

IPI UC5 – Banco de dados: Manipulação de arquivos e gerenciamento de dados

Modelos de banco de dados e linguagens

[WILSON]

Olá!

Tudo bem?

Estamos falando diretamente da rádio Senac EAD.

Hoje, vamos ajudar você a entender e a desvendar os maravilhosos mistérios da informática.

Eu, tutor Wilson Cardoso, sou o âncora do programa. Também temos a participação do tutor Ronald Ramos.

Olá, Ronald! Tudo bem?

[RONALD]

Tudo bem, Wilson. E você?

[WILSON]

Tudo ótimo! Vamos seguir nossa conversa sobre banco de dados. Ronald, sabemos que atualmente existem vários tipos de banco de dados, e precisamos manter nossos ouvintes informados sobre o assunto. Então, neste podcast vamos falar sobre banco de dados relacional, banco de dados orientado a objetos e NoSQL.

Ronald, vou começar falando sobre o banco de dados relacional e logo passo a bola para você falar sobre banco de dados orientado a objetos, tudo bem?

[RONALD]

Claro, professor Wilson. Você começa o bate-papo e vamos dialogando.

[WILSON]

O banco de dados relacional é o modelo de banco de dados mais utilizado no mundo, fazendo a modelagem de forma com que o usuário perceba a estrutura como tabelas. Nesse sentido, nosso estudo será sobre banco de dados relacional, ou simplesmente banco de dados.

O banco de dados relacional é composto por tabelas; as tabelas, por sua vez, são compostas por campos, que armazenam as informações.

Para que estas informações sejam armazenadas no banco de dados, precisamos tratar esses dados em uma linguagem que faça sua manipulação. A linguagem em questão é a linguagem SQL (*Structured Query Language*), ou seja, linguagem de consulta estruturada.

Por meio da linguagem SQL, podemos criar tabelas, colunas, índices e dar aos usuários permissões restritas. Também com esta linguagem realizamos consultas e “conversamos” com o banco de dados.

Dominar a linguagem SQL é requisito para um bom uso e um desempenho adequado do banco de dados.

Esta linguagem é organizada em subconjuntos, que têm propósitos bem definidos e fazem o gerenciamento de dados.

Ronald, você quer falar sobre os subconjuntos DQL, DML e DDL?

[RONALD]

Claro que sim, Wilson. Além destes, vou acrescentar mais alguns subconjuntos não mencionados por você. Vamos deixar nossos ouvintes craques nos comandos dos subconjuntos.

Vou começar falando sobre o DQL, que é uma linguagem de consulta de dados e define o comando utilizado para que possamos consultar e verificar (*select*) informações armazenadas no banco de dados. O objetivo principal dessa linguagem é popular nossos *grids* em formulários. Este subconjunto é sempre utilizado nas aplicações. Quem trabalha como DBA (administrador do banco de dados) sempre utiliza as consultas para poder pesquisar e posteriormente tratar e ou manipular esses dados. A consulta *select* apresenta ao DBA todas as linhas que estão inseridas no banco de dados em determinada tabela.

A DML (linguagem de manipulação de dados), por sua vez, define os comandos utilizados para manipulação de dados no banco (*insert*, *update* e *delete*). Este subconjunto é responsável por inserir ou persistir os dados no banco de dados. Toda aplicação que utiliza banco de dados precisa fazer a inserção de algum ou vários itens no banco de dados. O comando *insert* é utilizado para inserir os dados, o *update* tem a função de alterar os dados e o *delete* é usado para deletar os dados persistidos no banco de dados.

A DDL (linguagem de definição dos dados) é usada na criação das tabelas com a função “*create*”, mas também pode criar “*views*”, que são tabelas espelhos e servem para otimizar consultas. Também podemos realizar a atualização da tabela com o comando “*alter*”, atualização ou alteração necessária quando percebemos que faltou algum campo na nossa aplicação ou, por algum critério legal, for necessário incluir algum campo novo. Recentemente, todos os sistemas que trabalham com emissão de nota fiscal eletrônica precisaram incluir um novo campo no cadastro do produto, indicando qual porcentagem da alíquota do ICMS se destina ao fundo de amparo à pobreza. Ainda temos o comando DROP, utilizado para eliminar, ou seja, apagar a tabela e seus registros.

Ok, Wilson! Era isso que eu tinha para falar sobre os conjuntos, mas ainda falta conversarmos sobre os bancos de dados orientados a objetos e NoSQL.

[WILSON]

Perfeito, Ronald. Pode deixar que desse assunto eu entendo.

O banco de dados orientado a objetos armazena as informações na forma de objetos, ou seja, usa o paradigma de orientação a objetos, que por sua vez utiliza conceitos de abstração, encapsulamento, herança e polimorfismo.

A abstração é a capacidade de modelar coisas do mundo real. O encapsulamento tem como principal benefício evitar que interferências externas influenciem na manipulação dos dados. A herança é o mecanismo que permite a reutilização de códigos. Já o polimorfismo se refere às várias formas que o objeto pode conter.

Ao longo do curso, vamos estudar mais profundamente o paradigma de orientação a objetos e suas atribuições.

O banco de dados NoSql utiliza diversos modelos para acessar e gerenciar dados. Esses tipos de bancos de dados são otimizados especificamente para aplicativos que exigem modelos de grande volume de dados, baixa latência e flexibilidade. Esses requisitos são atendidos mediante o relaxamento de algumas restrições de consistência de dados dos outros bancos.

É importante lembrar que o tipo de banco de dados utilizado no curso técnico em **Informática para Internet** será o banco de dados relacional.

Mais uma vez, agradeço ao tutor Ronald pela ótima explicação sobre os subconjuntos.

Ronald, quer deixar uma mensagem final?

[RONALD]

Entender os subconjuntos e as funções SQL são essenciais para todo programador, pois sempre precisaremos validar a persistência de dados e aferir se o nosso simples cadastro está de acordo com o que pretendemos salvar.

Wilson, obrigado pela oportunidade e um grande abraço a todos.

[WILSON]

Você também gostou da explicação do Ronald? Gostou do nosso podcast? Esperamos que sim. Em breve, a gente se comunica novamente aqui, na rádio Senac EAD.