

Tarea 1 definición y análisis de SGBD

Instrucciones:

1. Tipos de SGBD a investigar: Deberás buscar información sobre al menos cinco (5) SGBD pertenecientes a los siguientes tipos:

- Relacionales (ejemplo: MySQL, PostgreSQL).
- NoSQL (ejemplo: MongoDB, Cassandra).
- Orientados a grafos (ejemplo: Neo4j, ArangoDB).
- Almacenamiento en la nube (ejemplo: Amazon RDS, Google BigQuery).
- Otros (por ejemplo: orientados a objetos, distribuidos, u otros tipos especializados).

2. Aspectos a investigar: Para cada SGBD, responde las siguientes preguntas:

- ¿Cuál es su modelo de datos principal (relacional, NoSQL, grafos, etc.)?
- ¿Cuáles son sus principales características técnicas?
- ¿En qué casos de uso se recomienda utilizarlo?
- ¿Qué ventajas tiene frente a otros SGBD similares?
- ¿Cuáles son sus limitaciones o desventajas?

3. Formato del informe: Presenta la información en un cuadro comparativo con las siguientes columnas:

- Nombre del SGBD
- Tipo de SGBD
- Modelo de datos
- Características principales
- Ventajas
- Limitaciones
- Casos de uso recomendados

4. Entrega:

- El cuadro comparativo debe estar acompañado de una conclusión personal de máximo 200 palabras, explicando cuál de los SGBD investigados te parece más interesante o útil, y por qué.

Nombre del SGBD	Tipo de SGBD	Modelo de datos	Características principales	Ventajas	Limitaciones	Casos de uso recomendados
MySQL	Relacional	Tablas	soporte	fácil de usar,	Limitado en	Aplicaciones

		relacionales	completo para SQL, compatibilidad con múltiples plataformas, permite replicación y clustering.	ampliamente adoptado, rendimiento alto en consultas estructuradas, gran comunidad de soporte,	funcionalidades NoSQL, menor rendimiento para grandes volúmenes de datos no estructurados.	web, comercio electrónico, sistemas ERP.
MongoDB	NoSQL	Documentos JSON/BSON	almacenamiento flexible en documentos JSON, escalabilidad horizontal, consultas ad-hoc, soporte para esquemas dinámicos.	Ideal para datos no estructurados, manejo eficiente de grandes volúmenes de datos, flexible en cambios de esquema	Menos eficiente en consultas complejas y transacciones, mayor consumo de espacio en disco.	Aplicaciones móviles, IoT, gestión de big data.
Neo4j	Orientado a grafos	Grafos	modelo basado en nodos, relaciones y propiedades, soporte para consultas complejas con Cypher, transacciones ACID, visualización de grafos integrada.	optimizado para análisis de redes y grafos, consultas rápidas en estructuras interconectadas, fácil visualización.	No ideal para datos tabulares, escalabilidad horizontal limitada.	análisis de redes sociales, motores de recomendación, detección de fraude.
Amazon RDS	Almacenamiento en la nube	Relacional (columnar)	data warehouse totalmente administrado, optimización para análisis de big data, consultas SQL estándar, integración con ecosistema de Google Cloud.	consultas extremadamente rápidas para grandes volúmenes, sin necesidad de infraestructura, modelo de pago por uso.	limitado a datos en Google Cloud, costos incrementales por uso intensivo, menor personalización que soluciones locales.	análisis de big data, machine learning, informes de datos en tiempo real.

Google BigQuery	Almacenamiento en la nube	Relacional (columnar)	data warehouse totalmente administrado, optimización para análisis de big data, consultas SQL estándar, integración con ecosistema de Google Cloud.	consultas extremadamente rápidas para grandes volúmenes, sin necesidad de infraestructura, modelo de pago por uso.	limitado a datos en Google Cloud, costos incrementales por uso intensivo, menor personalización que soluciones locales.	análisis de big data, machine learning, informes de datos en tiempo real.
ArangoDB	Orientado a grafos	Grafos, documentos, clave-valor	soporte multi-modelo (grafos, documentos, clave-valor), lenguaje de consulta AQL, escalabilidad horizontal, sistema de transacciones ACID.	flexibilidad para manejar múltiples modelos de datos en un solo sistema, consultas poderosas y optimización automática.	Configuración más compleja, menor adopción en la comunidad comparado con Neo4j o MongoDB.	Aplicaciones multi-modelo, análisis de grafos, sistemas híbridos.
PostgreSQL	Relacional	Tablas relacionales	compatible con SQL estándar y extensiones, soporte para JSON/JSONB, robustez en transacciones, indexación avanzada, funciones definidas por el usuario.	alta capacidad para consultas complejas, escalabilidad, buena integración con datos semiestructurados.	configuración inicial más compleja, menor rendimiento en operaciones simples comparado con MySQL.	análisis de datos, aplicaciones empresariales, sistemas GIS.