***OBJETIVO***

“Compreender o papel dos Bancos de Dados apresentando SQL e NoSQL”

***INTRODUÇÃO***

Banco de Dados ou B**ase de dados** é um arquivo de texto visualizado em formato de tabela, sendo um repositório de informações, se encontram armazenadas em um sistema de computador, organizadas e estruturadas de acordo com o que for estabelecido pela empresa. Os mais utilizados são formatados em linhas e colunas de tabelas, possibilitando assim o processamento e a consulta das informações ali presentes mais fáceis.

Nessas plataformas é possível realizar as seguintes ações, caso necessário:

* Acessar;
* Modificar;
* Atualizar;
* Controlar;
* Organizar;
* Gerenciar.

***SQL***

SQL é uma linguagem padrão de comunicação com bancos de dados relacionais, significa “Structured Query Language”, ou “Linguagem de Consulta Estruturada”. É uma linguagem de programação para lidar com banco de dados relacional (baseado em tabelas). Criado para que vários desenvolvedores pudessem acessar e modificar dados de uma empresa simultaneamente, de maneira descomplicada e unificada Ela é uma linguagem declarativa e que não necessita de profundos conhecimentos de programação para que alguém possa começar a escrever queries, as consultas e pedidps, que trazem resultados de acordo com o que você está buscando, pode ser aprendida por profissionais que não são necessariamente desenvolvedores, mas trabalham com bancos de dados com frequência, é a linguagem universal, usada por cientistas de dados para manipular bancos de dados por meio dos SGBDs.

A programação SQL pode ser usada para analisar ou executar tarefas em tabelas, principalmente através dos seguintes comandos: inserir (‘insert’), pesquisar (‘search’), atualizar (‘update’) e excluir (‘delete’). Porém, isso não significa que o SQL não possa fazer coisas mais avançadas, como escrever queries (comandos de consulta) com múltiplas informações.

 A linguagem SQL é utilizada de maneira relativamente parecida entre os principais bancos de dados relacionais do mercado: Oracle, MySQL, MariaDB, PostgreSQL, Microsoft SQL Server, entre muitos outros.

***NoSQL***

O termo 'NoSQL' se refere a tipos não relacionais de bancos de dados, e esses bancos de dados armazenam dados em um formato diferente das tabelas relacionais, permitindo que os dados sejam armazenados de maneiras mais intuitivas e fáceis de entender, ou mais próximas da maneira como os dados são usados pelos aplicativos - com menos transformações necessárias ao armazenar ou recuperar usando APIs no estilo NoSQL. Podendo aproveitar ao máximo a nuvem para oferecer tempo de inatividade zero. Nele são criados para modelos de dados específicos e têm esquemas flexíveis para a criação de aplicativos modernos. Amplamente reconhecidos por sua facilidade de desenvolvimento, funcionalidade e performance em escala, eles naturalmente aceitam um paradigma de desenvolvimento ágil, adaptando-se rapidamente aos requisitos em constante mudança, mediante o relaxamento de algumas restrições de consistência de dados dos outros bancos.

Somente em meados dos anos 2000 que outros modelos de dados começaram a ser adotados e ter um uso mais significativo. Para diferenciar e categorizar essas novas classes de bancos e modelos de dados, o termo “NoSQL” foi criado. Muitas vezes, o termo “NoSQL” é usado de forma intercambiável com “não relacional”.

Bancos de dados relacionais x Bancos de dados NoSQL

Cargas de trabalho ideais

* Bancos de dados relacionais são projetados para aplicativos transacionais e fortemente consistentes de processamento de transações online (OLTP) e são bons para processamento analítico online (OLAP).
* Os bancos de dados do NoSQL são projetados para vários padrões de acesso aos dados que incluem aplicativos de baixa latência. Os bancos de dados de pesquisa NoSQL são projetados para análise de dados semiestruturados.

Modelo de dados

* O modelo relacional normaliza dados em tabelas, compostas por linhas e colunas. Um esquema define estritamente tabelas, colunas, índices, relações entre tabelas e outros elementos do banco de dados. O banco de dados impõe a integridade referencial nos relacionamentos entre as tabelas.
* Os bancos de dados NoSQL fornecem uma variedade de modelos de dados, como chave-valor, documento e gráfico, que são otimizados para performance e escala.

Propriedades ACID

* Bancos de dados relacionais fornecem propriedades de atomicidade, consistência, isolamento e durabilidade (ACID):
* A atomicidade exige uma transação para executar completamente ou não é executada de forma alguma.
* A consistência exige que, quando uma transação é confirmada, os dados devem estar em conformidade com o esquema do banco de dados.
* O isolamento exige que as transações simultâneas sejam executadas separadamente umas das outras.
* A resiliência exige a capacidade de se recuperar de uma falha do sistema ou falta de energia inesperada para o último estado conhecido.
* Os bancos de dados NoSQL geralmente fazem compensações relaxando algumas das propriedades ACID dos bancos de dados relacionais para um modelo de dados mais flexível que pode ser escalado horizontalmente. Isso torna os bancos de dados NoSQL uma excelente opção para casos de uso de baixa latência e alta taxa de transferência que precisam ser escalados horizontalmente além das limitações de uma única instância.

Performance

* A performance normalmente depende do subsistema do disco. A otimização de consultas, índices e estrutura de tabela é necessária para alcançar máxima performance.
* A performance geralmente é uma função do tamanho do cluster do hardware subjacente, da latência de rede e do aplicativo que faz a chamada.

Escala

* Os bancos de dados relacionais geralmente escalam verticalmente o tamanho ao aumentar os recursos de computação do hardware, ou escalam horizontalmente o tamanho ao adicionar réplicas para cargas de trabalho somente leitura.
* Os bancos de dados NoSQL normalmente são particionáveis porque os padrões de acesso podem escalar horizontalmente o tamanho usando arquitetura distribuída para aumentar a taxa de transferência que fornece performance consistente em escala quase ilimitada.

APIs

* As solicitações para armazenar e recuperar dados são comunicadas usando consultas compatíveis com uma Structured Query Language (SQL – Linguagem de consultas estruturadas). Essas consultas são analisadas e executadas pelo banco de dados relacional.
* APIs baseadas em objetos permitem que desenvolvedores de aplicativos armazenem e restaurem facilmente estruturas de dados. As chaves de partição permitem que os aplicativos procurem pares de chave-valor, conjuntos de colunas ou documentos semiestruturados que contenham objetos e atributos de aplicativos serializados.

***Conclusão***

A melhor linguagem para o banco de dados não existe, mas sim a linguagem de acordo com a ferramenta que o profissional precisa. A mais atualizada no mercado nem sempre será o melhor para a empresa, então a indicação é encontrar qual se adequa mais aos padrões, ramo e porte e disponibilidade financeira da instituição.

Referencia

<https://kenzie.com.br/blog/sql/#:~:text=Para%20que%20serve%20o%20SQL,s%C3%A3o%20alguns%20adeptos%20do%20SQL>. - visitado em 14/09/2022.

[https://www.tecmundo.com.br/software/146482-sql-que-ele-serve.htm - visitado em 14/09/2022](https://www.tecmundo.com.br/software/146482-sql-que-ele-serve.htm%20-%20visitado%20em%2014/09/2022).

<https://www.certifiquei.com.br/banco-dados/> - Atualizado em 12.1.21 as 22:01

<https://aws.amazon.com/pt/nosql/> - visitado em 14/09/2022

<https://www.oracle.com/br/database/nosql/what-is-nosql/> - visitado 14/09/2022