.

Faculdade de Informática e Administração Paulista

DATABASE APPLICATION & DATA SCIENCE

**Challenge – Sprint 3**

**INTEGRANTES**

|  |  |
| --- | --- |
| **RM**  **(SOMENTE NÚMEROS)** | **NOME COMPLEMENTO**  **(SEM ABREVIAR)** |
| **93613** | **Vinicius de Oliveira** |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |

Sumário

[1 – Descrição do Projeto e Regras de Negócio 5](#_Toc67673069)

[2 – Dicionário de Dados 6](#_Toc67673070)

[3 – Projeto Lógico do Banco de Dados 7](#_Toc67673071)

[4 – Projeto Físico do Banco de Dados 8](#_Toc67673072)

[5 – Data Definition Language – DDL 9](#_Toc67673073)

[6 – Data Manupulation Language – DML (INSERT) 10](#_Toc67673074)

[7 – Data Manupulation Language – DML (UPDATE / DELETE) 11](#_Toc67673075)

[8 – Data Query Language – DQL (SELECT) 12](#_Toc67673076)

[8.1 – Relatório simples contendo apenas uma tabela com dados ordenados. 12](#_Toc67673077)

# 1 – Descrição do Projeto e Regras de Negócio

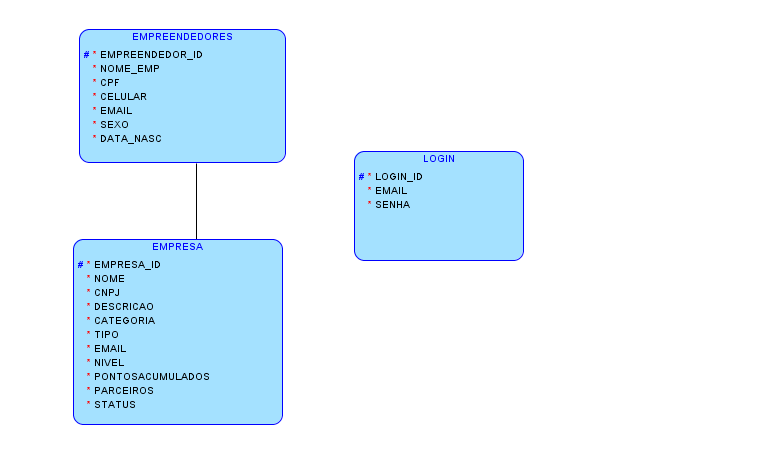
**A inovação e a transformação tecnológica podem acabar trazendo diversos impactos práticos aos empreendedores brasileiros. Alguns desses impactos seriam a redução de tarefas manuais, como gestão de tarefas, redução de mão de obra, diminuição de desperdícios, além da melhoria na qualidade dos produtos, e outros. Desse modo, o investimento em processos de automação e em tecnologia transformam a realidade de diversas indústrias, desde a matéria-prima até o consumidor final.**

**A AgroSync coleta e analisa as informações do ambiente de produção de agricultura brasileira e tem como objetivo principal auxiliar os agentes do agro nacional a realizar uma gestão automatizada e personalizada de suas plantações. Ademais, essa plataforma auxilia no monitoramento remoto de plantações, na previsão de safras e condições climáticas, disponibilizando aos seus clientes uma orientação personalizada para otimização da produção e uma comunicação direta com especialistas em agricultura, se necessário. Por fim, o AgroSync analisa dados fornecidos à ele, para que possua uma tomada de decisões mais precisa.**

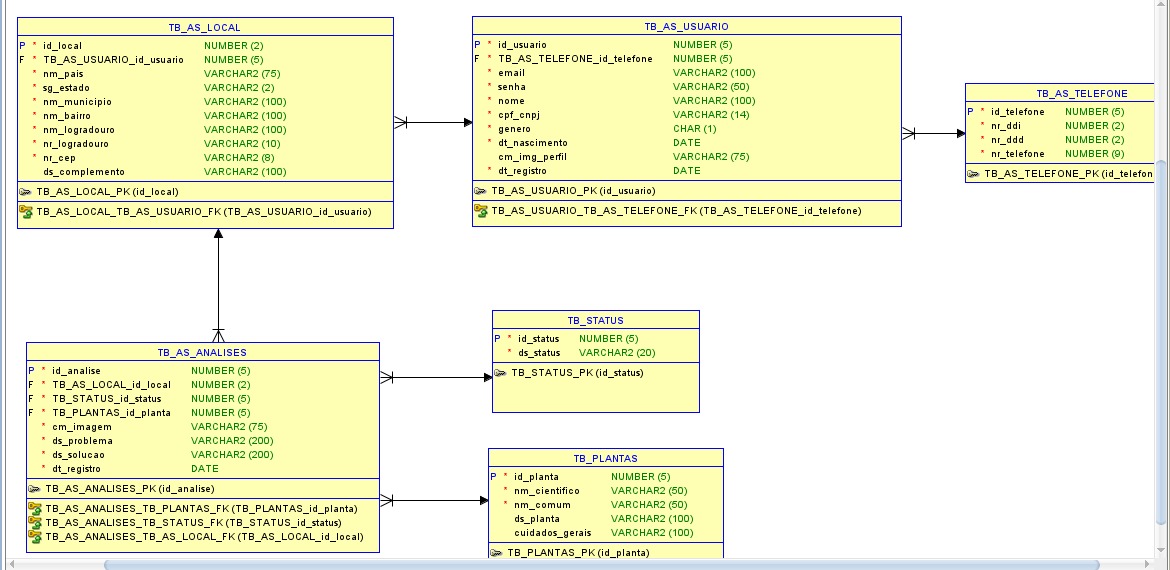
## 2 – Dicionário de Dados

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Tabela** | EMPRESA | | | |
| **Descrição** | Tabela de Empresas. Relaciona-se com a tabela EMPREENDEDOR | | | |
| **Coluna** | **Tipo de Dados** | **Tamanho** | **Constraint** | **Descrição** |
| empresa\_id | NUMBER | 7 | PK | Identificador único da empresa |
| nome | VARCHAR | 50 | NN | nome da empresa |
| cnpj | VARCHAR | 50 | NN | CNPJ da empresa |
| descricao | VARCHAR | 100 | NN | Tipo da empresa |
| categoria | VARCHAR | 20 | NN | Categoria da empresa |
| tipo | VARCHAR | 20 | NN | Tipo da empresa |
| email | VARCHAR | 200 | NN | E-mail da empresa |
| nivel | NUMBER | 7 | NN | Nível da empresa |
| pontosacumulados | NUMBER | 7 | NN | Pontos da empresa |
| parceiros | NUMBER | 7 | NN | Parceiros da empresa |
| status | NUMBER | 7 | NN | Status da conta da empresa (1 = Ativo, 0 = Inativo) |

## 3 – Projeto Lógico do Banco de Dados



## 4 – Projeto Físico do Banco de Dados



## 5 – Data Definition Language – DDL

|  |
| --- |
| -- Removendo tabelas e sequences  DROP TABLE tb\_as\_analise CASCADE CONSTRAINTS;  DROP TABLE tb\_as\_local CASCADE CONSTRAINTS;  DROP TABLE tb\_as\_telefone CASCADE CONSTRAINTS;  DROP TABLE tb\_as\_usuario CASCADE CONSTRAINTS;  DROP TABLE tb\_as\_planta CASCADE CONSTRAINTS;  DROP TABLE tb\_as\_status CASCADE CONSTRAINTS;  DROP TABLE tb\_as\_auditoria CASCADE CONSTRAINTS;  DROP TABLE TB\_AS\_ERRO;  DROP SEQUENCE seq\_id\_analise;  DROP SEQUENCE seq\_id\_local;  DROP SEQUENCE seq\_id\_telefone;  DROP SEQUENCE seq\_id\_usuario;  DROP SEQUENCE seq\_id\_planta;  DROP SEQUENCE seq\_id\_status;  DROP SEQUENCE seq\_id\_registro;  DROP SEQUENCE seq\_id\_erro;  -- Criação de tabelas e sequences  -- Gerado por Oracle SQL Developer Data Modeler 22.2.0.165.1149  -- em: 2023-08-19 12:32:53 BRT  -- site: Oracle Database 11g  -- tipo: Oracle Database 11g  -- predefined type, no DDL - MDSYS.SDO\_GEOMETRY  -- predefined type, no DDL - XMLTYPE  CREATE TABLE tb\_as\_analise (  id\_analise NUMBER(5) NOT NULL,  id\_local NUMBER(2) NOT NULL,  id\_status NUMBER(5) NOT NULL,  id\_planta NUMBER(5) NOT NULL,  cm\_imagem VARCHAR2(75) NOT NULL,  ds\_problema VARCHAR2(200) NOT NULL,  ds\_solucao VARCHAR2(200),  dt\_registro DATE NOT NULL  );  COMMENT ON COLUMN tb\_as\_analise.id\_analise IS  'Este campo sera para armazenar o id da analise. Ele e obrigatorio e chave primaria.';  COMMENT ON COLUMN tb\_as\_analise.id\_local IS  'Este campo sera para armazenar o id do local. Ele e obrigatorio e chave estrageira.';  COMMENT ON COLUMN tb\_as\_analise.id\_status IS  'Este campo sera para armazenar o id do status. Ele e obrigatorio e chave estrageira.';  COMMENT ON COLUMN tb\_as\_analise.id\_planta IS  'Este campo sera para armazenar o id da planta. Ele e obrigatorio e chave estrageira.';  COMMENT ON COLUMN tb\_as\_analise.cm\_imagem IS  'Este campo sera para armazenar o caminho da imagem da analise. Ele e obrigatorio.';  COMMENT ON COLUMN tb\_as\_analise.ds\_problema IS  'Este campo sera para armazenar a descrição da analise. Ele e obrigatorio.';  COMMENT ON COLUMN tb\_as\_analise.ds\_solucao IS  'Este campo sera para armazenar a descrição da solucao. Ele e obrigatorio.';  COMMENT ON COLUMN tb\_as\_analise.dt\_registro IS  'Este campo sera para armazenar a data de registro da solucao. Ele e obrigatorio.';  ALTER TABLE tb\_as\_analise ADD CONSTRAINT tb\_as\_analises\_pk PRIMARY KEY ( id\_analise );  CREATE TABLE tb\_as\_local (  id\_local NUMBER(2) NOT NULL,  id\_usuario NUMBER(5) NOT NULL,  nm\_pais VARCHAR2(75) NOT NULL,  sg\_estado VARCHAR2(2) NOT NULL,  nm\_municipio VARCHAR2(100) NOT NULL,  nm\_bairro VARCHAR2(100) NOT NULL,  nm\_logradouro VARCHAR2(100) NOT NULL,  nr\_logradouro VARCHAR2(10) NOT NULL,  nr\_cep VARCHAR2(8) NOT NULL,  ds\_complemento VARCHAR2(100)  );  COMMENT ON COLUMN tb\_as\_local.id\_local IS  'Este campo sera para armazenar o id do local. Ele e obrigatorio e chave primaria.';  COMMENT ON COLUMN tb\_as\_local.id\_usuario IS  'Este campo sera para armazenar o id do usuario. Ele e obrigatorio e chave estrangeira..';  COMMENT ON COLUMN tb\_as\_local.nm\_pais IS  'Este campo sera para armazenar o nome do pais do local. Ele e obrigatorio.';  COMMENT ON COLUMN tb\_as\_local.sg\_estado IS  'Este campo sera para armazenar a sigla do estado do local. Ele e obrigatorio.';  COMMENT ON COLUMN tb\_as\_local.nm\_municipio IS  'Este campo sera para armazenar o nome do municipio do local. Ele e obrigatorio.';  COMMENT ON COLUMN tb\_as\_local.nm\_bairro IS  'Este campo sera para armazenar o nome do bairro do local. Ele e obrigatorio.';  COMMENT ON COLUMN tb\_as\_local.nm\_logradouro IS  'Este campo sera para armazenar o nome do logradouro do local. Ele e obrigatorio.';  COMMENT ON COLUMN tb\_as\_local.nr\_logradouro IS  'Este campo sera para armazenar o numero do logradouro do local. Ele e obrigatorio.';  COMMENT ON COLUMN tb\_as\_local.nr\_cep IS  'Este campo sera para armazenar o cep do local. Ele e obrigatorio.';  COMMENT ON COLUMN tb\_as\_local.ds\_complemento IS  'Este campo sera para armazenar o complemento local. Ele e obrigatorio.';  ALTER TABLE tb\_as\_local ADD CONSTRAINT tb\_as\_local\_pk PRIMARY KEY ( id\_local );  CREATE TABLE tb\_as\_telefone (  id\_telefone NUMBER(5) NOT NULL,  nr\_ddi NUMBER(2) NOT NULL,  nr\_ddd NUMBER(2) NOT NULL,  nr\_telefone NUMBER(9) NOT NULL  );  COMMENT ON COLUMN tb\_as\_telefone.id\_telefone IS  'Este campo sera para armazenar o id do telefone. Ele e obrigatorio e chave primaria.';  COMMENT ON COLUMN tb\_as\_telefone.nr\_ddi IS  'Este campo sera para armazenar o ddi do telefone. Ele e obrigatorio';  COMMENT ON COLUMN tb\_as\_telefone.nr\_ddd IS  'Este campo sera para armazenar o ddd do telefone. Ele e obrigatorio.';  COMMENT ON COLUMN tb\_as\_telefone.nr\_telefone IS  'Este campo sera para armazenar o número do telefone. Ele e obrigatorio.';  ALTER TABLE tb\_as\_telefone ADD CONSTRAINT tb\_as\_telefone\_pk PRIMARY KEY ( id\_telefone );  CREATE TABLE tb\_as\_usuario (  id\_usuario NUMBER(5) NOT NULL,  id\_telefone NUMBER(5) NOT NULL,  email VARCHAR2(100) NOT NULL,  senha VARCHAR2(50) NOT NULL,  nome VARCHAR2(100) NOT NULL,  cpf\_cnpj VARCHAR2(14) NOT NULL,  genero CHAR(1) NOT NULL,  dt\_nascimento DATE NOT NULL,  cm\_img\_perfil VARCHAR2(75),  dt\_registro DATE NOT NULL  );  COMMENT ON COLUMN tb\_as\_usuario.id\_usuario IS  'Este campo sera para armazenar o id do usuário. Ele e obrigatorio e chave primaria.';  COMMENT ON COLUMN tb\_as\_usuario.id\_telefone IS  'Este campo sera para armazenar o id do telefone do usuario. Ele e obrigatorio e chave estrangeira.';  COMMENT ON COLUMN tb\_as\_usuario.email IS  'Este campo sera para armazenar o email do usuario e sera usado como login. Ele e obrigatorio.';  COMMENT ON COLUMN tb\_as\_usuario.senha IS  'Este campo sera para armazenar a senha do usuario e sera usado como senha para o login. Ele e obrigatorio.';  COMMENT ON COLUMN tb\_as\_usuario.nome IS  'Este campo sera para armazenar o nome do usuario. Ele e obrigatorio.';  COMMENT ON COLUMN tb\_as\_usuario.cpf\_cnpj IS  'Este campo sera para armazenar o cpd ou cnpj do usuario. Ele e obrigatorio.';  COMMENT ON COLUMN tb\_as\_usuario.genero IS  'Este campo sera para armazenar o genero do usuario. Ele e obrigatorio.';  COMMENT ON COLUMN tb\_as\_usuario.dt\_nascimento IS  'Este campo sera para armazenar a data de nascimento do usuario. Ele e obrigatorio.';  COMMENT ON COLUMN tb\_as\_usuario.cm\_img\_perfil IS  'Este campo sera para armazenar o caminho da foto de perfil do usuario.';  COMMENT ON COLUMN tb\_as\_usuario.dt\_registro IS  'Este campo sera para armazenar a data de registro do usuario. Ele e obrigatorio.';  ALTER TABLE tb\_as\_usuario ADD CONSTRAINT tb\_as\_usuario\_pk PRIMARY KEY ( id\_usuario );  CREATE TABLE tb\_as\_planta (  id\_planta NUMBER(5) NOT NULL,  nm\_comum VARCHAR2(50) NOT NULL,  nm\_cientifico VARCHAR2(50),  ds\_planta VARCHAR2(100),  cuidados\_gerais VARCHAR2(100)  );  COMMENT ON COLUMN tb\_as\_planta.id\_planta IS  'Este campo sera para armazenar o id da planta. Ele e obrigatorio e chave primaria.';  COMMENT ON COLUMN tb\_as\_planta.nm\_comum IS  'Este campo sera para armazenar o nome comum da planta. Ele e obrigatorio';  COMMENT ON COLUMN tb\_as\_planta.nm\_cientifico IS  'Este campo sera para armazenar o nome cientifico da planta.';  COMMENT ON COLUMN tb\_as\_planta.ds\_planta IS  'Este campo sera para armazenar a descricao da planta.';  COMMENT ON COLUMN tb\_as\_planta.cuidados\_gerais IS  'Este campo sera para armazenar os cuidados gerais da planta.';  ALTER TABLE tb\_as\_planta ADD CONSTRAINT tb\_as\_plantas\_pk PRIMARY KEY ( id\_planta );  CREATE TABLE tb\_as\_status (  id\_status NUMBER(5) NOT NULL,  ds\_status VARCHAR2(20) NOT NULL  );  COMMENT ON COLUMN tb\_as\_status.id\_status IS  'Este campo sera para armazenar o id do status. Ele e obrigatorio e chave primaria.';  COMMENT ON COLUMN tb\_as\_status.ds\_status IS  'Este campo sera para armazenar a descricao do status. Ele e obrigatorio.';  ALTER TABLE tb\_as\_status ADD CONSTRAINT tb\_as\_status\_pk PRIMARY KEY ( id\_status );  ALTER TABLE tb\_as\_analise  ADD CONSTRAINT tb\_as\_analises\_tb\_as\_local\_fk FOREIGN KEY ( id\_local )  REFERENCES tb\_as\_local ( id\_local );  ALTER TABLE tb\_as\_analise  ADD CONSTRAINT tb\_as\_analises\_tb\_as\_plantas\_fk FOREIGN KEY ( id\_planta )  REFERENCES tb\_as\_planta ( id\_planta );  ALTER TABLE tb\_as\_analise  ADD CONSTRAINT tb\_as\_analises\_tb\_as\_status\_fk FOREIGN KEY ( id\_status )  REFERENCES tb\_as\_status ( id\_status );  ALTER TABLE tb\_as\_local  ADD CONSTRAINT tb\_as\_local\_tb\_as\_usuario\_fk FOREIGN KEY ( id\_usuario )  REFERENCES tb\_as\_usuario ( id\_usuario );  ALTER TABLE tb\_as\_usuario  ADD CONSTRAINT tb\_usuario\_telefone\_fk FOREIGN KEY ( id\_telefone )  REFERENCES tb\_as\_telefone ( id\_telefone );  CREATE TABLE tb\_as\_auditoria (  id\_registro NUMBER(5),  nm\_usuario VARCHAR2(30),  dt\_hora DATE,  acao\_realizada VARCHAR2(10),  tabela\_afetada VARCHAR2(50)  );  CREATE TABLE TB\_AS\_ERRO (  ID\_ERRO NUMBER(5) NOT NULL,  CD\_ERRO NUMBER(10) NOT NULL,  NM\_ERRO VARCHAR2(300) NOT NULL,  DT\_REGISTRO DATE NOT NULL,  USUARIO VARCHAR2(100) NOT NULL,  PROCEDIMENTO VARCHAR2(50) NOT NULL  );  CREATE SEQUENCE seq\_id\_analise  START WITH 1  INCREMENT BY 1;  CREATE SEQUENCE seq\_id\_local  START WITH 1  INCREMENT BY 1;  CREATE SEQUENCE seq\_id\_telefone  START WITH 1  INCREMENT BY 1;  CREATE SEQUENCE seq\_id\_usuario  START WITH 1  INCREMENT BY 1;  CREATE SEQUENCE seq\_id\_planta  START WITH 1  INCREMENT BY 1;  CREATE SEQUENCE seq\_id\_status  START WITH 1  INCREMENT BY 1;  CREATE SEQUENCE seq\_id\_registro  START WITH 1  INCREMENT BY 1;  CREATE SEQUENCE seq\_id\_erro  START WITH 1  INCREMENT BY 1; |

Exemplo - DDL do banco

## 6 – Data Manupulation Language – DML (INSERT)

|  |
| --- |
| -- Inserindo dados iniciais  INSERT INTO tb\_as\_telefone (ID\_TELEFONE, NR\_DDI, NR\_DDD, NR\_TELEFONE)  VALUES (seq\_id\_telefone.NEXTVAL, 55, 11, 935684267);  INSERT INTO tb\_as\_telefone (ID\_TELEFONE, NR\_DDI, NR\_DDD, NR\_TELEFONE)  VALUES (seq\_id\_telefone.NEXTVAL, 55, 21, 932689367);  INSERT INTO tb\_as\_telefone (ID\_TELEFONE, NR\_DDI, NR\_DDD, NR\_TELEFONE)  VALUES (seq\_id\_telefone.NEXTVAL, 55, 31, 932974268);  INSERT INTO tb\_as\_telefone (ID\_TELEFONE, NR\_DDI, NR\_DDD, NR\_TELEFONE)  VALUES (seq\_id\_telefone.NEXTVAL, 55, 41, 939286832);  INSERT INTO tb\_as\_telefone (ID\_TELEFONE, NR\_DDI, NR\_DDD, NR\_TELEFONE)  VALUES (seq\_id\_telefone.NEXTVAL, 55, 41, 949382859);  INSERT INTO tb\_as\_usuario (ID\_USUARIO, ID\_TELEFONE, EMAIL, SENHA, NOME, CPF\_CNPJ, GENERO, DT\_NASCIMENTO, DT\_REGISTRO)  VALUES (seq\_id\_usuario.NEXTVAL, 1, 'vinicius@email.com', 'Vincius123', 'Vinicius de Oliveira', '03954810000135', 'M', TO\_DATE('15/05/2003', 'DD-MM-YYYY'), SYSDATE);  INSERT INTO tb\_as\_usuario (ID\_USUARIO, ID\_TELEFONE, EMAIL, SENHA, NOME, CPF\_CNPJ, GENERO, DT\_NASCIMENTO, DT\_REGISTRO)  VALUES (seq\_id\_usuario.NEXTVAL, 2, 'maria@email.com', 'Maria123', 'Maria da Lima Silva', '26942875369', 'F', TO\_DATE('21/03/1930', 'DD-MM-YYYY'), SYSDATE);  INSERT INTO tb\_as\_usuario (ID\_USUARIO, ID\_TELEFONE, EMAIL, SENHA, NOME, CPF\_CNPJ, GENERO, DT\_NASCIMENTO, DT\_REGISTRO)  VALUES (seq\_id\_usuario.NEXTVAL, 3, 'jose@email.com', 'Jose123', 'Jose Carlos Souza', '36256938000123', 'M', TO\_DATE('30/09/1999', 'DD-MM-YYYY'), SYSDATE);  INSERT INTO tb\_as\_usuario (ID\_USUARIO, ID\_TELEFONE, EMAIL, SENHA, NOME, CPF\_CNPJ, GENERO, DT\_NASCIMENTO, DT\_REGISTRO)  VALUES (seq\_id\_usuario.NEXTVAL, 4, 'camila@email.com', 'Camila123', 'Camila da Silva de Souza', '39658423698', 'F', TO\_DATE('29/07/2004', 'DD-MM-YYYY'), SYSDATE);  INSERT INTO tb\_as\_usuario (ID\_USUARIO, ID\_TELEFONE, EMAIL, SENHA, NOME, CPF\_CNPJ, GENERO, DT\_NASCIMENTO, DT\_REGISTRO)  VALUES (seq\_id\_usuario.NEXTVAL, 5, 'ricardo@email.com', 'Ricardo123', 'Ricardo dos Santos Cruz', '03954810000135', 'M', TO\_DATE('01/03/1960', 'DD-MM-YYYY'), SYSDATE);  INSERT INTO tb\_as\_status (ID\_STATUS, DS\_STATUS)  VALUES (seq\_id\_status.NEXTVAL, 'Morta');  INSERT INTO tb\_as\_status (ID\_STATUS, DS\_STATUS)  VALUES (seq\_id\_status.NEXTVAL, 'Em progresso');  INSERT INTO tb\_as\_status (ID\_STATUS, DS\_STATUS)  VALUES (seq\_id\_status.NEXTVAL, 'Em tratamento');  INSERT INTO tb\_as\_status (ID\_STATUS, DS\_STATUS)  VALUES (seq\_id\_status.NEXTVAL, 'Analisando');  INSERT INTO tb\_as\_status (ID\_STATUS, DS\_STATUS)  VALUES (seq\_id\_status.NEXTVAL, 'Curada');  INSERT INTO tb\_as\_planta (ID\_PLANTA, NM\_COMUM, NM\_CIENTIFICO, DS\_PLANTA, CUIDADOS\_GERAIS)  VALUES (seq\_id\_planta.NEXTVAL, 'Rosa', 'Rosa spp.', 'A rosa é uma planta conhecida por suas flores delicadas e perfumadas.', 'A rosa requer rega regular e luz solar direta para florescer bem.');  INSERT INTO tb\_as\_planta (ID\_PLANTA, NM\_COMUM, NM\_CIENTIFICO, DS\_PLANTA, CUIDADOS\_GERAIS)  VALUES (seq\_id\_planta.NEXTVAL, 'Lírio', 'Lilium spp.', 'O lírio é uma planta de flores vistosas e variadas cores.', 'Os lírios preferem solo bem drenado e devem ser regados regularmente.');  INSERT INTO tb\_as\_planta (ID\_PLANTA, NM\_COMUM, NM\_CIENTIFICO, DS\_PLANTA, CUIDADOS\_GERAIS)  VALUES (seq\_id\_planta.NEXTVAL, 'Cacto', 'Cactaceae', 'Os cactos são plantas suculentas conhecidas por sua resistência à seca.', 'Os cactos precisam de pouca água e luz solar direta.');  INSERT INTO tb\_as\_planta (ID\_PLANTA, NM\_COMUM, NM\_CIENTIFICO, DS\_PLANTA, CUIDADOS\_GERAIS)  VALUES (seq\_id\_planta.NEXTVAL, 'Orquídea', 'Orchidaceae', 'As orquídeas são conhecidas por suas flores elegantes e variadas.', 'As orquídeas necessitam de umidade constante e luz indireta.');  INSERT INTO tb\_as\_planta (ID\_PLANTA, NM\_COMUM, NM\_CIENTIFICO, DS\_PLANTA, CUIDADOS\_GERAIS)  VALUES (seq\_id\_planta.NEXTVAL, 'Samambaia', 'Nephrolepis exaltata', 'A samambaia é uma planta que gosta de ambientes úmidos e sombreados.', 'Mantenha o solo da samambaia sempre úmido e evite luz solar direta.');  INSERT INTO tb\_as\_local (ID\_LOCAL, ID\_USUARIO, NM\_PAIS, SG\_ESTADO, NM\_MUNICIPIO, NM\_BAIRRO, NM\_LOGRADOURO, NR\_LOGRADOURO, NR\_CEP, DS\_COMPLEMENTO)  VALUES (seq\_id\_local.NEXTVAL, 1, 'Brasil', 'SP', 'São Paulo', 'Centro', 'Avenida Paulista', '123', '01234567', 'Apto 101');  INSERT INTO tb\_as\_local (ID\_LOCAL, ID\_USUARIO, NM\_PAIS, SG\_ESTADO, NM\_MUNICIPIO, NM\_BAIRRO, NM\_LOGRADOURO, NR\_LOGRADOURO, NR\_CEP, DS\_COMPLEMENTO)  VALUES (seq\_id\_local.NEXTVAL, 2, 'Brasil', 'RJ', 'Rio de Janeiro', 'Copacabana', 'Rua Nossa Senhora de Copacabana', '456', '22334455', 'Casa 2A');  INSERT INTO tb\_as\_local (ID\_LOCAL, ID\_USUARIO, NM\_PAIS, SG\_ESTADO, NM\_MUNICIPIO, NM\_BAIRRO, NM\_LOGRADOURO, NR\_LOGRADOURO, NR\_CEP, DS\_COMPLEMENTO)  VALUES (seq\_id\_local.NEXTVAL, 3, 'Brasil', 'MG', 'Belo Horizonte', 'Savassi', 'Avenida Getúlio Vargas', '789', '30123456', NULL);  INSERT INTO tb\_as\_local (ID\_LOCAL, ID\_USUARIO, NM\_PAIS, SG\_ESTADO, NM\_MUNICIPIO, NM\_BAIRRO, NM\_LOGRADOURO, NR\_LOGRADOURO, NR\_CEP, DS\_COMPLEMENTO)  VALUES (seq\_id\_local.NEXTVAL, 4, 'Brasil', 'RS', 'Porto Alegre', 'Moinhos de Vento', 'Rua Padre Chagas', '101', '90450123', 'Sala 3');  INSERT INTO tb\_as\_local (ID\_LOCAL, ID\_USUARIO, NM\_PAIS, SG\_ESTADO, NM\_MUNICIPIO, NM\_BAIRRO, NM\_LOGRADOURO, NR\_LOGRADOURO, NR\_CEP, DS\_COMPLEMENTO)  VALUES (seq\_id\_local.NEXTVAL, 5, 'Brasil', 'DF', 'Brasília', 'Asa Norte', 'Quadra 102', 'Apto 501', '70000123', 'Bloco B');  INSERT INTO tb\_as\_analise (ID\_ANALISE, ID\_LOCAL, ID\_STATUS, ID\_PLANTA, CM\_IMAGEM, DS\_PROBLEMA, DS\_SOLUCAO, DT\_REGISTRO)  VALUES (seq\_id\_analise.NEXTVAL, 1, ROUND(DBMS\_RANDOM.VALUE(1, 5)), ROUND(DBMS\_RANDOM.VALUE(1, 5)), 'planta1.jpg', 'A planta apresenta folhas murchas.', 'Aumentar a rega e verificar a exposição solar.', SYSDATE);  INSERT INTO tb\_as\_analise (ID\_ANALISE, ID\_LOCAL, ID\_STATUS, ID\_PLANTA, CM\_IMAGEM, DS\_PROBLEMA, DS\_SOLUCAO, DT\_REGISTRO)  VALUES (seq\_id\_analise.NEXTVAL, 2, ROUND(DBMS\_RANDOM.VALUE(1, 5)), ROUND(DBMS\_RANDOM.VALUE(1, 5)), 'planta2.png', 'A planta está com folhas amareladas.', 'Adicionar fertilizante e ajustar a rega.', SYSDATE);  INSERT INTO tb\_as\_analise (ID\_ANALISE, ID\_LOCAL, ID\_STATUS, ID\_PLANTA, CM\_IMAGEM, DS\_PROBLEMA, DS\_SOLUCAO, DT\_REGISTRO)  VALUES (seq\_id\_analise.NEXTVAL, 3, ROUND(DBMS\_RANDOM.VALUE(1, 5)), ROUND(DBMS\_RANDOM.VALUE(1, 5)), 'planta3.jpg', 'A planta possui pragas nas folhas.', 'Usar inseticida e remover as folhas afetadas.', SYSDATE);  INSERT INTO tb\_as\_analise (ID\_ANALISE, ID\_LOCAL, ID\_STATUS, ID\_PLANTA, CM\_IMAGEM, DS\_PROBLEMA, DS\_SOLUCAO, DT\_REGISTRO)  VALUES (seq\_id\_analise.NEXTVAL, 4, ROUND(DBMS\_RANDOM.VALUE(1, 5)), ROUND(DBMS\_RANDOM.VALUE(1, 5)), 'planta4.jpg', 'A planta está com excesso de água.', 'Reduzir a rega e melhorar a drenagem do solo.', SYSDATE);  INSERT INTO tb\_as\_analise (ID\_ANALISE, ID\_LOCAL, ID\_STATUS, ID\_PLANTA, CM\_IMAGEM, DS\_PROBLEMA, DS\_SOLUCAO, DT\_REGISTRO)  VALUES (seq\_id\_analise.NEXTVAL, 5, ROUND(DBMS\_RANDOM.VALUE(1, 5)), ROUND(DBMS\_RANDOM.VALUE(1, 5)), 'planta5.png', 'A planta está com manchas nas folhas.', 'Identificar a causa das manchas e tratá-las.', SYSDATE);  COMMIT; |

Exemplo de Script em SQL para carga inicial do banco de dados no banco

# 7 – Data Manupulation Language – DML (UPDATE / DELETE)

|  |
| --- |
| UPDATE tb\_as\_analise  SET CM\_IMAGEM = 'nova\_imagem\_planta.png'  WHERE ID\_ANALISE = 3;  UPDATE tb\_as\_telefone  SET NR\_TELEFONE = '966309946'  WHERE ID\_TELEFONE = 2;  COMMIT; |
| DELETE FROM tb\_as\_analise  WHERE ID\_ANALISE = 4;  COMMIT; |

Exemplo de Comandos de Atualização e de Deleção

# 8 – Data Query Language – DQL (SELECT)

|  |
| --- |
| select \* from tb\_as\_usuario u  inner join tb\_as\_telefone t on (u.id\_telefone = t.id\_telefone);  SELECT ID\_STATUS, COUNT(\*) FROM tb\_as\_analise GROUP BY ID\_STATUS;  SELECT COUNT(\*)  FROM TB\_AS\_ANALISE a  INNER JOIN TB\_AS\_STATUS s ON (a.id\_status = s.id\_status)  WHERE UPPER(s.DS\_STATUS) = UPPER('ANALISANDO');  SELECT DS\_SOLUCAO  FROM TB\_AS\_ANALISE  WHERE DS\_PROBLEMA LIKE '%' || 'murchas' || '%'; |

Exemplo de Relatório

# 9 – Procedures

9.1 - Procedure para cadastrar um usuario junto com o telefone dele

*CREATE OR REPLACE PROCEDURE PRC\_CADASTRO\_USUARIO (*

*p\_email IN VARCHAR2,*

*p\_senha IN VARCHAR2,*

*p\_nome IN VARCHAR2,*

*p\_cpf\_cnpj IN VARCHAR2,*

*p\_genero IN CHAR,*

*p\_dt\_nascimento IN DATE,*

*p\_cm\_img\_perfil IN VARCHAR2,*

*p\_nr\_ddi IN NUMBER,*

*p\_nr\_ddd IN NUMBER,*

*p\_nr\_telefone IN NUMBER*

*) IS*

*idTelefone NUMBER;*

*v\_codigo\_erro NUMBER;*

*v\_mensagem\_erro VARCHAR2(250);*

*BEGIN*

*IF INSTR(p\_email, '@') = 0 THEN*

*RAISE\_APPLICATION\_ERROR(-20001, 'O email deve conter o caracter @.');*

*END IF;*

*IF LENGTH(p\_cpf\_cnpj) != 11*

*AND LENGTH(p\_cpf\_cnpj) != 14 THEN*

*RAISE\_APPLICATION\_ERROR(-20002, 'O conteudo do CPF/CNPJ deve ter 11 ou 14 caracteres.');*

*END IF;*

*IF p\_genero != 'M'*

*AND p\_genero != 'F' THEN*

*RAISE\_APPLICATION\_ERROR(-20003, 'O valor de genero deve ser M (Masculino) ou F (Feminino).');*

*END IF;*

*IF INSTR(UPPER(p\_cm\_img\_perfil), '.PNG') = 0*

*AND INSTR(UPPER(p\_cm\_img\_perfil), '.JPEG') = 0 THEN*

*RAISE\_APPLICATION\_ERROR(-20004, 'Os formatos de imagem permitos sao: png ou jpeg.');*

*END IF;*

*idTelefone := seq\_id\_telefone.NEXTVAL;*

*INSERT INTO tb\_as\_telefone (ID\_TELEFONE, NR\_DDI, NR\_DDD, NR\_TELEFONE)*

*VALUES (idTelefone, p\_nr\_ddi, p\_nr\_ddd, p\_nr\_telefone);*

*INSERT INTO tb\_as\_usuario (ID\_USUARIO, ID\_TELEFONE, EMAIL, SENHA, NOME, CPF\_CNPJ, GENERO, DT\_NASCIMENTO, CM\_IMG\_PERFIL, DT\_REGISTRO)*

*VALUES (seq\_id\_usuario.NEXTVAL, idTelefone, p\_email, p\_senha, p\_nome, p\_cpf\_cnpj, p\_genero, TO\_DATE(p\_dt\_nascimento, 'DD-MM-YYYY'), p\_cm\_img\_perfil, SYSDATE);*

*EXCEPTION*

*WHEN DUP\_VAL\_ON\_INDEX THEN*

*DBMS\_OUTPUT.PUT\_LINE('Erro: Valor duplicado em coluna unica.');*

*v\_codigo\_erro := SQLCODE;*

*v\_mensagem\_erro := SQLERRM;*

*INSERT INTO TB\_AS\_ERRO (ID\_ERRO, CD\_ERRO, NM\_ERRO, DT\_REGISTRO, USUARIO, PROCEDIMENTO)*

*VALUES (seq\_id\_erro.NEXTVAL, v\_codigo\_erro, v\_mensagem\_erro, SYSDATE, USER, 'PRC\_CADASTRO\_USUARIO');*

*WHEN NO\_DATA\_FOUND THEN*

*DBMS\_OUTPUT.PUT\_LINE('Erro: Chave estrangeira nao encontrada.');*

*v\_codigo\_erro := SQLCODE;*

*v\_mensagem\_erro := SQLERRM;*

*INSERT INTO TB\_AS\_ERRO (ID\_ERRO, CD\_ERRO, NM\_ERRO, DT\_REGISTRO, USUARIO, PROCEDIMENTO)*

*VALUES (seq\_id\_erro.NEXTVAL, v\_codigo\_erro, v\_mensagem\_erro, SYSDATE, USER, 'PRC\_CADASTRO\_USUARIO');*

*WHEN VALUE\_ERROR THEN*

*DBMS\_OUTPUT.PUT\_LINE('Erro: Violou a restricao de tamanho.');*

*v\_codigo\_erro := SQLCODE;*

*v\_mensagem\_erro := SQLERRM;*

*INSERT INTO TB\_AS\_ERRO (ID\_ERRO, CD\_ERRO, NM\_ERRO, DT\_REGISTRO, USUARIO, PROCEDIMENTO)*

*VALUES (seq\_id\_erro.NEXTVAL, v\_codigo\_erro, v\_mensagem\_erro, SYSDATE, USER, 'PRC\_CADASTRO\_USUARIO');*

*WHEN OTHERS THEN*

*DBMS\_OUTPUT.PUT\_LINE('Ocorreu um erro: ' || SQLERRM);*

*v\_codigo\_erro := SQLCODE;*

*v\_mensagem\_erro := SQLERRM;*

*INSERT INTO TB\_AS\_ERRO (ID\_ERRO, CD\_ERRO, NM\_ERRO, DT\_REGISTRO, USUARIO, PROCEDIMENTO)*

*VALUES (seq\_id\_erro.NEXTVAL, v\_codigo\_erro, v\_mensagem\_erro, SYSDATE, USER, 'PRC\_CADASTRO\_USUARIO');*

*END;*

Interface gráfica do usuário, Aplicativo, Word

Descrição gerada automaticamente

9.2 - Procedure para cadastrar um local

*CREATE OR REPLACE PROCEDURE PRC\_CADASTRO\_LOCAL (*

*p\_id\_usuario IN NUMBER,*

*p\_nm\_pais IN VARCHAR2,*

*p\_sg\_estado IN VARCHAR2,*

*p\_nm\_municipio IN VARCHAR2,*

*p\_nm\_bairro IN VARCHAR2,*

*p\_nm\_logradouro IN VARCHAR2,*

*p\_nr\_logradouro IN VARCHAR2,*

*p\_nr\_cep IN VARCHAR2,*

*p\_ds\_complemento IN VARCHAR2*

*) IS*

*v\_verificaUsuario NUMBER;*

*v\_codigo\_erro NUMBER;*

*v\_mensagem\_erro VARCHAR2(250);*

*BEGIN*

*SELECT COUNT(\*)*

*INTO v\_verificaUsuario*

*FROM tb\_as\_usuario*

*WHERE ID\_USUARIO = p\_id\_usuario;*

*IF v\_verificaUsuario = 0 THEN*

*RAISE\_APPLICATION\_ERROR(-20005, 'Usuario nao encontrado.');*

*END IF;*

*INSERT INTO tb\_as\_local (ID\_LOCAL, ID\_USUARIO, NM\_PAIS, SG\_ESTADO, NM\_MUNICIPIO, NM\_BAIRRO, NM\_LOGRADOURO, NR\_LOGRADOURO, NR\_CEP, DS\_COMPLEMENTO)*

*VALUES (seq\_id\_local.NEXTVAL, p\_id\_usuario, p\_nm\_pais, p\_sg\_estado, p\_nm\_municipio, p\_nm\_bairro, p\_nm\_logradouro, p\_nr\_logradouro, p\_nr\_cep, p\_ds\_complemento);*

*EXCEPTION*

*WHEN DUP\_VAL\_ON\_INDEX THEN*

*DBMS\_OUTPUT.PUT\_LINE('Erro: Valor duplicado em coluna unica.');*

*v\_codigo\_erro := SQLCODE;*

*v\_mensagem\_erro := SQLERRM;*

*INSERT INTO TB\_AS\_ERRO (ID\_ERRO, CD\_ERRO, NM\_ERRO, DT\_REGISTRO, USUARIO, PROCEDIMENTO)*

*VALUES (seq\_id\_erro.NEXTVAL, v\_codigo\_erro, v\_mensagem\_erro, SYSDATE, USER, 'PRC\_CADASTRO\_LOCAL');*

*WHEN NO\_DATA\_FOUND THEN*

*DBMS\_OUTPUT.PUT\_LINE('Erro: Chave estrangeira nao encontrada.');*

*v\_codigo\_erro := SQLCODE;*

*v\_mensagem\_erro := SQLERRM;*

*INSERT INTO TB\_AS\_ERRO (ID\_ERRO, CD\_ERRO, NM\_ERRO, DT\_REGISTRO, USUARIO, PROCEDIMENTO)*

*VALUES (seq\_id\_erro.NEXTVAL, v\_codigo\_erro, v\_mensagem\_erro, SYSDATE, USER, 'PRC\_CADASTRO\_LOCAL');*

*WHEN VALUE\_ERROR THEN*

*DBMS\_OUTPUT.PUT\_LINE('Erro: Violou a restricao de tamanho.');*

*v\_codigo\_erro := SQLCODE;*

*v\_mensagem\_erro := SQLERRM;*

*INSERT INTO TB\_AS\_ERRO (ID\_ERRO, CD\_ERRO, NM\_ERRO, DT\_REGISTRO, USUARIO, PROCEDIMENTO)*

*VALUES (seq\_id\_erro.NEXTVAL, v\_codigo\_erro, v\_mensagem\_erro, SYSDATE, USER, 'PRC\_CADASTRO\_LOCAL');*

*WHEN OTHERS THEN*

*DBMS\_OUTPUT.PUT\_LINE('Ocorreu um erro: ' || SQLERRM);*

*v\_codigo\_erro := SQLCODE;*

*v\_mensagem\_erro := SQLERRM;*

*INSERT INTO TB\_AS\_ERRO (ID\_ERRO, CD\_ERRO, NM\_ERRO, DT\_REGISTRO, USUARIO, PROCEDIMENTO)*

*VALUES (seq\_id\_erro.NEXTVAL, v\_codigo\_erro, v\_mensagem\_erro, SYSDATE, USER, 'PRC\_CADASTRO\_LOCAL');*

*END;*

Tabela

Descrição gerada automaticamente

9.3 - Procedure para cadastrar uma analise

*CREATE OR REPLACE PROCEDURE PRC\_CADASTRO\_ANALISE (*

*p\_id\_local IN NUMBER,*

*p\_id\_status IN NUMBER,*

*p\_id\_planta IN NUMBER,*

*p\_cm\_imagem IN VARCHAR2,*

*p\_ds\_problema IN VARCHAR2,*

*p\_ds\_solucao IN VARCHAR2*

*) IS*

*v\_verificaLocal NUMBER;*

*v\_verificaStatus NUMBER;*

*v\_verificaPlanta NUMBER;*

*v\_codigo\_erro NUMBER;*

*v\_mensagem\_erro VARCHAR2(250);*

*BEGIN*

*SELECT COUNT(\*)*

*INTO v\_verificaLocal*

*FROM tb\_as\_local*

*WHERE ID\_LOCAL = p\_id\_local;*

*IF v\_verificaLocal = 0 THEN*

*RAISE\_APPLICATION\_ERROR(-20006, 'Local nao encontrado.');*

*END IF;*

*SELECT COUNT(\*)*

*INTO v\_verificaStatus*

*FROM tb\_as\_status*

*WHERE ID\_STATUS = p\_id\_status;*

*IF v\_verificaStatus = 0 THEN*

*RAISE\_APPLICATION\_ERROR(-20007, 'Status nao encontrado.');*

*END IF;*

*SELECT COUNT(\*)*

*INTO v\_verificaPlanta*

*FROM tb\_as\_planta*

*WHERE ID\_PLANTA = p\_id\_planta;*

*IF v\_verificaPlanta = 0 THEN*

*RAISE\_APPLICATION\_ERROR(-20008, 'Planta nao encontrado.');*

*END IF;*

*IF INSTR(UPPER(p\_cm\_imagem), '.PNG') = 0*

*AND INSTR(UPPER(p\_cm\_imagem), '.JPEG') = 0 THEN*

*RAISE\_APPLICATION\_ERROR(-20009, 'Os formatos de imagem permitos s�o: png ou jpeg.');*

*END IF;*

*INSERT INTO tb\_as\_analise (ID\_ANALISE, ID\_LOCAL, ID\_STATUS, ID\_PLANTA, CM\_IMAGEM, DS\_PROBLEMA, DS\_SOLUCAO, DT\_REGISTRO)*

*VALUES (seq\_id\_analise.NEXTVAL, p\_id\_local, p\_id\_status, p\_id\_planta, p\_cm\_imagem, p\_ds\_problema,*

*p\_ds\_solucao, SYSDATE);*

*EXCEPTION*

*WHEN DUP\_VAL\_ON\_INDEX THEN*

*DBMS\_OUTPUT.PUT\_LINE('Erro: Valor duplicado em coluna unica.');*

*v\_codigo\_erro := SQLCODE;*

*v\_mensagem\_erro := SQLERRM;*

*INSERT INTO TB\_AS\_ERRO (ID\_ERRO, CD\_ERRO, NM\_ERRO, DT\_REGISTRO, USUARIO, PROCEDIMENTO)*

*VALUES (seq\_id\_erro.NEXTVAL, v\_codigo\_erro, v\_mensagem\_erro, SYSDATE, USER, 'PRC\_CADASTRO\_ANALISE');*

*WHEN NO\_DATA\_FOUND THEN*

*DBMS\_OUTPUT.PUT\_LINE('Erro: Chave estrangeira nao encontrada.');*

*v\_codigo\_erro := SQLCODE;*

*v\_mensagem\_erro := SQLERRM;*

*INSERT INTO TB\_AS\_ERRO (ID\_ERRO, CD\_ERRO, NM\_ERRO, DT\_REGISTRO, USUARIO, PROCEDIMENTO)*

*VALUES (seq\_id\_erro.NEXTVAL, v\_codigo\_erro, v\_mensagem\_erro, SYSDATE, USER, 'PRC\_CADASTRO\_ANALISE');*

*WHEN VALUE\_ERROR THEN*

*DBMS\_OUTPUT.PUT\_LINE('Erro: Violou a restricao de tamanho.');*

*v\_codigo\_erro := SQLCODE;*

*v\_mensagem\_erro := SQLERRM;*

*INSERT INTO TB\_AS\_ERRO (ID\_ERRO, CD\_ERRO, NM\_ERRO, DT\_REGISTRO, USUARIO, PROCEDIMENTO)*

*VALUES (seq\_id\_erro.NEXTVAL, v\_codigo\_erro, v\_mensagem\_erro, SYSDATE, USER, 'PRC\_CADASTRO\_ANALISE');*

*WHEN OTHERS THEN*

*DBMS\_OUTPUT.PUT\_LINE('Ocorreu um erro: ' || SQLERRM);*

*v\_codigo\_erro := SQLCODE;*

*v\_mensagem\_erro := SQLERRM;*

*INSERT INTO TB\_AS\_ERRO (ID\_ERRO, CD\_ERRO, NM\_ERRO, DT\_REGISTRO, USUARIO, PROCEDIMENTO)*

*VALUES (seq\_id\_erro.NEXTVAL, v\_codigo\_erro, v\_mensagem\_erro, SYSDATE, USER, 'PRC\_CADASTRO\_ANALISE');*

*END;*

Interface gráfica do usuário

Descrição gerada automaticamente com confiança média

9.4 – Exemplo de exception

Interface gráfica do usuário, Aplicativo, Word

Descrição gerada automaticamente

# 10 – Functions

10.1 – Função para retornar o porcentual de analises com o valor do parametro passado referente ao total de analises.

*CREATE OR REPLACE FUNCTION FNC\_PORCENT\_ANALISES(*

*p\_status IN VARCHAR*

*)*

*RETURN NUMBER*

*IS*

*v\_total NUMBER;*

*v\_porStatus NUMBER;*

*v\_resultado NUMBER;*

*v\_codigo\_erro NUMBER;*

*v\_mensagem\_erro VARCHAR2(250);*

*BEGIN*

*IF UPPER(p\_status) IN ('MORTA', 'EM PROGRESSO', 'EM TRATAMENTO', 'ANALISANDO', 'CURADA') THEN*

*SELECT COUNT(\*)*

*INTO v\_total*

*FROM TB\_AS\_ANALISE;*

*SELECT COUNT(\*)*

*INTO v\_porStatus*

*FROM TB\_AS\_ANALISE a*

*INNER JOIN TB\_AS\_STATUS s ON (a.id\_status = s.id\_status)*

*WHERE UPPER(s.DS\_STATUS) = UPPER(p\_status);*

*v\_resultado := (v\_porStatus / v\_total) \* 100;*

*RETURN v\_resultado;*

*ELSE*

*RAISE\_APPLICATION\_ERROR(-20001, 'O valor deve ser um valor entre esses: Morta, Em progresso, Em tratamento, Analisando e Curada');*

*end if;*

*EXCEPTION*

*WHEN NO\_DATA\_FOUND THEN*

*DBMS\_OUTPUT.PUT\_LINE('Nenhuma solucao encontrada para o problema: ' || p\_status);*

*v\_codigo\_erro := SQLCODE;*

*v\_mensagem\_erro := SQLERRM;*

*INSERT INTO TB\_AS\_ERRO (ID\_ERRO, CD\_ERRO, NM\_ERRO, DT\_REGISTRO, USUARIO, PROCEDIMENTO)*

*VALUES (seq\_id\_erro.NEXTVAL, v\_codigo\_erro, v\_mensagem\_erro, SYSDATE, USER, 'FNC\_PORCENT\_ANALISES');*

*RETURN 0;*

*WHEN OTHERS THEN*

*DBMS\_OUTPUT.PUT\_LINE('Ocorreu um erro: ' || SQLERRM);*

*v\_codigo\_erro := SQLCODE;*

*v\_mensagem\_erro := SQLERRM;*

*INSERT INTO TB\_AS\_ERRO (ID\_ERRO, CD\_ERRO, NM\_ERRO, DT\_REGISTRO, USUARIO, PROCEDIMENTO)*

*VALUES (seq\_id\_erro.NEXTVAL, v\_codigo\_erro, v\_mensagem\_erro, SYSDATE, USER, 'FNC\_PORCENT\_ANALISES');*

*RETURN NULL;*

*END;*

Interface gráfica do usuário, Texto

Descrição gerada automaticamente com confiança média

10.2 – Funcao para retornar uma solucao para o problema passado como parametro com base em palavra ou pequenas frases chaves.

*CREATE OR REPLACE FUNCTION FNC\_RETORNA\_SOLUCAO(*

*p\_problema IN VARCHAR2*

*)*

*RETURN VARCHAR2*

*IS*

*v\_retorno VARCHAR2(500);*

*v\_codigo\_erro NUMBER;*

*v\_mensagem\_erro VARCHAR2(250);*

*BEGIN*

*SELECT DS\_SOLUCAO*

*INTO v\_retorno*

*FROM TB\_AS\_ANALISE*

*WHERE UPPER(DS\_PROBLEMA) LIKE '%' || UPPER(p\_problema) || '%'*

*AND DT\_REGISTRO = (SELECT MAX(DT\_REGISTRO) FROM TB\_AS\_ANALISE)*

*AND ROWNUM = 1;*

*RETURN v\_retorno;*

*EXCEPTION*

*WHEN NO\_DATA\_FOUND THEN*

*DBMS\_OUTPUT.PUT\_LINE('Nenhuma solucao encontrada para o problema: ' || p\_problema);*

*v\_codigo\_erro := SQLCODE;*

*v\_mensagem\_erro := SQLERRM;*

*INSERT INTO TB\_AS\_ERRO (ID\_ERRO, CD\_ERRO, NM\_ERRO, DT\_REGISTRO, USUARIO, PROCEDIMENTO)*

*VALUES (seq\_id\_erro.NEXTVAL, v\_codigo\_erro, v\_mensagem\_erro, SYSDATE, USER, 'FNC\_RETORNA\_SOLUCAO');*

*RETURN NULL;*

*WHEN OTHERS THEN*

*DBMS\_OUTPUT.PUT\_LINE('Ocorreu um erro: ' || SQLERRM);*

*v\_codigo\_erro := SQLCODE;*

*v\_mensagem\_erro := SQLERRM;*

*INSERT INTO TB\_AS\_ERRO (ID\_ERRO, CD\_ERRO, NM\_ERRO, DT\_REGISTRO, USUARIO, PROCEDIMENTO)*

*VALUES (seq\_id\_erro.NEXTVAL, v\_codigo\_erro, v\_mensagem\_erro, SYSDATE, USER, 'FNC\_RETORNA\_SOLUCAO');*

*RETURN NULL;*

*END;*

10.3 – Exemplo de exception

Texto

Descrição gerada automaticamente com confiança média

Interface gráfica do usuário, Texto, Aplicativo, Email

Descrição gerada automaticamente

# 11 – Triggers (Uma trigger para cada principal tabela do banco)

*SET SERVEROUTPUT ON;*

*-- Uma trigger para as principais tabelas do banco*

*CREATE OR REPLACE TRIGGER TRG\_AUDITORIA\_ANALISE*

*AFTER INSERT OR UPDATE OR DELETE ON tb\_as\_analise*

*FOR EACH ROW*

*DECLARE*

*v\_comando VARCHAR2(50);*

*v\_codigo\_erro NUMBER;*

*v\_mensagem\_erro VARCHAR2(250);*

*BEGIN*

*IF INSERTING THEN*

*v\_comando := 'INSERT';*

*ELSIF UPDATING THEN*

*v\_comando := 'UPDATE';*

*ELSIF DELETING THEN*

*v\_comando := 'DELETE';*

*END IF;*

*INSERT INTO tb\_as\_auditoria (ID\_REGISTRO, NM\_USUARIO, DT\_HORA, ACAO\_REALIZADA, TABELA\_AFETADA)*

*VALUES (seq\_id\_registro.NEXTVAL, USER, SYSDATE, v\_comando, 'TB\_AS\_ANALISE');*

*EXCEPTION*

*WHEN OTHERS THEN*

*DBMS\_OUTPUT.PUT\_LINE('Ocorreu um erro: ' || SQLERRM);*

*v\_codigo\_erro := SQLCODE;*

*v\_mensagem\_erro := SQLERRM;*

*INSERT INTO TB\_AS\_ERRO (ID\_ERRO, CD\_ERRO, NM\_ERRO, DT\_REGISTRO, USUARIO, PROCEDIMENTO)*

*VALUES (seq\_id\_erro.NEXTVAL, v\_codigo\_erro, v\_mensagem\_erro, SYSDATE, USER, 'TRG\_AUDITORIA\_ANALISE');*

*END;*

*CREATE OR REPLACE TRIGGER TRG\_AUDITORIA\_LOCAL*

*AFTER INSERT OR UPDATE OR DELETE ON tb\_as\_local*

*FOR EACH ROW*

*DECLARE*

*v\_comando VARCHAR2(50);*

*v\_codigo\_erro NUMBER;*

*v\_mensagem\_erro VARCHAR2(250);*

*BEGIN*

*IF INSERTING THEN*

*v\_comando := 'INSERT';*

*ELSIF UPDATING THEN*

*v\_comando := 'UPDATE';*

*ELSIF DELETING THEN*

*v\_comando := 'DELETE';*

*END IF;*

*INSERT INTO tb\_as\_auditoria (ID\_REGISTRO, NM\_USUARIO, DT\_HORA, ACAO\_REALIZADA, TABELA\_AFETADA)*

*VALUES (seq\_id\_registro.NEXTVAL, USER, SYSDATE, v\_comando, 'TB\_AS\_ANALISE');*

*EXCEPTION*

*WHEN OTHERS THEN*

*DBMS\_OUTPUT.PUT\_LINE('Ocorreu um erro: ' || SQLERRM);*

*v\_codigo\_erro := SQLCODE;*

*v\_mensagem\_erro := SQLERRM;*

*INSERT INTO TB\_AS\_ERRO (ID\_ERRO, CD\_ERRO, NM\_ERRO, DT\_REGISTRO, USUARIO, PROCEDIMENTO)*

*VALUES (seq\_id\_erro.NEXTVAL, v\_codigo\_erro, v\_mensagem\_erro, SYSDATE, USER, 'TRG\_AUDITORIA\_LOCAL');*

*END;*

*CREATE OR REPLACE TRIGGER TRG\_AUDITORIA\_USUARIO*

*AFTER INSERT OR UPDATE OR DELETE ON tb\_as\_usuario*

*FOR EACH ROW*

*DECLARE*

*v\_comando VARCHAR2(50);*

*v\_codigo\_erro NUMBER;*

*v\_mensagem\_erro VARCHAR2(250);*

*BEGIN*

*IF INSERTING THEN*

*v\_comando := 'INSERT';*

*ELSIF UPDATING THEN*

*v\_comando := 'UPDATE';*

*ELSIF DELETING THEN*

*v\_comando := 'DELETE';*

*END IF;*

*INSERT INTO tb\_as\_auditoria (ID\_REGISTRO, NM\_USUARIO, DT\_HORA, ACAO\_REALIZADA, TABELA\_AFETADA)*

*VALUES (seq\_id\_registro.NEXTVAL, USER, SYSDATE, v\_comando, 'TB\_AS\_ANALISE');*

*EXCEPTION*

*WHEN OTHERS THEN*

*DBMS\_OUTPUT.PUT\_LINE('Ocorreu um erro: ' || SQLERRM);*

*v\_codigo\_erro := SQLCODE;*

*v\_mensagem\_erro := SQLERRM;*

*INSERT INTO TB\_AS\_ERRO (ID\_ERRO, CD\_ERRO, NM\_ERRO, DT\_REGISTRO, USUARIO, PROCEDIMENTO)*

*VALUES (seq\_id\_erro.NEXTVAL, v\_codigo\_erro, v\_mensagem\_erro, SYSDATE, USER, 'TRG\_AUDITORIA\_LOCAL');*

*END;*