



UNIVERSIDADE FEDERAL
DO ESPÍRITO SANTO

Centro Universitário Norte do Espírito Santo - CEUNES
Departamento de Computação e Eletrônica - DCEL
Disciplina de Banco de Dados - 2019/2
Professora - Maria das Graças da Silva Teixeira

DrTok

O Tok da Limpeza

Trabalho Final de Banco de Dados

Alunos: André Couto, Guilherme Panetto, Igor Pereira, Rafael Guimarães;

Resumo

O tema proposto para o nosso trabalho envolve uma empresa de terceirização de serviços de limpeza. O SGBD a ser elaborado permite gerenciar as questões de contratação do serviço (empresa-contratante), informações relacionadas aos funcionários (dados pessoais e alocação de trabalho) e estoque de produtos e materiais de limpeza.

Histórico de Revisão

Nome	Data	Motivo p/ alterações	Versão
André, Igor	23/10/2019	Criação do arquivo	1.0
Rafael, Guilherme	26/10/2019	Adição de descrições	1.1
André, Rafael	29/10/2019	Adição de Requisitos funcionais	1.2
Igor, Guilherme	31/10/2019	Correções gerais	1.3
André, Rafael	06/11/2019	Mudança de alguns requisitos	1.4
André, Guilherme	16/11/2019	Modificações/atualização de requisitos	1.5
Igor	23/11/2019	Limpeza e organização	1.6
André, Guilherme, Rafael	30/11/2019	Finalização	1.7

Atividade 1 – Especificação de Requisitos	3
1. Introdução	3
1.1 Objetivo do documento	3
1.2 Escopo do produto	3
1.3 Referências	3
1.4 Visão geral	3
2. Descrição geral	4
2.1 Descoberta de Requisitos	4
2.2 Funções do produto	4
2.4 Características do usuário	5
3. Requisitos	5
3.1 Interfaces de hardware	6
3.2 Interfaces de software	6
Atividade 2 – Modelagem	7
1. Construir o Modelo de Entidade e Relacionamento	7
2. Construir os Modelos Relacionais (Lógico e Físico)	8
2.1 Modelo Lógico	8
2.2 Modelo Físico	8
3. Dicionário de dados do Modelo Físico	9
4. Identificar as Restrições de Integridade	13
Atividade 3 – Scripts	14
1. Script de Criação	14
2. Script de Carga de Dados	14
3. Script de Testes	14
4. Script de Views	14
5. Script Stored Procedures	14
6. Script de Trigger	14
Atividade 4 – Elementos de Programação	15

Atividade 1 – Especificação de Requisitos

1. Introdução

Apresentamos, de uma forma sucinta, a definição, análise e especificação de requisitos de uma empresa de terceirização de serviços de limpeza, para permitir gerenciar as questões de contratação do serviço ofertado (empresa-contratante), informações relacionadas aos funcionários (dados pessoais e alocação de trabalho) e estoque de produtos e materiais de limpeza.

1.1 Objetivo do documento

O foco principal deste documento é o desenvolvimento de informações inerentes a sistemas gerenciadores de banco de dados aplicado à uma empresa prestadora de serviços de limpeza, faxina e higiene, utilizando-se dos conceitos e procedimentos abordados na disciplina Banco de Dados, elaborando e apresentando artefatos parciais.

1.2 Escopo do produto

O sistema a ser desenvolvido é a base da empresa DrTok, que atua no ramo de terceirização de limpeza. O sistema possibilita a gerência de seus funcionários e suas distinções, juntamente com o controle de estoque e a organização dos contratos com seus clientes. Todos estes recursos são de extrema importância para a gestão e funcionamento de cada setor da empresa, composta por cerca de trezentos funcionários, assim como seu desenvolvimento econômico e estrutural.

1.3 Referências

[Catálogo de base dos produtos](#)

1.4 Visão geral

Generalizando a pesquisa de produção do sistema, vamos desenvolver simplificada os requisitos para a construção parcial do software de gestão da empresa, focando principalmente nas modelagens que são necessárias para a manipulação do banco de dados.

2. Descrição geral

2.1 Descoberta de Requisitos

Na 1ª reunião, foi discutido abertamente o funcionamento da empresa e suas necessidades. O método de *brainstorming* foi escolhido, por ser mais dinâmico, recebendo opiniões diferentes com mais rapidez, assim, ajudando na evolução da pesquisa dos requisitos.

Após toda a discussão, obtivemos alguns pontos, que foram identificados, analisados e escolhidos como base para o sistema, são eles:

- Funcionário Administrativo
- Funcionário Externo
- Salário
- Contracheque
- Aloca
- Estoque
- Serviço
- Fornecedor
- Entrega
- Cliente
- Contrato
- Produto

Na seção 3 iremos aprofundar à cada requisito identificado, citado acima.

2.2 Funções do produto

Em resumo, algumas das funcionalidades que devem estar presentes no sistema.

- Gerenciamento de contratos
 - Gravando detalhes do Cliente
 - Atualização do registro com contrato
 - Impressão de despesas
 - Renovação de contrato
- Sistema de gerenciamento de funcionários e salários
 - Atendimento e horário dos funcionários
 - Aprovações de férias
 - Horário de trabalho
 - Cálculo impostos

- Gerenciamento de Estoque
 - Gerenciamento de estoque
 - Notificação de vencimento
 - Notificação de falta de produto
 - Cálculo de faturamento
- Gestão de Veículos e Transportes (**Invalidado**)
 - Gerenciamento de vans
 - Gerenciamento de tempo
 - Cálculos de faturamento

2.4 Características do usuário

- **Administrador** - Gerencia todas as funções do sistema
- **Gerente Comercial** - Gerencia as funcionalidades de contratos com clientes
- **Gerente Financeiro** - Gerencia as funcionalidades de compra de produtos
- **Gerente Recursos Humanos** - Gerencia as funcionalidades de funcionários
- **Gerente Operacional** - Gerencia as funcionalidades que envolvem os serviços
- **Sistema** - Gerencia os cálculos automáticos presentes no sistema (Preferencialmente em nível de Banco de dados)

3. Requisitos

O sistema DrTok é um sistema que deverá conter funcionalidades que atendam aos funcionários, agrupados em setores por suas categorias. Cada setor tem suas próprias funções, sendo eles Administrativo, Comercial, Financeiro, Recursos Humanos E Operacional. Os funcionários dos setores citados são alocados no escritório da empresa, exceto os de nível operacional, que prestam os serviços na empresa contratante do serviço (Cliente). De um funcionário precisamos saber Nome, CPF, Data De Nascimento, Os Vencimentos E Seu Endereço. Os vencimentos do funcionário ficam armazenados nos contracheques, que são divididos por mês. De um cliente, precisamos de CNPJ, Endereço, Inscrição Estadual, E-mail, Telefone Fixo, Celular, Nome, data de criação do cadastro (fidelidade). Cada cliente gera um contrato de serviço e precisamos armazenar as informações de descrição e preço. Para os produtos, precisamos saber seu Nome, Código de Barras, Validade e Preço, esse produto é comprado de algum fornecedor de produtos de limpeza, que precisamos saber o CNPJ, Endereço, Inscrição Estadual, E-mail, Telefone Fixo, Celular, Nome, especialidade de seus produtos e data de criação do cadastro (fidelidade). Todos os produtos ficam localizados em estoques, que podem estar espalhados por diversas cidades, precisamos saber o Nome e Endereço.

3.1 Interfaces de hardware

- Notebook/Desktop PC
 - Processador - Core i3
 - RAM - 4GB
 - HDD - 500GB
 - A proposta deste PC é para a utilização no ambiente interno da empresa, no qual todos os setores necessários para o funcionamento são afetados. Para a execução simples do sistema, precisamos de um PC razoavelmente rápido, para atender à mínima demanda, sem perda de desempenho por longos períodos, facilitado por ser um sistema implantado localmente.
- LED/LCD Monitor/TV
 - Com o intuito das equipes ficarem sempre a par de seus próximos trabalhos, tela em pública da empresa, onde os funcionários terão acesso fácil.
- Impressora Laser (P/B)
 - Dispositivo para impressão de relatórios.
- Roteador Wi-Fi
 - Dispositivo para possibilitar o acesso à internet, aos serviços do servidor de dados do sistema.

3.2 Interfaces de software

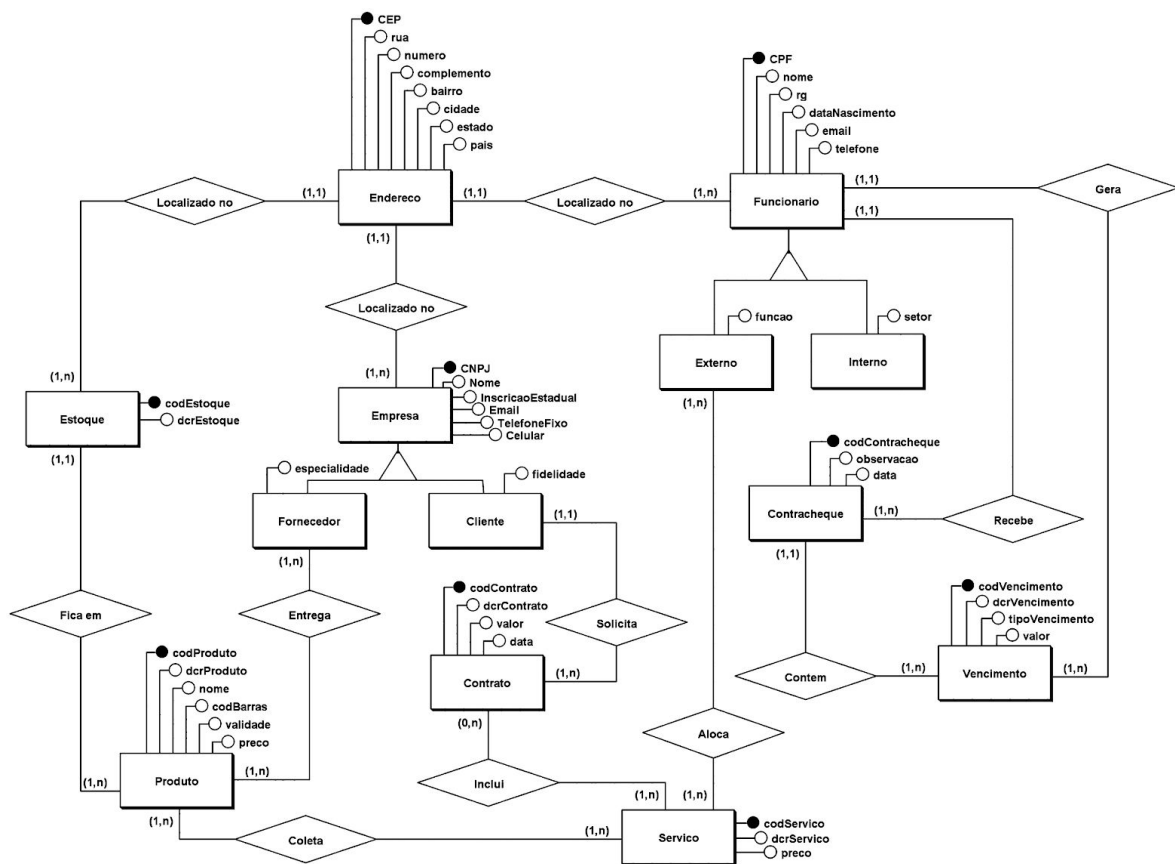
- OS
 - Windows 7 ou posterior, Linux Mint ou equivalente.
- JRE 1.8
 - JAVA Runtime Environment para uma possível implementação gráfica do sistema
- MySQL server
 - Conexão com o banco de Dados.

Atividade 2 – Modelagem

1. Construir o Modelo de Entidade e Relacionamento

O MER está localizado em anexo nos arquivos:

- BrModelo ([DrTok - MER Conceitual\).brm3](#))
- Imagem PNG (DrTok - MER (Conceitual).png).

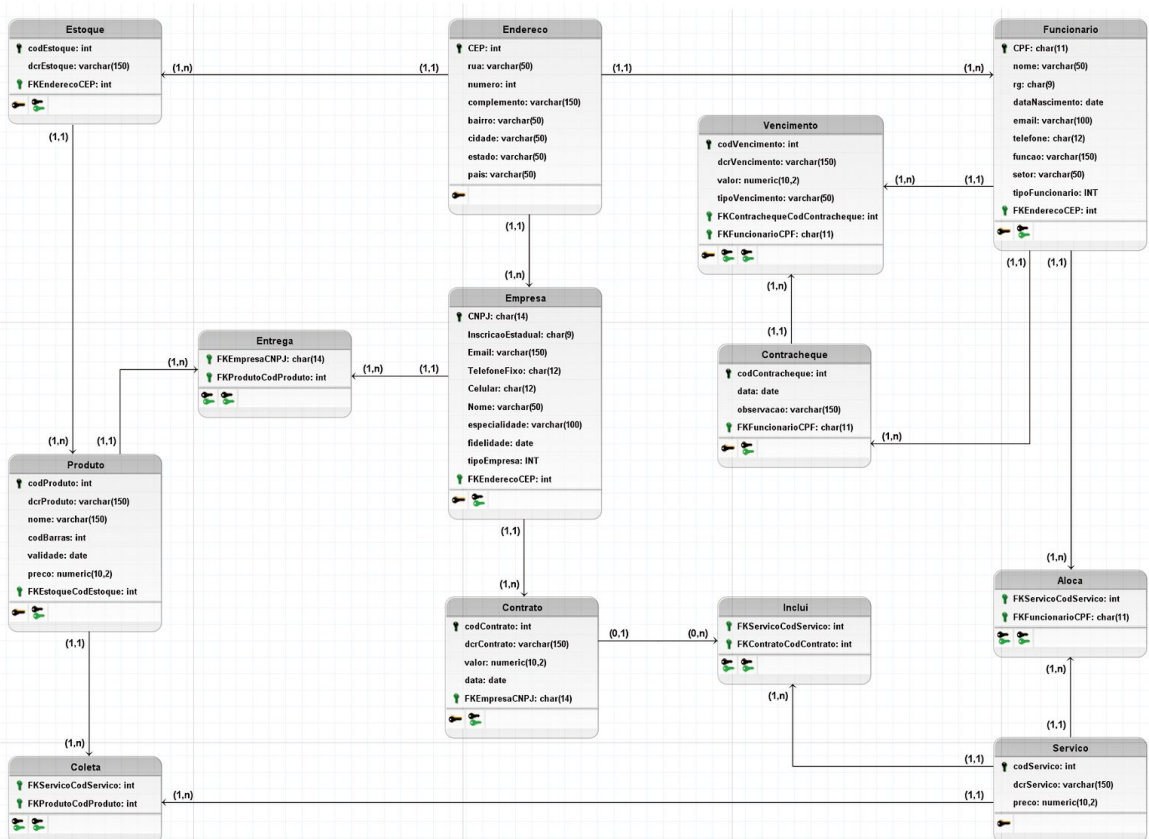


2. Construir os Modelos Relacionais (Lógico e Físico)

2.1 Modelo Lógico

O Modelo Relacional Lógico está localizado nos arquivos:

- BrModelo ([DrTok - MER \(Lógico\).brm3](#))
- Imagem PNG (DrTok - MER (Lógico).png).



2.2 Modelo Físico

O Modelo Relacional físico está localizado nos arquivos:

- SQL ([DrTok - Físico.SQL](#))

3. Dicionário de dados do Modelo Físico

Tabela	Empresa				
Descrição	Organização econômica, civil ou comercial, constituída para explorar um ramo de negócio, podendo ser cliente ou fornecedor				
Observações	Possui chave estrangeira da tabela Endereço				
Campos					
Nome	Descrição	Tipo	Tamanho	Restrições	
CNPJ	Número único que identifica uma pessoa jurídica	char	14	NOT NULL	
Nome	Nome da empresa	varchar	50	NOT NULL	
InscricaoEstadual	Registro formal do negócio junto à Receita Estadual.	char	9	NOT NULL	
Email	Endereço eletrônico da empresa	varchar	150	NOT NULL	
TelefoneFixo	Telefone de contato fixo da empresa	char	12	NOT NULL	
Celular	Telefone de contato móvel da empresa	char	12	NOT NULL	

Tabela	Fornecedor				
Descrição	Organização que fornece produtos para realização do serviço				
Observações	Tabela derivada de Empresa				
Campos					
Nome	Descrição	Tipo	Tamanho	Restrições	
especialidade	Especialidade do fornecedor	varchar	100	NOT NULL	

Tabela	Cliente				
Descrição	Organização que por meio de um contrato adquire serviço da empresa				
Observações	Tabela derivada de Empresa				
Campos					
Nome	Descrição	Tipo	Tamanho	Restrições	
fidelidade	Quando o primeiro contrato foi estabelecido	date		NOT NULL	

Tabela	Funcionario			
Descrição	Mão de obra assalariada contratada pela empresa para prestação de serviços			
Observações	Possui chave estrangeira da tabela Endereço			
Campos				
Nome	Descrição	Tipo	Tamanho	Restrições
CPF	Número único que identifica uma pessoa física	char	11	NOT NULL
nome	Nome do funcionário	varchar	50	NOT NULL
rg	Documento nacional de identificação civil	char	9	NOT NULL
dataNascimento	Data de nascimento do funcionário	date		NOT NULL
email	Endereço eletrônico do funcionário	varchar	100	NOT NULL
telefone	Telefone para contato do funcionário	char	12	NOT NULL

Tabela	Interno			
Descrição	Funcionário que trabalha na parte interna da empresa, escritório e administração.			
Observações	Tabela derivada de Funcionário			
Campos				
Nome	Descrição	Tipo	Tamanho	Restrições
setor	Setor do funcionário interno	varchar	50	NOT NULL

Tabela	Externo			
Descrição	Funcionário que trabalha na parte externa da empresa, faxineiro, motorista.			
Observações	Tabela derivada de Funcionário			
Campos				
Nome	Descrição	Tipo	Tamanho	Restrições
funcao	Função do funcionário externo	varchar	150	NOT NULL

Tabela	Contracheque				
Descrição	Documento que demonstra e certifica o pagamento do salário de um funcionário				
Observações	Possui uma chave estrangeira da tabela Funcionário				
Campos					
Nome	Descrição	Tipo	Tamanho	Restrições	
codContracheque	Código do contracheque	int		NOT NULL	
data	Data do contracheque	date		NOT NULL	
observacao	Observações relacionadas ao contracheque	varchar	150	NOT NULL	

Tabela	Vencimento				
Descrição	Tipos de pagamento do funcionário				
Observações	Possui uma chave estrangeira da tabela Funcionário e Contracheque				
Campos					
Nome	Descrição	Tipo	Tamanho	Restrições	
codVencimento	Código do vencimento	int		NOT NULL	
dcrVencimento	Descrição do vencimento	varchar	150	NOT NULL	
tipoVencimento	Tipo do vencimento, receita ou desconto	int		NOT NULL	
valor	Valor do vencimento	numeric	10.2	NOT NULL	

Tabela	Estoque			
Descrição	Materiais e suprimentos que a empresa mantém para fornecer insumos ou suprimentos para o processo de serviço			
Observações	Possui chave estrangeira da tabela Endereço			
Campos				
Nome	Descrição	Tipo	Tamanho	Restrições
codEstoque	Código do estoque	int		NOT NULL
dcrEstoque	Descrição do estoque	varchar	150	NOT NULL

Tabela	Servico			
Descrição	Produto da atividade dos funcionários externos que satisfaz a uma necessidade, sem assumir a forma de um bem material			
Observações				
Campos				
Nome	Descrição	Tipo	Tamanho	Restrições
codServico	Código do serviço	int		NOT NULL
dcrServico	Descrição do serviço	varchar	150	NOT NULL
preco	Valor do serviço	numeric	10.2	NOT NULL

Tabela	Produto			
Descrição	Um bem adquirido de um fornecedor, que é utilizado para a execução dos serviços pelos funcionários.			
Observações	Possui uma chave estrangeira da tabela Estoque			
Campos				
Nome	Descrição	Tipo	Tamanho	Restrições
codProduto	Código do produto	int		NOT NULL
dcrProduto	Descrição do produto	varchar	150	NOT NULL
nome	Nome do produto	varchar	150	NOT NULL
codBarras	Representação gráfica de dados numéricos do produto	int		NOT NULL
validade	Data de validade do produto	date		NOT NULL
preco	Valor do produto	numeric	10.2	NOT NULL

Tabela	Contrato			
Descrição	Vínculo jurídico entre a empresa e o cliente de direito correspondido pela vontade, da responsabilidade do ato firmado			
Observações	Possui uma chave estrangeira da tabela Empresa			
Campos				
Nome	Descrição	Tipo	Tamanho	Restrições
codContrato	Código do contrato	int		NOT NULL
dcrContrato	Descrição do contrato	varchar	150	NOT NULL
valor	Valor do contrato	numeric	10.2	NOT NULL

4. Identificar as Restrições de Integridade

- Na tabela de Produto, uma data de validade não pode ser inferior a data presente no momento da criação do registro;
- Na tabela Vencimento, o tipo de vencimento pode assumir apenas 2 valores, sendo eles receita e desconto;
- Na tabela Empresa o tipo de empresa pode assumir apenas 2 valores, sendo eles 1 e 2;
- Na tabela de Contracheque, uma data não pode ser inferior a data presente no momento da criação do registro;
- Na tabela de Vencimento, um valor não pode ser menor que 0 (zero);

Atividade 3 – Scripts

1. Script de Criação

A criação do esquema, das tabelas com as restrições de integridade presentes do Banco de dados DrTok, estão no arquivo [Modelo Físico.sql](#)

2. Script de Carga de Dados

A carga de dados está no arquivo [Inserts.sql](#)

3. Script de Testes

A carga de Testes básicos está no arquivo [Conferência.sql](#)

4. Script de Views

A carga de Views está no arquivo [Views.sql](#)

5. Script Stored Procedures

A carga de Stored Procedures está no arquivo [StoredProcedures.sql](#)

6. Script de Trigger

A carga dos Trigger está no arquivo [Triggers.sql](#)

Atividade 4 – Elementos de Programação

ANEXOS

A seguir temos alguns exemplos para uma leitura rápida, antes da execução dos arquivos SQL, melhorando a compreensão do que foi feito no trabalho, e o conhecimento adquirido por todos do grupo, sendo futuramente compartilhado em uma comunidade de códigos para programação.

```
1 DROP SCHEMA IF EXISTS `DRTOK`;
2 CREATE SCHEMA `DRTOK`;
3 USE `DRTOK`;
4
5 CREATE TABLE Endereco (
6     CEP INT NOT NULL,
7     rua VARCHAR(50) NOT NULL,
8     numero INT NOT NULL,
9     complemento VARCHAR(150) NOT NULL,
10    bairro VARCHAR(50) NOT NULL,
11    cidade VARCHAR(50) NOT NULL,
12    estado VARCHAR(50) NOT NULL,
13    pais VARCHAR(50) NOT NULL,
14    PRIMARY KEY (CEP)
15 );
16
17 CREATE TABLE Funcionario (
18     CPF CHAR(11) NOT NULL,
19     nome VARCHAR(50) NOT NULL,
20     rg CHAR(9) NOT NULL,
21     dataNascimento DATE NOT NULL,
22     email VARCHAR(100) NOT NULL,
23     telefone CHAR(12) NOT NULL,
24     funcao VARCHAR(150) NOT NULL,
25     setor VARCHAR(50) NOT NULL,
26     tipoFuncionario INT NOT NULL CHECK(tipoFuncionario in (1,2)),
27     FKEnderecoCEP INT NOT NULL,
28     PRIMARY KEY (CPF),
29     CONSTRAINT FKFuncionario FOREIGN KEY (FKEnderecoCEP)
30         REFERENCES Endereco (CEP) ON DELETE RESTRICT ON UPDATE RESTRICT
31 );
```

```

1 ## Insert
2 ##### Inserts, a Carga de dados para fazer o Banco de dados Girar, e possibilitar a construção de relações
  de dados reais
3
4 USE `DRTOK`;
5
6 INSERT INTO
7 Vencimento (dcrVencimento, valor, tipoVencimento, FKContrachequeCodContracheque, FKFuncionarioCPF) VALUES
8   ("SALÁRIO BASE", 5892.43, "RECEITA", 1, "77209625755"),
9   ("SALÁRIO FAMÍLIA", 647.83, "RECEITA", 1, "77209625755"),
10  ("FÉRIAS", 591.42, "RECEITA", 1, "77209625755"),
11  ("GRATIFICAÇÃO", 538.90, "RECEITA", 1, "77209625755"),
12  ("BONIFICAÇÃO", 883.61, "RECEITA", 1, "77209625755"),
13  ("HORAS EXTRAS", 719.41, "RECEITA", 1, "77209625755"),
14  ("ADICIONAL INSALUBRIDADE", 818.27, "RECEITA", 1, "77209625755"),
15  ("ADICIONAL NOTURNO", 920.54, "RECEITA", 1, "77209625755"),
16  ("ADICIONAL PONTUALIDADE", 188.46, "RECEITA", 1, "77209625755"),
17  ("AUXÍLIO ALIMENTAÇÃO", 472.95, "RECEITA", 1, "77209625755"),
18  ("FALTAS", 507.97, "DESCONTO", 1, "77209625755"),
19  ("INSS", 406.74, "DESCONTO", 1, "77209625755"),
20  ("VALE TRANSPORTE", 989.54, "DESCONTO", 1, "77209625755"),
21  ("CONVÊNIO", 864.07, "DESCONTO", 1, "77209625755"),
22  ("ADIANTAMENTO QUINZENAL", 664.68, "DESCONTO", 1, "77209625755"),
23  ("CONTRIBUIÇÃO SINDICAL", 384.74, "DESCONTO", 1, "77209625755"),
24  ("SALÁRIO BASE", 3461.48, "RECEITA", 2, "31906491704"),
25  ("SALÁRIO FAMÍLIA", 580.76, "RECEITA", 2, "31906491704"),
26  ("FÉRIAS", 973.56, "RECEITA", 2, "31906491704"),
27  ("GRATIFICAÇÃO", 861.50, "RECEITA", 2, "31906491704"),

```

```

1 # Atividade 4 – Elementos de Programação
2
3 USE `DRTOK`;
4 ## Teste de construção
5 ## Exemplos de Selects
6 ##### Identificar selects básicos para exemplificar o funcionamento do BD
7 ##### Ver todos os funcionários de uma mesma cidade
8 SELECT f.nome as Nome, e.cep as CEP, e.rua as Rua, e.cidade as Cidade
9 FROM Funcionario as f, Endereco as e
10 WHERE e.cep = f.FKenderecocep AND e.cidade = "Colatina"
11 ORDER BY f.nome;
12
13 ##### Ver o endereço dos Estoques
14 SELECT es.dcrestoque as Estoque, e.cep as CEP, e.rua as Rua, e.cidade as Cidade
15 FROM Estoque as es, Endereco as e
16 WHERE e.cep = es.FKenderecocep
17 ORDER BY e.cidade;
18
19 ##### Ver todos os Funcionários que são analistas
20 SELECT f.nome as Nome, f.setor as Setor, f.funcao as Função
21 FROM Funcionario as f
22 WHERE f.funcao = "analista"
23 ORDER BY f.setor;

```



```

1 ## Visões
2 ### Identificar partes do modelo bastante suscetíveis a consultas
3 ### Implementar pelo menos duas views para as partes do modelo
4
5 USE `DRTOK`;
6
7 #####
8 # 1 A descoberta dos produtos (ou localizados em um estoque),
9 # utilizado para fazer um pedido de reposição de produtos no estoque
10
11 DROP VIEW IF EXISTS Suprimentos;
12 CREATE VIEW Suprimentos AS
13     SELECT p.codproduto as Código,
14            p.nome as Nome,
15            e.dcrestoque as Estoque,
16            p.dcrproduto as Descrição,
17            p.preco as Preço,
18            p.validade as Validade
19     FROM Produto as p, Estoque as e
20     -- 1/2 Opcionalmente podemos modificar a linha para conseguir
21     -- produtos de todos os estoques
22     -- WHERE p.fkestoquecodestoque = (
23     --     SELECT es.codestoque
24     --     FROM estoque as es, endereco as e
25     --     WHERE es.fkenderecocep = e.cep AND e.cidade = "colatina")
26     WHERE p.fkestoquecodestoque = e.codestoque
27     ORDER BY e.dcrestoque;
28
29 SELECT * FROM Suprimentos;
30
31 #####
32 # 2 Descoberta de todos os produtos de um serviço e com a identificação do Fornecedor
33
34 DROP VIEW IF EXISTS FornecedorProdutoServico;
35 CREATE VIEW FornecedorProdutoServico AS
36     SELECT s.dcrServico as Serviço,
37            p.codProduto as Código,
38            p.nome as Produto,
39            em.nome as Fornecedor
40     FROM Produto as p, Entrega as e, coleta as c, servico as s, estoque as es, empresa as em
41     WHERE em.cnpj = e.fkempresacnpj
42            AND p.codProduto = e.FKProdutoCodProduto
43            AND p.codProduto = c.fkprodutocodproduto
44            AND c.fkservicocodservico = s.codServico
45            AND p.fkestoquecodestoque = es.codEstoque
46     ORDER BY s.codServico, em.cnpj;
47
48 SELECT * FROM FornecedorProdutoServico;

```

```

1 ## Stored Procedures
2 ### Identificar duas situações para implementação de procedures
3 ### Implementar as procedures e os casos de testes das procedures
4
5 USE `DRTOK`;
6
7 #####
8 # Procedure que calcula os ganhos de um funcionário com os descontos
9
10 DROP PROCEDURE IF EXISTS Pagamento;
11 DELIMITER //
12 CREATE PROCEDURE Pagamento(IN nomeFuncionario varchar(50))
13 BEGIN
14     SELECT f.Nome as Nome, f.funcao as Função, (
15         (SELECT SUM(v.valor)
16          FROM vencimento as v, contracheque as c, funcionario as f
17          WHERE v.fkcontrachequecodcontracheque = c.codcontracheque
18               AND v.fkfuncionariocpf = f.cpf
19               AND f.cpf = (
20                   SELECT cpf
21                   FROM Funcionario
22                   WHERE nome = nomeFuncionario) AND v.tipovencimento = 'RECEITA'
23         ) - (SELECT SUM(v.valor)
24             FROM vencimento as v, contracheque as c, funcionario as f
25             WHERE v.fkcontrachequecodcontracheque = c.codcontracheque
26                  AND v.fkfuncionariocpf = f.cpf
27                  AND f.cpf = (
28                      SELECT cpf
29                      FROM Funcionario
30                      WHERE nome = nomeFuncionario) AND v.tipovencimento = 'DESCONTO'
31         ))
32     as Pagamento
33 FROM vencimento as v, contracheque as c, funcionario as f
34 WHERE v.fkcontrachequecodcontracheque = c.codcontracheque
35      AND v.fkfuncionariocpf = f.cpf
36      AND f.cpf = (
37          SELECT cpf
38          FROM Funcionario
39          WHERE nome = nomeFuncionario)
40 GROUP BY f.nome;
41 END //
42 DELIMITER ;
43
44 CALL Pagamento('Cláudio Thales da Mota');

```

```

1 ## Triggers
2 ### Identificar duas situações para implementação de triggers
3 ### Implementar os triggers e os casos de testes dos triggers
4
5 USE `DRTOK`;
6
7 #####
8 # Trigger que trata o caso de restrição de integridade do atributo DATA na tabela Contrato
9
10 DROP TRIGGER IF EXISTS TgrContratoVerificaData;
11 DELIMITER //
12 CREATE TRIGGER TgrContratoVerificaData
13 BEFORE INSERT ON contrato
14 FOR EACH ROW
15 BEGIN
16     IF (CURDATE() > new.data) THEN
17         SET new.data = NULL;
18     END IF;
19 END //
20 DELIMITER ;
21
22 #####
23 # Acompanha o Trigger TgrContratoVerificaData, no qual este trata
24 ## o erro de inserção, caso a data ou outro campo esteja inválido
25
26 DROP PROCEDURE IF EXISTS spInsereContrato;
27 DELIMITER //
28 CREATE PROCEDURE spInsereContrato ( IN spdcrContrato VARCHAR(50),
29                                     IN spvalor NUMERIC(10,2),
30                                     IN spdata DATE,
31                                     IN spFKEmpresaCNPJ CHAR(14))
32 BEGIN
33     DECLARE EXIT HANDLER FOR SQLSTATE '23000'
34     BEGIN
35         SELECT 'Problemas com a inserção data, Data anterior à hoje!' as ERRO;
36     END;
37
38     INSERT INTO Contrato (dcrContrato, valor, data, FKEmpresaCNPJ) VALUES
39     (spdcrContrato, spvalor, spdata, spFKEmpresaCNPJ);
40
41 END//
42 DELIMITER ;
43
44 #####
45 Caso de teste para o TgrContratoVerificaData e spInsereContrato.
46 CALL spInsereContrato('MasterClean SupremeShine', 1432.22, '2019-11-30', '80367007000185');

```