**EXERCÍCIOS**

**1)Qual a importância dos bancos de dados para os Sistemas de Informação?**

**R:** De grande importância, visto que armazena informações necessárias para o funcionamento correto para tais sistemas.

**2) Umas das características de um SGBD é o gerenciamento de transações. O que é uma transação no contexto de banco de dados?**

**R:** É um conjunto de operações que são tratados como uma unidade logica indivisível, ou seja, todas as operações devem ser concluídas com êxito, caso o contrario o banco retorna ao estado inicial e a transação não é concluída.

**3) Explique cada uma das propriedades ACID do gerenciamento de transações do SGBD. (ACID)**

**R:**

**Atomicidade:** Todas as operações devem ser executadas com sucesso caso contrário a transação será anulada e o banco volta a seu estado anterior.

**Consistência:** Durante uma transação o banco de dados deve estar em um estado consistente, ou seja, informações sem divergência e ao final de uma transação bem sucedida o estado do banco ainda deve ser consistente, ou seja, o resultado da transação deve ser coerente com a solicitação.

**Exemplo:**

Saldo bancário:

Zezinho da silva: R$200,00

Execute uma operação de saque no valor de R$ 100,00:

Novo saldo de Zezinho da silva: R$100,00

O estado se manteve consistente mesmo após a transação já que a solicitação foi correspondida com sucesso.

**Isolamento:** A operação de uma transação só pertence a ela mesmo que haja operações simultâneas acontecendo, se mantendo isolada.

**Durabilidade:** Após o commit ser executado com sucesso as alterações são permanentes sem chance de retorno ao estado anterior

**4) Quando um SGBD executa um rollback?**

**R:** Quando alguma operação da transação não é executada com sucesso o banco retorna ao estado inicial(rollback).

**5) Explique com suas palavras cada uma das características do SGBD.**

**R:** De acordo com as que foram apresentadas no slide temos:

**Controle de redundância:** Garante que os dados sejam minimamente duplicados já que podemos utilizar um único dado em vários locais do banco o que de certa forma evita inconsistências no banco evitando falhas de transações por exemplo. Uma parte boa disso é que só é necessário atualizar a informação do dado uma única vez que onde ele for utilizado vai ser alterado também.

**Controle de concorrência:** Quando há múltiplos acessos ao BD, até mesmo advindos de diferentes contextos da aplicação, o controle de concorrência atua garantindo que as informações cheguem sem divergências e sem falhas que resultem em problemas como a anomalia de leitura que acontece basicamente por ter diversas solicitações de um único dado acarretando que em alguma dessas transações o valor do dado enviado não corresponde ao que se tem realmente no banco pelo fato da transação estar lendo um dado que esta nesse exato momento sendo comitado por outra transação que ainda não validou o commite.

**Controle de acesso:** Oferece segurança e confiabilidade aos usuários do banco de dados já que nos SGBD os acessos são estritamente controlados e divididos em diferentes hierarquias que vão desde o usuário que vai inserir até o que vai ter permissão de acessar e alterar os dados inseridos.

**Controle de integridade:** Está se referindo ao pacote de medidas e tipos de mecanismos que estão em um determinado SGBD, que procura garantir que os dados que estão lá permaneçam corretos mantendo a consistência, evitando falhas de transações tornando assim aquele bd confiável e integro antes, durante e após seu uso.

**Backup:** É uma das ferramentas cruciais para um SGBD, já que mantem uma imagem de um momento em que o BD estava trabalhando sem falhas garantindo que mesmo após acidentes e catástrofes inesperadas o BD possa ser restaurado a uma versão funcional do mesmo.

**6) Quais as principais arquitetura de SGBD? Aponte vantagens e desvantagens de cada um deles.**

**R:**

**Arquitetura distribuída:**

**Vantagens:** são extremamente escalonáveis e tem um sistema de redundância absurdo visto que os servidores estão espalhados em vários locais e a queda de um não significa perda de comunicação como um todo.

**Desvantagens:** Difícil manter todas as partes(servidores) sincronizados e funcionando além de que o custo é muito alto para implementar e manter.

**Arquitetura stand-alone:**

**Vantagens:** É fácil de implementar, é barata, não necessita de nada externo.

**Desvantagens:** É muito difícil expandir o banco, já que fica limitado ao equipamento que vai ser usado, se o BD falhar não tem onde ir buscar redundância podendo até mesmo perder o BD, colaboração acaba se tornando limitada já que não é o ideal e nem suporta milhares de acessos ou transações simultâneas.

**Arquitetura centralizada:**

**Vantagens:** É simples de gerencias e manter as manutenções já que o mesmo encontra se em um único servidor central, quando se trata de pequenos sistemas pode manter um bom desempenho e fica fácil de implementar e manter politicas de segurança por se tratar de um único servidor.

**Desvantagens:** Acaba se tornando um ambiente de escalabilidade limitada justamente pela dificuldade que é escalonar para grandes fluxos de dados e acesso, não existe redundância naturalmente pois está centralizado em um único servidor e sofrem com sobrecarga se ainda assim vários usuários efetuarem uma solicitação ao mesmo tempo.

**Arquitetura cliente-servidor:**

**Vantagens:** Divide a carga de processamento com o cliente já que parte da solicitação é justamente executada na maquina do cliente, é amplamente escalonável e lida naturalmente com quantidades maiores de usuários e solicitações se comparada com a arquitetura anterior além de permitir o acesso de diferentes maneiras, como exemplo aplicações web.

**Desvantagens:** Alta complexidade quanto a manutenção e gerenciamento, maiores riscos que atentam contra a segurança e integridade dos dados, já que trafega entre cliente e servidor e necessita de uma conexão de rede solida e eficiente para que não haja falha no trafego desses dados.