

1:N =====

## CODIGO:

- CREATE:

```
create table TB_CLIENTE(idCliente integer primary key AUTOINCREMENT,nomeCliente text NOT NULL, endereco text NOT NULL);
```

```
create table TB_PEDIDOS(idPedidos integer primary key AUTOINCREMENT, nomePedido text NOT NULL, FOREIGN KEY (idPedidos) REFERENCES TB_CLIENTE (idPedidos));
```

```
insert into TB_CLIENTE( nomeCliente, endereco) values ( 'Maria Carla', 'Rua da Ajuda 99');
insert into TB_CLIENTE( nomeCliente, endereco) values ( 'Alberto Dantas', 'Rua da saudade 66');
insert into TB_CLIENTE( nomeCliente, endereco) values ( 'Dom Juan Ramon', 'Rua da Condolência 33');
```

```
insert into TB_PEDIDOS( nomePedido) values ( 'Frango assado com batatas');
insert into TB_PEDIDOS( nomePedido) values ( 'Ensopado de batatas');
insert into TB_PEDIDOS( nomePedido) values ( 'Batata assada ao molho');
```

```
sqlite> create table TB_CLIENTE(idCliente integer primary key AUTOINCREMENT,nomeCliente text NOT NULL, endereco text NOT NULL);
sqlite> create table TB_PEDIDOS(idPedidos integer primary key AUTOINCREMENT, nomePedido text NOT NULL, FOREIGN KEY (idPedidos) REFERENCES TB_CLIENTE (idPedidos));

sqlite> insert into TB_CLIENTE( nomeCliente, endereco) values ( 'Maria Carla', 'Rua da Ajuda 99');
sqlite> insert into TB_CLIENTE( nomeCliente, endereco) values ( 'Alberto Dantas', 'Rua da saudade 66');
sqlite> insert into TB_CLIENTE( nomeCliente, endereco) values ( 'Dom Juan Ramon', 'Rua da Condolência 33');
sqlite> insert into TB_PEDIDOS( nomePedido) values ( 'Frango assado com batatas');
sqlite> insert into TB_PEDIDOS( nomePedido) values ( 'Ensopado de batatas');
sqlite> insert into TB_PEDIDOS( nomePedido) values ( 'Batata assada ao molho');
sqlite>
```

- READ

```
select TB_CLIENTE.nomeCliente, TB_PEDIDOS.nomePedido from TB_CLIENTE, TB_PEDIDOS where TB_CLIENTE.idCliente = TB_PEDIDOS.idPedidos;
```

```
select * from TB_PEDIDOS;
select * from TB_CLIENTE;
```

```
sqlite> select TB_CLIENTE.nomeCliente, TB_PEDIDOS.nomePedido from TB_CLIENTE, TB_PEDIDOS where TB_CLIENTE.idCliente = TB_PEDIDOS.idPedidos;
Maria Carla|Frango assado com batatas
Alberto Dantas|Ensopado de batatas
Dom Juan Ramon|Batata assada ao molho
```

```
sqlite> select * from TB_PEDIDOS;
1|Frango assado com batatas
2|Ensopado de batatas
3|Batata assada ao molho
```

```
sqlite> select * from TB_CLIENTE;
1|Maria Carla|Rua da Ajuda 99
2|Alberto Dantas|Rua da saudade 66
3|Dom Juan Ramon|Rua da Condolência 33
```

- UPDATE:

update TB\_CLIENTE set nomeCliente= 'Lucas Andrade' where nomeCliente= 'Alberto Dantas' ;  
select \* from TB\_CLIENTE;

```
sqlite> update TB_CLIENTE set nomeCliente= 'Lucas Andrade' where nomeCliente= 'Alberto Dantas' ;
sqlite> select * from TB_CLIENTE;
1|Maria Carla|Rua da Ajuda 99
2|Lucas Andrade|Rua da saudade 66
3|Dom Juan Ramon|Rua da Condolência 33
```

- DELETE:

delete from TB\_PEDIDOS where nomePedido = 'Ensopado de batatas';  
select \* from TB\_PEDIDOS;

```
sqlite> delete from TB_PEDIDOS where nomePedido = 'Ensopado de batatas';
sqlite> select * from TB_PEDIDOS;
1|Frango assado com batatas
3|Batata assada ao molho
```

//alter table TB\_PEDIDOS add id\_cliente int foreign key references TB\_CLIENTE;

1:1 =====

CODIGO:

- CREATE:

create table TB\_ALUNO(idAluno integer primary key AUTOINCREMENT, nomeAluno text NOT NULL, endereço text NOT NULL);

create table TB\_MATRICULA(idMatricula integer primary key AUTOINCREMENT, numeroMatricula int, FOREIGN KEY (idMatricula) REFERENCES TB\_ALUNO (id));

insert into TB\_ALUNO(nomeAluno, endereço) values ( 'Marcos Paulo', 'Rua Marechal de Andrade');

insert into TB\_ALUNO(nomeAluno, endereço) values ( 'Maria Machado', 'Macunaima');

insert into TB\_ALUNO( nomeAluno, endereço) values ( 'Marcelo Abreu', 'Andrade de Sousa');

insert into TB\_MATRICULA(numeroMatricula) values ( 123);

insert into TB\_MATRICULA(numeroMatricula) values ( 456);

insert into TB\_MATRICULA( numeroMatricula) values ( 789);

```
sqlite> create table TB_ALUNO(idAluno integer primary key AUTOINCREMENT, nomeAluno text NOT NULL, endereço text NOT NULL);
sqlite> create table TB_MATRICULA(idMatricula integer primary key AUTOINCREMENT, numeroMatricula int, FOREIGN KEY (idMatricula) REFERENCES TB_ALUNO (id));
```

```
sqlite> insert into TB_ALUNO(nomeAluno, endereço) values ( 'Marcos Paulo', 'Rua Marechal de Andrade');
sqlite> insert into TB_ALUNO(nomeAluno, endereço) values ( 'Maria Machado', 'Macunaima');
sqlite> insert into TB_ALUNO( nomeAluno, endereço) values ( 'Marcelo Abreu', 'Andrade de Sousa');
sqlite> insert into TB_MATRICULA(numeroMatricula) values ( 123);
sqlite> insert into TB_MATRICULA(numeroMatricula) values ( 456);
sqlite> insert into TB_MATRICULA( numeroMatricula) values ( 789);
```

- READ

select \* from TB\_ALUNO;

select \* from TB\_MATRICULA;

```
sqlite> select * from TB_ALUNO;
1|Marcos Paulo|Rua Marechal de Andrade
2|Maria Machado|Macunaima
3|Marcelo Abreu|Andrade de Sousa
```

```
sqlite> select * from TB_MATRICULA;
1|123
2|456
3|789
```

- UPDATE:

update TB\_ALUNO set nomeAluno= 'Marta' where nomeAluno= 'Marcos Paulo';  
select \* from TB\_ALUNO;

```
sqlite> update TB_ALUNO set nomeAluno= 'Marta' where nomeAluno= 'Marcos Paulo';
sqlite> select * from TB_ALUNO;
1|Marta|Rua Marechal de Andrade
2|Maria Machado|Macunaima
3|Marcelo Abreu|Andrade de Sousa
```

- DELETE:

delete from TB\_ALUNO where nomeAluno= 'Marcelo Abreu';  
select \* from TB\_ALUNO;

```
sqlite> delete from TB_ALUNO where nomeAluno= 'Marcelo Abreu';
sqlite> select * from TB_ALUNO;
1|Marta|Rua Marechal de Andrade
2|Maria Machado|Macunaima
```

N:N =====

### CODIGO:

- CREATE:

```
create table TB_ATOR(idAtor integer primary key AUTOINCREMENT , nomeAtor text NOT NULL, generoAtor text NOT NULL);
```

```
create table TB_FILME(idFilme integer primary key AUTOINCREMENT , nomeFilme text NOT NULL, generoFilme text NOT NULL);
```

```
create table ator_filme( idAtor integer, idFilme integer, primary key (idAtor,idFilme), foreign key (idAtor) REFERENCES TB_ATOR (idAtor) ON DELETE CASCADE ON UPDATE NO ACTION, foreign key(idFilme) REFERENCES TB_FILME(idFilme) ON DELETE CASCADE ON UPDATE NO ACTION);
```

```
insert into TB_ATOR(nomeAtor,generoAtor) values ('Leonardo Ramos', 'masculino');
insert into TB_ATOR(nomeAtor,generoAtor) values ('Margot Robbie', 'feminino');
insert into TB_ATOR(nomeAtor,generoAtor) values ('Leonardo DiCaprio', 'Masculino');
```

```
insert into TB_FILME(nomeFilme, generoFilme) values ('OFMD','comedia');
insert into TB_FILME(nomeFilme, generoFilme) values ('wwits','sobrenatural');
insert into TB_FILME(nomeFilme, generoFilme) values ('Toy Story','animacao');
```

```
insert into ator_filme values (1,2);
insert into ator_filme values (3,2);
insert into ator_filme values (3,1);
```

```
sqlite> create table TB_ATOR(idAtor integer primary key AUTOINCREMENT , nomeAtor text NOT NULL, generoAtor text NOT NULL);
sqlite> create table TB_FILME(idFilme integer primary key AUTOINCREMENT , nomeFilme text NOT NULL, generoFilme text NOT NULL);
sqlite> create table ator_filme( idAtor integer, idFilme integer, primary key (idAtor,idFilme), foreign key (idAtor) REFERENCES TB_ATOR (idAtor) ON DELETE CASCADE ON UPDATE NO ACTION, foreign key(idFilme) REFERENCES TB_FILME(idFilme) ON DELETE CASCADE ON UPDATE NO ACTION);
```

```

sqlite> insert into TB_ATOR(nomeAtor,generoAtor) values ('Leonardo Ramos', 'masculino');
sqlite> insert into TB_ATOR(nomeAtor,generoAtor) values ('Margot Robbie', 'feminino');
sqlite> insert into TB_ATOR(nomeAtor,generoAtor) values ('Leonardo DiCaprio', 'Masculino');
sqlite> insert into TB_FILME(nomeFilme, generoFilme) values ('OFMD','comedia');
sqlite> insert into TB_FILME(nomeFilme, generoFilme) values ('wwits','sobrenatural');
sqlite> insert into TB_FILME(nomeFilme, generoFilme) values ('Toy Story','animacao');
sqlite> insert into ator_filme values (1,2);
sqlite> insert into ator_filme values (3,2);
sqlite> insert into ator_filme values (3,1);

```

- READ:

```

select * from TB_FILME;
select * from TB_ATOR;
select * from ator_filme;

```

```

sqlite> select * from TB_FILME;
1|OFMD|comedia
2|wwits|sobrenatural
3|Toy Story|animacao

```

```

sqlite> select * from TB_ATOR;
1|Leonardo Ramos|masculino
2|Margot Robbie|feminino
3|Leonardo DiCaprio|Masculino

```

```

sqlite> select * from ator_filme;
1|2
3|2
3|1

```

- UPDATE:

```

update TB_ATOR set nomeAtor = 'Will Smith' where idAtor = 3;
select * from TB_ATOR;

```

```

sqlite> update TB_ATOR set nomeAtor = 'Will Smith' where idAtor = 3;
sqlite> select * from TB_ATOR;
1|Leonardo Ramos|masculino
2|Margot Robbie|feminino
3|Will Smith|Masculino

```

- DELETE:

```

delete from TB_FILME where nome = 'OFMD';

```

```
select * from TB_FILME;
```

```
sqlite> delete from TB_FILME where nomeFilme = 'OFMD';  
sqlite> select * from TB_FILME;  
2|wwits|sobrenatural  
3|Toy Story|animacao  
sqlite>
```