

PREGUNTAS “Práctica 01”

Equipo “Luk”

¿Qué otros SMBD existen actualmente en el mercado?

Existen muchos Sistemas de Gestión de Bases de Datos (SMBD) en el mercado. Algunos de los más conocidos son:

- **MySQL:** Amplia popularidad, especialmente en aplicaciones web, ahora bajo la propiedad de Oracle.
- **Oracle Database:** Una base de datos robusta y escalable, usada comúnmente en grandes empresas para aplicaciones críticas.
- **Microsoft SQL Server:** Muy utilizado en entornos corporativos, especialmente en combinación con aplicaciones de Microsoft.
- **MongoDB:** Un SMBD NoSQL basado en documentos, popular en aplicaciones que manejan grandes volúmenes de datos no estructurados.
- **SQLite:** Ligero y utilizado comúnmente en aplicaciones móviles y pequeñas aplicaciones de escritorio.
- **Redis:** Un SMBD NoSQL basado en memoria, ideal para aplicaciones que requieren baja latencia.
- **Firebase:** Una base de datos NoSQL basada en la nube, desarrollada por Google, utilizada principalmente en aplicaciones móviles.

2. ¿Cuáles son las principales diferencias con PostgreSQL?

PostgreSQL es conocido por ser un SMBD relacional muy completo y extendido. Algunas diferencias clave en comparación con otros SMBD son:

- **MySQL vs. PostgreSQL:**
 - **Conformidad con el estándar SQL:** PostgreSQL se apega más estrictamente a los estándares SQL, lo que lo hace más adecuado para aplicaciones que requieren funcionalidades avanzadas de SQL.
 - **Extensibilidad:** PostgreSQL es altamente extensible, permitiendo a los usuarios definir tipos de datos personalizados, funciones y operadores.
 - **Soporte de transacciones:** PostgreSQL tiene un sistema de transacciones ACID robusto, con soporte nativo para transacciones anidadas (a través de puntos de guardado).
 - **Desempeño:** MySQL puede ser más rápido para lecturas simples, mientras que PostgreSQL es más eficiente en operaciones complejas y consultas que requieren manipulación de datos.
- **Oracle vs. PostgreSQL:**

- **Costo:** Oracle es una solución comercial con licencias costosas, mientras que PostgreSQL es open source y gratuito.
- **Características empresariales:** Oracle ofrece un conjunto muy amplio de características empresariales, como particionamiento avanzado, optimización automática, y replicación en tiempo real, aunque muchas de estas características también están disponibles en PostgreSQL a través de extensiones.
- **Escalabilidad y soporte técnico:** Oracle es conocido por su capacidad de manejar bases de datos extremadamente grandes y por su soporte técnico de clase mundial.
- **MongoDB vs. PostgreSQL:**
 - **Modelo de datos:** MongoDB es una base de datos NoSQL que almacena datos en documentos JSON, lo que le permite manejar datos no estructurados, mientras que PostgreSQL sigue un modelo relacional basado en tablas.
 - **Escalabilidad horizontal:** MongoDB está diseñado para escalar horizontalmente más fácilmente que PostgreSQL, aunque PostgreSQL también puede escalar con la configuración adecuada.
 - **Flexibilidad de esquema:** MongoDB permite cambios en el esquema sin necesidad de migraciones complejas, lo que puede ser ventajoso en entornos donde el esquema de datos evoluciona rápidamente.

3. ¿Por qué una empresa debería escoger una base de datos open source?

- **Costo:** Las bases de datos open source, como PostgreSQL o MySQL, no tienen costos de licencias, lo que reduce significativamente el gasto, especialmente para startups y pequeñas empresas.
- **Flexibilidad:** Las soluciones open source permiten una mayor personalización y adaptación a las necesidades específicas de la empresa, ya que el código fuente está disponible para su modificación.
- **Comunidad y soporte:** Las bases de datos open source suelen tener comunidades activas que contribuyen al desarrollo y ofrecen soporte gratuito, además de recursos educativos abundantes.
- **Independencia de proveedores:** Las empresas no están atadas a un solo proveedor, lo que permite evitar el “vendor lock-in” y da mayor libertad para cambiar o adaptar la infraestructura según sea necesario.
- **Innovación rápida:** Las tecnologías open source tienden a evolucionar rápidamente, incorporando nuevas características y mejoras con mayor agilidad en comparación con soluciones propietarias.

4. ¿Cuáles son las ventajas, para un DBA el trabajar con un SMBD, open source?

- **Acceso al código fuente:** Permite a los DBA comprender a fondo cómo funciona la base de datos, personalizarla y solucionar problemas específicos que podrían no ser abordados en un SMBD propietario.
- **Amplia documentación y recursos:** Los SMBD open source suelen tener una abundante cantidad de recursos de aprendizaje, incluyendo foros, blogs, y tutoriales, lo que facilita la resolución de problemas.
- **Flexibilidad en la elección de herramientas:** Los DBA tienen la libertad de elegir herramientas de administración, monitoreo y optimización, ya que no están limitados a las soluciones propietarias que puedan venir con un SMBD comercial.
- **Rapidez en la adopción de nuevas tecnologías:** Los SMBD open source tienden a adoptar nuevas tecnologías y metodologías más rápido que las soluciones propietarias, lo que permite a los DBA mantenerse a la vanguardia en cuanto a prácticas de administración de bases de datos.
- **Colaboración y contribución:** Los DBA pueden contribuir a la comunidad, desarrollando y compartiendo sus propias mejoras o extensiones, lo que no solo beneficia a su organización, sino también al ecosistema en general.

5. ¿Qué son las bases de datos NoSQL? Menciona 3 ventajas y desventajas contra las bases relacionales.

Las bases de datos NoSQL son sistemas de gestión de bases de datos que no se basan en el modelo relacional tradicional de tablas y SQL. En su lugar, utilizan diferentes modelos de almacenamiento como documentos, grafos, columnas o pares clave-valor para manejar grandes volúmenes de datos, datos no estructurados, y aplicaciones distribuidas.

Ventajas de las bases de datos NoSQL:

1. **Escalabilidad horizontal:** NoSQL está diseñado para escalar fácilmente añadiendo más servidores, lo que es ideal para aplicaciones que requieren manejar grandes volúmenes de datos distribuidos.
2. **Flexibilidad de esquema:** NoSQL permite almacenar datos sin un esquema fijo, lo que facilita adaptarse a cambios en la estructura de datos sin necesidad de realizar migraciones complejas.
3. **Alto rendimiento para ciertas cargas de trabajo:** NoSQL puede ofrecer un rendimiento superior en aplicaciones que requieren grandes cantidades de lectura y escritura, especialmente cuando se trata de datos no estructurados o semi-estructurados.

Desventajas de las bases de datos NoSQL:

1. **Consistencia limitada:** Muchas bases de datos NoSQL sacrifican la consistencia a favor de la disponibilidad y la partición (según el teorema CAP), lo que puede llevar a problemas de consistencia eventual en los datos.
2. **Falta de estándares y herramientas:** NoSQL carece de un estándar unificado como SQL, lo que puede dificultar la migración de datos entre diferentes sistemas y la disponibilidad de herramientas de administración y monitoreo.
3. **Curva de aprendizaje:** Las bases de datos NoSQL pueden requerir una nueva forma de pensar en el diseño de aplicaciones, lo que implica una curva de aprendizaje para los desarrolladores y administradores acostumbrados a las bases de datos relacionales.

REFERENCIAS BIBLIOGRÁFICAS

- Elmasri, R., & Navathe, S. B. (2016). *Fundamentals of database systems* (7th ed.). Pearson.
- Connolly, T., & Begg, C. (2015). *Database systems: A practical approach to design, implementation, and management* (6th ed.). Pearson.
- Redmond, E., & Wilson, J. R. (2012). *Seven databases in seven weeks: A guide to modern databases and the NoSQL movement*. Pragmatic Bookshelf.
- Oracle Corporation. (n.d.). *Oracle Database documentation*. Oracle. <https://docs.oracle.com/en/database/>
- MySQL AB. (n.d.). *MySQL documentation*. Oracle. <https://dev.mysql.com/doc/>
- Stonebraker, M., & Özsu, M. T. (2011). *Readings in database systems* (5th ed.). MIT Press.