

FCT-Universidade Nova de Lisboa



Relatório do Projecto “AutoNova”

Professor A. Eduardo Dias, Turno P9

Área Curricular: Bases de Dados

**Ana Cavadas (nº 31543)
Miguel Aniceto (nº 31411)
Tiago Santos (nº 31562)**

Índice

0. Introdução ao projecto;
1. Objectivos;
2. Alterações desde a versão preliminar;
3. Análise do Modelo ER (Ver Anexo 8.1);
4. Análise do Modelo Relacional (Ver Anexo 8.2);
5. Análise da Interface da Base de Dados;
6. Análise de Triggers (Ver Anexo 8.3);
7. Conclusão;
8. Anexos:
 - 8.1 Modelo ER;
 - 8.2 Modelo Relacional;
 - 8.3 Listagem do código de criação de objectos.

0:: Introdução ao projecto

O nosso Projecto consiste na organização dos dados de um stand automóvel. Como tal será incluído no nosso trabalho dados relativamente a:

- **Pessoas:** NIF, Nome, Morada, Telefone, Sexo, Data-nsc;
 - **Clientes:** Número de Cliente, número de compras;
 - **Funcionários:** número de Funcionário, Data de Entrada e Data de saída;
 - **Vendedores do stand:** percentagem de venda, salário;
 - **Gestor de equipa do stand:** salário;
- **Especificações:** Atributos como o identificador da especificação, Modelo, Marca e Tara;
- **Carros:** Identificador do carro, cor, preço, ano do carro e combustível;
- **Vendas:** código de vendas e a data referente a esta;
- **Pagamento:** nº pagamento, montante, data e tipo;

Por exemplo: Eu, como comprador pretendo adquirir um carro. Dirijo-me ao stand, onde sou atendido por um vendedor (que trabalha para um gerente), digo-lhe as características que pretendo que o meu carro tenha (especificação do carro).

Caso exista tal carro, o vendedor entra em contacto comigo, e no caso de desejar poderei realizar test-drives para o testar. Se tudo correr bem e pretender comprar o carro em questão, então irei proceder ao seu pagamento.

A escolha deste tema recai sobre várias razões, para aprendermos como funciona este ramo, uma vez que um stand tem vários elementos dinâmicos que interagem entre si e também porque nos interessamos pelo ramo automóvel.

1:: Objectivos

O objectivo deste projecto é desenvolver de raiz uma base de dados funcional, ao longo deste projecto vamos obter as bases necessárias à concepção, construção e análise de bases de dados relacionais.

Nesta fase é necessário saber modelar o modelo Entidade-Relação, o modelo relacional de dados e também saber os fundamentos da linguagem SQL, para realizar o script que cria as tabelas juntamente com as suas restrições.

2:: Alterações desde a Versão preliminar

Ao longo do desenvolvimento deste projecto, verificou-se que existiam situações em que o Modelo criado não estava optimizado, assim sendo foram efectuadas alterações e melhoramentos que constam na seguinte lista:

- Na entidade Funcionários, acrescentou-se os atributos “Data_entrada” e “Data_Saída”, que têm a utilidade de indicar se um funcionário se encontra activo no Stand. Deste modo ao apagarmos um funcionário obtemos uma data de saída e as vendas do funcionário podem ser mantidas ao invés no anterior Modelo. Se o funcionário ainda está na empresa então a Data_Saída fica a NULL.
- Na entidade Gerentes e Vendedores foi removido o atributo “Comissão”, verificou-se que é um atributo desnecessário pois podemos obter a comissão dos funcionários através de cálculos entre outros campos.
- No modelo Relacional estava presente uma relação incorrecta entre as especificações e os carros, do modo que estava se n carros tivessem uma especificação igual, seria necessário criar n especificações iguais, portanto esta situação foi corrigida.
- Colocação do novo atributo Num_Compras na tabela Clientes.

3: Análise do Modelo ER (Ver anexo 7.1)

- A partir das Pessoas temos uma relação total porque pode haver pessoas das quais não são nem funcionárias nem clientes;
- Uma venda tem obrigatoriamente um cliente, mas um cliente pode não ter uma venda associada;
- No Pagamento, existe obrigatoriamente uma venda associada, mas poderá existir vendas para as quais o pagamento ainda não foi efectuado;
- Em cada venda há sempre um vendedor associado;
- Um gestor tem uma relação com os vendedores para ser possível obter a sua comissão (visto que é em função das vendas dos vendedores);
- Vários carros diferentes podem pertencer a uma só venda;
- Vários carros podem ter uma só especificação;
- Um cliente pode fazer diversos testes-drives a carros diferentes, e um carro pode ser testado por clientes diferentes (N:M)

4: Análise do Modelo Relacional (Ver anexo 7.2)

- Acrescentou-se os atributos Data_Entrada e Data_Saída na tabela Funcionários.
- Retirou-se a relação entre Pagamentos e Vendas através da chave externa NumP, pois havia redundância entre estas duas tabelas. Deste modo, retirou-se a chave externa NumP da tabela Vendas.
- Verificou-se que Vendas não pode ter como chave externa NumF mas sim NIF da tabela Vendedores.
- Verificou-se que Vendas não pode ter como chave externa NumC mas sim NIF da tabela Clientes.
- Ao colocar o atributo IdEspecificação como chave primária na tabela Especificações a tabela Carros fica com a chave externa IdEspecificação.
- Acrescentou-se na tabela Clientes o novo atributo, Num_Compras.
- Retirou-se Marca e Modelo como chave primária da tabela Especificações, ficando IdEspecificação como chave primária desta tabela.
- Retirou-se da tabela Vendedores e Gerentes o atributo comissão.

5:: Análise da Interface da Base de Dados

As funcionalidades da Interface da Base de Dados pedidas podem ser observadas de acordo com os seguintes passos:

- “Existência de uma página de entrada onde existam ligações para as páginas subsequentes;”

A página Home é a página inicial e contém as ligações para as restantes páginas.

- “Listagem de dados da Base de Dados onde códigos referentes a chaves externas sejam substituídos pelo valor de outros atributos de fácil compreensão (e.g. lista de alunos onde figura o nome dos seus cursos em vez dos respectivos códigos);”

A partir da página Home, no link Carros_Form ao clicarmos em combustível aparece as 2 opções disponíveis, e ao clicarmos numa delas somos redireccionados para uma lista com todos os veículos com o combustível seleccionado.

- “Listagem de dados da Base de Dados onde sejam apresentados valores derivados (e.g. a média de cada aluno);”

A partir da Home, está disponível um link com o nome “Numero de Carros”, aqui é possível verificar o valor total de carros no stand, a partir da tabela carros.

- “Possibilidade de inserir, remover e actualizar tuplos da Base de Dados;”

A partir da Home, é visível um link para um Form com o nome Carros_Form, aqui é possível inserir, remover e actualizar tuplos da Base de Dados.

- “Possibilidade de preencher valores de atributos correspondentes a relações (chaves externas) sem se ter conhecimento de códigos (e.g. seleccionando um valor de uma *LOV* baseada numa consulta) e/ou restringindo o domínio dos valores seleccionados (e.g. seleccionando o valor numa *select list* cujas opções são obtidas dinamicamente por uma *LOV* baseada numa consulta);”

A partir da Home, é visível um link com o nome “Carros por combustível”, que satisfaz as funcionalidades pedidas (Devido a complicações, apenas a 3ª primeira e 3ª última selecção funcionam).

- Existência de links de navegação (*breadcrumbs*) nas várias páginas;

A partir de qualquer página é visível a presença de breadcrumbs portanto nesta situação não há um exemplo específico.

- Existência de dois reports interligados, onde um apresenta detalhes do outro (*drill-down*) (e.g. página com lista de todos os cursos onde, ao seleccionar um deles, somos guiados a uma página com detalhes sobre esse curso, por exemplo a lista dos seus alunos);

A partir da application builder ao seleccionar na página “Gerentes” e ao clicar “run”, surge uma tabela, onde ao clicar em “NIF” surgem os vendedores associados ao “NIF” desse gerente.

- Existência de um detalhe condicional (e.g. listar apenas alunos do departamento seleccionado numa select list, mostrando o número total de cadeiras desse curso, quando existe um seleccionado);

Na página vendedores, no topo é apresentada uma select list onde surgem todos os gerentes, ao seleccionar um, surgem os vendedores associados a esse gerente.

- Existência de um form *master-detail*, onde seja possível inserir, remover e alterar dados (pelo menos) do *detail* (e.g. form com informação sobre um aluno e lista de inscrições, onde se possa acrescentar uma inscrição, remover uma inscrição, e alterar os dados de uma inscrição existente)

A partir da application builder ao seleccionar a página “Vendas” (Número 14) e clicar “run” seguidamente ao clicar no link “edit” é apresentado o Master Detail.

6:: Análise de Triggers

Para garantir a integridade da nossa base de dados e a afectação de alguns dados depois da inserção criou-se alguns triggers, dos quais se destacam:

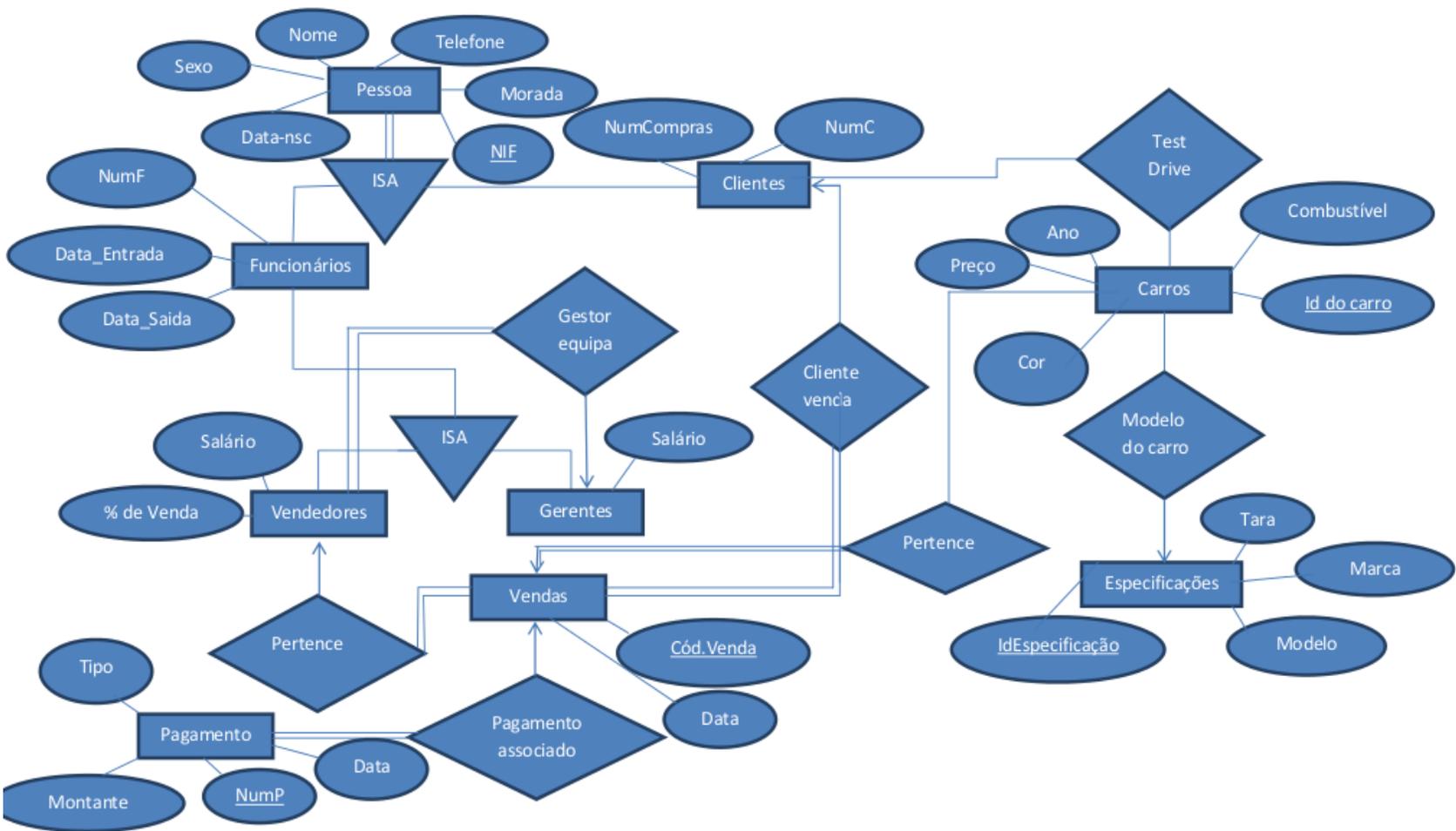
- Triggers para ISA's
Garantem que os dados criados nas tabelas depois da generalização têm de constar na tabela “mãe”. Por exemplo não queremos ter clientes ou funcionários que não constem na tabela de pessoas.
- Triggers para Vistas
Garantem que a alteração de dados não afecta as vistas mas sim as tabelas que fazem parte desta.
- Triggers de afectação:
 1. Ao eliminar um funcionário para que este passe a ter a data de saída.
 2. Ao Inserir uma venda, incrementa o número de compras do cliente em questão e altera a percentagem de vendas dos funcionários em questão (sendo estes independentes).

7::Conclusão

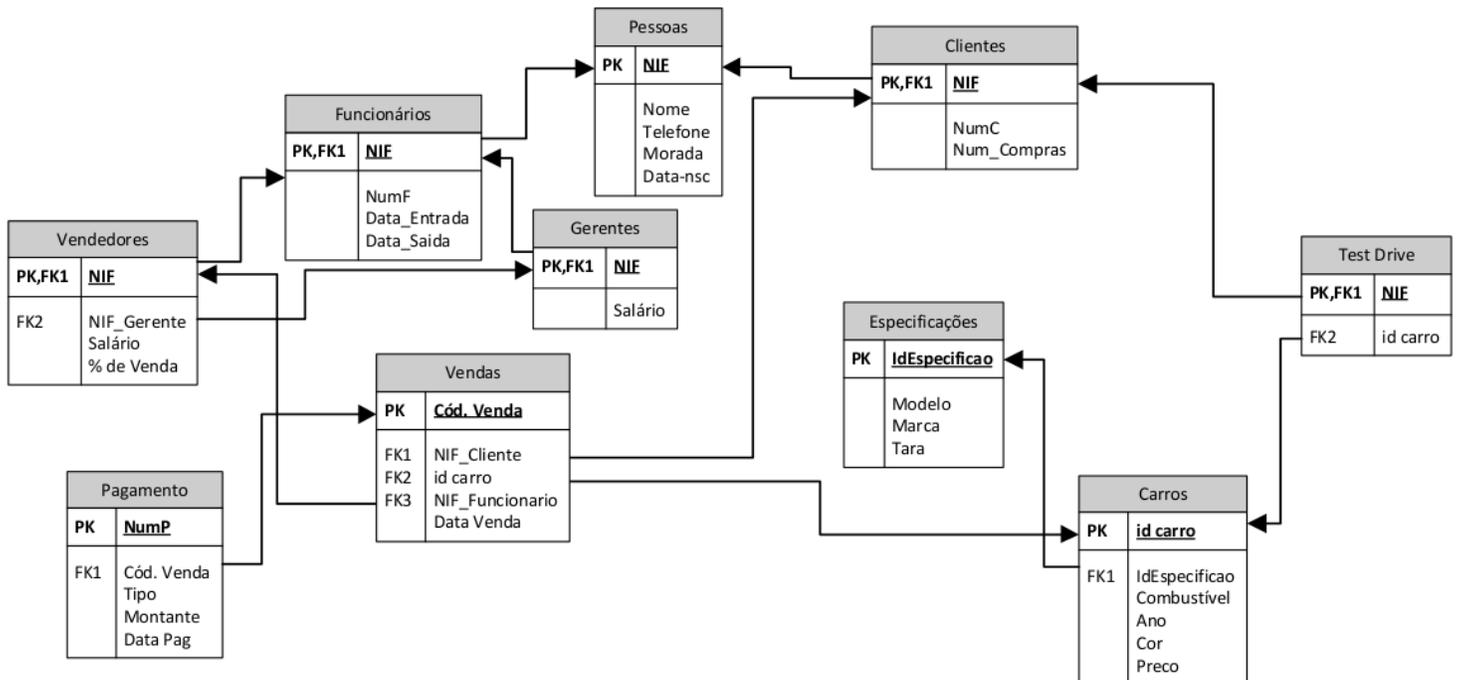
Durante a realização deste projecto adquirimos variados conhecimentos, desde a modelar Modelos ER juntamente com modelos relacionais. Inclusive a criar scripts de criação de dados e objectos em sql, bem como triggers, view e todos os outros objectos necessários numa base de dados funcional. Ainda se implementou uma interface da base de dados recorrendo à Apex.

8:: Anexos

8.1:: Modelo ER



8.2:: Modelo Relacional



8.3:: Listagem do código de criação de objectos

--Tabelas

```
drop table Pessoas cascade constraints;
```

```
create table Pessoas(
NIF number(9) not null,
Nome varchar2(50) not null,
Morada varchar2(50) not null,
Telefone number(9) not null,
Data_nsc date,
sexo char(1) not null check ( sexo in ( 'F' , 'M' ) ),
primary key (NIF)
);
```

```
drop table Funcionarios cascade constraints;
```

```
create table Funcionarios(
NIF number(9) not null,
NumF number(5) not null,
Data_Entrada date not null,
Data_Saida date,
primary key (NIF),
foreign key (NIF) references Pessoas
);
```

```

drop table Clientes cascade constraints;

create table Clientes(
NIF number(9) not null,
NumC number(5) not null,
Num_Compras number (3) not null,
primary key (NIF),
foreign key (NIF) references Pessoas
);

drop table Gerentes cascade constraints;

create table Gerentes(
NIF number(9) not null,
Salario number(5) not null,
primary key (NIF),
foreign key (NIF) references Funcionarios
);

drop table Vendedores cascade constraints;

create table Vendedores(
NIF number(9) not null,
Salario number(5) not null,
Porcentagem_Vendas number(5) not null,
NIF_Gerente number(9) not null,
primary key (NIF),
foreign key (NIF) references Funcionarios,
foreign key (NIF_Gerente) references Gerentes
);

drop table Especificacoes cascade constraints;

create table Especificacoes(
IdEspecificacao number(5) not null,
Modelo varchar2(10) not null,
Marca varchar2(10) not null,
Tara number(4) not null,
primary key (IdEspecificacao)
);

drop table Carros cascade constraints;

create table Carros(
IdCarro number(5) not null,
Combustivel varchar2(15) not null,
Ano number(4) not null,
Cor varchar2(10) not null,
Preco number(8) check (Preco > 0),
IdEspecificacao number(5) not null,
primary key (IdCarro),
foreign key (IdEspecificacao) references Especificacoes
);

```

```

drop table Vendas cascade constraints;

create table Vendas(
CodVenda number(5) not null,
NIF_Cliente number(9) not null,
NIF_Funcionario number(9) not null,
DataVenda date not null,
IdCarro number(5) not null,
primary key (CodVenda),
foreign key (NIF_Cliente) references Clientes(NIF),
foreign key (IdCarro) references Carros,
foreign key (NIF_Funcionario) references Funcionarios(NIF)
);

drop table Pagamentos cascade constraints;

create table Pagamentos(
NumP number(5) not null,
Tipo varchar2(15) not null,
Montante number(8) not null,
DataPagamento date not null,
CodVenda number(5) not null,
primary key (NumP),
foreign key (CodVenda) references Vendas
);

drop table Test_Drive cascade constraints;

create table Test_Drive(
NIF number(9) not null,
IdCarro number(5) not null,
primary key (NIF),
foreign key (NIF) references Clientes,
foreign key (IdCarro) references Carros);
--correr duas vezes para criar todas as sequencias
drop sequence seq_funcionario;
create sequence seq_funcionario increment by 1 start with 1;

drop sequence seq_cliente;
create sequence seq_cliente increment by 1 start with 1;

drop sequence seq_vendas;
create sequence seq_vendas increment by 1 start with 1;

drop sequence seq_pagamento;
create sequence seq_pagamento increment by 1 start with 1;

drop sequence seq_especificacao;
create sequence seq_especificacao increment by 1 start with 1;

drop sequence seq_carro;
create sequence seq_carro increment by 1 start with 1;

```

```

create or replace view Func_Vendas as
(select NIF, Nome, NumF, Salario, NIF_Cliente, Percentagem_Vendas, CodVenda,
DataVenda, IdCarro
from Pessoas natural inner join Funcionarios natural inner join Vendedores
natural inner join Vendas);

create or replace view Vendas_Pagamentos as
(select NumP, Tipo, Montante, DataPagamento, CodVenda, NIF_Cliente,
NIF_Funcionario, DataVenda, IdCarro
from Vendas natural inner join Pagamentos);

--Triggers
--trigger para actualizar um pagamento na respectiva tabela a partir da vista
vendas_pagamentos
drop trigger actualiza_Vendas_Pagamentos;
create trigger actualiza_Vendas_Pagamentos
instead of update on Vendas_Pagamentos
for each row
begin
update Pagamentos
set Tipo =:new.Tipo, Montante = :new.Montante, DataPagamento =
:new.DataPagamento, CodVenda = :new.CodVenda
where NumP =:old.NumP;
end;
/

--trigger para inserir um pagamento na respectiva tabela a partir da vista
vendas_pagamentos
drop trigger insere_Vendas_Pagamentos;
create trigger insere_Vendas_Pagamentos
instead of insert on Vendas_Pagamentos
for each row

begin
insert into Pagamentos
values (seq_pagamento.nextval, :new.Tipo, :new.Montante,
:new.DataPagamento, :new.CodVenda);
end;
/

--trigger para actualizar uma venda na respectiva tabela a partir da vista
vendas_funcionarios
drop trigger actualiza_VendasFunc;
create trigger actualiza_VendasFunc
instead of update on Func_Vendas
for each row

begin
update Vendas
set NIF_Cliente=:new.NIF_Cliente, NIF_Funcionario=:new.NIF,
DataVenda=:new.DataVenda, IdCarro = :new.IdCarro
where CodVenda=:old.CodVenda;
end;
/

```

```

--trigger para inserir uma venda na respectiva tabela a partir da vista
vendas_funcionarios
drop trigger insere_VendasFunc;
create trigger insere_VendasFunc
instead of insert on Func_Vendas
for each row
begin
    insert into Vendas
    values (seq_vendas.nextval, :new.NIF_Cliente, :new.NIF, :new.DataVenda,
:new.IdCarro);
end;
/
--so se permite criar um cliente se este estiver contido na tabela Pessoas
drop trigger insert_cliente;
create or replace trigger insert_cliente
before insert or update on Clientes
for each row
declare Num_C number;
begin
    select count(*) into Num_C from Pessoas
    where NIF = :new.NIF;
if(Num_C=0) then
    raise_application_error(-20000,'Pessoa não existente!');
end if;
end;
/
--so se permite criar um gerente se este estiver contido na tabela
Funcionarios
drop trigger insert_gerente;
create or replace trigger insert_gerente
before insert or update on Gerentes
for each row
declare Num_F number;
begin
    select count(*) into Num_F from Funcionarios
    where NIF = :new.NIF;
if(Num_F=0) then
    raise_application_error(-20000,'Pessoa não existente!');
end if;
end;
/
--so se permite criar um vendedor se este estiver contido na tabela
Funcionarios
drop trigger insert_vendedores;
create or replace trigger insert_vendedores
before insert or update on Vendedores
for each row
declare Num_F number;
begin
    select count(*) into Num_F from Funcionarios
    where NIF = :new.NIF;
if(Num_F=0) then
    raise_application_error(-20000,'Pessoa não existente!');
end if;
end;
/

```

```

--so se permite criar um funcionario se este estiver contido na tabela
Pessoas
drop trigger insert_funcionario;
create or replace TRIGGER insert_funcionario
before insert or update on Funcionarios
for each row
declare Num_F number;
begin
    select count(*) into Num_F from Pessoas
    where NIF = :new.NIF;
If (Num_F=0) then
    raise_application_error(-20000,'Pessoa não existente!');
end if;
end;
/

drop trigger elimina_gerentes;
create or replace trigger elimina_gerentes
after delete on Gerentes
for each row
begin
    update Funcionarios set Data_Saida = sysdate-1
    where NIF= :old.NIF;
end;
/

drop trigger aumento_percentagem_insert;
create or replace trigger aumento_percentagem_insert after insert on Vendas
for each row
declare
num_vendas number;
total_vendas number;
begin
select count(*) into total_vendas from Vendas;
select count(*) into num_vendas from Vendedores
where Vendedores.NIF = :new.NIF_Funcionario;
update Vendedores set Percentagem_Vendas = ((num_vendas/total_vendas)*100)
where Vendedores.NIF=:new.NIF_Funcionario;

end;
/

drop trigger elimina_funcionario;
create or replace trigger elimina_funcionario
after delete on Funcionarios
for each row
begin
    update Funcionarios set Data_Saida = sysdate-1
    where NIF= :old.NIF;
end;
/

```

```

drop trigger aumento_percentagem_update;
create or replace trigger aumento_percentagem_update after update on Vendas
for each row
declare
num_vendas number;
total_vendas number;
begin
select count(*) into total_vendas from Vendas;
select count(*) into num_vendas from Vendedores
where Vendedores.NIF = :new.NIF_Funcionario;
update Vendedores set Percentagem_Vendas = ((num_vendas/total_vendas)*100)
where Vendedores.NIF=:new.NIF_Funcionario;

end;
/

drop trigger elimina_vendedores;
create or replace trigger elimina_vendedores
after delete on Vendedores
for each row
begin
update Funcionarios set Data_Saida = sysdate-1
where NIF= :old.NIF;
end;
/

drop trigger num_comprascliente_insert;
create or replace trigger num_comprascliente_insert after insert on Vendas
for each row
declare
num_compras number;
begin
select count(*) into num_compras from Clientes
where Clientes.NIF = :new.NIF_Cliente;
update Clientes set Num_Compras = num_compras
where Clientes.NIF = :new.NIF_Cliente;
end;
/

```

```

--Dados
insert into Pessoas values(153765518,'Alfredo da Cunha ALmeida','Av. 25 de
Abril n° 8 3° Esq', 917839456,to_date('1965.01.01','YYYY.MM.DD'),'M');
insert into Pessoas values(215122496,'Bruno Fernando da Silva Delgado','Rua
das Flores N° 25 8° B', 933569412,to_date('1967.03.01','YYYY.MM.DD'),'M');
insert into Pessoas values(217179193,'Carla Cristina da Costa Ferreira','Rua
Rafael Bordalo Pinheiro n° 8 A', 968490345,
to_date('1974.07.13','YYYY.MM.DD'),'F');
insert into Pessoas values(152662898,'Daniel da Silva Costa','Qt da Herdeira
Lt 798', 934567890 ,to_date('1974.09.23','YYYY.MM.DD'),'M');
insert into Pessoas values(227285786,'Diana Patricia Mendes Pereira','Praceta
Antônio Sebastiao n° 4 2° Drt',
927849104,to_date('1974.08.15','YYYY.MM.DD'),'F');
insert into Pessoas values(118101099,'Eduardo Evora Simoes','Av. Luis de
Camoos n°5 6°Esq', 965321789,to_date('1981.11.16','YYYY.MM.DD'),'M');
insert into Pessoas values(183497457,'Fernanda da Cunha Faria','Rua Jose L
Rego n°16 7° Frt',917845978,to_date('1966.01.07','YYYY.MM.DD'),'F');
insert into Pessoas values(128327820,'Gabriel Dias Fortuna','Av. Camilo
Castelo Branco Lt. 25',967634543,to_date('1974.01.07','YYYY.MM.DD'),'M');
insert into Pessoas values(198038925,'Helder Antonio Filipe Francisco','Praça
Bartolomeu Dias n° 8 R/ch
B',934782934,to_date('1978.02.16','YYYY.MM.DD'),'M');
insert into Pessoas values(141065281,'Ines Maria Teixeira Martins','Rua João
Dias n° 5 4° Drt',928349234,to_date('1979.10.29','YYYY.MM.DD'),'F');
insert into Pessoas values(127242546,'Jaime Galvao Fernandes','Av. 25 de
Abril n° 6 1° Drt',309916059,to_date('1965.01.01','YYYY.MM.DD'),'M');
insert into Pessoas values(189245808,'Lina Maria Magalhaes Teixeira','Rua Dr.
Lacerda Almeida n° 7 8° A',212289348,to_date('1967.03.01','YYYY.MM.DD'),'F');
insert into Pessoas values(148090505,'Manuela Marinheiro Monteiro','Av. da
Liberdade n°123 2° Esq',934753473,to_date('1974.07.13','YYYY.MM.DD'),'F');
insert into Pessoas values(136204155,'Natalia Maria Ferreira
Fernandes','Praça dos Olivais n°3 6°
Frt',912345678,to_date('1974.09.23','YYYY.MM.DD'),'F');
insert into Pessoas values(216495636,'Nelson Pereira Silva','Praceta Jose
Costa Barras n°1 3° Frt ',960823647,to_date('1974.08.15','YYYY.MM.DD'),'M');
insert into Pessoas values(126136858,'Orlando Barbosa de Jesus','Av da
Republica n°3 1° Drt',212224856,to_date('1981.11.16','YYYY.MM.DD'),'M');
insert into Pessoas values(161793223,'Palmira Pereira Miranda','Travessa do
Carmo n°5 6° C',936965938,to_date('1966.01.07','YYYY.MM.DD'),'F');
insert into Pessoas values(190333634,'Ricardo Joao da Silva Carvalho','Quinta
das Laranjeiras n° 25',210855934,to_date('1974.01.07','YYYY.MM.DD'),'M');
insert into Pessoas values(182467066,'Sergio da Silva Reis', 'Rua Fernando
Pessoa n° 5',960038764,to_date('1978.02.16','YYYY.MM.DD'),'M');
insert into Pessoas values(194164519,'Teresa da Silva Gomes','Av de Roma Lt
35',925743985,to_date('1979.10.29','YYYY.MM.DD'),'F');

insert into Clientes values(153765518,seq_cliente.nextval, 0);
insert into Clientes values(152662898,seq_cliente.nextval, 0);
insert into Clientes values(227285786,seq_cliente.nextval, 0);
insert into Clientes values(183497457,seq_cliente.nextval, 0);
insert into Clientes values(128327820,seq_cliente.nextval, 0);
insert into Clientes values(141065281,seq_cliente.nextval, 0);
insert into Clientes values(127242546,seq_cliente.nextval, 0);
insert into Clientes values(189245808,seq_cliente.nextval, 0);
insert into Clientes values(136204155,seq_cliente.nextval, 0);
insert into Clientes values(216495636,seq_cliente.nextval, 0);
insert into Clientes values(161793223,seq_cliente.nextval, 0);

```

```

insert into Clientes values(182467066,seq_cliente.nextval, 0);
insert into Clientes values(194164519,seq_cliente.nextval, 0);

insert into Funcionarios
values(217179193,seq_funcionario.nextval,to_date('2000.02.20','YYYY.MM.DD'),N
ULL);
insert into Funcionarios
values(198038925,seq_funcionario.nextval,to_date('2000.05.17','YYYY.MM.DD'),N
ULL);
insert into Funcionarios
values(215122496,seq_funcionario.nextval,to_date('2001.09.01','YYYY.MM.DD'),N
ULL);
insert into Funcionarios
values(118101099,seq_funcionario.nextval,to_date('2002.01.14','YYYY.MM.DD'),N
ULL);
insert into Funcionarios
values(148090505,seq_funcionario.nextval,to_date('2002.03.26','YYYY.MM.DD'),N
ULL);
insert into Funcionarios
values(126136858,seq_funcionario.nextval,to_date('2005.04.11','YYYY.MM.DD'),N
ULL);
insert into Funcionarios
values(190333634,seq_funcionario.nextval,to_date('2009.11.15','YYYY.MM.DD'),N
ULL);

insert into Gerentes values(217179193,2000);
insert into Gerentes values(198038925,1500);

insert into Vendedores values(215122496,1200,0,217179193);
insert into Vendedores values(118101099,1100,0,217179193);
insert into Vendedores values(148090505,1100,0,217179193);
insert into Vendedores values(126136858,900,0,198038925);
insert into Vendedores values(190333634,600,0,198038925);

insert into Especificacoes
values(seq_especificacao.nextval,'Corsa','Opel',1400);
insert into Especificacoes
values(seq_especificacao.nextval,'Aveo','Chevrolet',1100);
insert into Especificacoes
values(seq_especificacao.nextval,'Liana','Suzuki',1170);
insert into Especificacoes
values(seq_especificacao.nextval,'350z','Nissan',1518);
insert into Especificacoes
values(seq_especificacao.nextval,'A4','Audi',1700);
insert into Especificacoes
values(seq_especificacao.nextval,'Picasso','Citroen',1300);
insert into Especificacoes
values(seq_especificacao.nextval,'Laguna','Renault',1350);
insert into Especificacoes
values(seq_especificacao.nextval,'Clio','Renault',1050);
insert into Especificacoes
values(seq_especificacao.nextval,'C4','Citroen',1500);
insert into Especificacoes
values(seq_especificacao.nextval,'ForTwo','Smart',780);
insert into Especificacoes
values(seq_especificacao.nextval,'Benz','Mercedes',3735);

```

```

insert into Especificacoes
values(seq_especificacao.nextval,'Cooper','Mini',1140);

insert into Carros values(seq_carro.nextval,'Gasolina',2005,'Verde',10000,1);
insert into Carros
values(seq_carro.nextval,'Gasolina',2006,'Cinzento',11000,2);
insert into Carros
values(seq_carro.nextval,'Gasolina',2006,'Amarelo',12000,3);
insert into Carros
values(seq_carro.nextval,'Gasolina',2006,'Amarelo',13000,4);
insert into Carros
values(seq_carro.nextval,'Gasolina',2007,'Vermelho',14500,5);
insert into Carros
values(seq_carro.nextval,'Gasolina',2007,'Branco',16000,6);
insert into Carros values(seq_carro.nextval,'Gasoleo',2007,'Preto',15000,7);
insert into Carros
values(seq_carro.nextval,'Gasolina',2008,'Branco',17500,8);
insert into Carros values(seq_carro.nextval,'Gasoleo',2009,'Azul',18000,9);
insert into Carros values(seq_carro.nextval,'Gasoleo',2009,'Azul',17000,10);
insert into Carros values(seq_carro.nextval,'Gasoleo',2009,'Preto',17500,11);
insert into Carros
values(seq_carro.nextval,'Gasolina',2010,'Vermelho',23000,12);
insert into Carros
values(seq_carro.nextval,'Gasolina',2010,'Vermelho',23000,12);

insert into Vendas
values(seq_vendas.nextval,153765518,148090505,to_date('2005.05.15','YYYY.MM.D
D'),1);
insert into Vendas
values(seq_vendas.nextval,183497457,126136858,to_date('2006.02.08','YYYY.MM.D
D'),3);
insert into Vendas
values(seq_vendas.nextval,141065281,215122496,to_date('2007.04.14','YYYY.MM.D
D'),5);
insert into Vendas
values(seq_vendas.nextval,189245808,118101099,to_date('2008.12.27','YYYY.MM.D
D'),8);
insert into Vendas
values(seq_vendas.nextval,153765518,148090505,to_date('2009.08.25','YYYY.MM.D
D'),9);
insert into Vendas
values(seq_vendas.nextval,194164519,190333634,to_date('2010.11.15','YYYY.MM.D
D'),12);

insert into Pagamentos
values(seq_pagamento.nextval,'Dinheiro',10000,to_date('2005.05.15','YYYY.MM.D
D'),1);
insert into Pagamentos
values(seq_pagamento.nextval,'Transferencia',12000,to_date('2006.02.27','YYYY
.MM.DD'),2);
insert into Pagamentos
values(seq_pagamento.nextval,'Dinheiro',14500,to_date('2007.04.20','YYYY.MM.D
D'),3);
insert into Pagamentos
values(seq_pagamento.nextval,'Transferencia',17500,to_date('2009.01.04','YYYY
.MM.DD'),4);

```

```
insert into Pagamentos
values(seq_pagamento.nextval,'Transferencia',18000,to_date('2009.09.15','YYYY
.MM.DD'),5);
insert into Pagamentos
values(seq_pagamento.nextval,'Transferencia',23000,to_date('2010.11.30','YYYY
.MM.DD'),6);

insert into Test_Drive values(152662898,1);
insert into Test_Drive values(227285786,3);
insert into Test_Drive values(136204155,4);
insert into Test_Drive values(141065281,5);
insert into Test_Drive values(194164519,12);

commit;
```