

- Calendario de 6 meses para experta en bases de datos

- Objetivo: Convertirte en experta en bases de datos (relacionales, NoSQL, nube) en 6 meses.

Duración: 24 semanas (6 meses), ~15-20 horas semanales.

Equipo: MacBook Air 2022 (M2, 8 GB RAM, 512 GB SSD).

Entregables clave: 3 proyectos en GitHub (relacional, NoSQL, ETL/nube), certificación (opcional), CV, perfil LinkedIn.

Recursos: PostgreSQL, MySQL, MongoDB Atlas, AWS Free Tier, DBeaver, VS Code, Draw.io, Python, GitHub.

- 

- Meses 1-2: Fundamentos y SQL avanzado (Semanas 1-8)

- Objetivo: Dominar conceptos básicos, SQL y diseño inicial de bases de datos relacionales.

Horas semanales: ~15-20 horas (10 horas teoría/práctica, 5-10 horas proyectos).

- Semana 1: Introducción a bases de datos y configuración

- - Temas: Modelos de datos (relacional vs. no relacional), estructura de bases de datos (tablas, claves primarias/foráneas).

- - Tareas:

- - - Instalar PostgreSQL y MySQL (brew install postgresql mysql).

- - - Configurar DBeaver y VS Code con SQLTools.

- - - Ver “Database Fundamentals” (YouTube, FreeCodeCamp, ~2 horas).

- - - Leer: Normalización (1NF, 2NF, 3NF) en “Database System Concepts” (cap. 1-2, ~50 páginas).

- - Práctica: Crear base de datos simple en PostgreSQL (ej. tabla de usuarios).

- • Entregable: Base de datos con 2-3 tablas en GitHub.

- • Horas: 15 horas.

- Semana 2: SQL básico

- • Temas: SELECT, INSERT, UPDATE, DELETE, WHERE, LIKE, ORDER BY.

- • Tareas:

- • ○ Curso: “SQL for Data Science” (Coursera, ~5 horas, semana 1).

- • ○ Resolver 10-15 problemas SQL en LeetCode (nivel fácil).

- • Práctica: Consultas para insertar y filtrar datos (ej. usuarios mayores de 18 años).

- • Entregable: 10 consultas SQL en GitHub.

- • Horas: 15 horas.

- Semana 3: SQL intermedio (JOINS y agregaciones)

- • Temas: JOINS (INNER, LEFT, RIGHT), COUNT, SUM, AVG, GROUP BY.

- • Tareas:

- • ○ Curso: “SQL for Data Science” (Coursera, ~5 horas, semana 2).

- • ○ Resolver 10-15 problemas SQL en HackerRank (nivel medio).

- • Práctica: Crear base de datos para biblioteca (libros, usuarios, préstamos). Consultas con JOINS.

- • Entregable: 5 consultas con JOINS y agregaciones en GitHub.

- • Horas: 15 horas.

- Semana 4: Proyecto inicial y normalización

- • Temas: Normalización hasta 3NF, diagramas ER básicos.

- • Tareas:

- - Leer: Normalización en “SQL Performance Explained” (cap. 1, ~20 páginas).
- - Crear diagrama ER para tienda online (usuarios, productos, pedidos) en Draw.io.
- - Práctica: Implementar base de datos de tienda online en PostgreSQL, normalizada hasta 3NF.
- - Entregable: Diagrama ER y base de datos en GitHub.
- - Horas: 20 horas.
- Semana 5: SQL avanzado (subconsultas y transacciones)
- - Temas: Subconsultas, CTEs, transacciones (BEGIN, COMMIT, ROLLBACK).
- - Tareas:
- - Curso: “Advanced SQL” (DataCamp, ~5 horas).
- - Resolver 10 problemas SQL avanzados en LeetCode.
- - Práctica: Consultas con subconsultas para tienda online (ej. productos más vendidos).
- - Entregable: 5 consultas avanzadas en GitHub.
- - Horas: 15 horas.
- Semana 6: Introducción a índices
- - Temas: Índices (B-Tree, únicos), impacto en rendimiento.
- - Tareas:
- - Curso: “Database Design and Basic SQL in PostgreSQL” (Udemy, ~3 horas, índices).
- - Crear índice en tabla de pedidos en PostgreSQL.
- - Práctica: Comparar tiempos de consultas con y sin índices.
- - Entregable: Informe breve (1 página) sobre impacto de índices.
- - Horas: 15 horas.

---

- Semana 7: Proyecto intermedio

---

- - Temas: Consolidar SQL y diseño.
- 

- - Tareas:
- 

- - Diseñar base de datos para sistema de reservas (pacientes, doctores, citas).
- 

- - Escribir consultas para reportes (citas por doctor, pacientes por día).
- 

- - Práctica: Implementar en PostgreSQL con índices y transacciones.
- 

- - Entregable: Base de datos y 5 consultas en GitHub.
- 

- - Horas: 20 horas.
- 

- Semana 8: Revisión y práctica intensiva

---

- - Tareas:
- 

- - Revisar normalización, JOINS, índices.
- 

- - Resolver 20 problemas SQL (LeetCode/HackerRank, fácil a medio).
- 

- - Práctica: Optimizar consultas lentas del proyecto de reservas.
- 

- - Entregable: Repositorio GitHub actualizado.
- 

- - Horas: 15 horas.
- 

- 

---

- Meses 3-4: Diseño, optimización y NoSQL (Semanas 9-16)

---

- Objetivo: Dominar diseño avanzado, optimización de consultas y bases de datos no relacionales.
- 

Horas semanales: ~15-20 horas.

---

- Semana 9: Diseño avanzado de bases de datos

---

- - Temas: Modelado ER avanzado, UML, desnormalización.
- 

- - Tareas:
-

- - Curso: “Database Design and Basic SQL in PostgreSQL” (Udemy, ~5 horas, diseño).
- 

- - Leer: Desnormalización en “SQL Performance Explained” (~20 páginas).
- 

- - Práctica: Crear diagrama ER para app de eventos (eventos, asistentes, organizadores).
- 

- - Entregable: Diagrama ER en Draw.io.
- 

- - Horas: 15 horas.
- 

- Semana 10: Optimización de consultas

---

- - Temas: Planes de ejecución (EXPLAIN), optimización de JOINS e índices.
- 

- - Tareas:
- 

- - Curso: “Advanced SQL” (DataCamp, ~5 horas, optimización).
- 

- - Usar EXPLAIN en PostgreSQL para analizar consultas del proyecto de eventos.
- 

- - Práctica: Optimizar 3 consultas lentas con índices o reescritura.
- 

- - Entregable: Informe con planes de ejecución antes/después.
- 

- - Horas: 15 horas.
- 

- Semana 11: Transacciones avanzadas

---

- - Temas: Propiedades ACID, niveles de aislamiento, bloqueos.
- 

- - Tareas:
- 

- - Curso: “Database Systems” (Stanford Online, videos sobre transacciones, ~3 horas).
- 

- - Implementar transacciones en PostgreSQL (ej. transferencia bancaria).
- 

- - Práctica: Simular conflicto de transacciones y resolverlo.
- 

- - Entregable: Script SQL con transacciones.
- 

- - Horas: 15 horas.
- 

- Semana 12: Introducción a NoSQL (MongoDB)

---

- Temas: Bases de datos de documentos, colecciones, consultas en MongoDB.

- Tareas:

- ○ Curso: “MongoDB Basics” (MongoDB University, gratuito, ~5 horas).

- ○ Instalar MongoDB (brew install mongodb-community) o usar MongoDB Atlas.

- Práctica: Crear colección para blog (posts, usuarios, comentarios).

- Entregable: Base de datos MongoDB con 3 consultas.

- Horas: 15 horas.

- Semana 13: Proyecto NoSQL

- Temas: Diseño NoSQL vs. relacional.

- Tareas:

- ○ Diseñar base de datos NoSQL para app de recomendaciones (ej. películas).

- ○ Implementar en MongoDB Atlas.

- Práctica: Consultas para buscar películas por género/usuario.

- Entregable: Base de datos NoSQL en GitHub.

- Horas: 20 horas.

- Semana 14: Bases de datos en la nube (AWS)

- Temas: AWS RDS, DynamoDB, configuración en la nube.

- Tareas:

- ○ Curso: “Introduction to AWS RDS” (AWS Skill Builder, gratuito, ~3 horas).

- ○ Configurar PostgreSQL en AWS RDS (Free Tier).

- Práctica: Migrar base de datos de tienda online a RDS.

- Entregable: Base de datos en RDS funcional.



- • Horas: 15 horas.

- Semana 15: Pipelines ETL

- • Temas: Extract, Transform, Load con Python.

- • Tareas:

- • ○ Instalar bibliotecas Python: pip install pandas sqlalchemy pymongo.

- • ○ Curso: “Data Engineering with Python” (DataCamp, ~5 horas, ETL).

- • Práctica: Script Python para extraer datos de API (JSONPlaceholder), transformarlos y cargarlos en PostgreSQL.

- • Entregable: Script ETL en GitHub.

- • Horas: 15 horas.

- Semana 16: Proyecto integrado

- • Temas: Combinar relacional, NoSQL y nube.

- • Tareas:

- • ○ Diseñar app híbrida (e-commerce con productos en PostgreSQL, recomendaciones en MongoDB).

- • ○ Implementar pipeline ETL en AWS.

- • Entregable: Proyecto completo en GitHub con documentación.

- • Horas: 20 horas.

- 

- Meses 5-6: Habilidades avanzadas y preparación profesional (Semanas 17-24)

- Objetivo: Dominar administración, Big Data, certificaciones y construir portafolio.

Horas semanales: ~15-20 horas.

- Semana 17: Administración de bases de datos

- - Temas: Backups, restauración, monitoreo.
- - Tareas:
  - - Curso: “Database Administration” (Udemy, ~5 horas).
    - Configurar backup automatizado en PostgreSQL.
- - Práctica: Simular restauración desde backup.
- - Entregable: Script de backup/restauración.
- - Horas: 15 horas.

- Semana 18: Introducción a Big Data

- - Temas: Snowflake, BigQuery, integración con bases de datos.
- - Tareas:
  - - Curso: “Introduction to BigQuery” (Google Cloud, gratuito, ~3 horas).
    - Cargar dataset en BigQuery (GCP, créditos gratuitos).
- - Práctica: Escribir consultas SQL en BigQuery.
- - Entregable: Consulta BigQuery en GitHub.
- - Horas: 15 horas.

- Semana 19: Preparación para certificación

- - Temas: AWS Certified Database Specialty.
- - Tareas:
  - - Curso: “AWS Certified Database Specialty” (Udemy, ~10 horas).
    - Practicar con AWS Free Tier (RDS, DynamoDB).
- - Práctica: Configurar réplica en RDS y probar failover.



- • Entregable: Notas de estudio para certificación.

- • Horas: 20 horas.

- Semana 20: Proyecto final (portafolio)

- • Temas: Consolidar habilidades.

- • Tareas:

- ◦ Diseñar base de datos para gestión de inventarios con reportes en la nube.

- ◦ Integrar SQL, NoSQL, ETL y visualización (Tableau Public).

- • Entregable: Proyecto completo en GitHub con README.

- • Horas: 20 horas.

- Semana 21: Portafolio y networking

- • Temas: Presencia profesional.

- • Tareas:

- ◦ Crear portafolio en GitHub con 3 proyectos (relacional, NoSQL, ETL/nube).

- ◦ Escribir artículo en LinkedIn sobre un proyecto.

- ◦ Unirse a r/Database (Reddit), #SQL en X.

- • Entregable: Perfil de GitHub y LinkedIn actualizados.

- • Horas: 15 horas.

- Semana 22: Preparación para entrevistas

- • Temas: Preguntas técnicas, simulaciones.

- • Tareas:

- ◦ Resolver 20 problemas SQL avanzados en LeetCode.

- ◦ Practicar preguntas de diseño (ej. "Diseña base de datos para Uber").

- • Práctica: Simular entrevista técnica (Pramp.com).

- • Entregable: Lista de respuestas a 10 preguntas de entrevistas.

- • Horas: 15 horas.

- Semana 23: Aplicaciones laborales

- • Tareas:

- • ○ Aplicar a 10-15 puestos (Administrador de Bases de Datos, Ingeniero de Datos) en

LinkedIn/Indeed.

- • ○ Personalizar CV con proyectos y habilidades.

- • Entregable: CV actualizado, 5 aplicaciones enviadas.

- • Horas: 15 horas.

- Semana 24: Cierre y aprendizaje continuo

- • Tareas:

- • ○ Revisar proyectos y optimizarlos.

- • ○ Seguir Percona Database Blog, asistir a meetup virtual (Meetup.com).

- • ○ (Opcional) Tomar examen AWS Certified Database Specialty (~\$150).

- • Entregable: Plan de aprendizaje continuo.

- • Horas: 15 horas.

- 

- Resumen

- • Total horas: ~360-480 horas.

- • Costos estimados: \$0-\$200 (Udemy: \$15-\$30, certificación: \$150).

- • Notas:

- - Usa AWS Free Tier, MongoDB Atlas, Google Cloud (créditos gratuitos) para minimizar carga en tu MacBook.

- - Sube todos los entregables a GitHub.

- - Registra avances semanales en un documento para seguimiento.

- 

- Instrucciones para crear el PDF en tu MacBook Air

- - 1 Copia el texto: Selecciona y copia el calendario anterior.

- - 2 Pega en una herramienta:

- - Google Docs:

- - Abre docs.google.com y crea un nuevo documento.

- - Pega el texto, ajusta fuentes (ej. Arial, tamaño 11) y márgenes (1 cm).

- - Ve a Archivo > Descargar > PDF.

- - Microsoft Word:

- - Abre Word, pega el texto, organiza con encabezados (Título 1 para meses, Título 2 para semanas).

- - Ve a Archivo > Guardar como > Formato: PDF.

- - Notas (macOS):

- - Abre la app Notas, crea una nueva nota y pega el texto.

- - Ve a Archivo > Exportar como PDF.

- - 3 Formato sugerido:

- - Usa negritas para encabezados (semanas, temas, tareas).

- - Añade viñetas para tareas y entregables.

- - Divide en páginas por mes para mayor claridad.

- 4 Guarda el PDF: Almacénalo en tu MacBook (ej. carpeta /Documentos/

Calendario\_BasesDatos.pdf) y respáldalo en Google Drive.

- Si prefieres, puedo ajustar el formato del texto (ej. más compacto, con viñetas diferentes) para que sea más fácil de convertir, o puedo detallar alguna semana específica si necesitas más granularidad.

¿Quieres que modifique algo antes de que lo conviertas a PDF?