINS, 2 agregaciones, 1 combinada).
 - Script hackerrank_semana3.sql con 10 soluciones de problemas HackerRank.
 - Base de datos biblioteca_db con 3 tablas y datos de prueba.
 - Repositorio GitHub actualizado con README detallado.
- Notas:
 - Usa PostgreSQL local en tu MacBook para biblioteca_db, manteniendo el uso de RAM bajo.
 - Si tienes dificultades con HackerRank, consulta las discusiones de cada problema o la documentación de PostgreSQL.
 - Guarda los scripts en una carpeta (ej. /Documentos/BasesDatos/Semana3) y respáldala en Google Drive.

Instrucciones adicionales

- Seguimiento: Registra el tiempo y tareas completadas en tu documento de Google Docs o Notas para mantener el control. Añade esta distribución a tu calendario.
 - PDF: Copia este texto y pégalo en el documento de tu calendario (Google Docs/Word) para incluirlo como parte de la Semana 3. Usa viñetas y encabezados claros antes de exportar a PDF.
 - Soporte: Si necesitas ayuda con algún paso (ej. crear biblioteca_db, escribir consultas con JOINS, o usar HackerRank), dime y te guío paso a paso.
- ¡Mucho ánimo con el inicio de la Semana 1 mañana! ¿Quieres que detalle otra semana (ej. Semana 4) o necesitas ayuda con algo específico de la Semana 3, como configurar HackerRank o una consulta SQL concreta?