Regras de negócio:

FUNCIONARIO.NOME = Preenchimento obrigatório

FUNCIONARIO.CPF = Não pode haver duas instâncias com mesmo valor no campo CPF.FUNCIONARIO.SALARIO = O salário dos funcionários não pode ser menor que 1000,00.FUNCIONARIO.SEXO = Só aceitar os caracteres “M” ou “F”.FUNCIONARIO.COD\_SUPER = coluna usada para definição do auto relacionamento DEPARTAMENTO.NOME = Preenchimento obrigatório DER regras de negócio: FUNCIONARIO.NOME = Preenchimento obrigatório FUNCIONARIO.CPF = Não pode haver duas instâncias com mesmo valor no campo CPF.FUNCIONARIO.SALARIO = O salário dos funcionários não pode ser menor que 1000,00.FUNCIONARIO.SEXO = Só aceitar os caracteres “M” ou “F”.FUNCIONARIO.COD\_SUPER = coluna usada para definição do auto relacionamento DEPARTAMENTO.NOME = Preenchimento obrigatório DEPENDENTE.NOME = Preenchimento obrigatório

Regras de negócio: FUNCIONARIO.NOME = Preenchimento obrigatório FUNCIONARIO.CPF = não pode haver duas instâncias com mesmo valor no campo CPF.FUNCIONARIO. SALARIO = O salário dos funcionários não pode ser menor que 1000,00. FUNCIONARIO.SEXO = Só aceitar os caracteres “M” ou “F”. FUNCIONARIO.COD\_SUPER = coluna usada para definição do auto relacionamento DEPARTAMENTO.NOME = Preenchimento obrigatório DEPENDENTE.NOME = Preenchimento obrigatório

DEPENDENTE.PARENTESCO = Preenchimento obrigatório DEPENDENTE.PARENTESCO = Só aceitar valores dentro do conjunto (“PAI”, ”MÃE”, “IRMAOS”, ”FILHO”) PROJETO.TITULO = Preenchimento obrigatório PROJETO.DATA\_CAD = Data de cadastro dos projetos com valor padrão igual à data do dia. A função que retorna a data do dia no Oracle é Sysdate.

Exercícios

1.Altere a tabela PROJETO, fazendo as seguintes mudanças:

A adicionar uma coluna de nome STATUS, de preenchimento obrigatório, com dados de tipo numérico, aceitando apenas os valores 0 (flagque indica projeto inativo) e 1 (flagque indica projeto em andamento). ]

B alterar a coluna TITULO, aumentando seu tamanho para 252 caracteres.

C renomear a coluna DATA\_CAD para DATA\_CADASTRO2.Altere a tabela FUNCIONARIO, fazendo as seguintes mudanças:

1. Transformar a coluna ENDERECO numa tabela separada, sendo que cada endereço poderá estar vinculado a mais de um funcionário (Exemplo: marido, esposa e filho trabalhando na mesma empresa) e cada funcionário poderá informar mais de um endereço.

DICA: Dessa forma, teremos um relacionamento N x M entre funcionário e a nova entidade ENDERECO.

b. Excluir a restrição associada à coluna SALARIO.

c. Desativar e depois ativar a restrição associada à coluna CPF.

d. Desativar e depois ativar a restrição associada à coluna cod\_func.

3.Altere a tabela DEPENDENTE, fazendo as seguintes mudanças:

a. A coluna PARENTESCO deixa de ser texto e passa a ser controlada através de outra tabela, onde são armazenados o CODIGO e DESCRICAO do tipo de parentescos aceitos pela empresa. Dessa forma, uma chave estrangeira na tabela DEPENDENTE deverá referenciara chave primária dessa nova tabela, num relacionamento 1 x N (Parentesco →. Dependente).

b. Criar uma chave alternativa na tabela DEPENDENTE, onde será armazenado o CPF dos dependentes.

c. Automatizar o processo de exclusão de registros da tabela DEPENDENTE ao excluir um registro da tabela FUNCIONARIO. Para isso, utilizar o comando DELETE CASCADE na chave estrangeira da coluna DEPENDENTE.COD\_FUNC. (Lembre-se: não é possível modificar uma constraint!) Vide aula DDL –PARTE 2.

4.Altere a tabela PARTICIPA, fazendo asseguintes mudanças:

a. Renomear a tabela PARTICIPA para FUNC\_PROJETO

Atividade termina até aqui

V Regras de negócio: FUNCIONARIO.NOME = Preenchimento obrigatório FUNCIONARIO.CPF = não pode haver duas instâncias com mesmo valor no campo CPF.FUNCIONARIO. SALARIO = O salário dos funcionários não pode ser menor que 1000,00. FUNCIONARIO.SEXO = Só aceitar os caracteres “M” ou “F”. FUNCIONARIO.COD\_SUPER = coluna usada para definição do auto relacionamento DEPARTAMENTO.NOME = Preenchimento obrigatório DEPENDENTE.NOME = Preenchimento obrigatório

PENDENTE.NOME = Preenchimento obrigatório

Fora do contexto

Um homem pode ter vários RGS, mas apenas um e único CPF

EXEMPLOS DO PROGRAMA

select \* from pedido;

ROLLBACK;

create table pedido

(ped\_id number(10),

ped\_valor\_total number (9,2));

ALTER TABLE PEDIDO;

ADD CONSTRAINT PK\_PED\_ID PRIMARY KEY(PED\_ID);

SELECT \* FROM user\_constraints;

WHERE table\_name='PEDIDO';

DESC PEDIDO;

INSERT into pedido values (1,100.30);

commit;

ALTER TABLE PEDIDO ADD PED\_DATA DATE;

ALTER TABLE PEDIDO MODIFY PED\_ID varchar(15);

ALTER TABLE PEDIDO DROP COLUMN PED\_DATA;

Com modificações

select \* from pedido;

ROLLBACK;

create table pedido

(ped\_id number(10),

ped\_valor\_total number (9,2));

alter table pedido add cli\_cod number(10);

ALTER TABLE PEDIDO

ADD CONSTRAINT FK\_PEDIDO\_CLI\_COD FOREIGN KEY(CLI\_COD);

REFERENCES CLIENTE(CLI\_COD);

desc pedido;

create table cliente

(cli\_cod number (10) primary key,

cli\_nome varchar(30));

ALTER TABLE cliente DROP COLUMN cli\_cod;

ALTER TABLE cliente

DROP CONSTRAINT SYS\_C007066 CASCADE;

SELECT \* FROM CLIENTE;

INSERT INTO CLIENTE VALUES (1, 'PAULO');

[CASCADE]

SELECT \* FROM user\_constraints;

WHERE table\_name='PEDIDO';

DESC PEDIDO;

INSERT into pedido values (1,100.30);

commit;

ALTER TABLE PEDIDO ADD PED\_DATA DATE;

ALTER TABLE PEDIDO MODIFY PED\_ID varchar(15);

ALTER TABLE PEDIDO DROP COLUMN PED\_DATA;