

Instituto Superior Técnico

Projecto de Base de Dados

- 3ª Entrega -

Professor André Vasconcelos

Turno: BD22517957L08

Grupo 60

Número	Nome	Trabalho [%]	Esforço [horas]
86466	Madalena Pedreira	33	43
86496	Pedro Custódio	33	43
86508	Rita Fernandes	33	43

SQL

1.

```
SELECT num_processo_socorro
FROM acciona
GROUP BY num_processo_socorro
HAVING COUNT(*) >= ALL (
    SELECT COUNT(*)
    FROM acciona
    GROUP BY num_processo_socorro);
```

2.

```
SELECT nome_entidade
FROM acciona NATURAL JOIN (
    SELECT *
    FROM evento_emergencia
    WHERE instante_chamada BETWEEN '2018-06-21 00:00:00' AND '2018-09-23
23:59:59' ) AS t
GROUP BY nome_entidade
HAVING COUNT(*) >= all (
    SELECT COUNT(*)
    FROM acciona NATURAL JOIN (
        SELECT *
        FROM evento_emergencia
        WHERE instante_chamada BETWEEN '2018-06-21 00:00:00'
AND '2018-09-23 23:59:59') AS r GROUP BY nome_entidade);
```

3.

```
SELECT distinct num_processo_socorro
FROM evento_emergencia NATURAL JOIN (
    SELECT * FROM acciona
    EXCEPT
    SELECT num_meio, nome_entidade, num_processo_socorro
    FROM audita ) AS t
WHERE instante_chamada BETWEEN '2018-01-01 00:00:00' AND '2018-12-31
23:59:59' AND morada_local = 'Oliveira do Hospital';
```

4.

```
SELECT distinct num_camara, num_segmento, data_hora_inicio, duracao
FROM video NATURAL JOIN vigia NATURAL JOIN segmento_video
WHERE duracao > '00:00:60' AND morada_local = 'Monchique' AND data_hora_inicio
BETWEEN '2018-08-01 00:00:00' AND '2018-08-31 23:59:59' AND data_hora_fim
BETWEEN '2018-08-01 00:00:00' AND '2018-08-31 23:59:59';
```

5.

```
SELECT num_meio, nome_entidade
```

```

FROM meio_combate
EXCEPT
SELECT num_meio,nome_entidade
FROM alocado ;

```

[assumiu-se que se também querem listar os meios de combate que não foram accionados em processos de socorro]

6.

```

SELECT nome_entidade
FROM entidade_meio AS e
WHERE NOT EXISTS (
    SELECT num_processo_socorro
    FROM processo_socorro
    EXCEPT
    SELECT num_processo_socorro
    FROM (acciona NATURAL JOIN meio_combate ) AS a
    WHERE a.nome_entidade = e.nome_entidade );

```

Arquitectura da Aplicação PHP

A aplicação está estruturada da seguinte forma: há um menu inicial (menu_raiz.html) que redirecciona o utilizador para as várias funcionalidades do programa. Clicando em cada uma destas opções, as funcionalidades são asseguradas pelas seguintes páginas PHP, pela ordem listada:

Alinea a)

Pasta 'ALINEA_A'

1. alinea_a.php

Utilizador informa se pretende inserir ou remover;

2. inserir_remove.php

Utilizador selecciona em que tabela pretende inserir ou remover;

3.inserir_remove_tabelas.php

Utilizador preenche formulário de inserção ou de remoção na tabela (consoante escolha anterior)

3.1.concluir_pedido_inserir_remove.php

É feita e confirmada a inserção na tabela escolhida;

3.2 delete_from_table.php

É feita a remoção da entrada na tabela escolhida;

Alinea b)

Pasta 'ALINEA_B'

1. Alinea_b.php

Utilizador informa se pretende inserir, remover ou editar registos

2. `escolha_tabela.php`
Utilizador selecciona em que tabela (`meio_apoio`, `meio_socorro`, `meio_combate`) pretende inserir, remover ou editar registos.
3. `do.php`
 - a) Caso o utilizador queira inserir: é requisitada a informação necessária para criar um registo. Note-se que o utilizador apenas pode inserir registos que já existem na tabela `meio`.
 - b) Caso o utilizador queira remover: é pedido para que se seleccione o registo a remover. Note-se que o utilizador quando remove um registo remove também as respetivas associações nas tabelas `aloca` e `transporta`.
 - c) Caso o utilizador queira editar: é pedido para que seja seleccionado o registo a editar. Note-se que o utilizador apenas pode editar para registos que já existem na tabela `meio`.
4.
 - a) `concluir.php` - executa a inserção apresentando uma mensagem no fim.
 - b) `remover.php` - executa a remoção apresentando uma mensagem no fim.
 - c) `editar.php` - apresenta as opções para as quais o registo pode ser alterado.
5. c) `done.php` - executa a alteração apresentando uma mensagem no fim.

Alinea c)

Pasta 'ALINEA_C'

1. `c.php`
Lista ambas as tabelas (Processos de Socorro e Meios) na mesma página, seguidas.

Alinea d)

Pasta 'ALINEA_D'

1. `Alinea_d.php`
Utilizador informa se pretende associar um processo de socorro a um meio ou a um evento de emergência
2. `EE_Meio.php`
Apresenta ao utilizador o formulário adequado, consoante a escolha do utilizador em `Alinea_d.php`
3. `assocPSOStoEE.php` e `assoc PSOStoMeio.php`
Realizam a tarefa em si, sendo que o `assocPSOStoEE.php` apenas atualiza a entrada na tabela `evento_emergencia` caso exista mais do que um evento associado ao processo que esta para ser substituído. No `PSOStoMeio.php` é apenas inserido na tabela `acciona` esta nova associação.

Alinea e)

Pasta: 'ALINEA_E'

1. `listMeiosPerSOSRequest.php`
Apresenta ao utilizador o formulário para que este especifique o número do Processo de Socorro.
2. `listMeiosPerSOS.php`
Apresenta uma tabela dos meios (número do meio e o nome da entidade) accionados para esse processo de socorro.

Alinea f)

Pasta: 'ALINEA_F'

1. form_morada.php

Pede a morada ao utilizador;

2. show_tabela.php

Apresenta a tabela com os meios de socorro com o filtro pretendido;

A aplicação pode ser testada no link:

http://web.tecnico.ulisboa.pt/~ist186466/Projecto_Entrega_3/menu_raiz.html

Criação da Base de Dados

```
DROP TABLE camara CASCADE;  
DROP TABLE video CASCADE;  
DROP TABLE segmento_video CASCADE;  
DROP TABLE local CASCADE;  
DROP TABLE vigia CASCADE;  
DROP TABLE evento_emergencia CASCADE;  
DROP TABLE processo_socorro CASCADE;  
DROP TABLE entidade_meio CASCADE;  
DROP TABLE meio CASCADE;  
DROP TABLE meio_socorro CASCADE;  
DROP TABLE meio_apoio CASCADE;  
DROP TABLE meio_combate CASCADE;  
DROP TABLE transporta CASCADE;  
DROP TABLE alocado CASCADE;  
DROP TABLE audita CASCADE;  
DROP TABLE coordenador CASCADE;  
DROP TABLE acciona CASCADE;  
DROP TABLE solicita CASCADE;
```

```
create table camara(  
    num_camara integer not null,  
    constraint pk_camara primary  
key(num_camara)  
);
```

```
create table video(  
    data_hora_inicio timestamp not null,  
    data_hora_fim timestamp not null,  
    num_camara integer not null,  
    constraint pk_video primary  
key(data_hora_inicio,num_camara),  
    constraint fk_video_camara foreign  
key(num_camara)  
    references camara(num_camara)  
);
```

```
create table segmento_video(  
    num_segimento integer not null,
```

```
    duracao interval hour to second,  
    data_hora_inicio timestamp not null,  
    num_camara integer not null,  
    constraint pk_segimento_video primary  
key(num_segimento,data_hora_inicio,num_cam  
ara),  
    constraint fk_segimento_video_video foreign  
key(data_hora_inicio,num_camara)  
    references  
video(data_hora_inicio,num_camara)  
);  
create table local(  
    morada_local varchar(30) not null,  
    constraint pk_local primary key(morada_local)  
);
```

```
create table vigia(  
    morada_local varchar(30) not null,  
    num_camara integer not null,  
    constraint pk_vigia primary  
key(morada_local,num_camara),  
    constraint fk_vigia_local foreign  
key(morada_local)  
    references local(morada_local),  
    constraint fk_vigia_camara foreign  
key(num_camara)  
    references camara(num_camara)  
);
```

```
create table processo_socorro(  
    num_processo_socorro integer,  
    constraint pk_processo_socorro primary  
key(num_processo_socorro)  
);
```

```
create table evento_emergencia(  
    num_telefone varchar(15) not null,  
    instante_chamada timestamp not null,
```

```

    nome_pessoa varchar(80) not null,
    morada_local varchar(30) not null,
    num_processo_socorro integer not null,
    constraint pk_evento_emergencia primary
key(num_telefone,instante_chamada),
    constraint fk_evento_emergencia_local
foreign key(morada_local)
        references local(morada_local),
    constraint
fk_evento_emergencia_num_processo_socorro
foreign key(num_processo_socorro)
references
processo_socorro(num_processo_socorro),
    unique(num_telefone,nome_pessoa)
);
create table entidade_meio(
    nome_entidade varchar(30) not null,
    constraint pk_entidade_meio primary
key(nome_entidade)
);
create table meio(
    num_meio integer not null ,
    nome_meio varchar(30) not null,
    nome_entidade varchar(30) not null,
    constraint pk_meio primary
key(num_meio,nome_entidade),
    constraint fk_meio foreign
key(nome_entidade)
        references entidade_meio(nome_entidade)
);
create table meio_combate(
    num_meio integer not null,
    nome_entidade varchar(30) not null,
    constraint pk_meio_combate primary
key(num_meio,nome_entidade),
    constraint fk_meio_combate_meio foreign
key(num_meio,nome_entidade)
        references
meio(num_meio,nome_entidade)
);
create table meio_apoio(
    num_meio integer not null,
    nome_entidade varchar(30) not null,
    constraint pk_meio_apoio primary
key(num_meio,nome_entidade),
    constraint fk_meio_apoio_meio foreign
key(num_meio,nome_entidade)
        references
meio(num_meio,nome_entidade)
);
create table meio_socorro(
    num_meio integer not null,
    nome_entidade varchar(30) not null,

```

```

    constraint pk_meio_socorro primary
key(num_meio,nome_entidade),
    constraint fk_meio_socorro_meio foreign
key(num_meio,nome_entidade)
        references
meio(num_meio,nome_entidade)
);
create table transporta(
    num_meio integer not null,
    nome_entidade varchar(30) not null,
    num_vitimas integer not null,
    num_processo_socorro integer,
    constraint pk_transporta primary
key(num_meio,nome_entidade,num_processo_
socorro),
    constraint fk_transporta_meio_socorro foreign
key(num_meio,nome_entidade)
        references
meio_socorro(num_meio,nome_entidade),
    constraint fk_transporta_processo_socorro
foreign key(num_processo_socorro)
        references
processo_socorro(num_processo_socorro)
);
create table alocado(
    num_meio integer not null,
    nome_entidade varchar(30) not null,
    num_horas integer not null,
    num_processo_socorro integer,
    constraint pk_alocado primary
key(num_meio,nome_entidade,num_processo_
socorro),
    constraint fk_alocado_meio_apoio foreign
key(num_meio,nome_entidade)
        references
meio_apoio(num_meio,nome_entidade),
    constraint fk_alocado_processo_socorro
foreign key(num_processo_socorro)
        references
processo_socorro(num_processo_socorro)
);
create table acciona(
    num_meio integer not null,
    nome_entidade varchar(30) not null,
    num_processo_socorro integer,
    constraint pk_acciona primary
key(num_meio,nome_entidade,num_processo_
socorro),
    constraint fk_acciona_meio foreign
key(num_meio,nome_entidade)
        references
meio(num_meio,nome_entidade),

```

```

        constraint fk_acciona_processo_socorro
foreign key(num_processo_socorro)
references
processo_socorro(num_processo_socorro)
);
create table coordenador(
    id_coordenador integer not null,
    constraint pk_coordenador primary
key(id_coordenador)
);
create table audita(
    id_coordenador integer not null,
    num_meio integer not null,
    nome_entidade varchar(30) not null,
    num_processo_socorro integer,
    data_hora_inicio timestamp not null,
    data_hora_fim timestamp not null,
    data_auditoria date not null,
    texto varchar(120) not null,
    constraint pk_audita primary
key(nome_entidade,num_processo_socorro,id_
coordenador),
    constraint fk_audita_acciona foreign key
(num_meio,nome_entidade,num_processo_soc
orro)
references
acciona(num_meio,nome_entidade,num_proces
so_socorro),
    constraint fk_audita_coordenador foreign key
(id_coordenador)
references coordenador(id_coordenador),
    check(data_hora_inicio < data_hora_fim),
    check(data_auditoria <= current_date)
);
create table solicita(
    id_coordenador integer not null,
    data_hora_inicio timestamp not null,
    num_camara integer not null,
    data_hora_inicio_video timestamp not null,
    data_hora_fim timestamp not null,
    constraint pk_solicita primary
key(data_hora_inicio_video,num_camara,id_coo
rdenador),
    constraint fk_solicita_coordenador foreign
key(id_coordenador)
references coordenador(id_coordenador),
    constraint fk_solicita_video foreign
key(data_hora_inicio_video,num_camara)
references
video(data_hora_inicio,num_camara)
);

```