

Grupo MP25:

Ricardo Rovisco no.49487

João Mota no.49508

Jorge Silva no. 49504

Ex1

a)

```
create table PESSOA(      id integer
not null unique,      noident char(12)
not null,      nif char(12) not null,
primary key(id,noident,nif),
nproprio varchar(10) not null,
apelido varchar(10) not null,
morada varchar(150) not null,
      codpostal integer not null check(codpostal BETWEEN 999999 AND 10000000),
localidade varchar(150) not null,
      atrdisc char(2) not null check (atrdisc in ('P', 'C', 'CL'))
);
create table PROPRIETARIO(
      idpessoa integer not null primary key,
      dtnascimento date not null check (DATE_PART ('year', current_date) - DATE_PART
('year', dtnascimento) >= 18),
      foreign key (idpessoa) references PESSOA(id) on delete cascade on update
cascade );

create table CONTACTO(      idpessoa
integer not null,      ntelefone
varchar(15) not null,      primary
key(idpessoa,ntelefone),
      foreign key (idpessoa) references PESSOA(id) on delete cascade on update
cascade );
create table CONDUTOR(
      idpessoa integer not null primary key,
ncconducao varchar(20) not null unique
      check (ncconducao similar to '[A-Za-z][A-Za-z]-[0-9][0-9][0-9][0-9][0-9][0-9][0-9][0-9]'),
      dtnascimento date not null check (DATE_PART('year', current_date) -
DATE_PART('year', dtnascimento) >= 18),
      foreign key (idpessoa) references PESSOA(id) on delete cascade on update
cascade );
create table TIPOVEICULO(      tipo
integer not null primary key,
      nlugares integer not null check(nlugares between 0 and 8),
      multiplicador numeric(1),
designação char(10) not null
);
create table VEICULO(      id integer
not null unique,      matricula
varchar(10) not null      check (
matricula similar to '[A-Za-z][A-Za-
```

```

z][0-9][0-9][A-Za-z][A-Za-z]' or
matricula similar to '[0-9][0-9][A-
Za-z][A-Za-z][0-9][0-9]'),      tipo
integer not null, modelo
varchar(10) not null,      marca
varchar(10) not null,
      ano integer not null check (DATE_PART ('year', current_date) - ano >= 5),
proprietario integer not null,
      foreign key (proprietario) references PROPRIETARIO(idpessoa) on delete cascade on
update cascade,
      foreign key (tipo) references TIPOVEICULO(tipo) on delete cascade on update
cascade
); create table CONDHABILITADO(
condutor integer not null,
veiculo integer not null,
primary key(condutor,veiculo),
      foreign key (condutor) references CONDUTOR(idpessoa) on delete cascade on update
cascade,
      foreign key (veiculo) references VEICULO(id) on delete cascade on update
cascade ); create table CORVEICULO( veiculo integer not null,      cor
varchar(10) not null,      primary key(veiculo,cor),
      foreign key (veiculo) references VEICULO(id) on delete cascade on update
cascade );
create table PERIODOATIVO(      veiculo
integer not null unique,      condutor
integer not null unique,      dtinicio
timestamp not null unique,      primary
key(veiculo,condutor,dtinicio),
      dtfim timestamp,      lat
decimal(6,2) not null,      long
decimal(6,2) not null,
      foreign key (veiculo) references VEICULO(id) on delete cascade on update cascade,
foreign key (condutor) references CONDUTOR(idpessoa) on delete cascade on update
cascade
);
create table VIAGEM(
      idsistema integer not null primary key,
      hinicio time not null ,      hfim
time,      dtviagem date not null,
valestimado decimal(5,2) not null,
valfinal decimal(5,2) ,      latinicio
decimal(6,2) not null,      longinicio
decimal(6,2) not null,      latfim
decimal(6,2),
      longfim decimal(6,2),
      classificação integer not null check(classificação between 0 and 6 or NULL),
veiculo integer not null,      condutor integer not null,      dtinicio timestamp
not null,
      foreign key (veiculo) references PERIODOATIVO(veiculo) on delete cascade on
update cascade,
      foreign key (condutor) references PERIODOATIVO(condutor) on delete cascade on
update cascade,

```

```

    foreign key (dtinicio) references PERIODOACTIVO(dtinicio) on delete cascade on
update cascade
); create table CLIENTEVIAGEM(
idpessoa integer not null,
viagem integer not null,
primary key(idpessoa, viagem),
    foreign key (idpessoa) references PESSOA(id) on delete cascade on update
cascade,
    foreign key (viagem) references VIAGEM(idsistema) on delete cascade on update
cascade
);

```

b)

```

drop table CLIENTEVIAGEM cascade;
drop table VIAGEM cascade; drop
table PERIODOACTIVO cascade; drop
table CORVEICULO cascade; drop
table CONDHABILITADO cascade; drop
table VEICULO cascade; drop table
TIPOVEICULO cascade; drop table
CONDUTOR cascade; drop table
contacto cascade; drop table
PROPRIETARIO cascade; drop table
PESSOA cascade;

```

c)

```

insert into PESSOA(id, noident, nif, nproprio, apelido, morada, codpostal,
localidade,
atrdisc) values
(1, '123456789012', '123456789', 'Rodrigo', 'Rodrigues', 'morada1', '1234123', 'localidade1',
'C'),
(2, '210987654321', '987654321', 'Agosto', 'Agostinho', 'morada2', '1234123', 'localidade1',
'C'),
(3, '566771230982', '999999999', 'Antonio', 'Antunes', 'morada3', '1234123', 'localidade3', '
CL'),
(4, '987462910470', '173890128', 'Diogo', 'Diego', 'morada4', '1234123', 'localidade4', 'P'),
(5, '678781423451', '508128749', 'Peter', 'Parker', 'morada5', '1234123', 'localidade5', 'CL'
);

```

```

insert into CONTACTO(idpessoa, ntelefone) values

```

```

(1, '351213456789'),
(2, '351960322212'),
(3, '351356762831'),
(4, '351912437687'),
(5, '351524576543');

```

```

insert into PROPRIETARIO(idpessoa, dtnascimento) values

```

```

(1, '2002-02-23'),
(2, '2002-03-12'),
(3, '1996-10-01'),
(4, '1964-10-25'),

```

```
(5,'2002-11-28');
```

```
insert into TIPOVEICULO(tipo, nlugares, multiplicador, designação) values  
(1,2,1,'Normal'), (2,2,1,'Luxo'),  
(3,5,1,'Normal');
```

```
insert into CONDUTOR(idpessoa, ncconducao, dtnascimento) values  
(1,'aa-1234567','2002-11-26'),  
(2,'aa-0743736','2002-02-23');
```

```
insert into VEICULO(id, matricula, tipo, modelo, marca, ano, proprietario) values  
(1,'AA00AA',1,'RangeRover','Fiat',2015,1),  
(2,'GF67BF',2,'polo','Volkswagen',2012,2),  
(3,'50Z053',3,'a3','Audi',2013,3);
```

```
insert into CORVEICULO(veiculo,cor) values  
(1,'AMARELO'),  
(2,'VERDE'),  
(3,'AZUL');
```

```
insert into CONDHABILITADO(condutor, veiculo) values  
(1,1),  
(2,2),  
(1,3);
```

```
insert into PERIODOATIVO(veiculo,condutor, dtinicio, dtfim, lat, long) values  
(2,2,'2021-11-29 13:13:13','2021-11-29 18:14:44',72.99,8.62),  
(1,1,'2021-11-29 11:11:11','2021-11-29 15:12:15',234.12,126.1);
```

```
insert into VIAGEM(idsistema, hinicio, hfim, dtviagem, valestimado, valfinal,  
latinicio,  
longinicio, latfim, longfim, classificação, veiculo, condutor, dtinicio) values  
(1,'11:11:11','12:12:12','2021-11-29',124.12 , NULL,123.12 ,56.12,  
NULL,NULL,1,1,1,'2021-11-29 11:11:11'),  
(2,'12:12:12','13:13:13','2021-12-29',253.12 ,253.12  
,35.12,54.12,456.12,156.12,2,2,2,'2021-11-29 13:13:13'),  
(3,'13:13:13','14:14:14','2021-11-29',743.12 ,743.12  
,560.12,755.12,93.12,256.12,2,1,1,'2021-11-29 13:13:13'),  
(4,'14:14:14','15:15:15','2021-11-29',359.12 ,156.12  
,76.12,45.12,12.12,264.12,1,2,2,'2021-11-29 11:11:11'),  
(5,'15:15:15','16:16:16','2020-11-29',16.12 ,111.12  
,99.12,199.12,83.12,56.12,1,2,2,'2021-11-29 11:11:11');
```

```
insert into CLIENTEVIAGEM(idpessoa, viagem) values  
(2,1),  
(2,2),  
(3,3),  
(4,4),  
(3,5);
```

d)

```
delete from CLIENTEVIAGEM;
delete from VIAGEM; delete
from PERIODOACTIVO; delete
from CORVEICULO; delete
from CONDHABILITADO; delete
from VEICULO; delete from
TIPOVEICULO; delete from
CONDUTOR; delete from
CONTACTO; delete from
PROPRIETARIO; delete from
PESSOA;
```

Ex 2

π projeção \square seleção p renomeação \bowtie junção

\Join agregação \cap interseção X produto cartesiano

a) Implementação 1:

$\pi(\text{nid}, \text{nif}, \text{idsistema}, \text{dtviagem}, \text{hinicio}, \text{hfim}, \text{valfinal}) (\text{CLIENTEVIAGEM} \bowtie (\text{idpessoa} = \text{id}) (\sigma (\text{ATRdisc} = \text{cl}, \text{nproprio} = \text{antonio}, \text{apelido} = \text{silva}) (\text{PESSOA})))$

Implementação 2:

$\text{ID} \leftarrow \pi (\text{id}) (\sigma (\text{atrdisc} = \text{cl}, \text{nproprio} = \text{António}, \text{apelido} = \text{Silva}) (\text{PESSOA}))$

$\text{VIAGENS} \leftarrow \pi (\text{idsistema}, \text{dtviagem}, \text{hinicio}, \text{hfim}, \text{valfinal}) (\square (\text{ID} = \text{idpessoa}) (\text{CLIENTEVIAGEM}))$

$\text{PESS} \leftarrow \pi (\text{nident}, \text{nif}) (\sigma (\text{atrdisc} = \text{cl}, \text{nproprio} = \text{António}, \text{apelido} = \text{Silva}) (\text{PESSOA}))$

$\text{RESULT} \leftarrow \pi (\text{nld}, \text{nif}, \text{idsistema}, \text{dtviagem}, \text{hinicio}, \text{hfim}, \text{valfinal}) (\text{PESS} \times \text{VIAGENS})$

b) Implementação 2:

$\text{PESS1} \leftarrow \pi (\text{id}) (\sigma (\text{nproprio} = \text{Carlos}, \text{apelido} = \text{Brito}) (\text{PESSOA}))$

$\text{PESS2} \leftarrow \pi (\text{id}) (\sigma (\text{nproprio} = \text{Bruno}, \text{apelido} = \text{Ferreira}) (\text{PESSOA}))$

$\text{VEICULOS1} \leftarrow \pi (\text{id}) (\sigma (\text{PESS1.id} = \text{proprietario}) (\text{VEICULO}))$

$\text{VEICULOS2} \leftarrow \pi (\text{id}) (\sigma (\text{PESS2.id} = \text{proprietario}) (\text{VEICULO}))$

$\text{COND} \leftarrow \pi (\text{condutor}) (\sigma (\text{VEICULOS1.id} = \text{veiculo}, \text{VEICULOS2.id} = \text{veiculo}) (\text{CONDHABILITADO}))$

$\text{PESS} \leftarrow \pi (\text{idpessoa}) (\sigma (\text{COND.id} = \text{idpessoa}) (\text{CONDUTOR}))$

$\text{RESULT} \leftarrow \pi (\text{nproprio}, \text{apelido}, \text{nident}) (\sigma (\text{PESS.id} = \text{id}) (\text{PESSOA}))$

c) Implementação 2:

$\text{VIAGENS} \leftarrow \pi (\text{idsistema}) (\square (\text{YEAR}(\text{dtviagem}) = 2021) (\text{VIAGEM}))$

$\text{AUX1} \leftarrow \pi (\square (\text{VIAGENS} = \text{viagem}) (\text{CLIENTEVIAGEM}))$

$VIAGENS \leftarrow \pi (idsistema, dtviagem, inicio, hfim, valfinal) (\bowtie (ID=idpessoa) CLIENTEVIAGEM)$

$RESULT \leftarrow \pi (id, nproprio, apelido, nif) (\bowtie (idpessoa=id) PESSOA)$

d) Implementação 2:

$COND \leftarrow \pi (id) (\bowtie (atrdisc=c) PESSOA)$

$AUX1 \leftarrow \pi (condutor) (\bowtie (COUNT(COND)=o) VIAGEM)$

$AUX2 \leftarrow \pi (idpessoa) (\bowtie (AUX1=idpessoa) CONDUTOR)$

$RESULT \leftarrow \pi (id, nproprio, apelido, nif) (\bowtie (id=AUX2) PESSOA)$

e) Implementação 2:

$VIAGENS \leftarrow \pi (\bowtie (YEAR(dtviagem)=2021)) CONDUTOR)$

$COUNT \leftarrow \bowtie ((COUNT(condutor))(VIAGENS))$

$COND \leftarrow \pi (id, nproprio, apelido) (\bowtie (atrdisc=c) PESSOA)$

$RESULT \leftarrow \pi (nproprio, apelido, count) (COND \bowtie (id=condutor) COUNT)$

Ex 3

a)

--2a

```
select noident, idsistema, dtviagem, inicio, hfim, valfinal from PESSOA
join CLIENTEVIAGEM on (idpessoa = id)      join VIAGEM on (viagem =
idsistema)
where(atrdisc = 'CL' and nproprio = 'António' and apelido = 'Silva');
--2b
select nproprio, apelido, noident from CONDUTOR
join CONDHABILITADO on (condutor = idpessoa)
join VEICULO on (veiculo = VEICULO.id)
join PROPRIETARIO on (proprietario = PROPRIETARIO.idpessoa)
join PESSOA on (PROPRIETARIO.idpessoa = PESSOA.id and nproprio = 'Bruno' and
apelido = 'Ferreira' and apelido != 'Brito' and nproprio != 'Carlos')
where (atrdisc = 'C');
```

--2c

```
select id, nif, nproprio, apelido
from PESSOA
join (select idpessoa, count(idpessoa) total
from CLIENTEVIAGEM
join
VIAGEM on (viagem = idsistema)
where (DATE_PART('year', dtviagem)='2021'))
```

```

        group by idpessoa) as totalviagens on (totalviagens.idpessoa = id)
where (select max(total)
        from (
            select idpessoa, count(idpessoa) total
        from CLIENTEVIAGEM
            join VIAGEM on (viagem = idsistema)
where (DATE_PART('year', dtviagem)='2021')
        group by idpessoa) as maxtotalviagens) = totalviagens.total;

```

--2d

```

select distinct id, nif, nproprio, apelido from VIAGEM
join PESSOA on (atrdisc = 'C')
where id not in (select condutor from VIAGEM);

```

--2e

```

select nproprio, apelido, count(viagem) as num_viagens from Condutor
join PESSOA on (CONDUTOR.idpessoa = id)
join CLIENTEVIAGEM on (CONDUTOR.idpessoa = CLIENTEVIAGEM.idpessoa)
group by nproprio, apelido order by num_viagens;

```

```

b) select count(viagem)
as num_viagens from
CLIENTEVIAGEM join VIAGEM
on (viagem = idsistema)
join VEICULO on (VEICULO.ano = 2012 and VEICULO.id = veiculo) join
PESSOA on (proprietario = PESSOA.id and nif = '987654321');

```

```

c)
select nproprio, apelido, noident, morada from
PESSOA
join (select idpessoa, sum(valfinal) total
from VIAGEM
join CONDUTOR on (condutor = idpessoa)
group
by idpessoa) as ganhos on ganhos.idpessoa = id where
(select max(total)
from (select idpessoa, sum(valfinal) total
from VIAGEM
join CONDUTOR on (condutor = idpessoa)
group by idpessoa) as maxganhos) = ganhos.total;

```

```

d)
select nproprio, apelido, count(idsistema) as total_viagens from VIAGEM
join PESSOA on (VIAGEM.condutor = id and atrdisc = 'C') group by
nproprio, apelido, noident, morada having count(idsistema) < 3;

```

```

e)
create view CONDUTORVIAGENSPOANO as
select CONDUTOR.idpessoa as condutor, DATE_PART('year', VIAGEM.dtviagem) as ano,
count(idsistema) as num_viagens, sum(valfinal) as ganhos from CONDUTOR join
VIAGEM on (CONDUTOR.idpessoa = VIAGEM.condutor) group by CONDUTOR.idpessoa, ano;

```

f)

```
create table CLIENTEVIAGENSANUAIS as  
select CLIENTEVIAGEM.idpessoa as cliente, DATE_PART('year', VIAGEM.dtviagem) as ano,  
count(viagem) as num_viagens  
from CLIENTEVIAGEM join VIAGEM on (CLIENTEVIAGEM.viagem = VIAGEM.idsistema) group  
by CLIENTEVIAGEM.idpessoa, ano;
```

Ex4 a) insert into TIPOVEICULO(tipo, designação, nlugares) **values**
(4,'Premium',4);

b)

```
update VEICULO set tipo = 4 where modelo = 'RangeRover';
```