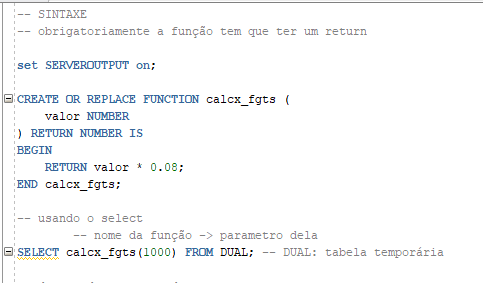
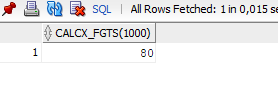
BD

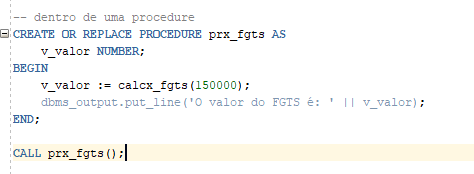
***FUNÇÕES***



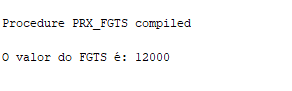
Usando o select:



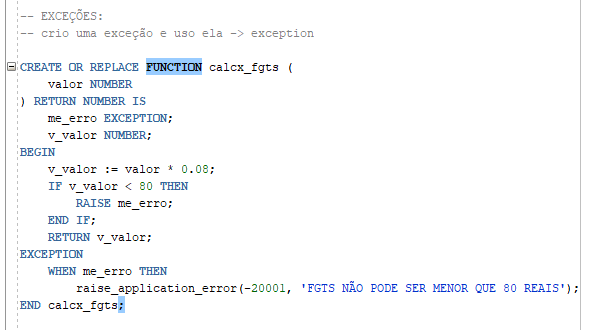
Usando o procedure



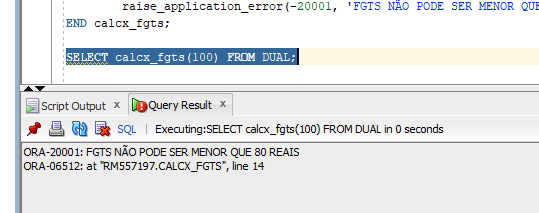
Retorno:



EXCEÇÕES:



Retorno, forçando o erro:



EXERCÍCIOS

1)

CREATE OR REPLACE PROCEDURE prc\_insere\_produto (

p\_cod\_produto NUMBER,

p\_nom\_produto VARCHAR2,

p\_cod\_barra NUMBER,

p\_sta\_ativo VARCHAR2,

p\_dat\_cadastro DATE,

p\_dat\_cancelamento DATE

) AS

BEGIN

-- Se o nome do produto tem mais de 3 caracteres

IF LENGTH(p\_nom\_produto) > 3 THEN

-- Verifica se o nome não contém números

IF NOT REGEXP\_LIKE(p\_nom\_produto, '[0-9]') THEN

-- Inserindo o produto na tabela

INSERT INTO produto (cod\_produto, nom\_produto, cod\_barra, sta\_ativo, dat\_cadastro, dat\_cancelamento)

VALUES (p\_cod\_produto, p\_nom\_produto, p\_cod\_barra, p\_sta\_ativo, p\_dat\_cadastro, p\_dat\_cancelamento);

COMMIT;

dbms\_output.put\_line('Produto inserido com sucesso!');

ELSE

-- Caso o nome contenha números, lançar erro

RAISE\_APPLICATION\_ERROR(-20001, 'Erro: O nome do produto não pode ter números.');

END IF;

ELSE

-- Caso o nome tenha 3 ou menos caracteres, lançar erro

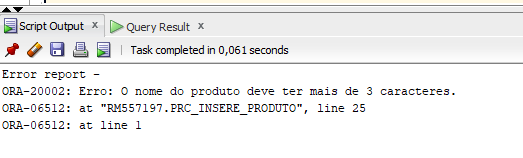
RAISE\_APPLICATION\_ERROR(-20002, 'Erro: O nome do produto deve ter mais de 3 caracteres.');

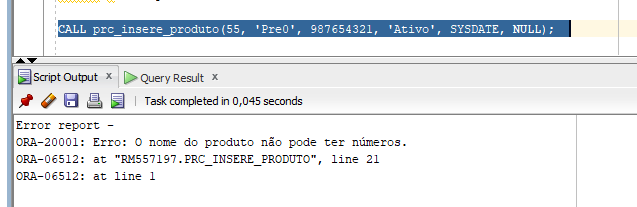
END IF;

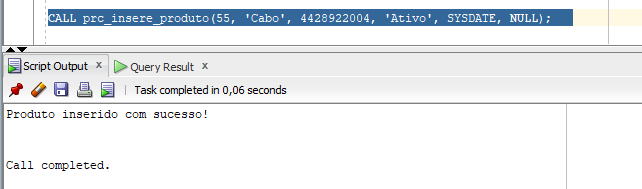
END;

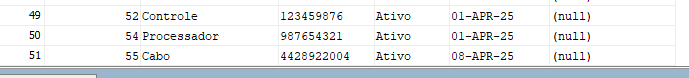
SELECT \* FROM produto;

CALL prc\_insere\_produto(55, 'Pre', 987654321, 'Ativo', SYSDATE, NULL);









2)

CREATE OR REPLACE PROCEDURE prc\_insere\_cliente (

p\_cod\_cliente NUMBER,

p\_nom\_cliente VARCHAR2,

p\_razao\_social VARCHAR2,

p\_tip\_pessoa VARCHAR2,

p\_num\_cpf\_cnpj VARCHAR2,

p\_dat\_cadastro DATE,

p\_dat\_cancelamento DATE,

p\_sta\_ativo VARCHAR2

) AS

BEGIN

-- Se o nome do cliente tem mais de 3 caracteres

IF LENGTH(p\_nom\_cliente) > 3 THEN

-- Se o nome do cliente não contém números usando REGEXP\_LIKE

IF NOT REGEXP\_LIKE(p\_nom\_cliente, '[0-9]') THEN

-- Inserindo o cliente na tabela

INSERT INTO cliente (cod\_cliente, nom\_cliente, des\_razao\_social, tip\_pessoa, num\_cpf\_cnpj, dat\_cadastro, dat\_cancelamento, sta\_ativo)

VALUES (p\_cod\_cliente, p\_nom\_cliente, p\_razao\_social, p\_tip\_pessoa, p\_num\_cpf\_cnpj, p\_dat\_cadastro, p\_dat\_cancelamento, p\_sta\_ativo);

COMMIT; -- Realiza o commit

dbms\_output.put\_line('Cliente inserido com sucesso!');

ELSE

-- Caso o nome contenha números, lançar erro

RAISE\_APPLICATION\_ERROR(-20001, 'Erro: O nome do cliente não pode ter números.');

END IF;

ELSE

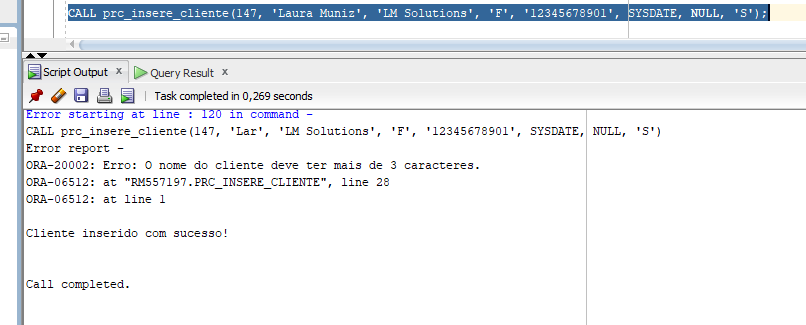
-- Caso o nome tenha 3 ou menos caracteres, lançar erro

RAISE\_APPLICATION\_ERROR(-20002, 'Erro: O nome do cliente deve ter mais de 3 caracteres.');

END IF;

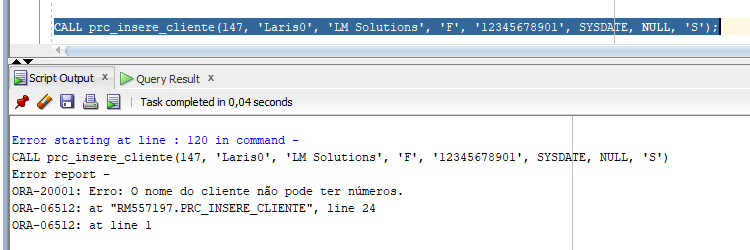
END;

CALL prc\_insere\_cliente(147, 'Laura Muniz', 'LM Solutions', 'F', '12345678901', SYSDATE, NULL, 'S');

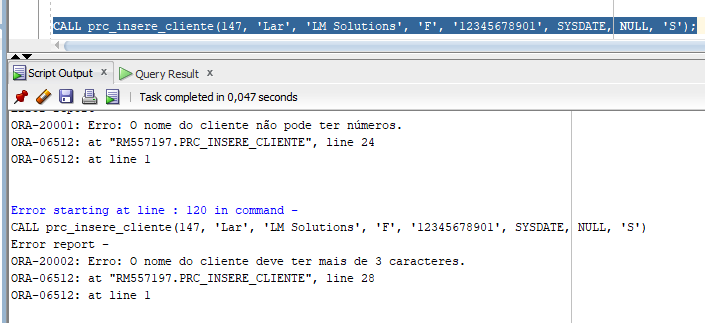
****

****

Primeiro erro:



Segundo erro:



3 )

CREATE OR REPLACE FUNCTION fun\_valida\_nome (

nome VARCHAR2

) RETURN VARCHAR2 IS

ME\_ERRO EXCEPTION;

BEGIN

-- Verifica se o nome tem mais de 3 caracteres

IF LENGTH(nome) <= 3 THEN

RAISE\_APPLICATION\_ERROR(-20001, 'Nome deve ter mais de 3 caracteres');

END IF;

-- Verifica se o nome contém números

IF REGEXP\_LIKE(nome, '[0-9]') THEN

RAISE\_APPLICATION\_ERROR(-20002, 'Nome não pode conter números');

END IF;

-- Se passou nas duas validações, retorna o nome como válido

RETURN nome;

EXCEPTION

WHEN OTHERS THEN

-- Captura qualquer erro inesperado

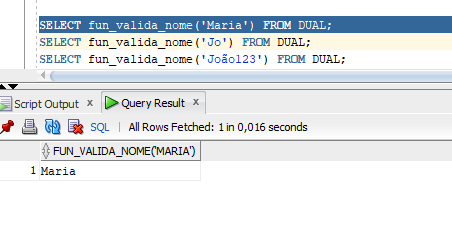
RAISE me\_erro;

END fun\_valida\_nome;

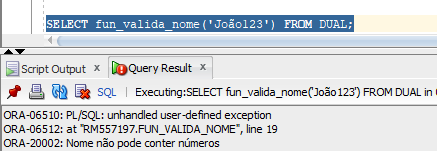
SELECT fun\_valida\_nome('Maria') FROM DUAL;

SELECT fun\_valida\_nome('Jo') FROM DUAL; -- o nome tem menos que 3 caracteres

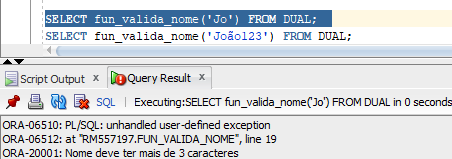
SELECT fun\_valida\_nome('João123') FROM DUAL; -- o nome tem numeros



Primeiro erro:



Segundo erro:



4)

-- exercicio 1 pra chamar a função do ex 3:

CREATE OR REPLACE PROCEDURE prc\_insere\_produto (

p\_cod\_produto NUMBER,

p\_nom\_produto VARCHAR2,

p\_cod\_barra NUMBER,

p\_sta\_ativo VARCHAR2,

p\_dat\_cadastro DATE,

p\_dat\_cancelamento DATE

) AS

v\_resultado VARCHAR2(100);

BEGIN

-- Chama a função de validação de nome do produto

v\_resultado := fun\_valida\_nome(p\_nom\_produto);

-- Se o nome do produto for válido, faz a inserção

IF v\_resultado = p\_nom\_produto THEN

-- Inserindo o produto na tabela

INSERT INTO produto (cod\_produto, nom\_produto, cod\_barra, sta\_ativo, dat\_cadastro, dat\_cancelamento)

VALUES (p\_cod\_produto, p\_nom\_produto, p\_cod\_barra, p\_sta\_ativo, p\_dat\_cadastro, p\_dat\_cancelamento);

COMMIT;

dbms\_output.put\_line('Produto inserido com sucesso!');

ELSE

-- Caso o nome do produto não seja válido, a função já gerará erro

NULL; -- Não é necessário realizar mais ações, pois a função já gera erro se inválido

END IF;

EXCEPTION

WHEN OTHERS THEN

-- Captura qualquer erro inesperado

RAISE\_APPLICATION\_ERROR(-20003, 'Erro ao inserir o produto' );

END;

CALL prc\_insere\_produto(56, 'Cabos', 44289220045, 'Ativo', SYSDATE, NULL);

SELECT \* FROM produto;

-- exercicio 2 para chamar a função do ex 3:

CREATE OR REPLACE PROCEDURE prc\_insere\_cliente (

p\_cod\_cliente NUMBER,

p\_nom\_cliente VARCHAR2,

p\_razao\_social VARCHAR2,

p\_tip\_pessoa VARCHAR2,

p\_num\_cpf\_cnpj VARCHAR2,

p\_dat\_cadastro DATE,

p\_dat\_cancelamento DATE,

p\_sta\_ativo VARCHAR2

) AS

v\_resultado VARCHAR2(100);

BEGIN

-- Chama a função de validação de nome do cliente

v\_resultado := fun\_valida\_nome(p\_nom\_cliente);

-- Se o nome do cliente for válido, faz a inserção

IF v\_resultado = p\_nom\_cliente THEN

-- Inserindo o cliente na tabela

INSERT INTO cliente (cod\_cliente, nom\_cliente, des\_razao\_social, tip\_pessoa, num\_cpf\_cnpj, dat\_cadastro, dat\_cancelamento, sta\_ativo)

VALUES (p\_cod\_cliente, p\_nom\_cliente, p\_razao\_social, p\_tip\_pessoa, p\_num\_cpf\_cnpj, p\_dat\_cadastro, p\_dat\_cancelamento, p\_sta\_ativo);

COMMIT; -- Realiza o commit

dbms\_output.put\_line('Cliente inserido com sucesso!');

ELSE

-- Caso o nome do cliente não seja válido, a função já gerará erro

NULL; -- Não é necessário realizar mais ações, pois a função já gera erro se inválido

END IF;

EXCEPTION

WHEN OTHERS THEN

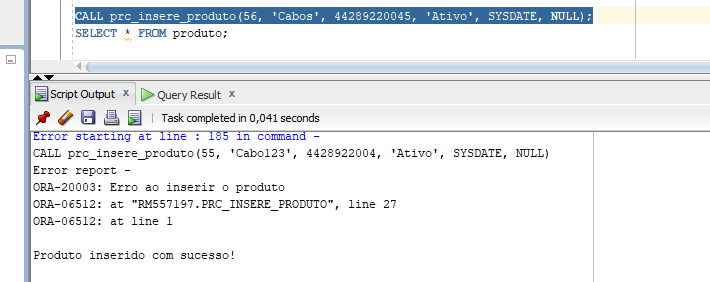
-- Captura qualquer erro inesperado

RAISE\_APPLICATION\_ERROR(-20003, 'Erro ao inserir o cliente ');

END;

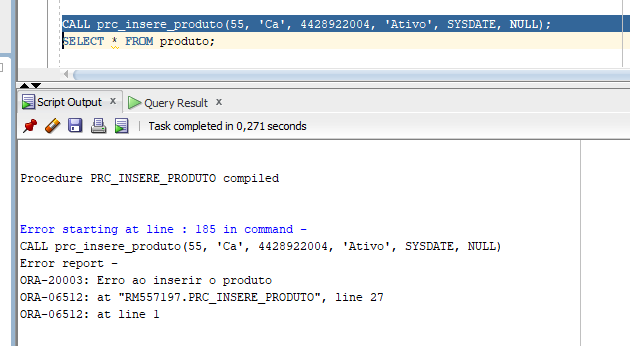
CALL prc\_insere\_cliente(148, 'Bru1234', 'BM Solutions', 'F', '12345678901', SYSDATE, NULL, 'S');

SELECT \* FROM cliente;

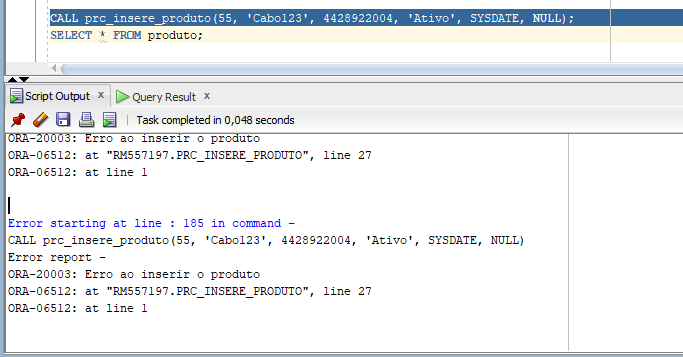


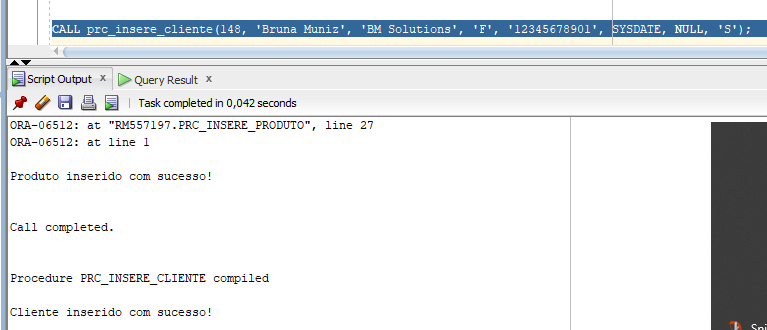


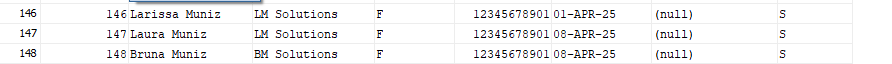
Deu erro porque o nome tem menos que 3 caracteres:



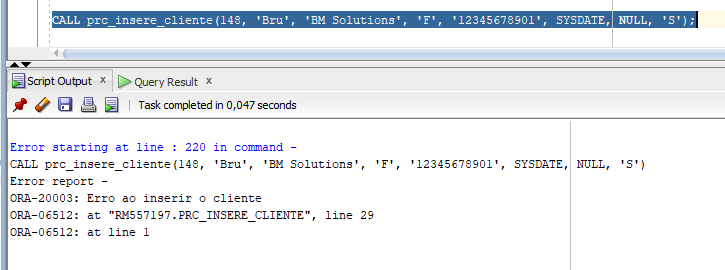
Erro porque o nome contem números:







Erro porque o nome tem apenas 3 caracteres:



Erro porque o nome do cliente contem números:

