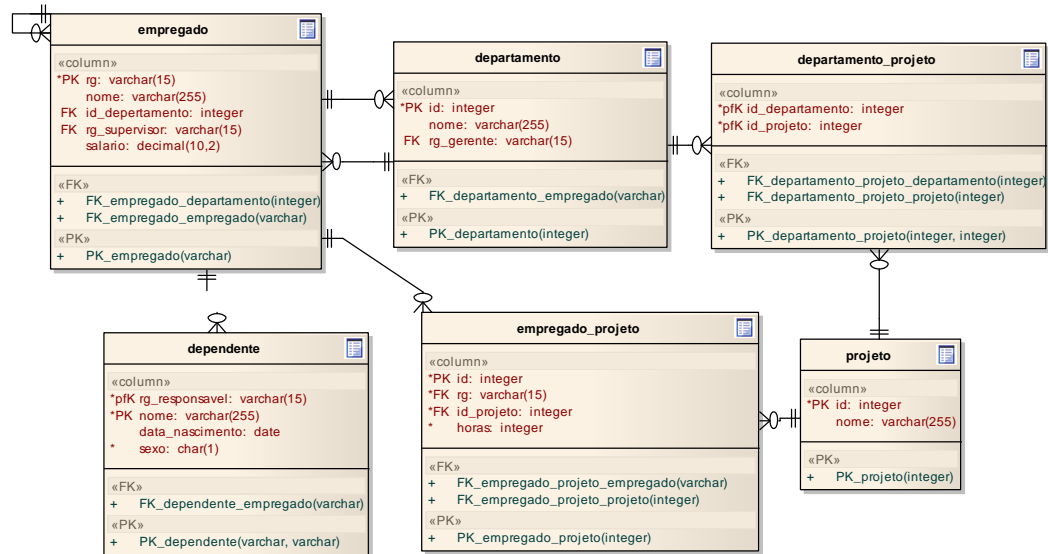


Formato: Enviar, pelo ambiente material didático, arquivo com extensão “.sql” contemplando as instruções para criação das *triggers* e das *triggers functions*.

Dado o modelo relacional abaixo crie *triggers* em SQL que executem as solicitações abaixo:



- Crie uma *trigger* que evite que sejam inseridos dependentes (na tabela dependentes) maiores de 18 anos com sexo masculino.
- Crie uma *trigger* que evite que um empregado seja supervisor dele mesmo.
- Crie uma *trigger* que faça exclusão em cascata todas as vezes que um projeto for excluído do banco de dados.
- Adicione o atributo mês (atributo do tipo inteiro) na tabela Empregado_Projeto. Em seguida, crie uma *trigger* que seja disparada sempre um registro for inserido/atualizado na tabela Empregado_Projeto. Essa trigger deve verificar se o número total de horas de um funcionário (fazer o somatório do número de horas para o empregado) no mês corrente não excede 180 horas. Se exceder, o registro não poderá ser inserido/atualizado.
- Crie uma *trigger* que, ao vincular um empregado a um projeto, atualize o valor do salário do empregado (adicione R\$50,00 ao salário) todas as vezes que o mesmo se envolver em um novo projeto.

```

-- Create Tables
CREATE SEQUENCE departamento_id_seq INCREMENT 1 START 1
;

CREATE TABLE departamento (
    id integer DEFAULT NEXTVAL("departamento_id_seq"::TEXT) NOT NULL,
    nome varchar(255),
    rg_gerente varchar(15)
)
;

CREATE TABLE departamento_projeto (
    id_departamento integer NOT NULL,
    id_projeto integer NOT NULL
)
;

CREATE TABLE dependente (
    rg_responsavel varchar(15) NOT NULL,
    nome varchar(255) NOT NULL,
    data_nascimento date,
    sexo char(1) NOT NULL -- F: Feminino, M: Masculino
)
;

COMMENT ON COLUMN dependente.sexo
IS 'F: Feminino, M: Masculino'
;

CREATE TABLE empregado (
    rg varchar(15) NOT NULL,
    nome varchar(255),
    id_departamento integer,
    rg_supervisor varchar(15),
    salario decimal(10,2)
)
;

CREATE SEQUENCE empregado_projeto_id_seq INCREMENT 1 START 1
;

CREATE TABLE empregado_projeto (
    id integer DEFAULT NEXTVAL("empregado_projeto_id_seq"::TEXT) NOT NULL,
    rg varchar(15) NOT NULL,
    id_projeto integer NOT NULL,
    horas integer NOT NULL
)
;

CREATE SEQUENCE projeto_id_seq INCREMENT 1 START 1
;

CREATE TABLE projeto (
    id integer DEFAULT NEXTVAL("projeto_id_seq"::TEXT) NOT NULL,
    nome varchar(255)
)
;

-- Create Primary Key Constraints
ALTER TABLE departamento ADD CONSTRAINT PK_departamento
PRIMARY KEY (id)
;

ALTER TABLE departamento_projeto ADD CONSTRAINT PK_departamento_projeto
PRIMARY KEY (id_departamento, id_projeto)
;

ALTER TABLE dependente ADD CONSTRAINT PK_dependente
PRIMARY KEY (rg_responsavel, nome)
;

ALTER TABLE empregado ADD CONSTRAINT PK_empregado
PRIMARY KEY (rg)
;

ALTER TABLE empregado_projeto ADD CONSTRAINT PK_empregado_projeto
PRIMARY KEY (id)
;

ALTER TABLE projeto ADD CONSTRAINT PK_projeto
PRIMARY KEY (id)
;

-- Create Foreign Key Constraints
ALTER TABLE departamento ADD CONSTRAINT FK_departamento_empregado
FOREIGN KEY (rg_gerente) REFERENCES empregado (rg)
;

ALTER TABLE departamento_projeto ADD CONSTRAINT FK_departamento_projeto_departamento
FOREIGN KEY (id_departamento) REFERENCES departamento (id)
;

ALTER TABLE departamento_projeto ADD CONSTRAINT FK_departamento_projeto_projeto
FOREIGN KEY (id_projeto) REFERENCES projeto (id)
;

ALTER TABLE dependente ADD CONSTRAINT FK_dependente_empregado
FOREIGN KEY (rg_responsavel) REFERENCES empregado (rg)
;

ALTER TABLE empregado ADD CONSTRAINT FK_empregado_departamento
FOREIGN KEY (id_departamento) REFERENCES departamento (id)
;

ALTER TABLE empregado ADD CONSTRAINT FK_empregado_empregado
FOREIGN KEY (rg_supervisor) REFERENCES empregado (rg)
;

ALTER TABLE empregado_projeto ADD CONSTRAINT FK_empregado_projeto_empregado
FOREIGN KEY (rg) REFERENCES empregado (rg)
;

ALTER TABLE empregado_projeto ADD CONSTRAINT FK_empregado_projeto_projeto
FOREIGN KEY (id_projeto) REFERENCES projeto (id)
;

```