

Structured Query Language

SQL

Data Definition Language

DDL

Linguagem de Definição de Dados

1. CREATE TABLE

Use a instrução CREATE TABLE para definir uma nova tabela e seus atributos, incluindo as restrições de atributos.

1.1. Sintaxe

```
CREATE TABLE tabela (atributo1 tipo [(tamanho)] [NOT NULL] [índice1] [, atributo2 tipo [(tamanho)] [NOT NULL] [índice2] [, ...]] [, CONSTRAINT índicedemultiatributos [, ...]]
```

1.2. A instrução CREATE TABLE tem estas partes:

<u>Parte</u>	<u>Descrição</u>
<i>tabela</i>	O nome da tabela a ser criada.
<i>atributo1</i> , <i>atributo2</i>	O nome do atributo ou atributos a serem criados na nova tabela. Uma tabela deve ter pelo menos um atributo.
<i>tipo</i>	O tipo de dados de atributo na nova tabela.
<i>tamanho</i>	O tamanho do atributo em caracteres (somente os atributos Texto e Numérico).
<i>índice1</i> , <i>índice2</i>	Uma cláusula CONSTRAINT que define um índice de atributo único. Para maiores informações vide a cláusula CONSTRAINT.
<i>Índicedemultiatributos</i>	Uma cláusula CONSTRAINT que define um índice de atributos múltiplos. Para maiores informações consulte o tópico CONSTRAINT.

1.2. Exemplos

- a) Criar a tabela: DEPARTAMENTO

```
CREATE TABLE DEPARTAMENTOS  
  (Codigo Text(3) NOT NULL,  
   Nome Text(30) NOT NULL,  
   CONSTRAINT PrkDep PRIMARY KEY (Codigo))
```

- b) Criar a tabela: CURSOS

```
CREATE TABLE CURSOS  
  (Codigo Text(3) NOT NULL,  
   Nome Text(30) NOT NULL,  
   CodigoDepartamento Text(3) NOT NULL,  
   CONSTRAINT PrkCurso PRIMARY KEY (Codigo),  
   CONSTRAINT FrkDepCurso FOREIGN KEY (CodigoDepartamento)  
   REFERENCES DEPARTAMENTOS (Codigo))
```

- c) Criar a tabela: DISCIPLINAS

```
CREATE TABLE DISCIPLINAS
(CodigoCurso Text(3) NOT NULL,
 Numero INTEGER NOT NULL,
 Nome Text(30) NOT NULL,
 Creditos Byte NOT NULL,
 Laboratorio Byte NOT NULL,
 Prelecao Byte NOT NULL,
 CodigoDepartamento Text(3) NOT NULL,
 CONSTRAINT PrkDisciplina PRIMARY KEY (CodigoCurso, Numero),
 CONSTRAINT FrkCursoDisciplina FOREIGN KEY (CodigoCurso)
 REFERENCES CURSOS (Codigo))
```

- d) Criar a Tabela: SEMESTRES

```
CREATE TABLE SEMESTRES
(Ano Integer NOT NULL,
 Numero Byte NOT NULL,
 CONSTRAINT PrkSemestre PRIMARY KEY (Ano,Numero))
```

- e) Criar a Tabela: TURMAS

```
CREATE TABLE TURMAS
(CodigoCurso Text(3) NOT NULL,
 NumeroDisciplina INTEGER NOT NULL,
 Numero Text(3) NOT NULL,
 NumeroSubturma Text (1) NOT NULL,
 AnoSemestre Integer NOT NULL,
 NumeroSemestre Byte NOT NULL,
 CONSTRAINT PrkTurma PRIMARY KEY (CodigoCurso, NumeroDisciplina, Numero,
 NumeroSubTurma, AnoSemestre,NumeroSemestre),
 CONSTRAINT FrkDisciplinaTurma FOREIGN KEY (CodigoCurso,NumeroDisciplina)
 REFERENCES DISCIPLINAS (CodigoCurso,Numero),
 CONSTRAINT FrkDisciplinaSemestre FOREIGN KEY (AnoSemestre,NumeroSemestre)
 REFERENCES SEMESTRES (Ano,Numero))
```

1.3. Comentários

Uma cláusula CONSTRAINT estabelece várias restrições em um atributo e pode ser utilizada para estabelecer a chave primária.

2. ALTER TABLE

Modifica a estrutura de uma tabela depois de ter sido criada com a instrução CREATE TABLE.

2.1. Sintaxe

```
ALTER TABLE tabela {ADD {COLUMN atributo tipo[(tamanho)] [NOT NULL]
[CONSTRAINT índice] | CONSTRAINT índice multiatributos} | DROP {COLUMN atributo |
CONSTRAINT nome do índice} }
```

2.2. A instrução ALTER TABLE tem estas partes:

<u>Parte</u>	<u>Descrição</u>
Tabela	O nome da tabela a ser alterada.
Atributo	O nome do atributo a ser adicionado ou excluído da tabela.
Tipo	O tipo de dados de atributo.
Tamanho	O tamanho do atributo em caracteres (somente os atributos Texto e Binário).
Índice	O índice para atributo. Consulte o tópico da cláusula CONSTRAINT para maiores informações sobre como construir este índice.
Índice multiatributos	A definição de um índice de atributos múltiplos a ser adicionado à tabela. Consulte o tópico da cláusula CONSTRAINT para maiores informações sobre como construir esta cláusula.
Nome do índice	O nome do índice de atributo múltiplo a ser removido.

2.3. Exemplos

- a) Acrescentar na tabela DEPARTAMENTOS, o atributo *DataDeCriacao* de preenchimento opcional e do tipo data.

```
ALTER TABLE DEPARTAMENTOS
ADD COLUMN DataDeCriacao Date
```

- b) Excluir da tabela DEPARTAMENTOS, o atributo *DataDeCriacao*

```
ALTER TABLE DEPARTAMENTOS DROP COLUMN DataDeCriacao
```

2.4. Comentários

Através da instrução ALTER TABLE, você pode alterar uma tabela existente de diversas maneiras.

Você pode:

Utilizar ADD COLUMN para adicionar um novo atributo à tabela. Você especifica o nome do atributo, tipo de dados e (para atributos Texto e Binário) um tamanho opcional.

Utilizar `ADD CONSTRAINT` para adicionar um índice de atributos múltiplos. Para maiores informações sobre índices de atributos múltiplos, consulte o tópico da cláusula `CONSTRAINT`.

Utilizar `DROP COLUMN` para excluir um atributo. Você especifica somente o nome do atributo.

Utilizar `DROP CONSTRAINT` para excluir um índice de atributos múltiplos. Você especifica somente o nome do índice após a palavra reservada `CONSTRAINT`.

Você não pode adicionar ou excluir mais de um atributo ou índice de cada vez.

Você pode utilizar `NOT NULL` em um atributo único ou dentro de uma cláusula `CONSTRAINT` nomeada que se aplica tanto a uma `CONSTRAINT` de atributo único ou atributos múltiplos. Contudo, você pode aplicar a restrição `NOT NULL` somente uma vez a um atributo pois, senão, ocorrerá um erro em tempo de execução.

3. DROP TABLE

Exclui uma tabela existente de um banco de dados ou exclui um índice existente de uma tabela.

3.1. Sintaxe

```
DROP {TABLE tabela | INDEX índice ON tabela}
```

3.2. A instrução DROP tem estas partes:

<u>Parte</u>	<u>Descrição</u>
Tabela	O nome da tabela a ser excluída ou a tabela a partir da qual um índice deve ser excluído.
Índice	O nome do índice a ser excluído da tabela.

3.3. Exemplos

- a) Excluir a Tabela: TABEXEMPLO criada pelo comando abaixo

Criando a tabela:

```
CREATE TABLE TABEXEMPLO (Codigo Text(3), Nome Text(30), DataNascimento  
Date,  
CONSTRAINT PrkTabExemplo PRIMARY KEY (Codigo))
```

Excluindo a tabela:

```
DROP TABLE TABEXEMPLO
```

- b) Excluir o Índice IdxNome criado pelos comandos abaixo:

Criando o índice (Vide comando CREATE INDEX):

```
CREATE INDEX IdxNome ON DEPARTAMENTOS (Nome ASC)
```

Excluindo o Índice IdxNome:

```
DROP INDEX IdxNome ON DEPARTAMENTOS
```

3.4. Comentários

Você deve fechar a tabela para poder excluí-la ou remover um índice dela.

Você também pode utilizar ALTER TABLE para excluir um índice de uma tabela.

Você pode utilizar CREATE TABLE para criar uma tabela e CREATE INDEX ou ALTER TABLE para criar um índice. Para modificar uma tabela, use ALTER TABLE.

4. CREATE INDEX

Cria um novo índice em uma tabela existente.

4.1. Sintaxe

```
CREATE [ UNIQUE ] INDEX índice ON tabela (atributo [ASC|DESC][, atributo [ASC|DESC], ...])  
[WITH { PRIMARY | DISALLOW NULL | IGNORE NULL }]
```

4.2. A instrução CREATE INDEX tem estas partes:

<u>Parte</u>	<u>Descrição</u>
índice	O nome do índice a ser criado.
tabela	O nome da tabela existente que conterá o índice.
atributo	O nome do atributo ou atributos a serem indexados. Para criar um índice de atributo único, liste o nome do atributo entre parênteses após o nome da tabela. Para criar um índice de atributos múltiplos, liste o nome de cada atributo a ser incluído no índice. Para criar índices descendentes, use a palavra reservada DESC; caso contrário, assume-se que os índices são ascendentes.

4.3. Exemplos

- a) Criar um Índice único para o atributo nome da tabela: **DEPARTAMENTOS**

```
CREATE UNIQUE INDEX IdxNome  
ON DEPARTAMENTOS (Nome ASC)
```

A cláusula UNIQUE indica que o índice criado não pode ter valores duplicados.

- b) Criar um índice para o atributo nome da tabela DEPARTAMENTOS proibindo a entrada de dados nulos.

```
CREATE INDEX IdxNome  
ON DEPARTAMENTOS (Nome ASC)  
WITH DISALLOW NULL
```

A cláusula WITH DISALLOW NULL proíbe a entrada de dados nulos no atributo nome.

- c) Criar um índice para o atributo nome da tabela DEPARTAMENTOS que não contenha valores nulos

```
CREATE INDEX IdxNome  
ON DEPARTAMENTOS (Nome ASC)  
WITH IGNORE NULL
```

A cláusula WITH IGNORE NULL proíbe valores nulos no índice.

4.4. Comentários

Para proibir valores duplicados no atributo ou atributos indexados de diferentes registros, use a palavra reservada **UNIQUE**.

Na cláusula **WITH** opcional, você pode impor regras de validação de dados. Você pode:

Proibir entradas **Null** no atributo ou atributos indexados dos novos registros, utilizando a opção **DISALLOW NULL**

Impedir que registros com valores **Null** no atributo ou atributos indexados sejam incluídos no índice utilizando a opção **IGNORE NULL**.

Designar o atributo ou atributos indexados como a chave primária utilizando a palavra reservada **PRIMARY**. Isto significa que a chave é exclusiva e, portanto, você pode omitir a palavra reservada **UNIQUE**.

Você também pode utilizar a instrução **ALTER TABLE** para adicionar um índice de atributo único ou de atributos múltiplos a uma tabela e pode utilizar a instrução **ALTER TABLE** ou a instrução **DROP** para remover um índice criado com **ALTER TABLE** ou **CREATE INDEX**.

5. CONSTRAINT

Uma restrição é semelhante a um índice, embora também possa ser utilizada para estabelecer uma relação com uma outra tabela.

Você utiliza a cláusula CONSTRAINT nas instruções ALTER TABLE e CREATE TABLE para criar ou excluir restrições. Há dois tipos de cláusulas CONSTRAINT: um para criar uma restrição em um atributo único e outro para criar uma restrição em mais de um atributo.

5.4. Sintaxe

Restrição de atributo único:

```
CONSTRAINT nome {PRIMARY KEY | UNIQUE | NOT NULL | REFERENCES tabelaexterna  
[(atributoexterno1, atributoexterno2)]}
```

Restrição de atributos múltiplos:

```
CONSTRAINT nome  
{PRIMARY KEY (primária1[, primária2 [, ...]]) |  
UNIQUE (exclusiva1[, exclusiva2 [, ...]]) |  
NOT NULL (nãonulo1[, nãonulo2 [, ...]]) |  
FOREIGN KEY (ref1[, ref2 [, ...]])  
REFERENCES tabelaexterna [(atributoexterno1 [, atributoexterno2 [, ...]])]}
```

5.5. A cláusula CONSTRAINT tem estas partes:

<u>Parte</u>	<u>Descrição</u>
Nome	O nome da restrição a ser criada.
primária1, primária2	O nome do atributo ou atributos a ser designado(s) à chave primária.
exclusiva1, exclusiva2	O nome do atributo ou atributos a ser designado(s) como uma chave exclusiva.
nãonulo1, nãonulo2	O nome do atributo ou atributos que estão restritos a valores não-nulo
ref1, ref2	O nome do atributo ou atributos de uma chave externa que fazem referência a atributos em uma outra tabela.
Tabelaexterna	O nome da tabela externa contendo o atributo ou atributos especificados por atributoexterno.
atributoexterno1, atributoexterno2	O nome do atributo ou atributos na tabelaexterna especificados por ref1, ref2. Você pode omitir esta cláusula se o atributo referenciado for a chave primária de tabelaexterna.

5.6. Comentários

Você utiliza a sintaxe para uma restrição de atributo único na cláusula de definição de atributo de uma instrução ALTER TABLE ou CREATE TABLE que se segue imediatamente à especificação do tipo de dados do atributo.

Você utiliza a sintaxe para uma restrição de atributos múltiplos sempre que utilizar a palavra reservada CONSTRAINT fora de uma cláusula de definição de atributo em uma instrução ALTER TABLE ou CREATE TABLE.

Utilizando CONSTRAINT, você pode designar um atributo como um dos seguintes tipos de restrições:

Você pode utilizar a palavra reservada UNIQUE para designar um atributo como chave exclusiva. Isto significa que não pode haver dois registros em uma tabela que tenham o mesmo valor neste atributo. Você pode restringir qualquer atributo ou lista de atributos como exclusivo. Se uma restrição de atributos múltiplos for designada como uma chave exclusiva, os valores combinados de todos os atributos no índice devem ser exclusivos, mesmo que dois ou mais registros tenham o mesmo valor em apenas um dos atributos.

Você pode utilizar as palavras reservadas PRIMARY KEY para designar um atributo ou conjunto de atributos em uma tabela como uma chave primária. Todos os valores na chave primária devem ser exclusivos e não Nulos e só pode haver uma chave primária para uma tabela.

Você pode utilizar as palavras reservadas FOREIGN KEY para designar um atributo como uma chave externa. Se a chave primária da tabela externa consistir em mais de um atributo, você deverá utilizar uma definição de restrição de atributos múltiplos, listando todos os atributos referenciais, o nome da tabela externa e os nomes dos atributos referenciados na tabela externa, na mesma ordem em que os atributos referenciais são listados. Se o atributo ou atributos referenciados forem a chave primária da tabela externa, você não precisará especificar os atributos referenciados — por padrão, o mecanismo de banco de dados se comporta como se a chave primária da tabela externa fosse os atributos referenciados.