

Educação Profissional Paulista

Técnico em
**Desenvolvimento
de Sistemas**

DDL X DML

Diferenças entre DDL e DML

Aula 3: Criação de tabelas e índices

Código da aula: [SIS]ANO2C4B2S12A3

DDL X DML

Mapa da Unidade 5 Componente 4

semana

12

Você está aqui!

Diferenças entre
DDL e DML

semana

13

Comandos DML
avançados

semana

14

Análise e
modelagem de
dados

semana

19

DDL X DML

DDL X DML

Mapa da
Unidade 5
Componente 4

Você está aqui!

12

Diferenças entre DDL e DML

Aula 3: Criação de tabelas e índices

Código da aula: [SIS]C4B2S12A3



Objetivos da aula

- Criar e gerenciar tabelas e índices utilizando comandos DDL.



Recursos didáticos

- Recursos audiovisuais para exibição de vídeos e imagens;
- Caderno e lápis.



Duração da aula

50 minutos.



Habilidades técnicas

- Compreender os conceitos e diferenças entre DDL e DML.



Habilidades socioemocionais

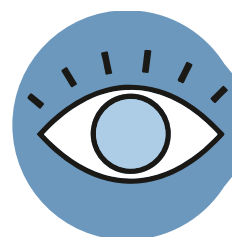
- Trabalhar a curiosidade ao explorar os diferentes comandos e seus usos.

Construindo o **conceito**

Comando **CREATE TABLE**

O comando CREATE TABLE é utilizado para criar uma nova tabela no banco de dados. Ao usar esse comando, você especifica o nome da tabela, as colunas e os tipos de dados que cada coluna aceitará. Também pode definir chaves primárias, estrangeiras e outras restrições.

Sintaxe do **CREATE TABLE**:



DE OLHO NO MODELO

```
CREATE TABLE nome_tabela (  
  coluna1 tipo_de_dado [restrições],  
  coluna2 tipo_de_dado [restrições],  
  ...,  
  PRIMARY KEY (coluna1),  
  FOREIGN KEY (colunaX) REFERENCES outra_tabela (colunaY)  
);
```

Construindo o **conceito**

Comando CREATE TABLE

Exemplo prático: criando a tabela de clientes

```
CREATE TABLE clientes (  
  id_cliente INT AUTO_INCREMENT PRIMARY KEY,  
  nome_cliente VARCHAR(100) NOT NULL,  
  email VARCHAR(100) UNIQUE,  
  telefone VARCHAR(15),  
  data_cadastro DATE );
```

Em que:

- **id_cliente:** coluna de tipo INT, que será a chave primária da tabela e autoincrementada.
- **nome_cliente:** coluna de tipo VARCHAR(100), que armazena o nome do cliente e não aceita valores nulos (NOT NULL).
- **email:** coluna de tipo VARCHAR(100) com a restrição UNIQUE, garantindo que não haja dois clientes com o mesmo e-mail.
- **data_cadastro:** coluna de tipo DATE, armazenando a data em que o cliente foi cadastrado.

Construindo o conceito

Índices

Índices são estruturas que otimizam o desempenho de consultas no banco de dados. Ao criar índices, você facilita a busca por registros específicos. No MySQL, existem diferentes tipos de índices.

1

Índice regular

O mais básico, melhora a performance em consultas específicas.

2

Índice único (UNIQUE)

Garante que os valores de uma coluna sejam únicos, além de melhorar a performance.

3

Índice de texto completo (FULLTEXT)

Usado para consultas de pesquisa textual, em colunas do tipo TEXT ou VARCHAR.

4

Índice de chave primária (PRIMARY KEY)

Garante a exclusividade e a ordem dos registros. Normalmente, é criado de forma automática ao definir uma coluna como chave primária.

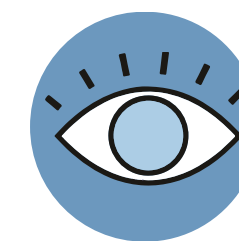
5

Índice composto

Criado em várias colunas ao mesmo tempo para otimizar consultas que dependem de múltiplas condições.

Construindo o **conceito**

Índices



DE OLHO NO MODELO

Índice regular

- ▶ O comando CREATE INDEX é utilizado para criar índices nas colunas da tabela, acelerando as buscas de dados.

Sintaxe do CREATE INDEX:

```
CREATE INDEX nome_indice  
ON nome_tabela (coluna1, coluna2, ...);
```

Exemplo prático: criando um índice regular

```
CREATE INDEX idx_nome_cliente  
ON clientes (nome_cliente);
```

Nesse exemplo, criamos um índice chamado idx_nome_cliente na coluna nome_cliente da tabela de clientes. Isso melhora a performance ao realizar consultas que busquem clientes pelo nome.

Construindo o **conceito**

Índices

► Índice único (UNIQUE INDEX)

Um **índice único** garante que os valores de uma coluna ou combinação de colunas sejam **únicos** em toda a tabela, ou seja, não permite duplicações. Esse tipo de índice pode ser útil para garantir a exclusividade de informações, como CPF, e-mail ou número de identificação de um cliente.

Exemplo prático:

```
CREATE UNIQUE INDEX idx_email_cliente  
ON clientes (email);
```

Criamos um índice único na coluna e-mail da tabela clientes. Esse índice garante que nenhum cliente tenha o mesmo e-mail registrado mais de uma vez.

Construindo o conceito

Índices

► Índice de texto completo (FULLTEXT INDEX)

Um índice de texto completo (FULLTEXT) é usado para realizar buscas eficientes em grandes textos. Ele pode ser aplicado em colunas de tipos TEXT, VARCHAR e CHAR, tornando as consultas de pesquisa mais rápidas em campos que armazenam grandes quantidades de texto.

Exemplo prático:

```
CREATE FULLTEXT INDEX idx_descricao_livro  
ON livros (descricao);
```

Esse comando cria um índice de texto completo na coluna descrição da tabela livros. Com isso, você pode usar o operador MATCH ... AGAINST para fazer buscas mais rápidas e precisas em textos descritivos, como resenhas de livros.

Continua ...

Construindo
o **conceito**

Índices

Exemplo de busca com FULLTEXT

```
SELECT * FROM livros  
WHERE MATCH (descricao) AGAINST ('aventura');
```

Essa consulta busca por todos os livros cuja descrição contenha a palavra "aventura".

Construindo o **conceito**

Índices

► Índice de chave primária (PRIMARY KEY)

Quando você define uma chave primária em uma tabela, o MySQL cria automaticamente um índice de chave primária. Esse índice é único e não permite valores NULL, além de ser utilizado para identificar unicamente cada linha da tabela.

Exemplo: criando uma tabela com chave primária

```
CREATE TABLE autores (  
  id_autor INT AUTO_INCREMENT,  
  nome_autor VARCHAR(100) NOT NULL,  
  PRIMARY KEY (id_autor)  
);
```

Construindo o **conceito**

Índices

▶ Índice composto

Exemplo:

```
CREATE INDEX idx_nome_email ON clientes (nome_cliente, email);
```

O índice composto melhora a performance de consultas que precisam buscar registros baseados tanto no nome quanto no e-mail do cliente.

Exemplo: criando um índice composto nas colunas nome_cliente e telefone

```
CREATE INDEX idx_nome_telefone  
ON clientes (nome_cliente, telefone);
```

Esse índice composto é útil quando a livraria precisa realizar consultas que filtram registros pelo nome do cliente e pelo número de telefone simultaneamente.

Continua...

Construindo
o **conceito**

Índices

Exemplo de consulta que usa um índice composto:

```
SELECT * FROM clientes  
WHERE nome_cliente = 'João Silva' AND telefone = '(11) 98765-4321';
```

Como a consulta envolve as duas colunas que fazem parte do índice composto, o desempenho dessa consulta será melhorado.

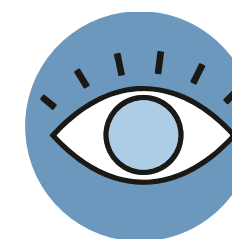
Continua...

Construindo o **conceito**

Índices

Caso você tenha uma tabela já criada e precise adicionar um índice posteriormente, pode usar o comando ALTER TABLE.

Exemplo:



DE OLHO NO MODELO

```
ALTER TABLE nome_tabela  
ADD INDEX nome_indice (coluna);
```

Exemplo prático: adicionando um índice em uma tabela existente

```
ALTER TABLE clientes  
ADD INDEX idx_telefone (telefone);
```

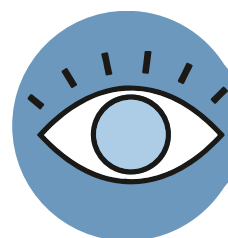
Um índice é adicionado na coluna telefone da tabela de clientes, o que melhora a performance de consultas por número de telefone.

Continua ...

Construindo o **conceito**

Índices

Se um índice não é mais necessário ou está impactando negativamente a performance, você pode removê-lo com o comando DROP INDEX.



DE OLHO NO MODELO

Sintaxe do DROP INDEX:

```
DROP INDEX nome_indice  
ON nome_tabela;
```



Tome nota

Se um índice não está sendo utilizado por consultas frequentes, ele ocupa espaço desnecessário e pode prejudicar o desempenho das operações de escrita.

Exemplo prático: removendo um índice

```
DROP INDEX idx_nome_cliente  
ON clientes;
```

Esse comando remove o índice `idx_nome_cliente` da tabela `clientes`.

Ser
sempre +

Situação

Otimização de banco de dados em um ambiente corporativo

Você trabalha como analista de dados em uma empresa de e-commerce chamada **TechShop**, que está crescendo rapidamente. O banco de dados da empresa armazena informações sobre produtos, clientes, pedidos e inventário.

No entanto, o sistema começou a apresentar lentidão, especialmente quando os usuários realizam buscas no catálogo de produtos, e as operações de inserção e atualização estão se tornando cada vez mais demoradas.

O gerente de TI pediu para sua equipe realizar uma auditoria no banco de dados e sugerir melhorias para otimizar o desempenho, sem prejudicar a integridade dos dados.

Como parte do processo, você percebe que algumas tabelas possuem muitos índices, enquanto outras não têm nenhum. Além disso, há índices duplicados e consultas que não aproveitam os índices existentes.

Situação fictícia produzida pela SEDUC-SP.

Ser
sempre +

Ação

Sua equipe precisa resolver a lentidão das consultas e melhorar a eficiência do banco de dados, considerando o impacto das suas ações no ambiente corporativo. Além disso, durante o processo, você enfrenta desafios como pressão por parte da gerência para resultados rápidos, divergências de opinião entre colegas de equipe sobre a melhor abordagem e necessidade de comunicar suas soluções de forma clara e assertiva para outros departamentos, que não possuem tanto conhecimento técnico.

- ▶ Como você priorizaria a criação ou eliminação de índices, considerando o impacto na performance das operações de leitura e escrita?
- ▶ Como lidar com a pressão da gerência, que exige resultados rápidos sem comprometer a qualidade e a integridade dos dados no processo de otimização?
- ▶ Qual é a melhor maneira de explicar as mudanças técnicas propostas para pessoas que não possuem conhecimento técnico, garantindo que todos entendam os benefícios?

Situação fictícia produzida pela SEDUC-SP.

Então, ficamos assim:

- 1** Aprendemos que o comando CREATE TABLE permite estruturar as informações de forma lógica, e os índices ajudam a otimizar o acesso a essas informações;
- 2** Compreendemos que o uso adequado dos índices pode melhorar significativamente a performance das consultas, especialmente em bases de dados com grande volume de dados;
- 3** Vimos que eliminar índices em um banco de dados pode ser necessário em algumas situações, pois, embora os índices melhorem a performance das consultas de leitura, eles também ocupam espaço e podem prejudicar o desempenho de operações de escrita.

O que nós
**aprendemos
hoje?**

© Getty Images

Referências da aula

Identidade visual: imagens © Getty Images

ALURA. **Alterando e restringindo o formato de nossas tabelas**, [s.d.]. Disponível em: <https://www.alura.com.br/apostila-sql-e-modelagem-com-banco-de-dados/Alterando-e-restringindo-o-formato-de-nossas-tabelas?srsId=AfmBOor3VvtE3BmAl4t3Y0YEZH1Zu6fERVn5aV4a4pvt2pdQsRcFS4ik>. Acesso em: 8 dez. 2024.

ELMASRI, R.; NAVATHE, S. B. **Sistema de banco de dados**. São Paulo: Pearson Addison Wesley, 2011.

OLIVEIRA, D.; DIAS, G. **Saiba tudo sobre SQL**: a linguagem padrão para trabalhar com banco de dados relacionais!, 29 nov. 2023. Disponível em https://www.alura.com.br/artigos/o-que-e-sql?srsId=AfmBOoqEtg3JKy_GMj7ePg3tuXniEHAfnw0xh-Acuy7e8Vnb4b0XxmY. Acesso em: 8 dez. 2024.

VILA, V. **Comandos DML**: manipulação de dados com MySQL, [s.d.]. Disponível em: <https://www.alura.com.br/conteudo/mysql-dml-manipulacao-de-dados>. Acesso em: 8 dez. 2024.

Orientações ao professor

Slide 6



Tempo seção: 34 minutos.

Professor, o modelo é uma modelagem básica, que os alunos podem seguir para entender a estrutura geral do comando.

Slide 13



Neste exemplo, a coluna id_autor é definida como **chave primária**, e o MySQL criará um índice automaticamente para essa coluna.

Slide 14 e 15



Um **índice composto** envolve mais de uma coluna. Ele é útil quando você frequentemente realiza consultas que utilizam várias colunas em condições de busca. O índice composto melhora a performance de consultas que filtram registros com base em múltiplas colunas ao mesmo tempo.

Slide 17



Ter vários índices semelhantes ou duplicados (em colunas ou combinações de colunas iguais ou muito semelhantes) pode ser desnecessário, impactando negativamente as operações de escrita, como inserções, atualizações e exclusões.

Slide 18



Orientações: a seção **Ser sempre +** tem como objetivo desenvolver e aprimorar as competências socioemocionais dos estudantes, focando especificamente as situações desafiadoras que podem surgir no ambiente profissional.



Tempo: 15 minutos



Gestão de sala de aula:

Mantenha um ambiente de diálogo aberto e respeitoso;

Assegure a participação equitativa, promovendo uma discussão inclusiva;

Reconheça a complexidade do tema e a diversidade de perspectivas que os estudantes podem trazer;

Forneça feedback construtivo e direcionamento à medida que os estudantes exploram possíveis soluções para o cenário proposto;

Ajude os estudantes a refinarem suas ideias e a considerarem todas as implicações de suas sugestões.



Condução da dinâmica:

O professor deve estimular o diálogo e a resolução de problemas.

Introdução ao problema: o professor apresenta o cenário desafiador (como a lentidão do banco de dados) e explica a importância de otimizar tabelas e índices, relacionando isso a um ambiente corporativo real.

- **Formação de grupos:** os alunos são divididos em pequenos grupos para fomentar a colaboração e a troca de ideias. Cada grupo deve discutir como resolver a situação, sugerindo mudanças nas tabelas e índices.
- **Facilitação e perguntas orientadoras:** o professor circula entre os grupos fazendo perguntas para guiar a análise e a tomada de decisão dos alunos (ex.: “Como essa mudança vai impactar a performance de escrita?”).
- **Apresentação de soluções:** cada grupo apresenta suas soluções. O professor incentiva a comunicação clara e o uso de uma linguagem acessível, como fariam em um ambiente corporativo.
- **Feedback e reflexão:** o professor dá feedback sobre as soluções, destaca boas práticas e promove uma reflexão sobre os desafios emocionais e técnicos enfrentados durante a atividade, como a pressão por resultados ou divergências de opinião.

Continua

Slide 18



Expectativas de respostas:

• Como você priorizaria a criação ou eliminação de índices, considerando o impacto na performance das operações de leitura e escrita?

Para priorizar a criação ou eliminação de índices, o primeiro passo é **analisar o padrão de uso do banco de dados**:

- Identifique as consultas que são executadas com mais frequência. Se uma consulta envolve colunas que não possuem índices ou que fazem varreduras completas de tabelas, a criação de um índice nessas colunas pode aumentar significativamente o desempenho;
- Considere o impacto dos índices nas operações de escrita. Muitos índices podem **tornar as operações de escrita mais lentas**, pois cada atualização de dados precisa modificar os índices relacionados. Por isso, elimine ou evite criar índices que não são usados em consultas frequentes;
- Priorize a criação de **índices compostos**, que otimizam múltiplas colunas frequentemente usadas juntas em filtros. Isso melhora a leitura sem comprometer tanto as escritas;
- Ferramentas como o **EXPLAIN**, no MySQL, podem ser usadas para verificar se um índice está sendo utilizado em uma consulta ou se uma varredura completa de tabela ainda está ocorrendo. Índices que não são usados devem ser removidos.

2. Como lidar com a pressão da gerência, que exige resultados rápidos sem comprometer a qualidade e a integridade dos dados no processo de otimização?

Lidar com a pressão da gerência envolve **gestão de expectativas** e **comunicação clara**.

- Explique claramente que, embora a otimização do desempenho seja possível, **mudanças rápidas** podem comprometer a integridade dos dados ou resultar em efeitos colaterais inesperados. É importante enfatizar que a **qualidade** deve ser priorizada para evitar problemas futuros, como corrupção de dados ou downtime;
- Proponha um **plano de ação em etapas**. Divida a otimização em pequenas melhorias que podem ser implementadas rapidamente (ex.: ajuste de índices em tabelas específicas), gerando resultados visíveis em curto prazo, enquanto trabalha em otimizações maiores;
- **Priorização de áreas críticas**: identifique as operações ou consultas mais críticas e comece a otimização por essas áreas. Isso demonstra que há progresso tangível, aliviando a pressão por resultados rápidos;
- **Relatórios frequentes**: mantenha a gerência informada sobre os **resultados parciais** à medida que as mudanças são implementadas, mostrando como o desempenho está sendo melhorado progressivamente. Isso ajuda a balancear a pressão por resultados rápidos com a necessidade de manter a qualidade e a integridade dos dados.

Continua...

Slide 18



3. Qual é a melhor maneira de explicar as mudanças técnicas propostas para pessoas que não possuem conhecimento técnico, garantindo que todos entendam os benefícios?

Ao explicar mudanças técnicas para pessoas não técnicas, é importante simplificar o jargão e focar os **benefícios práticos** que as mudanças trarão.

- **Use uma linguagem simples:** evite termos técnicos e explique as mudanças de forma acessível. Por exemplo, em vez de falar sobre "índices", explique que está sendo feito um "ajuste no sistema para que as buscas e operações sejam mais rápidas";
- **Use analogias:** utilize analogias do cotidiano para explicar o conceito. Por exemplo, você pode comparar um índice a um "índice de livro", que facilita encontrar uma informação sem precisar folhear todas as páginas.
- **Foque os benefícios de negócio:** em vez de detalhar como o índice foi criado, explique como isso resultará em **pesquisas mais rápidas, menos tempo de espera para o cliente e maior eficiência para a equipe**, o que leva a uma **melhor experiência do usuário e economia de tempo e recursos**;
- **Mostre resultados visíveis:** sempre que possível, apresente números ou exemplos práticos de como a mudança melhorou o desempenho (ex.: "Agora, uma pesquisa de produto que levava 5 segundos será feita em 1 segundo").

Slide 20



Orientações: professor, a seção **O que nós aprendemos hoje?** tem o objetivo de reforçar e esclarecer os conceitos principais discutidos na aula. Essa revisão pode ser uma ferramenta de avaliação informal do aprendizado dos estudantes, identificando áreas que podem precisar de mais atenção em aulas futuras.



Tempo: 1 minuto.



Gestão de sala de aula:

- Mantenha um tom positivo e construtivo, reforçando o aprendizado em vez de focar em correções;
- Seja direto e objetivo nas explicações para manter a atividade dentro do tempo estipulado;
- Engaje os estudantes rapidamente, pedindo confirmações ou reações breves às definições apresentadas.



Condução da dinâmica:

- Explique que esta parte da seção, “Então, ficamos assim”, é um momento de reflexão e esclarecimento sobre os conceitos abordados na aula;
- Informe que será uma rápida revisão para assegurar que os entendimentos dos estudantes estão alinhados com as definições corretas dos conceitos;
- Apresente o slide com a definição sintética de cada conceito principal discutido na aula, ampliando-os em forma de frases completas;
- Finalize resumindo os pontos principais e reiterando a importância de cada conceito e como eles se encaixam no contexto maior da aula;
- Reforce a ideia de que essa revisão ajuda a solidificar o entendimento dos estudantes e a prepará-los para aplicar esses conceitos em situações práticas.



Expectativas de respostas:

Os estudantes devem sair da aula com um entendimento claro e preciso dos conceitos principais. A atividade serve como uma verificação rápida do entendimento dos estudantes e uma oportunidade para corrigir quaisquer mal-entendidos.

Educação Profissional Paulista

Técnico em
**Desenvolvimento
de Sistemas**