



Unidade 2


Bancos de Dados Relacionais e Linguagem SQL

2.3 Gerenciando os dados relacionais: DDL, DML, DCL e TCL.

Instruções SQL

- As instruções SQL são agrupadas em algumas categorias, dependendo de suas funções.
- Existem quatro categorias principais de instruções SQL:
 - *Data Definition Language – DDL*
 - *Data Manipulation Language – DML*
 - *Data Control Language – DCL*
 - *Transaction Control Language – TCL*


Data Definition Language - DDL

- As instruções DDL criam, modificam e removem as estruturas de dados do banco de dados e são iniciadas pelas palavras chave **CREATE, ALTER, DROP, RENAME** e **TRUNCATE**.
-  Para criar o banco de dados a primeira categoria que se utiliza é a DDL que geralmente a toolkit que utilizaremos para desenhar o modelo de dados gera o script contendo estas instruções. **Não se esqueça de definir os tipos de dados!**


Data Manipulation Language - DML

- As instruções DML são usadas para modificar os dados das tabelas fazendo inclusão, alteração ou remoção de linhas existentes e são iniciadas pelas palavras chave **INSERT**, **UPDATE**, **DELETE** e **MERGE**.
- 😊 Após criado o banco de dados esta é a categoria que mais teremos que trabalhar que possui uma série de recursos e nos permite fazer a mágica da transformação dos dados.

Data Control Language - DCL

- As instruções DCL são usadas para fornecer ou remover direitos de acesso ao banco de dados e às estruturas dentro dele e são iniciadas pelas palavras chave **GRANT**, **DENY** e **REVOKE**.
-  Para garantir a segurança do banco de dados é fundamental conhecermos esta categoria de forma a evitar acessos de usuários indevidos.

Transaction Control Language - TCL

- As instruções TCL são usadas para gerenciar as alterações feitas pelas instruções DML, geralmente para garantir a consistência dos dados.
- As modificações nos dados são autorizadas ou não pelas palavras chave **COMMIT** e **ROLLBACK**.
-  Esta categoria nos ajuda a garantir que um débito em uma conta bancária será executado após as operações de crédito.

*Now, we need to create
databases from examples!*

Vamos importar os bancos de dados

- Vamos criar a estrutura a partir do arquivo .mwb
- <https://downloads.mysql.com/docs/sakila-db.zip>
- E importar os dados dos bancos:
- https://downloads.mysql.com/docs/world_x-db.zip
- <https://downloads.mysql.com/docs/menagerie-db.zip>

Data Definition Language - DDL

- Vamos iniciar esta abordagem apresentando quais são os objetos de banco de dados mais usados, como observar a estrutura da tabela e como criar novas tabelas;
- **Apesar das tabelas que criaremos serem menores elas envolvem as mesmas instruções e sintaxe SQL para operações em tabelas muito grandes.**

Data Definition Language - DDL

- Apresentaremos então os objetos de banco de dados mais utilizados e com funciona o acesso ao dicionário de dados.
- Os principais objetos de banco de dados são:
- **Tabela, Índice, Constraint, View, Sequence, Sinônimo.**
- Importante ressaltar que alguns destes objetos são dependentes e outros não.

Objetos de schemas de banco de dados

- Objetos que ocupam espaço em disco são conhecido como segmentos.
- Tabelas e índices ocupam espaço significativo no disco físico e portanto são classificados como segmentos.
- *Views, constraints, Sequences e sinônimos* não ocupam significativo ocupando somente o espaço para sua definição.


Objetos de schemas de banco de dados

- Objetos que ocupam espaço em disco são conhecido como segmentos.
- Tabelas e índices ocupam espaço significativo no disco físico e portanto são classificados como segmentos.
- *Views, constraints, Sequences e sinônimos* não ocupam significativo ocupando somente o espaço para sua definição.

Objetos de schemas de banco de dados

- Objetos que ocupam espaço em disco são conhecido como segmentos.
- Tabelas e índices ocupam espaço significativo no disco físico e portanto são classificados como segmentos.
- *Views*, *constraints*, *Sequences* e *sinônimos* não ocupam significativo ocupando somente o espaço para sua definição pois nenhum deles tem linhas de dados associadas.

Dicionário de dados

- Todas as definições de todos os objetos são registradas no dicionário de dados.
- O dicionário é acessado pelo próprio assim como pelo usuário para tratar algumas questões específicas de estrutura.
-  Ele funciona como um banco de dados do banco de dados, ou melhor de todos os bancos de dados de uma certa instância.

Dicionário de dados

- O banco de dados usa o Dicionário de Dados para todas as instruções que você executa, mesmo que liste os nomes das colunas.
- Ele verifica se as tabelas que você referenciou na sua instrução existem, se os nomes das colunas estão corretos, se você tem os privilégios apropriados para executar a ação que está solicitando além de definir o plano de execução da operação.

Dicionário de dados

- O banco de dados usa o Dicionário de Dados para todas as instruções que você executa, mesmo que liste os nomes das colunas.
- Ele verifica se as tabelas que você referenciou na sua instrução existem, se os nomes das colunas estão corretos, se você tem os privilégios apropriados para executar a ação que está solicitando além de definir o plano de execução da operação.


Dicionário de dados

- O banco de dados usa o Dicionário de Dados para todas as instruções que você executa, mesmo que liste os nomes das colunas.
- Ele verifica se as tabelas que você referenciou na sua instrução existem, se os nomes das colunas estão corretos, se você tem os privilégios apropriados para executar a ação que está solicitando além de definir o plano de execução da operação.

Data Definition Language - DDL

- Os dados são todos armazenados em tabelas e para criá-las existem algumas regras para criação das colunas que vale o reforço:
- Deve começar com uma letra e possuir de 1 a 30 caracteres.
- O nome só pode conter A-Z, a-z, 0-9, _, \$ e # e recomenda-se não repetir o nome de outro objeto existente mesmo sendo de usuários diferentes.

Data Definition Language - DDL - CREATE

- Para criar uma nova tabela, você deve ter o privilégio CREATE TABLE e uma área de armazenamento definida.
- Geralmente o DBA já fará isto para você e lhe entregará o acesso à instância com o banco já criado.
- No nosso caso, trabalharemos com o usuário do DBA  mas não é comum nas organizações.

Data Definition Language - DDL - CREATE

- Observe os elementos para que se crie uma tabela:
 - Tabela é o nome da tabela.
 - Coluna é o nome da coluna.
 - Tipo de dados é o tipo de dados e o tamanho da coluna.
 - A expressão default expressa um valor padrão que será registrado caso não seja passado o parâmetro correto na instrução SQL.

```
CREATE TABLE IF NOT EXISTS `LOTO`.`tblJogo` (  
  `idtblJogo` INT(11) NOT NULL AUTO_INCREMENT,  
  `nome` VARCHAR(45) NULL DEFAULT NULL,  
  `descricao` VARCHAR(256) NULL DEFAULT NULL,  
  `qtdeNumeros` INT(11) NULL DEFAULT NULL,  
  `numInicial` INT(11) NULL DEFAULT NULL,  
  `numFinal` INT(11) NULL DEFAULT NULL,  
  `dataCriacao` TIMESTAMP NULL DEFAULT CURRENT_TIMESTAMP,  
  `ultimaAtualizacao` TIMESTAMP NULL DEFAULT CURRENT_TIMESTAMP,  
  PRIMARY KEY (`idtblJogo`))
```

Data Definition Language - DDL - CREATE

- Possuímos também dois tipos de tabela, as de usuário (que são criadas nos schemas) e as de dicionário de dados (que precisam de permissão para acesso).
- No caso do MySQL para acessá-las precisaremos de acesso ao banco de dados do sistema que se chama mysql.
- [Acesse a documentação sobre o dicionário de dados do Mysql.](#)

Data Definition Language - DDL - CREATE

- **ATENÇÃO: NÃO É RECOMENDADO QUE USUÁRIOS QUE NÃO SÃO DBA TENHAM ACESSO AO BANCO DE DADOS DE SISTEMA.**
- **QUALQUER MANIPULAÇÃO EQUIVOCADA DESTRUIRÁ A CONSISTÊNCIA DE TODOS OS BANCO DE DADOS.**

Data Definition Language - DDL - ALTER

- As instruções ALTER TABLE são usadas para:
 - Adicionar uma coluna.
 - Modificar uma coluna existente.
 - Apontar um valor padrão para a coluna.
 - Eliminar uma coluna.

```
1  -- swap a and b
2  ALTER TABLE t1 RENAME COLUMN a TO b,
3                      RENAME COLUMN b TO a;
4  -- "rotate" a, b, c through a cycle
5  ALTER TABLE t1 RENAME COLUMN a TO b,
6                      RENAME COLUMN b TO c,
7                      RENAME COLUMN c TO a;
```


Data Definition Language - DDL - ALTER

- Destaca-se porém as regras para eliminação de colunas:
 - É possível eliminar uma coluna que contém dados mas sempre uma coluna por vez
 - Não é possível eliminar todas as colunas de uma tabela, pelo menos uma deverá permanecer.
 - Caso elimine a coluna os valores de dados contidos nela não poderão ser recuperados.

```
1 | ALTER TABLE t2 DROP COLUMN c, DROP COLUMN d;
```

Data Definition Language - DDL - DROP

- A instrução DROP TABLE remove a definição das tabelas:
 - Todos os dados da tabela são excluídos
 - A descrição da tabela é removida do dicionário de dados.

```
1 DROP [TEMPORARY] TABLE [IF EXISTS]
2     tbl_name [, tbl_name] ...
3     [RESTRICT | CASCADE]
```

Data Definition Language - DDL - RENAME

- Para mudar o nome de uma tabela, use a instrução RENAME
- Isso pode ser feito apenas pelo proprietário do objeto ou pelo DBA.

```
1  RENAME TABLE old_table TO new_table;
```

Data Definition Language - DDL - TRUNCATE

- A instrução TRUNCATE remove todas as linhas de uma tabela e libera o espaço de armazenamento utilizado por ela porém, mantém a estrutura para recebimento de novos dados.

```
1 TRUNCATE [TABLE] tbl_name
```

*Ready! We can continue
and begin with
manipulation data!*

Referências bibliográficas

MySQL, Channel, 2021. MySQL Workbench Tutorial. Disponível em:
<https://www.youtube.com/watch?v=X_umYKqKaF0>. Acesso em: 04 out. 2021.

MySQL, Manual 2021. MySQL Documentation Archive. Disponível em:
<<https://dev.mysql.com/doc/index-archive.html>>. Acesso em: 04 out. 2021.