9.4 LISTA DE CODIGOS DAS FUNÇÕES, ASSERÇOES E TRIGGERS

* TRIGGER NA TABELA PESSOA

Objetivo: Evitar que seja criada várias contas com o mesmo e-mail

CREATE OR REPLACE FUNCTION insert\_existing\_pessoa() RETURNS trigger AS $$

BEGIN

PERFORM 1 FROM pessoa WHERE email = NEW.email;

IF NOT FOUND THEN

RETURN NEW;

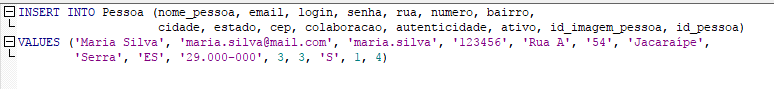
ELSE

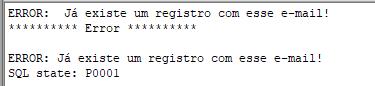
RAISE EXCEPTION 'Já existe uma pessoa com esse e-mail registrada!';

END IF;

END $$ LANGUAGE plpgsql;

CREATE TRIGGER a\_insert\_existing\_pessoa BEFORE INSERT ON pessoa

FOR EACH ROW EXECUTE PROCEDURE insert\_existing\_pessoa();



* TRIGGER NA TABELA LUGAR

CREATE OR REPLACE FUNCTION insert\_existing\_lugar() RETURNS trigger AS $$

BEGIN

PERFORM 1 FROM lugar WHERE nome\_lugar = NEW.nome\_lugar;

IF NOT FOUND THEN

RETURN NEW;

ELSE

RAISE EXCEPTION 'Já existe um lugar com esse nome registrado!';

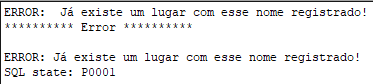
END IF;

END $$ LANGUAGE plpgsql;

CREATE TRIGGER a\_insert\_existing\_lugar BEFORE INSERT ON lugar

FOR EACH ROW EXECUTE PROCEDURE insert\_existing\_lugar();





* TRIGGER CATEGORIA

CREATE OR REPLACE FUNCTION insert\_existing\_categoria() RETURNS trigger AS $$

BEGIN

PERFORM 1 FROM categoria WHERE nome\_categoria = NEW.nome\_categoria;

IF NOT FOUND THEN

RETURN NEW;

ELSE

RAISE EXCEPTION 'Já existe uma categoria com esse nome registrado!';

END IF;

END $$ LANGUAGE plpgsql;

CREATE TRIGGER a\_insert\_existing\_categoria BEFORE INSERT ON categoria

FOR EACH ROW EXECUTE PROCEDURE insert\_existing\_categoria();

