

## Para saber mais: Singular e plural

Algumas vezes algo que parece trivial, como criar nomes para as tabelas de um banco, pode virar uma questão um pouco mais complexa - uma dessas questões é o singular x plural nas tabelas SQL.

Quando usamos ORMs (como o Sequelize) ou outros frameworks (como o Rails se estiver trabalhando com a linguagem Ruby) para trabalhar com os bancos de dados, o **comportamento padrão** dessas ferramentas é “pluralizar” automaticamente todos os nomes de tabelas. Ou seja, quando criamos a tabela `Person` (Pessoa em inglês) através do Sequelize, ele vai se conectar ao banco e criar a tabela com o nome de `People` (Pessoas em inglês). Isso vale para qualquer nome: `Name` (Nome) se torna `Names` (Nomes) e por aí vai.

Quando se trabalha com nomes em inglês isso não costuma ser um problema em si, pois a pessoa não precisa se preocupar em avisar ao Sequelize “olha, a tabela que eu chamei de `Person` você procura no banco como `People`, ok?”; o próprio ORM já faz essa conversão e sabe exatamente como transformar palavras em inglês do singular para o plural e vice-versa.

Bom, e quando estamos usando termos em português? Aí temos que ficar de olho para prevenir este comportamento padrão do Sequelize (e de outros ORMs e frameworks). Por esse motivo, ao invés de criar a tabela com o nome de `Pessoa`, já fizemos isso no plural, `Pessoas`. Isso evita que o Sequelize tente fazer essa conversão sozinho... O que não seria um problema em `Pessoa` x `Pessoas`, mas uma das tabelas do nosso projeto se chama `Nivel`, e se deixarmos o Sequelize resolver, ele irá transformar em `Niveis` e não `Niveis`.

Há também alguns recursos do Sequelize que você pode usar para controlar alguns comportamentos padrão do ORM:

- Utilizando a opção `freezeTableName` para “congelar” o nome da tabela, assim o Sequelize não vai tentar pluralizar:

```
sequelize.define('User', {  
  // ... (attributes)  
}, {  
  freezeTableName: true  
});
```

- Definindo o nome da tabela manualmente:

```
sequelize.define('User', {  
  // ... (attributes)  
}, {  
  tableName: 'Employees'  
});
```

Se, durante o desenvolvimento do seu projeto, você encontrar bugs relacionados a nomes de tabelas, dê uma olhada no código dos modelos e confira os nomes! :)