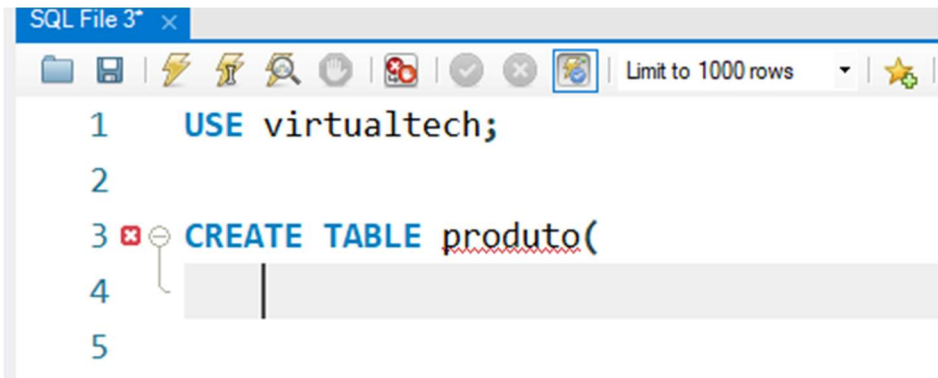


Criando uma tabela no Banco de Dados

No Workbench use a seguinte sintaxe:

CREATE TABLE produto



The screenshot shows the SQL Workbench interface with a file named 'SQL File 3*.x'. The toolbar includes icons for file operations, execution, and a 'Limit to 1000 rows' dropdown. The SQL editor contains the following code:

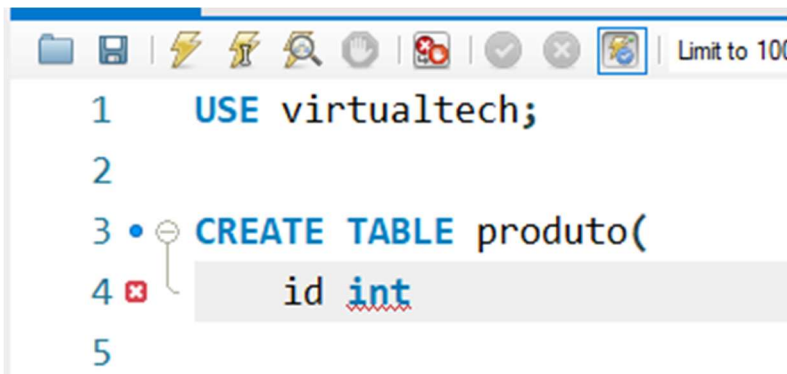
```
1  USE virtualtech;  
2  
3  CREATE TABLE produto(  
4  
5
```

O comando **CREATE** cria novas estruturas no Banco de Dados.

O **CREATE** com o parâmetro **TABLE** cria uma nova tabela no Banco de Dados, após esses parâmetros devemos colocar o nome da Tabela.

Ex: produto

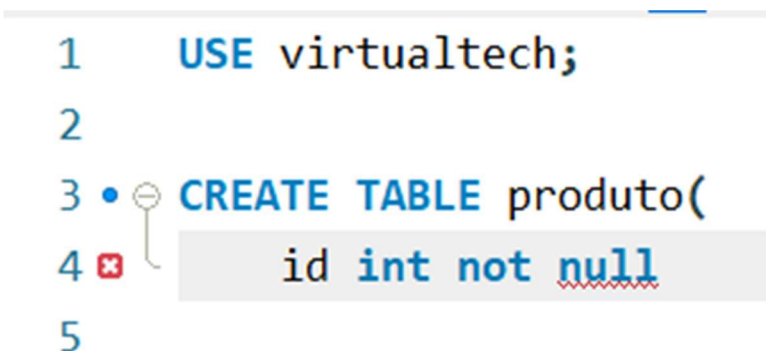
Abaixo colocamos o nome da coluna que será **id**, com valores do tipo **int** (Inteiros)



The screenshot shows the SQL Workbench interface with the same toolbar. The SQL editor now includes the column definition:

```
1  USE virtualtech;  
2  
3  CREATE TABLE produto(  
4      id int  
5
```

Abaixo colocamos que ela não poderá ter valores nulos com o parâmetro **not null**



The screenshot shows the SQL Workbench interface with the same toolbar. The SQL editor now includes the 'not null' constraint:

```
1  USE virtualtech;  
2  
3  CREATE TABLE produto(  
4      id int not null  
5
```

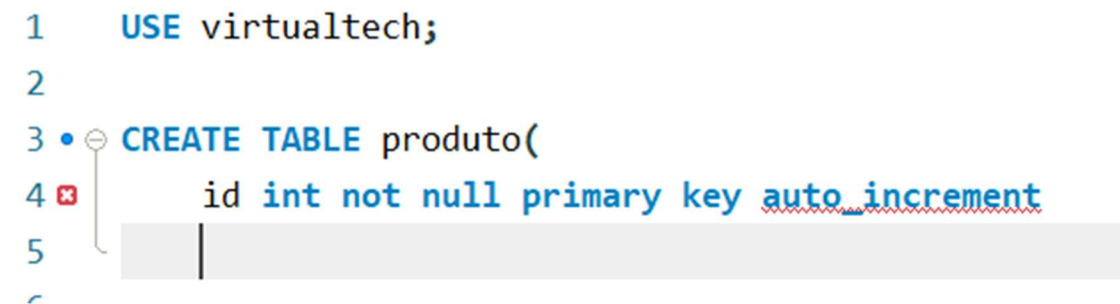
Essa coluna será a chave primária, passamos o parâmetro **primary key**



The screenshot shows a SQL editor interface with a toolbar at the top containing icons for file operations, execution, and search. Below the toolbar, the SQL code is displayed on a line-numbered background. The code consists of two lines: 'USE virtualtech;' on line 1 and 'CREATE TABLE produto(' on line 3. On line 4, the column definition 'id int not null primary key' is being entered, with the cursor at the end of the line. A red 'x' icon is visible next to the line number 4, indicating a syntax error or warning.

```
1  USE virtualtech;
2
3  CREATE TABLE produto(
4  id int not null primary key
5
```

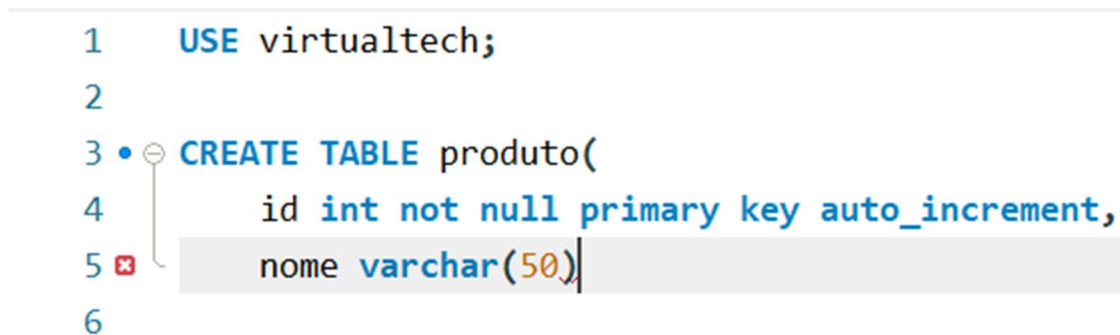
Ela será automaticamente acrescida de novos registros com o parâmetro **auto_increment** e finalizada colocamos a vírgula,



This screenshot shows the same SQL editor as before, but now the line 'id int not null primary key auto_increment' is complete on line 4. The cursor is at the end of the line. The red 'x' icon remains next to line 4.

```
1  USE virtualtech;
2
3  CREATE TABLE produto(
4  id int not null primary key auto_increment
5
```

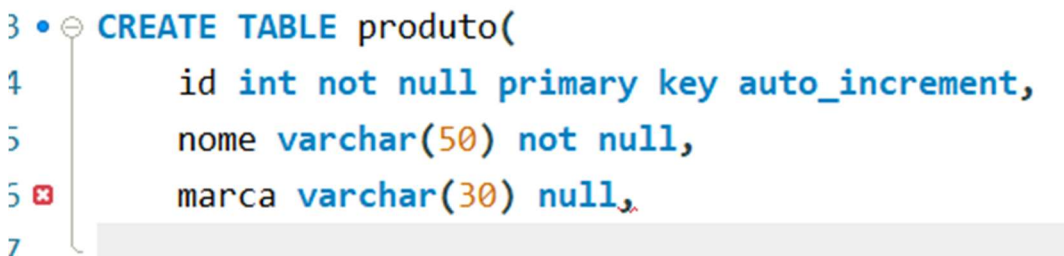
Na coluna nome usaremos o parâmetro **varchar(50)**, onde o tamanho é variável mas o máximo são 50 caracteres.



The screenshot shows the SQL code being updated. On line 5, the column definition 'nome varchar(50)' is being entered, followed by a closing parenthesis and a comma. The cursor is at the end of the line. The red 'x' icon is still present next to line 5.

```
1  USE virtualtech;
2
3  CREATE TABLE produto(
4  id int not null primary key auto_increment,
5  nome varchar(50)
6
```

Na coluna marca usaremos os mesmos parâmetros, quando não declaramos que não aceitamos valores nulos por padrão é aceito.



The screenshot shows the final SQL code. On line 6, the column definition 'marca varchar(30) null' is being entered. The cursor is at the end of the line. The red 'x' icon is still present next to line 6.

```
3  CREATE TABLE produto(
4  id int not null primary key auto_increment,
5  nome varchar(50) not null,
6  marca varchar(30) null,
7
```

Na coluna preço, usamos o parâmetro **decimal(8,2)** onde a quantidade máxima de caracteres será 8 e terá **2** casas decimais

```
CREATE TABLE produto(  
    id int not null primary key auto_increment,  
    nome varchar(50) not null,  
    marca varchar(30) null,  
    quantidade int,  
    preco decimal(8,2)  
);
```

Agora vamos verificar a tabela criada com os comandos **SELECT * FROM produto**

The screenshot shows a SQL client window with the following SQL commands entered:

```
1 USE virtualtech;  
2  
3 • SELECT *  
4 FROM produto;
```

Below the query editor, the 'Result Grid' is displayed, showing the columns and their current values:

id	nome	marca	quantidade	preco
*	NULL	NULL	NULL	NULL