

Banco de Dados

Definição: Armazena dados de forma estruturada, assim facilitando o acesso.

Tipos de Dados: Textos, Dados numéricos, gráficos e outros.

Estrutura: Organizados Normalmente em tabelas

Sistema Gerenciador de Banco de Dados (SGBD)

Definição: É um software que proporciona uma interface entre os usuários e o banco de dados, permitindo o gerenciamento dos dados de forma eficiente e segura.

Funções:

- **Criação e Manutenção:** Permite a criação de bancos de dados, bem como a manutenção contínua dos dados armazenados.
- **Consulta e Atualização:** Facilita a consulta aos dados e a atualização das informações existentes.
- **Segurança:** Implementa medidas de segurança para proteger os dados contra acesso não autorizado.
- **Integridade:** Garante a integridade e a consistência dos dados.
- **Recuperação:** Oferece ferramentas para a recuperação de dados em caso de falhas ou perda.

Relação entre Banco de Dados e SGBD

Interdependência: O banco de dados depende do SGBD para funcionar de maneira eficaz. O SGBD gerencia todas as operações de acesso, manipulação e gerenciamento dos dados no banco de dados.

Eficiência: O SGBD otimiza a maneira como os dados são armazenados e recuperados, melhorando a performance do sistema.

O Modelo Relacional de Banco de Dados

É uma abordagem poderosa e amplamente utilizada para organizar e manipular dados. Ele foi proposto por Edgar F. Codd em 1970 e é baseado na teoria matemática dos conjuntos e na lógica de predicados.

Conceitos Básicos do Modelo Relacional

- **Tabela (ou Relação):** A tabela é a estrutura básica de armazenamento de dados no modelo relacional. Cada tabela é uma coleção de linhas (ou registros) e colunas (ou campos). Cada coluna representa um atributo dos dados, e cada linha representa uma instância do conjunto de atributos.
- **Tupla:** Uma tupla é uma única linha em uma tabela, representando um registro específico.
- **Atributo:** Um atributo é uma coluna em uma tabela, representando uma característica dos dados armazenados na tabela.
- **Chave Primária:** Um atributo ou conjunto de atributos que identifica unicamente cada tupla em uma tabela.
- **Chave Estrangeira:** Um atributo ou conjunto de atributos em uma tabela que referencia a chave primária de outra tabela, criando uma relação entre as duas tabelas.
- **Esquema:** O esquema é a definição da estrutura do banco de dados, incluindo tabelas, colunas, tipos de dados e relações entre as tabelas.

Propriedades do Modelo Relacional

- **Integridade de Dados:** O modelo relacional impõe regras de integridade que garantem a consistência e a precisão dos dados. Por exemplo, uma chave estrangeira deve sempre referenciar uma chave primária existente.
- **Independência dos Dados:** Os usuários podem interagir com os dados de maneira abstrata, sem precisar conhecer os detalhes físicos de armazenamento.
- **Flexibilidade:** É fácil adicionar, modificar ou excluir dados sem afetar a estrutura geral do banco de dados.

Linguagem de Manipulação de Dados

- **SQL (Structured Query Language):** SQL é a linguagem padrão para interação com bancos de dados relacionais. Ela permite realizar operações como criação de tabelas, inserção, atualização, exclusão e consulta de dados.

Micro serviços

São uma abordagem arquitetônica para o desenvolvimento de software, onde uma aplicação é estruturada como um conjunto de serviços pequenos e independentes que comunicam entre si. Cada micro serviço é responsável por uma funcionalidade específica e é projetado para ser altamente desacoplado dos outros.

Conceitos Principais dos Micro Serviços

- **Desacoplamento:** Micro serviços são independentes uns dos outros, o que facilita a manutenção e a escalabilidade.
- **Funcionalidade Específica:** Cada micro serviço tem uma responsabilidade clara e bem definida.

- **Comunicação:** Micro serviços se comunicam uns com os outros através de APIs ou mensagens. Protocolos comuns incluem HTTP/REST, gRPC, e mensagens assíncronas.
- **Autonomia:** Cada micro serviço pode ser desenvolvido, implementado e escalado de forma independente.
- **Desenvolvimento Ágil:** Facilita a implementação de práticas ágeis, permitindo que equipes menores e multifuncionais trabalhem em diferentes partes da aplicação de forma simultânea.

Benefícios dos Micro Serviços

- **Escalabilidade:** Fácil de escalar serviços individuais conforme a necessidade, ao invés de escalar toda a aplicação.
- **Resiliência:** Falhas em um serviço específico não necessariamente afetam toda a aplicação.
- **Flexibilidade Tecnológica:** Diferentes serviços podem ser desenvolvidos com diferentes tecnologias e linguagens de programação.
- **Facilidade de Atualização:** Atualizações e novas funcionalidades podem ser implementadas rapidamente, sem impactar a aplicação inteira.

Desafios dos Micro Serviços

- **Gerenciamento de Dados:** Manter a consistência dos dados entre diferentes serviços pode ser complexo.
- **Comunicação Inter-Serviços:** Garantir que a comunicação entre os serviços seja eficiente e segura.
- **Monitoramento e Depuração:** Monitorar, depurar e solucionar problemas pode ser mais difícil devido à natureza distribuída da aplicação.

- **Desempenho:** Pode haver uma sobrecarga de desempenho devido à comunicação entre serviços.

Exemplos de Uso de Micro Serviços

- **Netflix:** Utiliza micro serviços para oferecer serviços de streaming escaláveis e resilientes.
- **Amazon:** Usa micro serviços para gerenciar suas diversas funcionalidades, desde o catálogo de produtos até o processamento de pagamentos.
- **Spotify:** Utiliza micro serviços para gerenciar diferentes aspectos da experiência do usuário, como recomendações de música e gestão de playlists.

Ferramentas e Tecnologias Comuns

- **Docker:** Utilizado para contêineres que isolam os micro serviços.
- **Kubernetes:** Para orquestrar e gerenciar contêineres.
- **API Gateway:** Para gerenciar a comunicação entre os clientes e os micro serviços.

Os micro serviços são uma abordagem poderosa para o desenvolvimento de aplicações modernas, especialmente quando se busca escalabilidade, flexibilidade e resiliência.