

UNIVERSIDADE CATÓLICA DO SALVADOR - NOVA PÓS

MBA EM TECNOLOGIAS E APLICAÇÕES DE BUSINESS INTELLIGENCE

Disciplina: Linguagem de Programação SQL – Transact SQL

Professor: Juracy Almeida

Data: 13/10/2018

Equipe: Camila Oliveira e Rosana Cardoso

Atividade da Disciplina – Parte 2 / 2

Carga Tabela LOG_CARGA

- Criar PACOTE PKG_LOG_CARGA

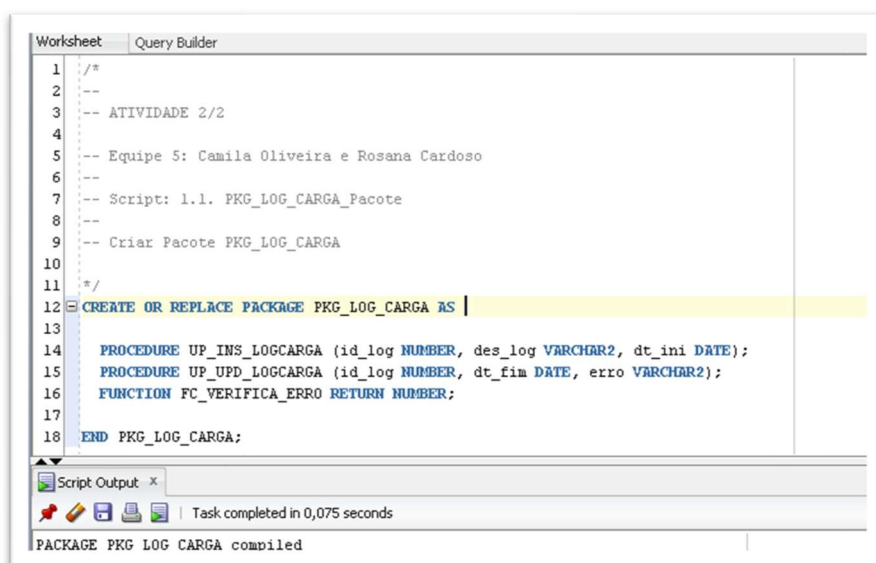
1. Criar procedure UP_INS_LOGCARGA para INSERIR dados na tabela LOG_CARGA. Os dados que serão inseridos irão para os campos, IDE_LOG_CARGA, DES_LOG_CARGA E DTC_INICIO.
2. Criar procedure UP_UPD_LOGCARGA para ATUALIZAR dados na tabela LOG_CARGA. Os campos que serão atualizados serão DTC_FIM e DES_ERRO;
3. Criar FUNÇÃO FC_VERIFICA_ERRO que verifica se na última carga ocorreu algum erro. O retorno da função é 1, se houve erro e 0 se não houve. PESQUISAR pelo registro mais recente da tabela LOG_CARGA e que DES_LOG_CARGA='CARGA DADOS'. Se no registro mais recente, o campo DES_ERRO FOR NULO, retorna 0. Senão houve erro e retorna 1.

Script:

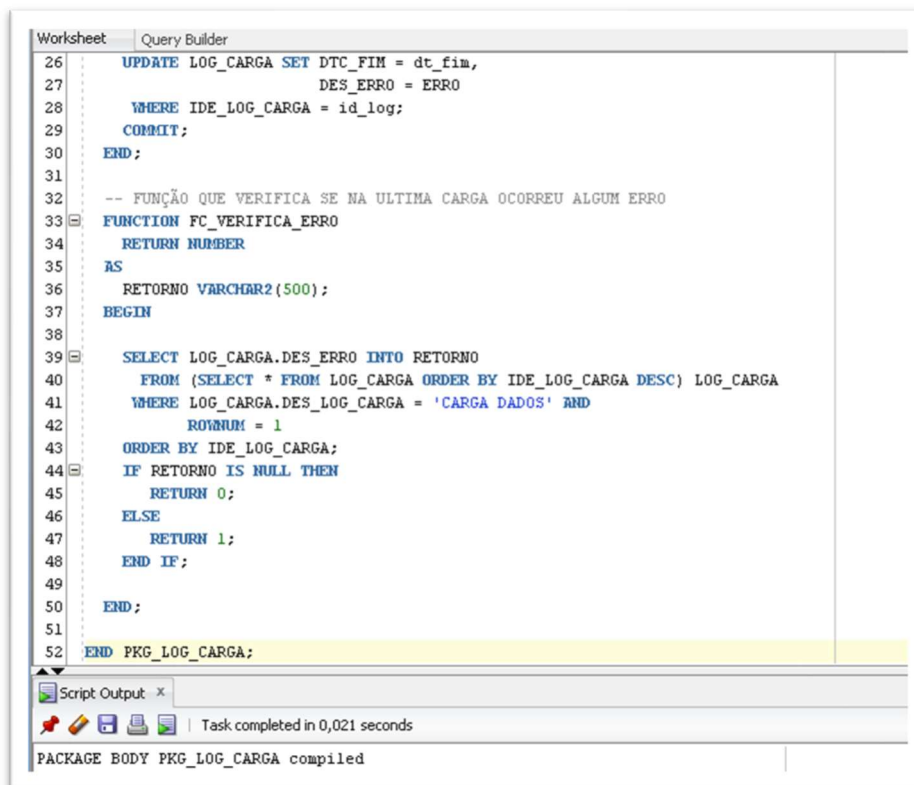
1. PKG_LOG_CARGA_Pacote.sql

2. PKG_LOG_CARGA_Corpo.sql

Validação da execução da carga PKG_LOG_CARGA_Pacote:



Validação da execução da carga do PKG_LOG_CARGA_Corpo:



```
26 UPDATE LOG_CARGA SET DTC_FIM = dt_fim,
27     DES_ERRO = ERRO
28 WHERE IDE_LOG_CARGA = id_log;
29 COMMIT;
30 END;
31
32 -- FUNÇÃO QUE VERIFICA SE NA ULTIMA CARGA OCORREU ALGUM ERRO
33 FUNCTION FC_VERIFICA_ERRO
34 RETURN NUMBER
35 AS
36     RETORNO VARCHAR2(500);
37 BEGIN
38
39     SELECT LOG_CARGA.DES_ERRO INTO RETORNO
40     FROM (SELECT * FROM LOG_CARGA ORDER BY IDE_LOG_CARGA DESC) LOG_CARGA
41     WHERE LOG_CARGA.DES_LOG_CARGA = 'CARGA DADOS' AND
42           ROWNUM = 1
43     ORDER BY IDE_LOG_CARGA;
44 IF RETORNO IS NULL THEN
45     RETURN 0;
46 ELSE
47     RETURN 1;
48 END IF;
49
50 END;
51
52 END PKG_LOG_CARGA;
```

Script Output x

Task completed in 0,021 seconds

PACKAGE BODY PKG_LOG_CARGA compiled

Carga Tabela TIPO_GASTO

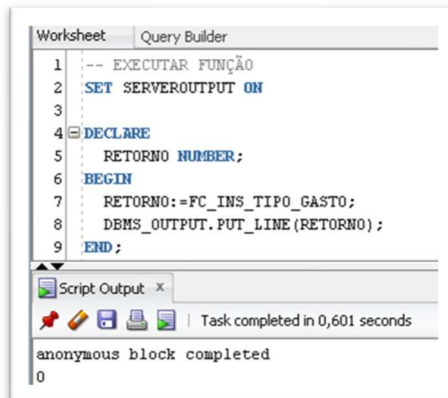
- Criar uma FUNÇÃO chamada FC_INS_TIPO_GASTO
 1. Carregar a tabela com os TIPO de Gastos existentes na tabela transparencia_gastos;
 2. Utilizar a tabela transparencia_gastos como fonte de dados para Inserir dados na tabela tipo_gasto;
 3. Registrar a carga na LOG_CARGA utilizando a PACKAGE e a SEQUENCE LOGCARGA_SEQ para inserção;
 4. A função deve retornar 0 se rodar com SUCESSO. Adicionar um tratamento de ERRO, no caso de algum erro, Retorne 1;
 5. Se executar tudo com sucesso armazenar apenas a DTC_FIM da LOG_CARGA. Se houver ERRO, registrar a DTC_FIM e o DES_ERRO, a MENSAGEM DE ERRO PADRÃO ORACLE deverá ser armazenada.

Script:

3. FC_INS_TIPO_GASTO.sql

Teste do código criado e validação da informação

1. execução da função:



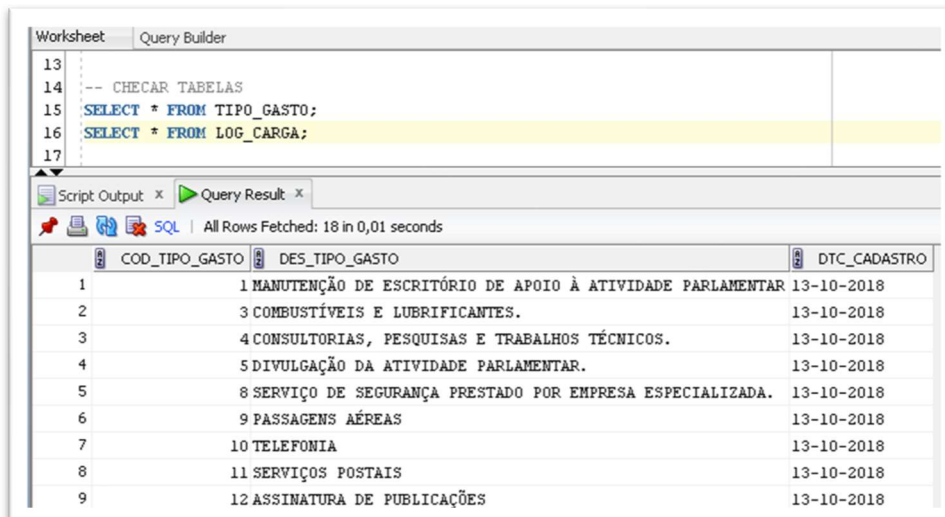
```
1  -- EXECUTAR FUNÇÃO
2  SET SERVEROUTPUT ON
3
4  DECLARE
5      RETORNO NUMBER;
6  BEGIN
7      RETORNO:=FC_INS_TIPO_GASTO;
8      DBMS_OUTPUT.PUT_LINE(RETORNO);
9  END;
```

Script Output x | Task completed in 0,601 seconds

anonymous block completed

0

2. Validação da carga da tabela tipo_gasto:

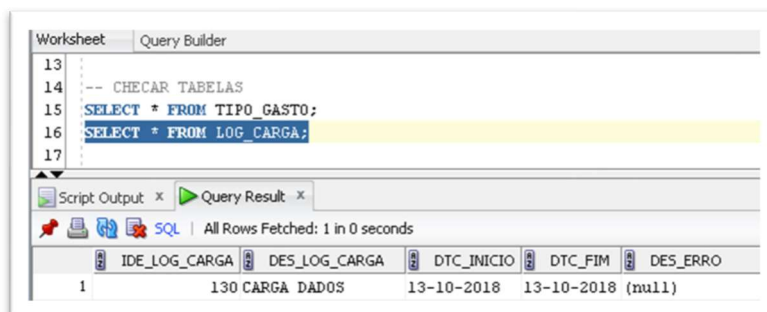


```
13
14 -- CHECAR TABELAS
15 SELECT * FROM TIPO_GASTO;
16 SELECT * FROM LOG_CARGA;
```

Script Output x | Query Result x | All Rows Fetched: 18 in 0,01 seconds

	COD_TIPO_GASTO	DES_TIPO_GASTO	DTC_CADASTRO
1	1	MANUTENÇÃO DE ESCRITÓRIO DE APOIO À ATIVIDADE PARLAMENTAR	13-10-2018
2	3	COMBUSTÍVEIS E LUBRIFICANTES.	13-10-2018
3	4	CONSULTORIAS, PESQUISAS E TRABALHOS TÉCNICOS.	13-10-2018
4	5	DIVULGAÇÃO DA ATIVIDADE PARLAMENTAR.	13-10-2018
5	8	SERVIÇO DE SEGURANÇA PRESTADO POR EMPRESA ESPECIALIZADA.	13-10-2018
6	9	PASSAGENS AÉREAS	13-10-2018
7	10	TELEFONIA	13-10-2018
8	11	SERVIÇOS POSTAIS	13-10-2018
9	12	ASSINATURA DE PUBLICAÇÕES	13-10-2018

3. Validação da carga da tabela log_carga:



```
13
14 -- CHECAR TABELAS
15 SELECT * FROM TIPO_GASTO;
16 SELECT * FROM LOG_CARGA;
```

Script Output x | Query Result x | All Rows Fetched: 1 in 0 seconds

	IDE_LOG_CARGA	DES_LOG_CARGA	DTC_INICIO	DTC_FIM	DES_ERRO
1	130	CARGA DADOS	13-10-2018	13-10-2018	(null)

Script utilizado para validação: FC_INS_TIPO_GASTO_Testes.sql

Carga Tabela PARLAMENTAR

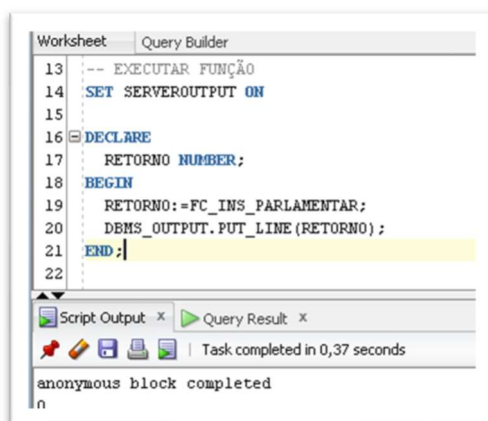
- Criar uma FUNÇÃO chamada FC_INS_PARLAMENTAR
 1. Utilizar Cursor para ler a tabela transparencia_gasto;
 2. Inserir dados na tabela_parlamentar;
 3. Registrar a carga na LOG_CARGA utilizando a PACKAGE e a SEQUENCE LOGCARGA_SEQ para inserção;
 4. A função deve retornar 0 se rodar com SUCESSO. Adicionar um tratamento de ERRO, no caso de algum erro, Retorne 1;
 5. Se executar tudo com sucesso armazenar apenas a DTC_FIM da LOG_CARGA. Se houver ERRO, registrar a DTC_FIM e o DES_ERRO, a MENSAGEM DE ERRO PADRÃO ORACLE deverá ser armazenada.

Script:

4. FC_INS_PARLAMENTAR.sql

Teste do código criado e validação da informação

1. execução da função:



2. Validação da carga da tabela tabela_parlamentar:

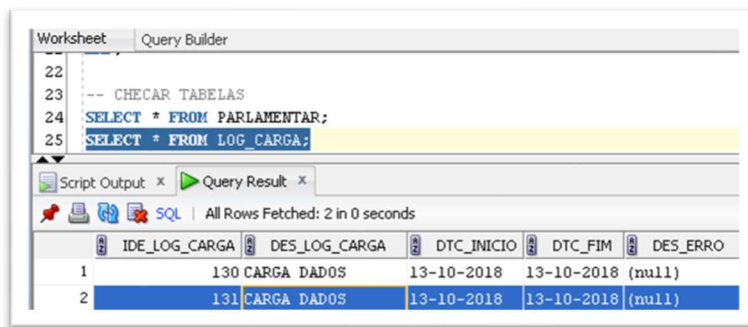
```
-- CHECAR TABELAS
SELECT * FROM PARLAMENTAR;
SELECT * FROM LOG_CARGA;
```

Script Output x Query Result x

SQL | Fetched 50 rows in 0,06 seconds

	IDE_PARLAMENTAR	DES_PARLAMENTAR	TIP_PARLAMENTAR	SGL_PARTIDO	SGL_ESTADO	DTC_CADASTRO
1	2934	MARCELO ÁLVARO ANTÔNIO	DEPUTADO	PSL	MG	13-10-2018
2	2945	TADEU ALENCAR	DEPUTADO	PSB	PE	13-10-2018
3	2964	MARINALDO ROSENDO	DEPUTADO	PP	PE	13-10-2018
4	2967	LAUDIVIO CARVALHO	DEPUTADO	PODE	MG	13-10-2018
5	2990	CAIO NARCIO	DEPUTADO	PSDB	MG	13-10-2018

3. Validação da carga da tabela log_carga:



The screenshot shows the SQL Developer interface with a query result for the log_carga table. The query is: `SELECT * FROM LOG_CARGA;`. The result is displayed in a table with 5 columns: IDE_LOG_CARGA, DES_LOG_CARGA, DTC_INICIO, DTC_FIM, and DES_ERRO. There are 2 rows of data.

IDE_LOG_CARGA	DES_LOG_CARGA	DTC_INICIO	DTC_FIM	DES_ERRO
1	130 CARGA DADOS	13-10-2018	13-10-2018	(null)
2	131 CARGA DADOS	13-10-2018	13-10-2018	(null)

Script utilizado para validação: FC_INS_PARLAMENTAR_Testes.sql

Carga Tabela GASTO_PARLAMENTAR

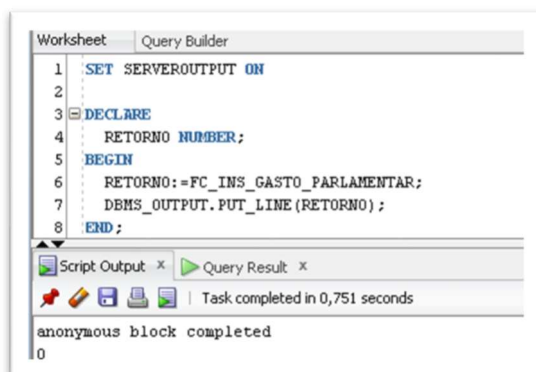
- Rotina para carregar os gastos acumulados do mês de cada parlamentar;
 - Criar uma FUNÇÃO chamada FC_INS_GASTO_PARLAMENTAR
1. Utilizar Cursor para ler a tabela , selecionando apenas 1 dos ESTADOS da lista;
 2. Inserir dados na tabela gasto_parlamentar;
Registrar a carga na LOG_CARGA utilizando a PACKAGE e a SEQUENCE LOGCARGA_SEQ para inserção;
 3. A função deve retornar 0 se rodar com SUCESSO. Adicionar um tratamento de ERRO, no caso de algum erro, Retorne 1;
 4. Se executar tudo com sucesso armazenar apenas a DTC_FIM da LOG_CARGA. Se houver ERRO, registrar a DTC_FIM e o DES_ERRO, a MENSAGEM DE ERRO PADRÃO ORACLE deverá ser armazenada.

Script:

5. FC_INS_GASTO_PARLAMENTAR.sql

Teste do código criado e validação da informação

1. execução da função:



The screenshot shows the SQL Developer interface with a script being executed. The script is: `SET SERVEROUTPUT ON; DECLARE RETORNO NUMBER; BEGIN RETORNO:=FC_INS_GASTO_PARLAMENTAR; DBMS_OUTPUT.PUT_LINE(RETORNO); END;`. The result is displayed in the Script Output window: `anonymous block completed` and `0`.

```
1 SET SERVEROUTPUT ON
2
3 DECLARE
4   RETORNO NUMBER;
5 BEGIN
6   RETORNO:=FC_INS_GASTO_PARLAMENTAR;
7   DBMS_OUTPUT.PUT_LINE(RETORNO);
8 END;
```

Script Output: anonymous block completed
0

2. Validação da carga da tabela gasto_parlamentar:

Worksheet

Query Builder

-- CHECAR TABELAS

SELECT * FROM LOG_CARGA;

SELECT * FROM GASTO_PARLAMENTAR;

Script Output x

Query Result x

SQL | Fetched 50 rows in 0,03 seconds

	COD_TIPO_GASTO	IDE_PARLAMENTAR	NUM_ANO	NUM_MES	VAL_GASTO_MES	VAL_ESTORNO_MES	DTC_CADASTRO
1	999	3077	2018	6	4379.45		0 13-10-2018
2	999	3077	2018	7	8825.96		0 13-10-2018
3	1	3118	2018	4	696.5		0 13-10-2018
4	3	3118	2018	1	3922.34		0 13-10-2018
5	3	3118	2018	2	3311.51		0 13-10-2018
6	3	3118	2018	3	3870.51		0 13-10-2018
7	3	3118	2018	4	5519.43		0 13-10-2018

3. Validação da carga da tabela log_carga:

Worksheet

Query Builder

```

10  -- CHECAR TABELAS
11  SELECT * FROM LOG_CARGA;
12  SELECT * FROM GASTO_PARLAMENTAR;

```

Script Output

Query Result

SQL | All Rows Fetched: 2 in 0,01 seconds

	IDE_LOG_CARGA	DES_LOG_CARGA	DTC_INICIO	DTC_FIM	DES_ERRO
1	134	CARGA DADOS	13-10-2018	13-10-2018	(null)
2	135	CARGA DADOS	13-10-2018	13-10-2018	(null)

Script utilizado para validação: FC_INS_PARLAMENTAR_Testes.sql

Carga Tabela UPDATE_EXECUTA_CARGA

- A procedure terá o parâmetro P_ORDEM do tipo NUMÉRICO;
- Armazenar o retorno da função FC_VERIFICA_ERRO na variável RET_ERRO;
- Registrar na tabela LOG_CARGA o processo GERAL da execução da CARGA e atribuir DES_LOG_CARGA='CARGA DADOS';
- Utilizar SEQUENCE LOGCARGA_SEQ para carregar o IDE_LOG_CARGA;
- Testar se a variável RET_ERRO tem o retorno diferente de 0. Caso afirmativo, realizar um DELETE nas tabelas de negócios abaixo. Esse passo também deverá ser registrado na tabela LOG_CARGA:
 1. TIPO_GASTO
 2. PARLAMENTAR
 3. GASTO_PARLAMENTAR
- Próximo passo, programar a execução da carga das tabelas de negócios na seguinte ORDEM
 1. Se o parâmetro P_ORDEM igual a 1
 - EXECUTAR A FUNÇÃO INS_TIPO_GASTO
 - EXECUTAR A FUNÇÃO PARLAMENTAR
 - EXECUTAR A FUNÇÃO GASTO_PARLAMENTAR
 2. Se o parâmetro P_ORDEM igual a 2
 - EXECUTAR A FUNÇÃO GASTO_PARLAMENTAR
 - EXECUTAR A FUNÇÃO TIPO_GASTO
 - EXECUTAR A FUNÇÃO PARLAMENTAR

- Ao final de todo processo, registrar o FINAL DA CARGA do processo GERAL na tabela LOG_CARGA. Se algum retorno da carga ocorreu algum erro, registrar em DES_ERRO do PROCESSO GERAL a mensagem 'OCORREU ERRO';
- Fazer uso das rotinas da Package PKG_LOG_CARGA;

Script:

6. UP_EXECUTA_CARGA.sql

Validação da execução da UPDATE_EXECUTA_CARGA:

The screenshot shows the SQL Developer interface with a 'Query Builder' window. The SQL script is as follows:

```

1  /*
2  --
3  -- ATIVIDADE 2/2
4  --
5  -- Equipe 5: Camila Oliveira e Rosana Cardoso
6  --
7  -- Script: 5. UP_EXECUTA_CARGA
8  --
9  -- Criar Procedure UP_EXECUTA_CARGA
10 --
11 */
12
13 CREATE OR REPLACE PROCEDURE UP_EXECUTA_CARGA (P_ORDEM NUMBER) AS
14     RET_ERRO NUMBER;
15     RET_ERRO_INT NUMBER;
16     RETORNO NUMBER;
17     REGISTRO NUMBER;
18     PROCESSO_GERAL NUMBER;
19     MENSAGEM VARCHAR2(50);
20     ERRO VARCHAR2(500);
21

```

The 'Script Output' window at the bottom shows the message: 'PROCEDURE UP_EXECUTA_CARGA compiled'.

PASSOS PARA VALIDAÇÃO DAS ROTINAS

Após construir todas as ROTINAS siga os PASSOS abaixo:

1. EXECUTE A FUNÇÃO FC_INS_TIPO_GASTO ATRAVÉS DE UM SELECT E CONSULTE SE CORREGOU A TABELA E A O LOG_CARGA
 - Execução do bloco da função

The screenshot shows the SQL Developer interface with a 'Query Builder' window. The SQL script is as follows:

```

13 -- 1. Executar a função FC_INS_TIPO_GASTO
14 DECLARE
15     RETORNO NUMBER;
16 BEGIN
17     RETORNO := FC_INS_TIPO_GASTO;
18 END;
19
20 SELECT * FROM TIPO_GASTO;
21 SELECT * FROM LOG_CARGA;
22

```

The 'Script Output' window at the bottom shows the message: 'anonymous block completed'.

A execução da função através da **SELECT** apresentou erro informando que o Oracle não pode executar operação DML em uma **SELECT**. Efetuamos a adequação do código retirando a execução através da **SELECT**.

```

1
2 DECLARE
3     RETORNO NUMBER;
4 BEGIN
5     SELECT FC_INS_TIPO_GASTO INTO RETORNO FROM DUAL;
6 END;

```

Script Output x | Task completed in 0,361 seconds

Error starting at line 2 in command:

```

DECLARE
  RETORNO NUMBER;
BEGIN
  SELECT FC_INS_TIPO_GASTO INTO RETORNO FROM DUAL;
END;

```

Error report:

```

ORA-14551: cannot perform a DML operation inside a query
ORA-06512: at "PRATICA.PKG_LOG_CARGA", line 14
ORA-06512: at "PRATICA.FC_INS_TIPO_GASTO", line 33
ORA-14551: cannot perform a DML operation inside a query
ORA-06512: at line 4
14551. 00000 - "cannot perform a DML operation inside a query "
*Cause:      DML operation like insert, update, delete or select-for-update
              cannot be performed inside a query or under a PDML slave.
*Action:     Ensure that the offending DML operation is not performed or
              use an autonomous transaction to perform the DML operation within
              the query or PDML slave.

```

- Evidência da carga da tabela TIPO_GASTO

```

13 -- 1. Executar a função FC_INS_TIPO_GASTO
14 DECLARE
15     RETORNO NUMBER;
16 BEGIN
17     RETORNO := FC_INS_TIPO_GASTO;
18 END;
19
20 SELECT * FROM TIPO_GASTO;
21 SELECT * FROM LOG_CARGA;
22

```

Script Output x | Query Result x | All Rows Fetched: 18 in 0,04 seconds

	COD_TIPO_GASTO	DES_TIPO_GASTO	DTC_CADASTRO
1		1 MANUTENÇÃO DE ESCRITÓRIO DE APOIO À ATIVIDADE PARLAMENTAR	13-10-2018
2		3 COMBUSTÍVEIS E LUBRIFICANTES.	13-10-2018
3		4 CONSULTORIAS, PESQUISAS E TRABALHOS TÉCNICOS.	13-10-2018
4		5 DIVULGAÇÃO DA ATIVIDADE PARLAMENTAR.	13-10-2018
5		8 SERVIÇO DE SEGURANÇA PRESTADO POR EMPRESA ESPECIALIZADA.	13-10-2018
6		9 PASSAGENS AÉREAS	13-10-2018

- Evidência da carga da tabela LOG_CARGA

The screenshot shows a SQL Developer window with a 'Query Builder' tab. The SQL editor contains a PL/SQL block:

```

12 */
13 -- 1. Executar a função FC_INS_TIPO_GASTO
14 DECLARE
15     RETORNO NUMBER;
16 BEGIN
17     RETORNO := FC_INS_TIPO_GASTO;
18 END;
19
20 SELECT * FROM TIPO_GASTO;
21 SELECT * FROM LOG_CARGA;
  
```

Below the editor, the 'Script Output' tab shows the execution results. The status bar indicates 'All Rows Fetched: 1 in 0 seconds'. The results table has the following columns: IDE_LOG_CARGA, DES_LOG_CARGA, DTC_INICIO, DTC_FIM, and DES_ERRO. The first row shows the value '1' for IDE_LOG_CARGA, '143 CARGA DADOS' for DES_LOG_CARGA, and '13-10-2018' for both DTC_INICIO and DTC_FIM, with DES_ERRO being null.

IDE_LOG_CARGA	DES_LOG_CARGA	DTC_INICIO	DTC_FIM	DES_ERRO
1	143 CARGA DADOS	13-10-2018	13-10-2018	(null)

- EXECUTAR A PROCEDURE UP_EXECUTA_CARGA passando o parâmetro 2;

The screenshot shows a SQL Developer window with a 'Query Builder' tab. The SQL editor contains the following PL/SQL block:

```

22
23 -- 2. Executar a Procedure UP_EXECUTA_CARGA passando o parametro 2
24 BEGIN
25     UP_EXECUTA_CARGA(2);
26 END;
27
  
```

Below the editor, the 'Script Output' tab shows the execution results. The status bar indicates 'Task completed in 1,352 seconds'. The results section shows 'anonymous block completed'.

- VERIFICAR SE todas as tabelas foram preenchidas;

- Análise da carga da tabela – Tipo_Gasto. Dados apareceram carregados.

The screenshot shows a SQL Developer window with a 'Query Builder' tab. The SQL editor contains the following query:

```

27
28 -- 3. Verificar se todas as tabelas foram preenchidas
29 SELECT * FROM TIPO_GASTO;
30 SELECT * FROM PARLAMENTAR;
31 SELECT * FROM GASTO_PARLAMENTAR;
32
  
```

Below the editor, the 'Script Output' tab shows the execution results. The status bar indicates 'All Rows Fetched: 18 in 0,02 seconds'. The results table has the following columns: COD_TIPO_GASTO, DES_TIPO_GASTO, and DTC_CADASTRO. The first five rows are visible:

COD_TIPO_GASTO	DES_TIPO_GASTO	DTC_CADASTRO
1	1 MANUTENÇÃO DE ESCRITÓRIO DE APOIO À ATIVIDADE PARLAMENTAR	13-10-2018
2	3 COMBUSTÍVEIS E LUBRIFICANTES.	13-10-2018
3	4 CONSULTORIAS, PESQUISAS E TRABALHOS TÉCNICOS.	13-10-2018
4	5 DIVULGAÇÃO DA ATIVIDADE PARLAMENTAR.	13-10-2018
5	8 SERVIÇO DE SEGURANÇA PRESTADO POR EMPRESA ESPECIALIZADA.	13-10-2018

- Análise da carga da tabela – Parlamentar. Dados apareceram carregados.

Worksheet Query Builder

```

27
28 -- 3. Verificar se todas as tabelas foram preenchidas
29 SELECT * FROM TIPO_GASTO;
30 SELECT * FROM PARLAMENTAR;
31 SELECT * FROM GASTO_PARLAMENTAR;
32

```

Script Output x Query Result x

SQL | Fetched 50 rows in 0,08 seconds

	IDE_PARLAMENTAR	DES_PARLAMENTAR	TIP_PARLAMENTAR	SGL_PARTIDO	SGL_ESTADO	DTC_CADASTRO
1	2934	MARCELO ÁLVARO ANTÔNIO	DEPUTADO	PSL	MG	13-10-2018
2	2945	TADEU ALENCAR	DEPUTADO	PSB	PE	13-10-2018
3	2964	MARINALDO ROSENDO	DEPUTADO	PP	PE	13-10-2018
4	2967	LAUDIVIO CARVALHO	DEPUTADO	PODE	MG	13-10-2018
5	2990	CAIO NARCIO	DEPUTADO	PSDB	MG	13-10-2018
6	2991	BETINHO GOMES	DEPUTADO	PSDB	PE	13-10-2018
7	3006	JOÃO FERNANDO COUTINHO	DEPUTADO	PROS	PE	13-10-2018
8	3007	BONFATO RACHECO	DEPUTADO	DEM	MG	13-10-2018

- Análise da carga da tabela – Gasto_Parlamentar. Dados apareceram carregados.

Worksheet Query Builder

```

27
28 -- 3. Verificar se todas as tabelas foram preenchidas
29 SELECT * FROM TIPO_GASTO;
30 SELECT * FROM PARLAMENTAR;
31 SELECT * FROM GASTO_PARLAMENTAR;
32

```

Script Output x Query Result x

SQL | Fetched 50 rows in 0,04 seconds

	COD_TIPO_GASTO	IDE_PARLAMENTAR	NUM_ANO	NUM_MES	VAL_GASTO_MES	VAL_ESTORNO_MES	DTC_CADASTRO
1	999	3077	2018	6	4379.45		0 13-10-2018
2	999	3077	2018	7	8825.96		0 13-10-2018
3	1	3118	2018	4	696.5		0 13-10-2018
4	3	3118	2018	1	3922.34		0 13-10-2018
5	3	3118	2018	2	3311.51		0 13-10-2018
6	3	3118	2018	3	3870.51		0 13-10-2018
7	3	3118	2018	4	5519.43		0 13-10-2018

4. Verificar se a tabela LOG_CARGA registrou corretamente;

Worksheet Query Builder

```

33 -- 4. Verificar Tabela LOG_CARGA
34 SELECT * FROM LOG_CARGA;
35

```

Script Output x Query Result x

SQL | All Rows Fetched: 6 in 0 seconds

	IDE_LOG_CARGA	DES_LOG_CARGA	DTC_INICIO	DTC_FIM	DES_ERRO
1	144	CARGA DADOS	13-10-2018	13-10-2018	OCORREU ERRO
2	145	CARGA DADOS	13-10-2018	13-10-2018	(null)
3	143	CARGA DADOS	13-10-2018	13-10-2018	(null)
4	146	CARGA DADOS	13-10-2018	13-10-2018	ORA-00001: unique constraint (PRATICA.PK_TIPO_GASTO) violated
5	147	CARGA DADOS	13-10-2018	13-10-2018	(null)
6	148	CARGA DADOS	13-10-2018	13-10-2018	OCORREU ERRO

Os dados da tabela foram carregados corretamente, conforme esperado, foi registrado erro na carga da tabela TIPO_GASTO em decorrência de já ter sido carregada previamente, na execução da função FC_INS_TIPO_GASTO.

5. EXECUTAR NOVAMENTE a PROCEDURE UP_EXECUTA_CARGA com o parâmetro 1.

```

35
36 -- 5. Executar a Procedure UP_EXECUTA_CARGA passando o parametro 1
37 BEGIN
38   UP_EXECUTA_CARGA(1);
39 END;

```

Script Output x Query Result x

Task completed in 4,006 seconds

anonymous block completed

6. VERIFICAR O QUE OCORRE E SE a tabela LOG_CARGA registrou corretamente.

- Análise da carga da tabela – Tipo_Gasto. Dados apareceram carregados.

```

40
41 -- 6. Verificar se todas as tabelas foram preenchidas
42 SELECT * FROM TIPO_GASTO;
43 SELECT * FROM PARLAMENTAR;
44 SELECT * FROM GASTO_PARLAMENTAR;
45 SELECT * FROM LOG_CARGA;
46

```

Script Output x Query Result x

All Rows Fetched: 18 in 0,04 seconds

	COD_TIPO_GASTO	DES_TIPO_GASTO	DTC_CADASTRO
1		1 MANUTENÇÃO DE ESCRITÓRIO DE APOIO À ATIVIDADE PARLAMENTAR	13-10-2018
2		3 COMBUSTÍVEIS E LUBRIFICANTES.	13-10-2018
3		4 CONSULTORIAS, PESQUISAS E TRABALHOS TÉCNICOS.	13-10-2018
4		5 DIVULGAÇÃO DA ATIVIDADE PARLAMENTAR.	13-10-2018
5		8 SERVIÇO DE SEGURANÇA PRESTADO POR EMPRESA ESPECIALIZADA.	13-10-2018
6		9 PASSAGENS AÉREAS	13-10-2018
7		10 TELEFONIA	13-10-2018

- Análise da carga da tabela – Parlamentar. Dados apareceram carregados.

```

40
41 -- 6. Verificar se todas as tabelas foram preenchidas
42 SELECT * FROM TIPO_GASTO;
43 SELECT * FROM PARLAMENTAR;
44 SELECT * FROM GASTO_PARLAMENTAR;
45 SELECT * FROM LOG_CARGA;
46

```

Script Output x Query Result x

Fetched 50 rows in 0,032 seconds

	IDE_PARLAMENTAR	DES_PARLAMENTAR	TIP_PARLAMENTAR	SGL_PARTIDO	SGL_ESTADO	DTC_CADASTRO
1	2934	MARCELO ÁLVARO ANTÔNIO	DEPUTADO	PSL	MG	13-10-2018
2	2945	TADEU ALENCAR	DEPUTADO	PSB	PE	13-10-2018
3	2964	MARINALDO ROSENDO	DEPUTADO	PP	PE	13-10-2018
4	2967	LAUDIVIO CARVALHO	DEPUTADO	PODE	MG	13-10-2018
5	2990	CAIO NARCIO	DEPUTADO	PSDB	MG	13-10-2018
6	2991	BETINHO GOMES	DEPUTADO	PSDB	PE	13-10-2018

- Análise da carga da tabela – Gasto_Parlamentar. Dados apareceram carregados.

Worksheet Query Builder

```

40
41 -- 6. Verificar se todas as tabelas foram preenchidas
42 SELECT * FROM TIPO_GASTO;
43 SELECT * FROM PARLAMENTAR;
44 SELECT * FROM GASTO_PARLAMENTAR;
45 SELECT * FROM LOG_CARGA;
46

```

Script Output x Query Result x

SQL | Fetched 50 rows in 0,05 seconds

	COD_TIPO_GASTO	IDE_PARLAMENTAR	NUM_ANO	NUM_MES	VAL_GASTO_MES	VAL_ESTORNO_MES	DTC_CADASTRO
1	999	3077	2018	6	4379.45		013-10-2018
2	999	3077	2018	7	8825.96		013-10-2018
3	1	3118	2018	4	696.5		013-10-2018
4	3	3118	2018	1	3922.34		013-10-2018
5	3	3118	2018	2	3311.51		013-10-2018
6	3	3118	2018	3	3870.51		013-10-2018
7	3	3118	2018	4	5519.43		013-10-2018

- Análise da carga da tabela – Log_Carga.

Worksheet Query Builder

```

40
41 -- 6. Verificar se todas as tabelas foram preenchidas
42 SELECT * FROM TIPO_GASTO;
43 SELECT * FROM PARLAMENTAR;
44 SELECT * FROM GASTO_PARLAMENTAR;
45 SELECT * FROM LOG_CARGA;
46

```

Script Output x Query Result x

SQL | All Rows Fetched: 14 in 0,01 seconds

	IDE_LOG_CARGA	DES_LOG_CARGA	DTC_INICIO	DTC_FIM	DES_ERRO
1	144	CARGA DADOS	13-10-2018	13-10-2018	OCORREU ERRO
2	145	CARGA DADOS	13-10-2018	13-10-2018	(null)
3	143	CARGA DADOS	13-10-2018	13-10-2018	(null)
4	146	CARGA DADOS	13-10-2018	13-10-2018	ORA-00001: unique constraint (PRATICA.PK_TIPO_GASTO) violated
5	147	CARGA DADOS	13-10-2018	13-10-2018	(null)
6	148	CARGA DADOS	13-10-2018	13-10-2018	OCORREU ERRO
7	149	CARGA DADOS	13-10-2018	13-10-2018	(null)
8	150	DELETA DADO TI...	13-10-2018	13-10-2018	(null)
9	151	DELETA DADO PA...	13-10-2018	13-10-2018	(null)
10	152	DELETA DADO GA...	13-10-2018	13-10-2018	(null)
11	153	CARGA DADOS	13-10-2018	13-10-2018	(null)
12	154	CARGA DADOS	13-10-2018	13-10-2018	(null)
13	155	CARGA DADOS	13-10-2018	13-10-2018	(null)
14	156	CARGA DADOS	13-10-2018	13-10-2018	(null)

Os dados da tabela foram carregados corretamente, sem erros (registro a partir de 149)

Script:

7. ROTINA_EXECUÇÃO.sql