

Introdução ao SQL/ Alterando Tabelas

Após criarmos uma tabela em um banco de dados, podemos alterar sua estrutura com o uso da cláusula ALTER TABLE. Deve-se notar que esta alteração é sempre sobre a estrutura da tabela, e não sobre os dados da tabela.

Esta alteração da tabela é efetuada de duas formas: ou acrescentando um novo campo (utilizando ADD) ou alterando-se as propriedades de um campo já existente (utilizando-se MODIFY). O comando ALTER TABLE segue a seguinte sintaxe:

```
ALTER TABLE Nome_Tabela  
ADD Nome_Campo Nova_Regra  
MODIFY Nome_Campo Nova_Regra;
```

ADD

Podemos utilizar a cláusula ADD para adicionar um novo campo em uma tabela, onde devemos definir seu tipo da mesma forma como fazemos ao criar um campo em uma nova tabela. Por exemplo, imaginemos que devemos inserir mais um campo chamado Telefone Comercial no final da tabela Cliente:

```
ALTER TABLE Cliente  
ADD Tel_Comercial INT;
```

Incluir o registro para que não fique null:

```
Update funcionario set tel_comercial = 965270372  
where cod_func = 1;
```

Da mesma forma, campos com tamanhos definidos, devem ter seu tamanho especificado. Assim se desejamos inserir mais um campo do tipo VarChar, chamado por exemplo de Nome da Mãe em nossa tabela Cliente fazemos:

```
ALTER TABLE Cliente  
ADD Nome_Mae VARCHAR(60);
```

Caso desejemos que um campo seja inserido antes ou depois de um determinado campo de nossa tabela, utilizamos a seguinte sintaxe:

```
ALTER TABLE Nome_Tabela  
ADD Nome_Campo Nova_Regra [FIRST|AFTER] Campo_Determinado;
```

Assim, se inserirmos FIRST, o novo campo será criado antes do campo escolhido da tabela. Se utilizarmos AFTER, depois. Desta forma se quisermos incluir o campo Nome_Mãe imediatamente depois do Nome do Cliente, digitamos:

```
ALTER TABLE Cliente  
ADD Nome_Pai VARCHAR(60) AFTER Nome;
```

Se desejarmos eliminar uma coluna de nossa tabela, utilizamos a seguinte sintaxe:

```
ALTER TABLE Nome_Tabela  
DROP nome_campo;
```

Alguns gerenciadores também utilizam a cláusula DROP COLUMN ao invés de somente DROP.

MODIFY

Podemos utilizar a cláusula MODIFY para modificar as propriedades de um campo em uma tabela. Por exemplo, imaginemos que devemos modificar o tamanho de nosso campo Nome na tabela Cliente, que é de 60 para 100. Então podemos escrever:

```
ALTER TABLE Cliente  
MODIFY Nome varchar(100);
```

Com a cláusula MODIFY, também podemos alterar o tipo de um campo. Imaginemos que por um motivo qualquer desejemos alterar o tipo do campo Nome de VarChar para Char:

```
ALTER TABLE Cliente  
MODIFY Nome CHAR (100);
```

Colocar as informações em ordem crescente

Select*from Cadastro order by nome;

Colocar as informações em ordem decrescente

Select*from Cadastro order by nome desc;