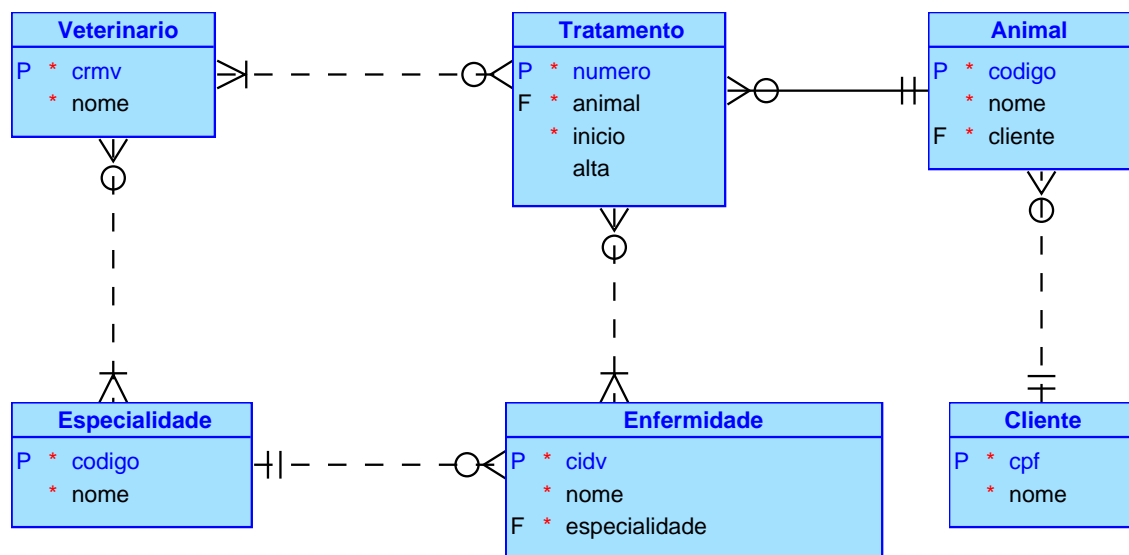


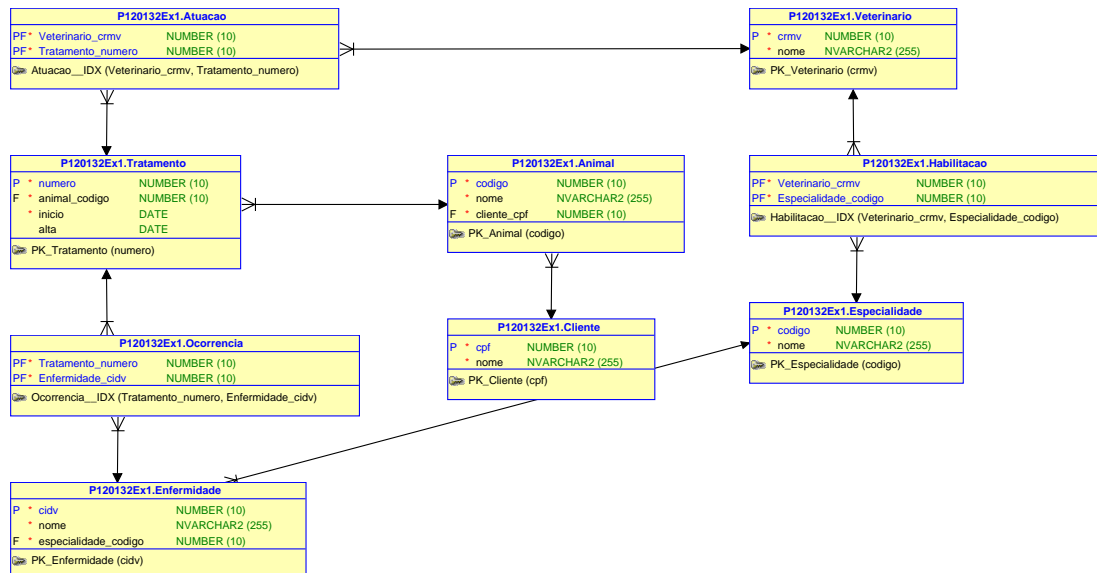
Q1	
Q2	
Q3	
Q4	
Nota	

Aluno: \_\_\_\_\_

1. [2,5 pontos] Desenvolver um modelo de dados para um hospital veterinário. Cada cliente pode possuir um ou vários animais em tratamento. Cada animal pode estar sofrendo de uma ou várias enfermidades. Os casos mais simples são resolvidos, geralmente, por um único veterinário, entretanto podem ocorrer casos em que um animal é atendido por mais de um veterinário. Cada veterinário possui uma ou várias especialidades. Cada especialidade habilita ao tratamento de uma ou mais enfermidades. Ao final do tratamento o animal recebe alta do tratamento.

R.:





2. [2,5 pontos] Escreva os comandos DDL para a criação do banco de dados do hospital veterinário.

R.:

```

CREATE TABLE P120132Ex1.Animal (
    codigo NUMBER (10) NOT NULL ,
    nome NVARCHAR2 NOT NULL ,
    cliente NUMBER (10) NOT NULL
) ;
ALTER TABLE P120132Ex1.Animal ADD
CONSTRAINT PK_Animal PRIMARY KEY (
    codigo
);

CREATE TABLE P120132Ex1.Atuacao (
    Veterinario_crmv NUMBER (10) NOT NULL ,
    Tratamento_codigo NUMBER (10) NOT NULL
) ;
ALTER TABLE P120132Ex1.Atuacao ADD
CONSTRAINT participa__IDX PRIMARY KEY (
    Veterinario_crmv , Tratamento_codigo
);

CREATE TABLE P120132Ex1.Cliente (
    cpf NUMBER (10) NOT NULL ,
    nome NVARCHAR2 (255) NOT NULL
) ;
  
```

```
ALTER TABLE P120132Ex1.Cliente ADD
CONSTRAINT PK_Cliente PRIMARY KEY(
    cpf
);
```

```
CREATE TABLE P120132Ex1.Enfermidade(
    cidv NUMBER (10) NOT NULL ,
    nome NVARCHAR2 (255) NOT NULL ,
    especialidade_codigo NUMBER (10) NOT NULL
) ;
```

```
ALTER TABLE P120132Ex1.Enfermidade ADD
CONSTRAINT PK_Enfermidade PRIMARY KEY (
    cidv
);
```

```
CREATE TABLE P120132Ex1.Especialidade (
    codigo NUMBER (10) NOT NULL ,
    nome NVARCHAR2 (255) NOT NULL
) ;
```

```
ALTER TABLE P120132Ex1.Especialidade ADD
CONSTRAINT PK_Especialidade PRIMARY KEY (
    codigo
);
```

```
CREATE TABLE P120132Ex1.Habilitacao (
    Veterinario_crmv NUMBER (10) NOT NULL ,
    Especialidade_codigo NUMBER (10) NOT NULL
) ;
```

```
ALTER TABLE P120132Ex1.Habilitacao ADD
CONSTRAINT habilitado__IDX PRIMARY KEY (
    Veterinario_crmv, Especialidade_codigo
);
```

```
CREATE TABLE P120132Ex1.Ocorrencia (
    Tratamento_codigo NUMBER (10) NOT NULL ,
    Enfermidade_cidv NUMBER (10) NOT NULL
) ;
```

```
ALTER TABLE P120132Ex1.Ocorrencia ADD CONSTRAINT
enfermidadev1__IDX PRIMARY KEY (
    Tratamento_codigo, Enfermidade_cidv
);
```

);

```
CREATE TABLE P120132Ex1.Tratamento (
    animal_codigo NUMBER (10) NOT NULL ,
    inicio DATE NOT NULL ,
    alta DATE
) ;
ALTER TABLE P120132Ex1.Tratamento ADD
CONSTRAINT PK_Tratamento PRIMARY KEY (
    codigo
);
```

```
CREATE TABLE P120132Ex1.Veterinario (
    crmv NUMBER (10) NOT NULL ,
    nome NVARCHAR2 (255) NOT NULL
) ;
ALTER TABLE P120132Ex1.Veterinario ADD
CONSTRAINT PK_Veterinario PRIMARY KEY (
    crmv
);
```

```
ALTER TABLE P120132Ex1.Ocorrencia ADD
CONSTRAINT FK_ASS_2 FOREIGN KEY ( Tratamento_codigo )
REFERENCES Tratamento ( codigo ) ;
```

```
ALTER TABLE P120132Ex1.Ocorrencia ADD
CONSTRAINT FK_ASS_3 FOREIGN KEY ( Enfermidade_cidv )
REFERENCES Enfermidade ( cidv ) ;
```

```
ALTER TABLE P120132Ex1.Habilitacao ADD
CONSTRAINT FK_ASS_4 FOREIGN KEY ( Veterinario_crmv )
REFERENCES Veterinario ( crmv ) ;
```

```
ALTER TABLE P120132Ex1.Habilitacao ADD
CONSTRAINT FK_ASS_5 FOREIGN KEY ( Especialidade_codigo )
REFERENCES Especialidade ( codigo ) ;
```

```
ALTER TABLE P120132Ex1.Atuacao ADD
CONSTRAINT FK_ASS_6 FOREIGN KEY ( Veterinario_crmv )
REFERENCES Veterinario ( crmv ) ;
```

```
ALTER TABLE P120132Ex1.Atuacao ADD
CONSTRAINT FK_ASS_7 FOREIGN KEY ( Tratamento_codigo )
REFERENCES Tratamento ( codigo ) ;
```

```
ALTER TABLE P120132Ex1.Tratamento ADD
CONSTRAINT FK_Animal_
Tratamento_1 FOREIGN KEY ( codigo )
REFERENCES Animal ( codigo ) ;
```

```
ALTER TABLE P120132Ex1.Animal ADD
CONSTRAINT FK_Cliente_Animal_1 FOREIGN KEY ( cliente )
REFERENCES Cliente ( cpf ) ;
```

```
ALTER TABLE P120132Ex1.Enfermidade ADD
CONSTRAINT FK_Especialidade_
Enfermidade_1 FOREIGN KEY ( codigo )
REFERENCES Especialidade ( codigo ) ;
```

3. [2,5 pontos] Escreva a implementação da restrição de integridade entre a especialidade do veterinário e as enfermidades do animal.

R.: **Um dos triggers a seguir. Não é necessário fazer todos!!**

```
create or replace
trigger P120132EX1.REGRA_ESPECIALIDADE_1
before insert or update on P120132EX1.ATUACAO
referencing old as ANTIGO new as NOVO
for each row
declare
    V_INCOMPATIVEL NUMBER(1,0);
begin
    V_INCOMPATIVEL:=0;
    begin
        select 1
        into V_INCOMPATIVEL
        from DUAL
        where exists (
            with
                ESP_MED as (
                    select T1.Especialidade_CODIGO CODIGO
                    from P120132EX1.HABILITACAO T1
```

```

        where T1.VETERINARIO_CRMV=:NOVO.VETERINARIO_CRMV),
    ESP_TRAT as (
        select T1.ESPECIALIDADE_CODIGO CODIGO
        from P120132EX1.ENFERMIDADE T1
        inner join P120132EX1.OCORRENCIA T2 on
            T2.ENFERMIDADE_CIDV=T1.CIDV
        inner join P120132EX1.TRATAMENTO T3 on
            T3.NUMERO=T2.TRATAMENTO_NUMERO
        where t3.numero=:novo.tratamento_numero)
    select T1.CODIGO
    from ESP_MED T1
    inner join ESP_TRAT T2 on T1.CODIGO=T2.CODIGO
    );
EXCEPTION
    when OTHERS then
        v_incompativel:=0;
end;

if (V_INCOMPATIVEL=1) then
    RAISE_APPLICATION_ERROR(-20003,'ERRO: conflito esp');
end if;

```

end;

## OU

```

create or replace
trigger P120132EX1.REGRA_especialidade_2
before delete on P120132EX1.OCORRENCIA
referencing old as ANTIGO new as NOVO
for each row
declare
    V_INCOMPATIVEL NUMBER(1,0);
begin
    V_INCOMPATIVEL:=0;
    begin
        select 1
        into V_INCOMPATIVEL
        from DUAL
        where exists (
            with

```

```

ESP_MED as (
    select T1.VETERINARIO_CRMV MED,
           T3.ESPECIALIDADE_CODIGO ESP
    from P120132EX1.ATUACAO T1
    inner join P120132EX1.VETERINARIO T2 on
           T2.CRMV=T1.VETERINARIO_CRMV
    inner join P120132EX1.HABILITACAO T3 on
           T3.VETERINARIO_CRMV=T2.CRMV
    where T1.TRATAMENTO_NUMERO=
           :ANTIGO.TRATAMENTO_NUMERO),
ESP_TRAT as (
    select T1.ESPECIALIDADE_CODIGO esp
    from P120132EX1.ENFERMIDADE T1
    inner join P120132EX1.OCORRENCIA T2 on
           T2.ENFERMIDADE_CIDV=T1.CIDV
    inner join P120132EX1.TRATAMENTO T3 on
           T3.NUMERO=T2.TRATAMENTO_NUMERO
    where T2.TRATAMENTO_NUMERO!=
           :ANTIGO.TRATAMENTO_NUMERO
    and T2.ENFERMIDADE_CIDV!=
           :ANTIGO.ENFERMIDADE_CIDV),
MED as (select distinct T1.MED med from ESP_MED T1),
MED_HAB as (
    select distinct T1.MED med
    from ESP_MED T1
    inner join ESP_TRAT T2 on T1.ESP=T2.ESP),
MED_NAO_HAB as (
    select T1.MED MED
    from MED T1
    left join MED_HAB T2 on T1.MED=T2.MED
    where T2.MED is null)
select * from MED_NAO_HAB
);
EXCEPTION
    when OTHERS then
        v_incompativel:=0;
end;

if (V_INCOMPATIVEL=1) then
    RAISE_APPLICATION_ERROR(-20003,'ERRO: conflito esp');

```

```
end if;
```

```
end;
```

OU

```
create or replace
```

```
trigger P120132EX1.REGRA_ESPECIALIDADE_3
```

```
before update on P120132EX1.OCORRENCIA
```

```
referencing old as ANTIGO new as NOVO
```

```
for each row
```

```
declare
```

```
    V_INCOMPATIVEL NUMBER(1,0);
```

```
begin
```

```
    V_INCOMPATIVEL:=0;
```

```
    if (:antigo.tratamento_numero!=  
        :novo.tratamento_numero) then
```

```
        begin
```

```
            select 1
```

```
            into V_INCOMPATIVEL
```

```
            from DUAL
```

```
            where exists (
```

```
            with
```

```
                ESP_MED as (
```

```
                    select T1.VETERINARIO_CRMV MED,
```

```
                           T3.ESPECIALIDADE_CODIGO ESP
```

```
                           from P120132EX1.ATUACAO T1
```

```
            inner join P120132EX1.VETERINARIO T2 on
```

```
                T2.CRMV=T1.VETERINARIO_CRMV
```

```
            inner join P120132EX1.HABILITACAO T3 on
```

```
                T3.VETERINARIO_CRMV=T2.CRMV
```

```
            where T1.TRATAMENTO_NUMERO=:
```

```
                ANTIGO.TRATAMENTO_NUMERO),
```

```
                ESP_TRAT as (
```

```
                    select T1.ESPECIALIDADE_CODIGO esp
```

```
                    from P120132EX1.ENFERMIDADE T1
```

```
            inner join P120132EX1.OCORRENCIA T2 on
```

```
                T2.ENFERMIDADE_CIDV=T1.CIDV
```

```
            inner join P120132EX1.TRATAMENTO T3 on
```

```
                T3.NUMERO=T2.TRATAMENTO_NUMERO
```

```
            where T2.TRATAMENTO_NUMERO!=:ANTIGO.TRATAMENTO_NUMERO
```



```

        and T2.ENFERMIDADE_CIDV!=:ANTIGO.ENFERMIDADE_CIDV),
        MED as (
            select distinct T1.MED med from ESP_MED T1),
        MED_HAB as (
            select distinct T1.MED med
            from ESP_MED T1
            inner join ESP_TRAT T2 on T1.ESP=T2.ESP),
        MED_NAO_HAB as (
            select T1.MED MED
from MED T1
left join MED_HAB T2 on T1.MED=T2.MED
where T2.MED is null)
        select * from MED_NAO_HAB
    );
EXCEPTION
    when OTHERS then
        V_INCOMPATIVEL:=0;
end;
else
begin
    select 1
    into V_INCOMPATIVEL
    from DUAL
    where exists (
        with
            ESP_MED as (
                select T1.VETERINARIO_CRMV MED,
                    T3.ESPECIALIDADE_CODIGO ESP
                    from P120132EX1.ATUACAO T1
                inner join P120132EX1.VETERINARIO T2 on
                    T2.CRMV=T1.VETERINARIO_CRMV
                inner join P120132EX1.HABILITACAO T3 on
                    T3.VETERINARIO_CRMV=T2.CRMV
                where T1.TRATAMENTO_NUMERO=:ANTIGO.TRATAMENTO_NUMERO),
            ESP_TRAT as (
                select T1.ESPECIALIDADE_CODIGO esp
from P120132EX1.ENFERMIDADE T1
inner join P120132EX1.OCORRENCIA T2 on
    T2.ENFERMIDADE_CIDV=T1.CIDV
inner join P120132EX1.TRATAMENTO T3 on

```

```

        T3.NUMERO=T2.TRATAMENTO_NUMERO
where T2.TRATAMENTO_NUMERO!=:ANTIGO.TRATAMENTO_NUMERO
      and T2.ENFERMIDADE_CIDV!=:ANTIGO.ENFERMIDADE_CIDV
union
select T1.ESPECIALIDADE_CODIGO
from P120132EX1.ENFERMIDADE T1
where T1.CIDV=:NOVO.ENFERMIDADE_CIDV),
      MED as (select distinct T1.MED med from ESP_MED T1),
      MED_HAB as (
        select distinct T1.MED med
from ESP_MED T1
inner join ESP_TRAT T2 on T1.ESP=T2.ESP),
      MED_NAO_HAB as (
        select T1.MED MED
from MED T1
left join MED_HAB T2 on T1.MED=T2.MED
where T2.MED is null)
      select * from MED_NAO_HAB
    );
EXCEPTION
  when OTHERS then
    V_INCOMPATIVEL:=0;
  end;
end if;

if (V_INCOMPATIVEL=1) then
  RAISE_APPLICATION_ERROR(-20003,'ERRO: conflito esp');
end if;

end;

```

4. [2,5 pontos] Escreva uma stored procedure para listar a especialidade mais praticada por cada veterinário.

R.:

```

create or replace
procedure LISTA_ESPECIALIDADE (P_RESULT in OUT
  SYS_REFCURSOR) as
begin
  open P_RESULT for
  'with

```

```

ESP_MED as (
    select T1.CRMV MED, T2.ESPECIALIDADE_CODIGO ESP
    from P120132EX1.VETERINARIO T1
    inner join P120132EX1.HABILITACAO T2 on
        T2.VETERINARIO_CRMV=T1.CRMV),
ESP_TRAT as (
    select T4.VETERINARIO_CRMV MED, T1.NUMERO TRAT,
        T3.ESPECIALIDADE_CODIGO ESP
    from P120132EX1.TRATAMENTO T1
    inner join P120132EX1.OCORRENCIA T2 on
        T2.TRATAMENTO_NUMERO=T1.NUMERO
    inner join P120132EX1.ENFERMIDADE T3 on
        T3.CIDV=T2.ENFERMIDADE_CIDV
    inner join P120132EX1.ATUACAO T4 on
        T4.TRATAMENTO_NUMERO=T1.NUMERO),
ATUA as (
    select T1.MED, T2.ESP, COUNT(*) CONT
    from ESP_MED T1
    inner join ESP_TRAT T2 on T1.MED=T2.MED
        and T2.ESP=T1.ESP
    group by T1.MED, T2.ESP),
MAX_ATUA as (
    select T1.MED, max(T1.CONT) MAX_CONT
    from ATUA T1
    group by T1.MED)
select T1.*
FROM ATUA T1 INNER JOIN MAX_ATUA T2 ON T1.MED=T2.MED
    AND T1.CONT=T2.MAX_CONT';
end LISTA_ESPECIALIDADE;

```