

# Projeto de Bases de Dados, Parte 3

Grupo 47

Daniel Fernandes 86400 &  
Francisco Sousa 86416 & Henrique Ferreira 86432

30 de Novembro de 2018

Turno BD22517957L09  
Sexta 12:30 Lab 8  
Prof. Taras Lykhenko

Número de Aluno	Nome	Esforço
86400	Daniel Fernandes	33% (20h)
86416	Francisco Sousa	33% (20h)
86432	Henrique Ferreira	33% (20h)

# 1 Criação e População da Base de Dados

Para a criação da base de dados foi usado o código em seguida apresentado. Em primeiro lugar é limpa a base de dados e em seguida são criadas as tabelas de acordo com o modelo relacional do enunciado.

```
drop table solicita cascade;
drop table audita cascade;
drop table coordenador cascade;
drop table acciona cascade;
drop table alocado cascade;
drop table transporta cascade;
drop table meio cascade;
drop table meio_socorro cascade;
drop table meio_apoio cascade;
drop table meio_combate cascade;
drop table entidade_meio cascade;
drop table evento_emergencia cascade;
drop table processo_socorro cascade;
drop table vigia cascade;
drop table localidade cascade;
drop table camara cascade;
drop table video cascade;
drop table segmento_video cascade;

create table camara
  (num_camara char(5) not null unique,
   constraint pk_camara primary key(num_camara));

create table video
  (data_hora_inicio timestamp not null,
   data_hora_fim      timestamp not null,
   num_camara         char(5)   not null,
   constraint pk_video primary key(data_hora_inicio, num_camara),
   constraint fk_num_camara foreign key(num_camara) references camara(num_camara),
   constraint ck_data_hora check (data_hora_inicio < data_hora_fim));

create table segmento_video
  (num_segmento      char(5)   not null,
   duracao           int not null,
   data_hora_inicio timestamp not null,
   num_camara        char(5)   not null,
   constraint pk_segmento_video primary key(num_segmento, data_hora_inicio, num_camara),
   constraint fk_segmento_video foreign key(data_hora_inicio, num_camara)
     references video(data_hora_inicio, num_camara));

create table localidade
  (morada_local varchar(255) not null,
   constraint pk_morada_local primary key(morada_local));

create table vigia
  (morada_local varchar(255) not null,
   num_camara   char(5) not null unique,
   constraint pk_vigia primary key(morada_local, num_camara),
   constraint fk_morada_local foreign key(morada_local) references localidade(morada_local),
   constraint fk_num_camara foreign key(num_camara) references camara(num_camara));

create table processo_socorro
  (num_processo_socorro char(5) not null unique,
   constraint pk_num_processo_socorro primary key(num_processo_socorro));

create table evento_emergencia
```

```

(num_telefone      char(9) not null,
 instante_chamada  timestamp not null,
 nome_pessoa       varchar(255) not null,
 morada_local      varchar(255) not null,
 num_processo_socorro char(5) not null,
 constraint pk_evento_emergencia primary key(num_telefone, instante_chamada),
 constraint fk_morada_local foreign key(morada_local) references localidade(morada_local),
 constraint fk_num_processo_socorro foreign key(num_processo_socorro)
 references processo_socorro(num_processo_socorro));

create table entidade_meio
(nome_entidade varchar(20) not null unique,
 constraint pk_nome_entidade primary key(nome_entidade));

create table meio
(num_meio      char(5)      not null,
 nome_meio     varchar(255) not null,
 nome_entidade varchar(20) not null,
 constraint pk_meio primary key(num_meio, nome_entidade),
 constraint fk_nome_entidade foreign key(nome_entidade) references entidade_meio(nome_entidade));

create table meio_combate
(num_meio      char(5)      not null,
 nome_entidade varchar(255) not null,
 constraint pk_meio_combate primary key(num_meio, nome_entidade),
 constraint fk_meio foreign key(num_meio, nome_entidade) references meio(num_meio, nome_entidade));

create table meio_apoio
(num_meio      char(5)      not null,
 nome_entidade varchar(255) not null,
 constraint pk_apoio primary key(nome_entidade, num_meio),
 constraint fk_meio foreign key(num_meio, nome_entidade) references meio(num_meio, nome_entidade));

create table meio_socorro
(num_meio      char(5)      not null,
 nome_entidade varchar(255) not null,
 constraint pk_socorro primary key(nome_entidade, num_meio),
 constraint fk_meio foreign key(num_meio, nome_entidade) references meio(num_meio, nome_entidade));

create table transporta
(num_meio      char(5)      not null,
 nome_entidade varchar(20) not null,
 num_vitimas   int,
 num_processo_socorro char(5),
 constraint pk_transporta primary key(num_processo_socorro, nome_entidade, num_meio),
 constraint fk_transporta_num_meio foreign key(num_meio, nome_entidade)
 references meio_socorro(num_meio, nome_entidade),
 constraint fk_transporta_num_proc foreign key(num_processo_socorro)
 references processo_socorro(num_processo_socorro));

create table alocado
(num_meio      char(5)      not null,
 nome_entidade varchar(20) not null,
 num_horas     decimal(5, 2),
 num_processo_socorro char(5),
 constraint pk_alocado primary key(num_processo_socorro, nome_entidade, num_meio),
 constraint fk_alocado_num_meio foreign key(num_meio, nome_entidade)
 references meio_apoio(num_meio, nome_entidade),
 constraint fk_alocado_num_proc foreign key(num_processo_socorro)
 references processo_socorro(num_processo_socorro));

```

```

create table acciona
(num_meio          char(5)    not null,
 nome_entidade     varchar(20) not null,
 num_processo_socorro char(5),
 constraint pk_acciona primary key(num_processo_socorro, nome_entidade, num_meio),
 constraint fk_acciona_num_meio foreign key(num_meio, nome_entidade)
      references meio(num_meio, nome_entidade),
 constraint fk_acciona_num_proc foreign key(num_processo_socorro)
      references processo_socorro(num_processo_socorro));

create table coordenador
(id_coordenador char(5) not null,
 constraint pk_coordenador primary key(id_coordenador));

create table audita
(id_coordenador      char(5)    not null,
 num_meio            char(5)    not null,
 nome_entidade        varchar(20) not null,
 num_processo_socorro char(5),
 data_hora_inicio     timestamp,
 data_hora_fim         timestamp,
 data_auditoria        date,
 texto                varchar(100),
 constraint pk_audita primary key(id_coordenador, num_meio, nome_entidade, num_processo_socorro),
 constraint fk_audita_num_meio foreign key(num_meio, nome_entidade, num_processo_socorro)
      references acciona(num_meio, nome_entidade, num_processo_socorro),
 constraint fk_audita_id_coord foreign key(id_coordenador) references coordenador(id_coordenador),
 constraint ck_data_autoria check (data_auditoria <= now()));

create table solicita
(id_coordenador      char(5)    not null,
 data_hora_inicio_video timestamp not null,
 num_camara           char(5)    not null unique,
 data_hora_inicio     timestamp not null,
 data_hora_fim         timestamp not null,
 constraint pk_solicita primary key(id_coordenador, data_hora_inicio_video, num_camara),
 constraint fk_solicita_id_coord foreign key(id_coordenador) references coordenador(id_coordenador),
 constraint fk_solicita_inicio_video foreign key(data_hora_inicio_video, num_camara)
      references video(data_hora_inicio, num_camara));

```

Para a população da base de dados foi criado um script em Python que cria o populate.sql com atenção às restrições de integridade.

## 2 SQL

1.

```

select * from (select num_processo_socorro, count(*) from acciona
group by num_processo_socorro) as c cross join
(select max(count) from (select num_processo_socorro, count(*) from acciona
group by num_processo_socorro) as b) as a where (count = max);

```

2.

```

select * from (select * from (select nome_entidade, count(*) from
(select * from (acciona natural join evento_emergencia) as b
where instante_chamada > '2018-06-21 00:00:00' and instante_chamada < '2018-09-22 23:59:59')
as c group by nome_entidade) as b cross join
(select max(count) from (select nome_entidade, count(*) from

```

```
(select * from (acciona natural join evento_emergencia) as b where instante_chamada > '2018-06-21 00:00:00'
and instante_chamada < '2018-09-22 23:59:59') as c group by nome_entidade) as d) as e) as ahhhhh
where count = max;
```

3.

```
select num_processo_socorro from (evento_emergencia natural join
(select num_processo_socorro from acciona except
select num_processo_socorro from (acciona natural join audita)) as b)
where (morada_local = 'Oliveira do Hospital'
and instante_chamada > '2018-01-01 00:00:00' and instante_chamada < '2018-12-31 23:59:59');
```

4.

```
select count(*) from
(select * from (vigia natural join segmento_video natural join video) as b
where (duracao > 60 and morada_local = 'Monchique' and
data_hora_inicio > '2018-08-01 00:00:00' and data_hora_fim < '2018-08-31 23:59:59')) as a;
```

5.

```
select nome_entidade from (meio_combate natural join
(select num_meio from meio_combate except select num_meio from
(select num_meio from acciona except select num_meio from
(select num_meio from acciona except select num_meio from meio_apoio) as b) as c) as marrucho);
```

6.

```
select nome_entidade from meio_combate where not exists
(select num_processo_socorro from processo_socorro except
select num_processo_socorro from (acciona natural join processo_socorro));
```

## 3 Desenvolvimento da Aplicação

A "entrada" da nossa aplicação é um ficheiro **index.html** que têm links para tudo o resto, separados por duas seções:

### 3.1 Ver

Tem todas as tabelas listadas, permitindo consultá-las na íntegra. Se tiver sido pedido, ainda é possível *adicionar*, *editar* ou *remover* entradas de cada tabela.

A consulta é conseguida com o ficheiro **list.php** que, por sua vez, constrói os links para as operações:

#### 3.1.1 Adicionar

Pede os valores necessários ao utilizador (**addform.php**) e adiciona (**add.php**);

#### 3.1.2 Editar

Mostra todos os campos alteráveis ao utilizador (**editform.php**) e edita o que este alterou (**edit.php**);

#### 3.1.3 Remove

Apaga todas as entradas necessárias de cada tabela, para manter as restrições (**remove.php**).

### 3.2 Consultar

Tem as duas consultas pedidas no enunciado. Para cada uma delas, pergunta ao utilizador o valor do campo a procurar (**consultform.php** e apresenta a consulta (**consult.php**).