

ST 767 A - Banco de Dados II

Prof. Dra. Gisele Busichia Baioco

Exercício 2 – SQL DDL e DML

Gabriel Marino - 247455

Matheus Roberto Mariano - 246029

Tauan Rodrigues Assunção de Jesus - 247599

Wara Inti Pardo - 218904

Questões

1. Criar o esquema físico da base de dados, ou seja, escrever o script para a criação das tabelas da base de dados no MS-SQL Server.

```
--garantindo que estamos no banco master--  
use master  
go
```

```
--criação do banco de dados principal--  
create database Reboque  
go
```

```
--garantindo que estamos no banco necessário--  
use Reboque
```

```
--criação das tabelas do banco--  
create table motorista (  
    codigo int not null,  
    nome char(33) not null,  
    nro_carteira int not null,  
    hora_entrada time not null,  
    hora_saida time not null,  
    primary key (codigo)  
)
```

```
create table cliente (  
    codigo int not null,  
    rg char(9) not null,  
    nome char(33) not null,  
    endereco char(44) not null,  
    primary key (codigo)  
)
```

```
create table veiculo (  
    placa char(7) not null,
```

```
marca char(11) not null,  
cor char(11) not null,  
primary key (placa)  
)  
  
create table ocorrencia (  
    codigo int not null,  
    end_busca char(44) not null,  
    end_entrega char (44) not null,  
    data date not null,  
    distancia numeric(8,2) not null,  
    preco money not null,  
    pago char(1) not null,  
    cod_motorista int not null,  
    cod_cliente int not null,  
    placa char(7) not null,  
    primary key (codigo),  
    foreign key (cod_motorista) references motorista,  
    foreign key (cod_cliente) references cliente,  
    foreign key (placa) references veiculo  
)  
go
```

--criação de índices para as chaves estrangeiras--

```
create index ix_ocorrencia_mot  
on ocorrencia (cod_motorista)
```

```
create index ix_ocorrencia_cli  
on ocorrencia (cod_cliente)
```

```
create index ix_ocorrencia_veiculo  
on ocorrencia (placa)
```

2. Fazer as seguintes operações em SQL:

- a. Inserir um novo motorista (escolha os valores para os atributos).

```
insert into motorista  
values ('1', 'Pedrinho', '123', '09:30', '13:45')
```

- b. Inserir um novo cliente (escolha os valores para os atributos).

```
insert into cliente  
values ('1', '123456789', 'Ciclano', 'Avenida Brasil 123')
```

- c. Inserir um novo veículo (escolha os valores para os atributos).

```
insert into veiculo  
values ('ABC1234', 'Renault', 'Amarelo')
```

- d. Inserir uma nova ocorrência (escolha os valores para os atributos).
`insert into ocorrencia`
`values ('1', 'Avenida Brasil 321', 'Avenida Paulista 9000', '2022-04-06', '250.12', '963.28', '0', '1', '1', 'ABC1234')`
- e. Atualizar o atributo pago para 'S' da ocorrência de código 10.
`update ocorrencia`
`set pago = '1'`
`where codigo = 10`
- f. Atualizar a hora de saída para 18:00h do motorista de código 5.
`update motorista`
`set hora_saida = '18:00'`
`where codigo = 5`
- g. Remover todas as ocorrências de um certo período (escolher o período) que já foram pagas.
`delete from ocorrencia`
`where data < '2022-01-01' and pago = '1'`
- h. Remover o veículo de placa 'AAA5555'.
`delete from veiculo`
`where placa = 'AAA5555'`
- i. Listar os motoristas que trabalham no período das 6:00h às 13:00h.
`select *`
`from motorista`
`where hora_entrada = '6:00' and hora_saida = '13:00'`
- j. Listar o número total de ocorrências já pagas (atributo pago = 'S').
`select count(*)`
`from ocorrencia`
`where pago = 'S'`
- k. Listar a data e a média de preço das ocorrências por data.
`select data 'Data', avg(preco) 'Preço Médio'`
`from ocorrencia`
`group by data`
`order by data desc`
- l. Listar o nome do cliente para todas as ocorrências que ainda não foram pagas (atributo pago = 'N').
`select nome`
`from cliente c inner join ocorrencia o`
`on c.codigo = cod_cliente`
`where pago = 'N'`

- m. Para todas as ocorrências, listar a placa do veículo, o nome do motorista que atendeu a ocorrência, a data e a distância, ordenando o resultado pela data.

```
select v.placa, nome, data, distancia
from ocorrencia o inner join veiculo v
    on o.codigo = v.placa
    inner join motorista m
    on o.cod_motorista = m.codigo
order by data desc
```

- n. Para todas as ocorrências que ainda não foram pagas (atributo pago = 'N'), listar o nome do cliente, o nome do motorista, a data e o preço do serviço solicitado, ordenando o resultado pela data e nome do cliente.

```
select c.nome, m.nome, data, preco
from ocorrencia o inner join cliente c
    on o.codigo = cod_cliente
    inner join motorista m
    on o.cod_motorista = m.codigo
order by data, c.nome desc
```