

1. ¿Qué se entiende por software libre considerando aspectos como gratuidad, código fuente y uso comercial?

1. **Gratuidad:** Aunque el software libre es comúnmente gratuito, "software libre" no significa necesariamente que no tenga costo. La libertad del software libre se refiere más a la libertad de uso, modificación y distribución que a su precio.
2. **Código fuente:** El software libre permite el acceso al código fuente, lo que significa que cualquier usuario puede ver, modificar y mejorar el software según sus necesidades. Esto promueve la transparencia y la colaboración entre desarrolladores.
3. **Uso comercial:** El software libre puede utilizarse en entornos comerciales. Empresas y desarrolladores pueden vender servicios relacionados con el software o incluso empaquetarlo y venderlo, siempre que respeten las licencias de software libre, como la GPL (Licencia Pública General).

2. Lista al menos 3 ventajas e inconvenientes de los productos de pago respecto a los libres.

1. **Ventajas de los productos de pago:**
2. **Soporte técnico dedicado:** Los productos de pago suelen ofrecer soporte técnico profesional, lo que ayuda a resolver problemas rápidamente.
3. **Actualizaciones regulares:** Normalmente, los productos de pago reciben actualizaciones constantes y mantenimiento por parte de su empresa desarrolladora.
4. **Seguridad y calidad controladas:** Al ser software desarrollado profesionalmente, suele ser probado exhaustivamente para garantizar su seguridad y rendimiento.
5. **Inconvenientes de los productos de pago:**
6. **Costo inicial y recurrente:** El software de pago requiere inversión inicial y, en muchos casos, pagos periódicos para seguir utilizando el producto.
7. **Dependencia del proveedor:** Al estar cerrado el código fuente, se depende del proveedor para mejoras o reparaciones.
8. **Limitaciones en personalización:** A diferencia del software libre, el software de pago suele ser menos flexible en términos de personalización y modificación del código.

3. ¿Qué tiene que ver la administración de un SGBD con el diseño de bases de datos?

La **administración de un Sistema de Gestión de Bases de Datos (SGBD)** está relacionada con el diseño de bases de datos en que ambas actividades deben colaborar para optimizar el funcionamiento, seguridad y rendimiento de la base de datos. Mientras que el diseño de bases de datos implica definir la estructura lógica y física, la administración se centra en gestionar el SGBD para asegurar que la base de datos funcione correctamente, esté segura y que sus recursos se usen eficientemente. Un buen diseño facilita la administración al estructurar los datos de forma eficiente, lo que reduce problemas de rendimiento y mantenimiento.

4. Cita al menos 3 ventajas de usar bases de datos frente a los tradicionales sistemas de ficheros.

1. **Redundancia y duplicación mínima:** Las bases de datos minimizan la duplicación de datos, centralizando la información en un sistema gestionado.
2. **Integridad y consistencia de datos:** Un SGBD permite reglas y restricciones que mantienen la consistencia de los datos, algo difícil de gestionar en sistemas de ficheros.
3. **Acceso concurrente seguro:** Las bases de datos permiten que múltiples usuarios accedan a los datos simultáneamente sin comprometer la integridad de la información.

5. Enumera al menos tres objetos típicos de una base de datos indicando su función.

1. **Tabla:** Estructura básica de una base de datos donde se almacenan los datos en filas (registros) y columnas (campos).
2. **Índice:** Objeto que mejora la velocidad de recuperación de datos, permitiendo un acceso más rápido a registros específicos sin recorrer toda la tabla.
3. **Vista:** Es una consulta almacenada que muestra datos de una o varias tablas de forma simplificada y personalizada para el usuario.

6. ¿Qué es una base de datos distribuida?

Una **base de datos distribuida** es un sistema en el que los datos están almacenados en varios servidores o ubicaciones físicas, pero funcionan como una única base de datos. Este tipo de sistema permite acceder a los datos de forma distribuida, lo que mejora el acceso, la disponibilidad y la tolerancia a fallos. Es común en aplicaciones de alta disponibilidad y sistemas que necesitan replicación de datos en diferentes ubicaciones geográficas.

7. Indica resumidamente las fases involucradas en el desarrollo de una base de datos desde su concepción hasta su puesta en marcha.

- **Recolección de requisitos:** Definir las necesidades de información y las funcionalidades requeridas por el sistema.
- **Diseño conceptual:** Crear un modelo conceptual de datos, como el modelo entidad-relación, para estructurar la información.
- **Diseño lógico:** Transformar el modelo conceptual en un modelo lógico compatible con el SGBD a usar (como modelo relacional).
- **Diseño físico:** Definir la estructura física y los detalles de almacenamiento, como índices y particiones.
- **Implementación:** Crear la base de datos en el SGBD y cargar los datos iniciales.
- **Pruebas:** Realizar pruebas para garantizar que la base de datos cumpla con los requisitos y funcione correctamente.
- **Puesta en marcha:** Implementar la base de datos en producción y comenzar su uso oficial.

8. ¿Para qué sirve un disparador en un SGBD?

Un **disparador** (trigger) en un SGBD es un procedimiento que se ejecuta automáticamente en respuesta a ciertos eventos en una tabla o vista, como una inserción, actualización o eliminación de registros. Los disparadores se utilizan para mantener la integridad de los datos, automatizar tareas repetitivas o registrar cambios en el sistema.

9. Explica con tus palabras qué es el diccionario de datos en un SGBD.

El **diccionario de datos** es una base de datos interna dentro del SGBD que almacena información sobre la estructura y organización de la base de datos. Incluye metadatos, como nombres de tablas, columnas, tipos de datos, restricciones y relaciones entre las tablas. Este diccionario es esencial para la administración del sistema, ya que ayuda a los usuarios y administradores a entender y gestionar la estructura de los datos.

10. Eres administrador de la base de datos. Indica un problema y su posible solución que te pueda surgir considerando dos casos:

- **Caso 1:** Base de datos con miles de usuarios y centrada en consultas, como un buscador.
- **Problema:** Alta carga de consultas, lo que puede ralentizar el rendimiento.
- **Solución:** Implementar **caché de consultas** y optimizar las consultas más frecuentes. Además, añadir índices en campos consultados frecuentemente puede reducir tiempos de respuesta.
- **Caso 2:** Base de datos de venta online con miles de usuarios y operaciones por segundo.
- **Problema:** Contención de datos y problemas de bloqueo cuando múltiples transacciones intentan acceder o modificar la misma información.
- **Solución:** Implementar técnicas de **aislamiento de transacciones** como el bloqueo de filas en lugar de tablas completas y optimizar el uso de índices para minimizar los bloqueos. Además, la **replicación de base de datos** puede ayudar a distribuir la carga.