

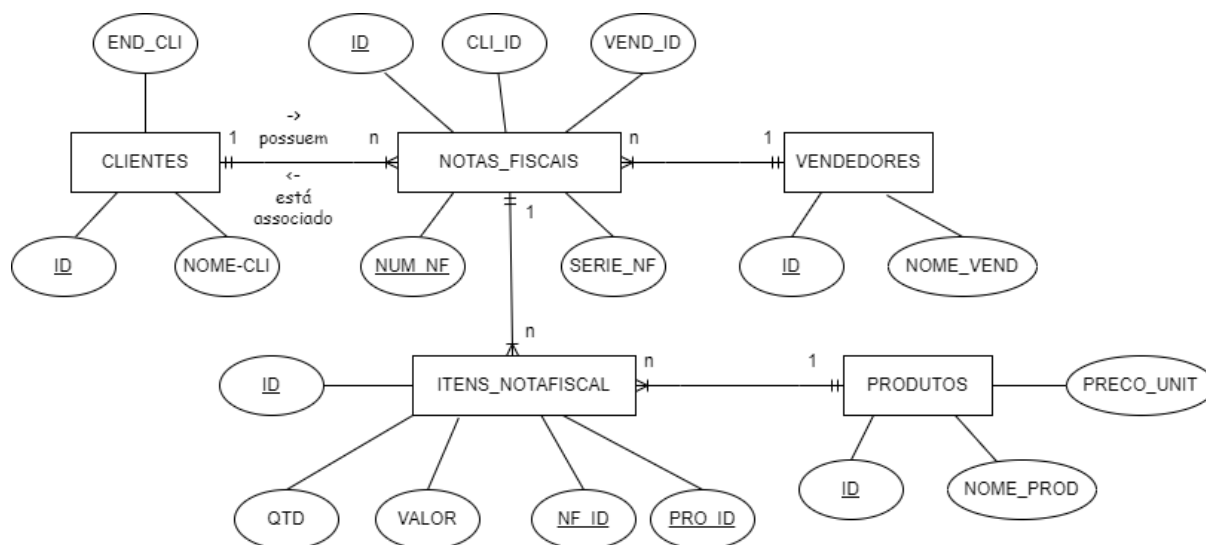
Instituto Federal de Educação, Ciência e Tecnologia do Ceará Campus Fortaleza -
Avaliação Prática

Alunas: Ana Beatriz Costa Gomes e Iara de Matos Galdino

Disciplina: Banco de Dados

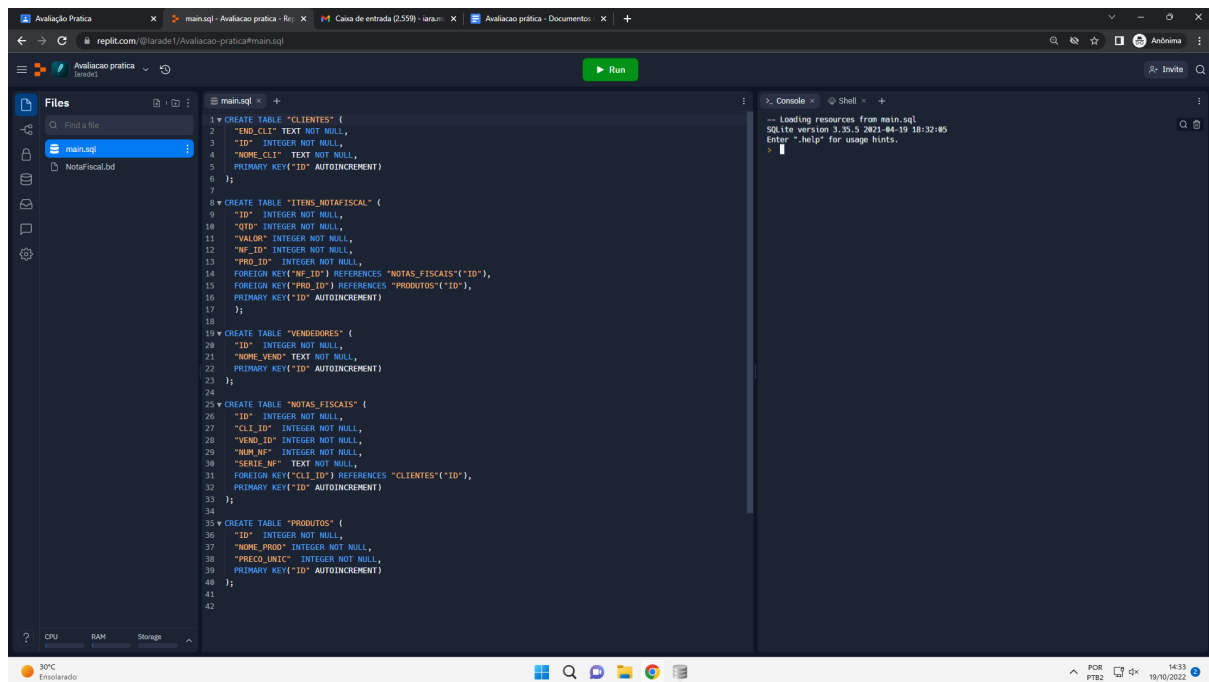
Professor: Ricardo Duarte Taveira

Curso Integrado de Informática - P4



Dado o Modelo de Entidade de Relacionamento (NOTA FISCAL) anexo abaixo pede-se:

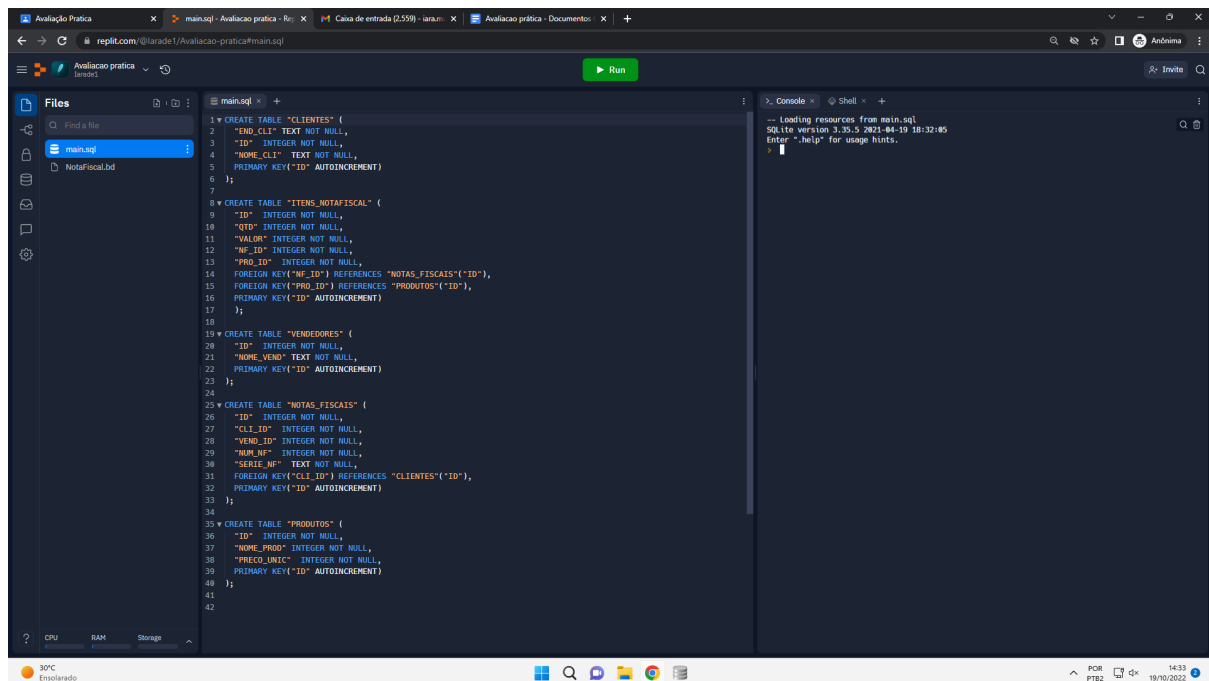
1) Crie a implementação das tabelas e relacionamentos usando o SQL;



The screenshot shows a Replit environment with a file explorer on the left containing 'main.sql' and 'NotaFiscal.tbl'. The main editor displays SQL code for creating several tables: 'CLIENTES', 'ITENS_NOTAFISCAL', 'VENDEDORES', 'NOTAS_FISCAIS', and 'PRODUTOS'. The code includes primary and foreign key constraints. The console on the right shows the output of running the script, indicating that the resources were loaded successfully and the SQLite version is 3.35.5.

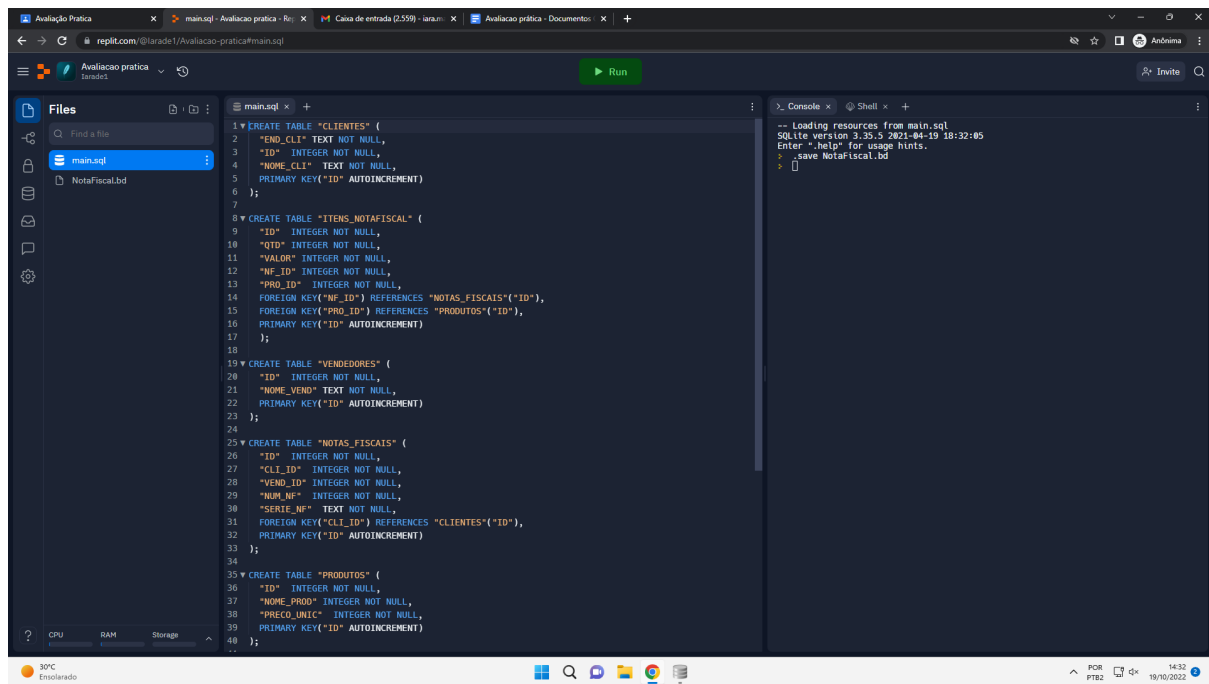
```
1 CREATE TABLE "CLIENTES" (  
2   "END_CLI" TEXT NOT NULL,  
3   "ID" INTEGER NOT NULL,  
4   "NOME_CLI" TEXT NOT NULL,  
5   PRIMARY KEY("ID" AUTOINCREMENT)  
6 );  
7  
8 CREATE TABLE "ITENS_NOTAFISCAL" (  
9   "ID" INTEGER NOT NULL,  
10  "QTD" INTEGER NOT NULL,  
11  "VALOR" INTEGER NOT NULL,  
12  "NF_ID" INTEGER NOT NULL,  
13  "PROD_ID" INTEGER NOT NULL,  
14  FOREIGN KEY("NF_ID") REFERENCES "NOTAS_FISCAIS"("ID"),  
15  FOREIGN KEY("PROD_ID") REFERENCES "PRODUTOS"("ID"),  
16  PRIMARY KEY("ID" AUTOINCREMENT)  
17 );  
18  
19 CREATE TABLE "VENDEDORES" (  
20   "ID" INTEGER NOT NULL,  
21   "NOME_VEND" TEXT NOT NULL,  
22   PRIMARY KEY("ID" AUTOINCREMENT)  
23 );  
24  
25 CREATE TABLE "NOTAS_FISCAIS" (  
26   "ID" INTEGER NOT NULL,  
27   "CLI_ID" INTEGER NOT NULL,  
28   "VEND_ID" INTEGER NOT NULL,  
29   "NUM_NF" INTEGER NOT NULL,  
30   "SERIE_NF" TEXT NOT NULL,  
31   FOREIGN KEY("CLI_ID") REFERENCES "CLIENTES"("ID"),  
32   PRIMARY KEY("ID" AUTOINCREMENT)  
33 );  
34  
35 CREATE TABLE "PRODUTOS" (  
36   "ID" INTEGER NOT NULL,  
37   "NOME_PROD" INTEGER NOT NULL,  
38   "PRECO_UNIC" INTEGER NOT NULL,  
39   PRIMARY KEY("ID" AUTOINCREMENT)  
40 );  
41  
42
```

2) Use o Replit para documentar os scripts SQL de forma que possam ser acessados via link registrado como entrega da tarefa no Google Classroom;



This screenshot is identical to the one above, showing the same Replit environment with the same SQL code for creating tables and relationships. The console output also matches, showing the successful loading of resources and the SQLite version.

3) Salve o banco de dados como NotaFiscal.bd no diretório do Replit;



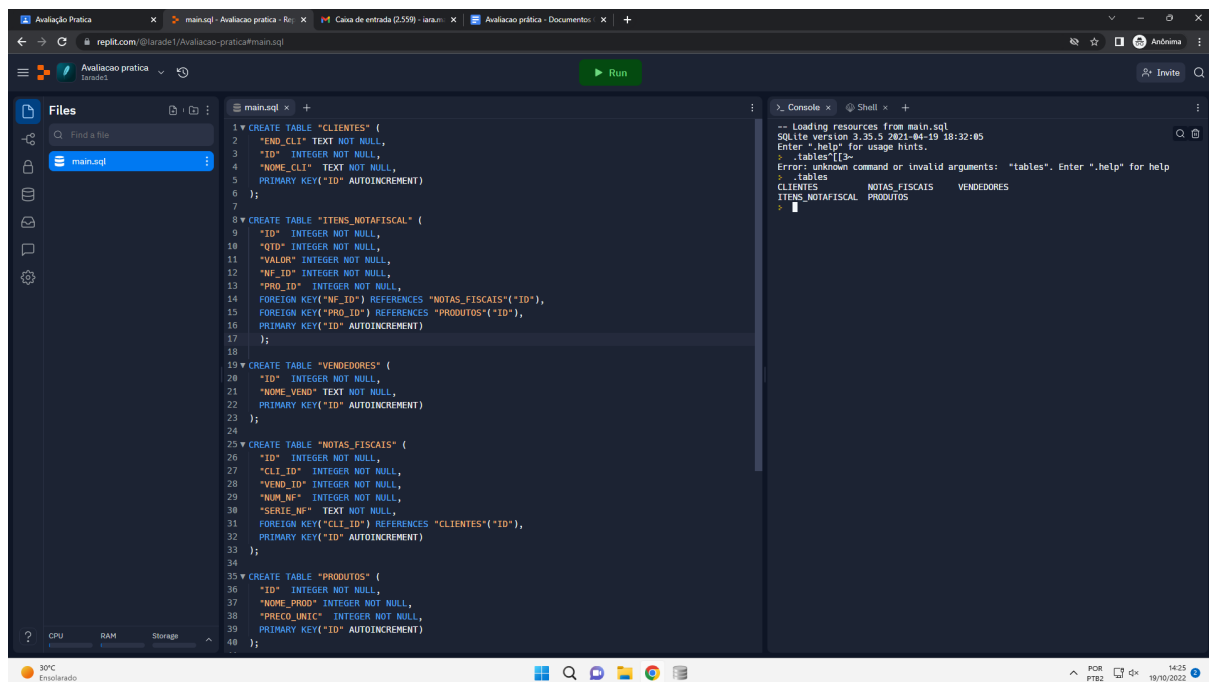
The screenshot shows a Replit environment with a file explorer on the left containing 'main.sql' and 'NotaFiscal.bd'. The main editor displays SQL code for creating several tables: 'CLIENTES', 'ITENS_NOTAFISCAL', 'VENDEDORES', 'NOTAS_FISCAIS', and 'PRODUTOS'. The console on the right shows the output of the SQL execution, including the SQLite version and a message indicating that the database was saved as 'NotaFiscal.bd'.

```
1 CREATE TABLE "CLIENTES" (
2   "END_CLI" TEXT NOT NULL,
3   "ID" INTEGER NOT NULL,
4   "NOME_CLI" TEXT NOT NULL,
5   PRIMARY KEY("ID" AUTOINCREMENT)
6 );
7
8 CREATE TABLE "ITENS_NOTAFISCAL" (
9   "ID" INTEGER NOT NULL,
10  "QTD" INTEGER NOT NULL,
11  "VALOR" INTEGER NOT NULL,
12  "NF_ID" INTEGER NOT NULL,
13  "PROD_ID" INTEGER NOT NULL,
14  FOREIGN KEY("NF_ID") REFERENCES "NOTAS_FISCAIS"("ID"),
15  FOREIGN KEY("PROD_ID") REFERENCES "PRODUTOS"("ID"),
16  PRIMARY KEY("ID" AUTOINCREMENT)
17 );
18
19 CREATE TABLE "VENDEDORES" (
20   "ID" INTEGER NOT NULL,
21   "NOME_VEND" TEXT NOT NULL,
22   PRIMARY KEY("ID" AUTOINCREMENT)
23 );
24
25 CREATE TABLE "NOTAS_FISCAIS" (
26   "ID" INTEGER NOT NULL,
27   "CLI_ID" INTEGER NOT NULL,
28   "VEND_ID" INTEGER NOT NULL,
29   "NUM_NF" INTEGER NOT NULL,
30   "SERIE_NF" TEXT NOT NULL,
31   FOREIGN KEY("CLI_ID") REFERENCES "CLIENTES"("ID"),
32   PRIMARY KEY("ID" AUTOINCREMENT)
33 );
34
35 CREATE TABLE "PRODUTOS" (
36   "ID" INTEGER NOT NULL,
37   "NOME_PROD" INTEGER NOT NULL,
38   "PRECO_UNIC" INTEGER NOT NULL,
39   PRIMARY KEY("ID" AUTOINCREMENT)
40 );
```

Console output:

```
-- Loading resources from main.sql
SQLite version 3.35.5 2021-04-19 18:32:05
Enter ".help" for usage hints.
> save NotaFiscal.bd
```

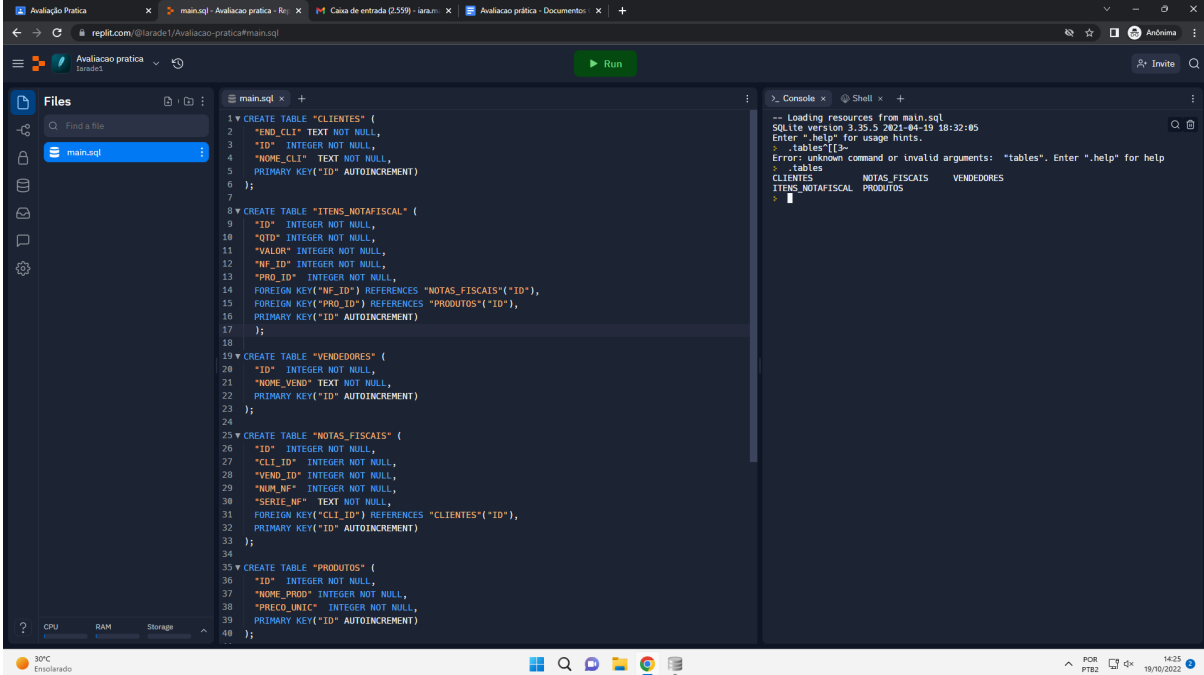
4) Mostre as tabelas criadas usando o comando .tables na interface de comando no SQLite;



The screenshot shows the same Replit environment as before, but the console now displays the output of the '.tables' command. It lists the names of the tables created: 'CLIENTES', 'NOTAS_FISCAIS', 'VENDEDORES', 'ITENS_NOTAFISCAL', and 'PRODUTOS'.

```
-- Loading resources from main.sql
SQLite version 3.35.5 2021-04-19 18:32:05
Enter ".help" for usage hints.
> .tables[[3-
Error: unknown command or invalid arguments: "tables". Enter ".help" for help
> .tables
CLIENTES      NOTAS_FISCAIS  VENDEDORES
ITENS_NOTAFISCAL  PRODUTOS
>
```

5) Faça um print da saída do comando e o anexe a entrega da tarefa no Google Classroom.



The screenshot shows a Replit IDE interface with a dark theme. The left sidebar contains a 'Files' panel with a file named 'main.sql'. The main editor area displays SQL code for creating several tables: 'CLIENTES', 'ITENS_NOTAFISCAL', 'VENDEDORES', 'NOTAS_FISCAIS', and 'PRODUTOS'. The code includes primary and foreign key constraints. The right sidebar features a 'Console' panel showing the output of the SQL execution. The output indicates that the resources were loaded successfully, the SQLite version is 3.35.5, and the command '.tables' was executed, listing the tables: 'CLIENTES', 'ITENS_NOTAFISCAL', 'NOTAS_FISCAIS', and 'PRODUTOS'. The bottom status bar shows the system temperature as 30°C and the environment as 'Ensolarado'.

```
1 CREATE TABLE "CLIENTES" (  
2   "END_CLI" TEXT NOT NULL,  
3   "ID" INTEGER NOT NULL,  
4   "NOME_CLI" TEXT NOT NULL,  
5   PRIMARY KEY("ID" AUTOINCREMENT)  
6 );  
7  
8 CREATE TABLE "ITENS_NOTAFISCAL" (  
9   "ID" INTEGER NOT NULL,  
10  "QTD" INTEGER NOT NULL,  
11  "VALOR" INTEGER NOT NULL,  
12  "NF_ID" INTEGER NOT NULL,  
13  "PRO_ID" INTEGER NOT NULL,  
14  FOREIGN KEY("NF_ID") REFERENCES "NOTAS_FISCAIS"("ID"),  
15  FOREIGN KEY("PRO_ID") REFERENCES "PRODUTOS"("ID"),  
16  PRIMARY KEY("ID" AUTOINCREMENT)  
17 );  
18  
19 CREATE TABLE "VENDEDORES" (  
20   "ID" INTEGER NOT NULL,  
21   "NOME_VEND" TEXT NOT NULL,  
22   PRIMARY KEY("ID" AUTOINCREMENT)  
23 );  
24  
25 CREATE TABLE "NOTAS_FISCAIS" (  
26   "ID" INTEGER NOT NULL,  
27   "CLI_ID" INTEGER NOT NULL,  
28   "VEND_ID" INTEGER NOT NULL,  
29   "NUM_NF" INTEGER NOT NULL,  
30   "SERIE_NF" TEXT NOT NULL,  
31   FOREIGN KEY("CLI_ID") REFERENCES "CLIENTES"("ID"),  
32   PRIMARY KEY("ID" AUTOINCREMENT)  
33 );  
34  
35 CREATE TABLE "PRODUTOS" (  
36   "ID" INTEGER NOT NULL,  
37   "NOME_PROD" INTEGER NOT NULL,  
38   "PRECO_UNIC" INTEGER NOT NULL,  
39   PRIMARY KEY("ID" AUTOINCREMENT)  
40 );
```

```
-- Loading resources from main.sql  
SQLite version 3.35.5 2021-04-19 18:32:05  
Enter ".help" for usage hints.  
> .tables  
Error: unknown command or invalid arguments: ".tables". Enter ".help" for help  
> .tables  
CLIENTES          NOTAS_FISCAIS      VENDEDORES  
ITENS_NOTAFISCAL  PRODUTOS
```