

Você está no meio do nada: NULL Value.

1. O Valor NULL

Olá, cientistas e entusiastas na área de dados, hoje o pai explicará para você o misterioso valor NULL. Certamente vocês conhecem os tipos de dados ‘bonitinhos’ como, por exemplo, numéricos, alfanuméricos, lógicos (VERDADEIRO OU FALSO), caracteres, e eles atendem praticamente tudo que precisamos para o nosso negócio. Porém há uma situação excepcional, um dado que precisa ser representado em alguns casos, porém não o conhece. Ele não é numérico, não é uma letra, também não um espaço vazio, um valor boolean, ele é algo que você não sabe, ele é o valor NULL. Bem-vindo, você está no meio do nada.



Para definir um tipo de dado como NULL, basta colocar a palavra reservada NULL depois do atributo quando estiver construindo o modelo lógico numa ferramenta CASE. E agora você pergunta: em que casos o NULL pode ser útil? Suponha que você tenha uma lista de pessoa, que possuem os seguintes dados: RG, nome, data de nascimento, idade, sexo e ‘alistável’ (um campo adicional booleano que diz se uma pessoa pode ser alistar no exercito ou não). A regra para se alistar é ser do sexo masculino e possuir a idade entre 18 a 45 anos. Nos demais casos, não é alistável. Definindo os dados podemos mapear da seguinte maneira:

tabelaPessoas	
rg	VARCHAR(9),
nome	VARCHAR(50),
dataNascimento	DATE,
idade	NUMBER,
sexo	CHARACTER,
eAlistavel	BOOLEAN NULL,
PRIMARY KEY (rg) NOT NULL	

tabelaPessoas					
rg	nome	dataNascimento	idade	sexo	eAlistavel

Dê atenção especial ao campo ‘eAlistavel’: precisamos dos demais dados (idade e sexo) para gerar ele, enquanto não acontecer, o valor é desconhecido, ou seja, precisamos dizer ao SGBD que o valor é NULL por padrão. Quando tivermos a idade e sexo, o ‘eAlistavel’ deixa de se NULL e se torna um BOOLEAN conforme definimos antes. Se dermos um comando ‘DESCRIBE tabelaPessoas’ o SGBD retornará exatamente isso, dizendo que ele é tipo com a restrição NULL.

Agora vamos começar a popular nossa tabela com os seguintes dados. Notaremos que se a idade e/ou sexo estiver ausente, não é possível gerar um valor para 'eAlistavel', sendo definido como NULL. Caso contrário, assume o valor de 'TRUE/FALSE'.

tabelaPessoas					
rg	nome	dataNascimento	idade	sexo	eAlistavel
<u>1</u>	Carlos	13/08/1989	30	M	TRUE
<u>2</u>	Ana	15/04/1990	-	F	NULL
<u>3</u>	Victor	20/03/1985	45	-	NULL
<u>4</u>	Paulo	31/01/1976	43	M	TRUE

E quando quisermos garantir que um valor não seja NULO, ou seja, sempre seja conhecido, basta apenas inserir a restrição 'NOT NULL' ainda na definição do esquema lógico. Inclusive isso é uma regra para as chaves primárias, que jamais podem ser NULAS.

Podemos comparar colunas com valores NULOS através das expressões 'IS NOT NULL' (significa que não é nulo) ou 'IS NULL' (é nulo). O SGBD entenderá perfeitamente o que estamos fazendo. Suponha, por exemplo, que queremos encontrar todas as pessoas que são passíveis de alistáveis ao exército (idade e sexo conhecidos), basta executar o seguinte comando.

```
SELECT nome, idade, sexo
FROM tabelaPessoas
WHERE eAlistavel IS NOT NULL
```

2. Função de Conversão NVL

Existem alguns cuidados quando trabalhamos com valores NULOS. Primeiramente, eles não podem ser comparados com qualquer outro tipo de dados. Especialmente se tentarmos realizar alguma operação aritmética que inclui valores NULOS, o resultado final será NULO. Algumas vezes precisamos converter esses valores nulos em outros valores: é aí que entra a função de conversão NVL.

NVL (Expression1, Expression2)

Se a 'Expression1' é NULA retornará a 'Expression2', caso contrário, retornará a 'Expression1'. Retornando ao nosso exemplo, suponha que queiramos encontrar todas as pessoas, porém com uma ressalva: quando o campo 'eAlistavel' for desconhecido (NULL), iremos inserir o valor 'FALSE', assim:

```
SELECT nome, idade, sexo, NVL(eAlistavel,FALSE)
FROM tabelaPessoas
```

O resultado será assim:

tabelaPessoas					
rg	nome	dataNascimento	idade	sexo	eAlistavel
<u>1</u>	Carlos	13/08/1989	30	M	TRUE
<u>2</u>	Ana	15/04/1990	-	F	FALSE
<u>3</u>	Victor	20/03/1985	45	-	FALSE

<u>4</u>	Paulo	31/01/1976	43	M	TRUE
----------	-------	------------	----	---	------

Há outras considerações sobre o valor NULL, porém os assuntos principais nós exploramos aqui. Quando não conhecemos o valor, podemos atribuir a restrição NULL ao campo e trabalhar com ela até termos conhecimento do valor. Com o NULL estamos no meio do nada, porém com um luzeiro para clarear o caminho depois das nossas explicações. Abraços e até a próxima.