

```

/*
use master
go
DROP DATABASE AC4_M2_Biblioteca;
GO

1

*/
--=X=-- --=X=-- --=X=-- --=X=-- --=X=-- --=X=-- --=X=-- --=X=--
-
--=X=-- DDL - Criação da base
--=X=-- --=X=-- --=X=-- --=X=-- --=X=-- --=X=-- --=X=-- --=X=--
-
CREATE DATABASE AC4_M2_Biblioteca;
GO
USE AC4_M2_Biblioteca;
GO
--Licença acadêmica para sugerir uma chave composta...
--Poderia ser chave artificial = ID
CREATE TABLE Livro (
Nome VARCHAR(255) NOT NULL
, Copia TINYINT NOT NULL --Não é auto incremental
, Status VARCHAR(30) NULL
, CONSTRAINT PK_Livro PRIMARY KEY ( Nome, Copia )
)
GO
CREATE TABLE Cliente (
ID INT NOT NULL IDENTITY(1,1)
, Nome VARCHAR(50) NOT NULL
, Telefone INT NULL --poderia ou deveria ser VARCHAR
, CONSTRAINT PK_Cliente PRIMARY KEY ( ID )
)
GO
CREATE TABLE Emprestimo (
ID INT NOT NULL IDENTITY(1,1)
, Livro VARCHAR(255) NOT NULL
, Copia TINYINT NOT NULL
, ID_Cliente INT NOT NULL
, DataEmprestimo DATETIME NOT NULL
, DataDevolucaoProposta DATETIME NULL
, DataDevolucaoEfetiva DATETIME NULL
, Multa Decimal(6,2)
, CONSTRAINT PK_Emprestimo PRIMARY KEY ( ID )
, CONSTRAINT FK_EmprestimoLivro FOREIGN KEY ( Livro, Copia )
REFERENCES Livro (
Nome, Copia )
, CONSTRAINT FK_EmprestimoCliente FOREIGN KEY ( ID_cliente )
REFERENCES Cliente (
ID )
)
GO
--=X=-- --=X=-- --=X=-- --=X=-- --=X=-- --=X=-- --=X=-- --=X=--
-
--=X=-- DML - Carga da base
--=X=-- --=X=-- --=X=-- --=X=-- --=X=-- --=X=-- --=X=-- --=X=--

```

```

-
INSERT INTO LIVRO ( nome, copia, [status] )
VALUES ('A arte da Guerra', 1, 'Atrasado')
, ('A arte da Guerra', 2, 'Livre')
, ('O Pequeno Principe', 1, 'Emprestado')
GO
INSERT INTO CLIENTE (/*ID, */ nome, telefone )
VALUES (/*1, */'José da Silva', '991235344')
, (/*2, */'Igor Pereira', '33120123')
GO
INSERT INTO Emprestimo ( Livro, Copia, ID_Cliente, DataEmprestimo,
DataDevolucaoProposta, DataDevolucaoEfetiva, Multa )

VALUES ( 'A arte da Guerra'

, 1
, (SELECT id FROM Cliente where nome = 'José da Silva')
, '20170810' --DataEmprestimo
, '20170910' --DataDevolucaoProposta
, NULL --DataDevolucaoEfetiva
, 11.00 --Ainda não devolveu, por isso a Multa Número sempre com '.'
)
, ( 'O Pequeno Principe'
, 1
, (SELECT id FROM Cliente where nome = 'Igor Pereira')
, '20170915' --DataEmprestimo
, '20171015' --DataDevolucaoProposta
, '20171015' --DataDevolucaoEfetiva
, NULL --Já devolveu, por isso sem Multa
)
--Conferências:
select * from Livro
select * from Cliente
select * from Emprestimo

```

Da AC3 temos -

- Lista com o nome dos livros disponíveis (sem duplicidades).

Lista com o nome dos livros

--disponíveis

(sem duplicidades).

```
select distinct nome
```

```
from livro
```

```
-- where status = 'Livre'
```

- Lista com o nome e o número da cópia, de cada livro+copia com status = 'livre'

Lista com o nome e o número

da cópia, de cada livro+copia

com status = 'livre'

```
SELECT Nome as [nome do livro]
```

```
, Copia as [Nr.exemplar]
```

```
FROM livro
```

```
WHERE [Status] = 'Livre'
```

- Lista com o nome do cliente, o livro+copia dos livros emprestados cuja data de devolução proposta já expirou. (ou seja, estou procurando quais livros estão atrasados, para poder atualizar seus status para atrasado e calcular sua multa).

Lista com o nome do cliente, o livro+copia dos livros emprestados cuja data de devolução proposta já expirou
(ou seja, estou procurando quais livros estão atrasados , para poder atualizar seus status para atrasado e calcular sua multa).

```
select cliente.nome as [nome do cliente],  
livro.nome as [nome do livro],  
livro.copia as [Numero Exemplar],  
emprestimo.DataDevolucaoProposta as [Data Proposta]
```

```
from livro inner join  
emprestimo on livro.nome = emprestimo.livro  
AND livro.Copia = emprestimo.Copia  
inner join cliente
```

3

```
ON emprestimo.ID_Cliente = cliente.id  
where DataDevolucaoProposta < getdate()  
and DataDevolucaoEfetiva IS NULL  
- Lista com os nomes dos clientes que pegaram emprestados livros no  
mês atual de Abril de 2018.  
Lista com os nomes dos clientes que pegaram emprestados  
livros no mês /*atual*/ de Abril de 2018.  
select Cliente.Nome as [nome do cliente],
```

```
Emprestimo.DataEmprestimo as [Data Emprestimo]
```

```
from Cliente inner join  
emprestimo on Cliente.ID = Emprestimo.ID_Cliente  
where MONTH(DataEmprestimo) = 04  
and year(DataEmprestimo) = 2018  
--and DataEmprestimo between '20180401' and '20180430 23:59:59.997'  
--and DataEmprestimo >= '20180401' and DataEmprestimo <= '20180430  
23:59:59.997'  
- Lista de Clientes que preencheram o campo Telefone no seguinte  
formato 912345678  
Lista de Clientes que preencheram o campo Telefone  
no seguinte formato 912345678  
-- Primeiro dígito 9, seguido de mais 8 números de 0 a 9  
SELECT *  
FROM Cliente  
WHERE Telefone like '[9][0-9][0-9][0-9][0-9][0-9][0-9][0-9][0-9]';  
-- um número que começa com 9  
SELECT *  
FROM Cliente  
WHERE Telefone LIKE '9%'  
-- um número com 9 chars  
SELECT *  
FROM Cliente  
WHERE Telefone LIKE '_____'
```

Na AC4 - Execute as seguintes consultas no banco de dados criado:

- Lista com o Ranking dos top 10 livros (pelo nome)
mais emprestados.

```

-- livros atualmente emprestados
truque é usar a coluna status do livro...

select  top 10
        nome as [nome do livro]
        , count(*) as Locações
from      livro
where status IN ('Atrasado', 'Emprestado') -- status <> 'Livre'
group by nome
order by Locações DESC

--teste, vou adicionar um terceiro exemplar de a arte da guerra
-- para entrar nos select ele já deve entrar emprestado.
INSERT INTO LIVRO ( nome, copia, [status] )
VALUES ('A arte da Guerra', 3, 'Emprestado')

INSERT INTO Emprestimo ( Livro, Copia, ID_Cliente, DataEmprestimo,
DataDevolucaoProposta, DataDevolucaoEfetiva, Multa )
VALUES ( 'A arte da Guerra'
, 3
, (SELECT id FROM Cliente where nome = 'Igor Pereira')
, '20170810' --DataEmprestimo
, '20170910' --DataDevolucaoProposta
, NULL --DataDevolucaoEfetiva
, 9.00 --Ainda não devolveu, por isso a Multa Número sempre com '.'
)

-- livros emprestados neste momento ou no passado...(historico)
truque é usar a tabela emprestimo....

select TOP 10 livro, count(*) as emprestimos
from emprestimo
group by livro
order by emprestimos DESC

- Lista com o nome do cliente,
o número de vezes em que ele já pagou uma multa,
e o valor total das multas já pagas.
select  cliente.nome
        , count(*) as emprestimos
        , count(emprestimo.multa) as totalMultas
        , sum(emprestimo.multa) as valorMultas
from cliente
    inner join emprestimo
    on cliente.id= emprestimo.id_cliente
--where emprestimo.multa > 0
group by cliente.nome

--lembrem-se que no exemplo acima eu adicionei multas
-- extras só para testar o select
José da Silva      1          11.00
Igor Pereira       2          100.00

- Lista dos Livros ( pelo nome )
que não foram emprestados nem uma única vez
no ano passado ( usar função de data ).

```

```

-- lista total de livros
-- exceto os que estão na
-- lista de livros emprestados no ano passado.

-- etapa 1
-- lista de livros emprestados no ano passado.
select livro
from cliente
    inner join emprestimo
    on cliente.id= emprestimo.id_cliente
where YEAR(DataEmprestimo) = YEAR(GETDATE())-1
-- YEAR( dateadd(year,-1,GETDATE()) )

-- etapa 2 - total de livros menos os emprestados
select nome
from livro
where nome NOT IN (
    select livro
    from cliente
        inner join emprestimo
        on cliente.id= emprestimo.id_cliente
    where YEAR(DataEmprestimo) = YEAR(GETDATE())-1
)

--select YEAR(GETDATE())-1 as [ ano passado ]
--      , YEAR(GETDATE()) as [ ano atual ]

--adicionando um livro no ano passado para teste
INSERT INTO Emprestimo ( Livro, Copia, ID_Cliente, DataEmprestimo,
DataDevolucaoProposta, DataDevolucaoEfetiva, Multa )
VALUES ( 'A arte da Guerra'
, 3
, (SELECT id FROM Cliente where nome = 'Igor Pereira')
, '20200810' --DataEmprestimo
, '20200910' --DataDevolucaoProposta
, NULL --DataDevolucaoEfetiva
, 9.00 --Ainda não devolveu, por isso a Multa Número sempre com '.'
)

```