

Consolidando o seu conhecimento

Chegou a hora de você pôr em prática o que foi visto na aula. Para isso, execute os passos listados abaixo.

- 1) Crie uma área de edição de comandos PL/SQL, com o usuário **user_dev**, e crie o cabeçalho do pacote de exceções, digitando e executando o comando abaixo:

```
CREATE OR REPLACE PACKAGE EXCEPTION_PAC
IS
    e_null exception;
    pragma exception_init (e_null, -1400);
    e_fk exception;
    pragma exception_init (e_fk, -2291);
END;
```

- 2) Use este pacote no outro pacote que manipula a tabela de clientes. Para isso, clique sobre o corpo do pacote **Cliente_PAC**. Na rotina **INCLUIR_CLIENTE**, onde as exceções são tratadas, antes da exceção **OTHERS**, inclua:

```
WHEN EXCEPTION_PAC.e_null THEN
    raise_application_error (-20012, 'Preenchimento de campo obrigatório');
```

- 3) Repita esta mesma edição na *procedure* **INCLUIR_CLIENTE**.

- 4) Clique na opção **Compilar** do editor do pacote.

- 5) Crie uma área de edição, agora associada ao usuário **user_app**.

- 6) Execute o comando abaixo, forçando a inclusão de um valor nulo para o identificador do cliente:

```
EXECUTE APP_INCLUIR_CLIENTE (NULL, 'INCLUIDO POR USER_APP PARA TESTAR VALOR NULO', '23456', 1, 10000)
```

- 7) Será possível observar o seguinte erro tratado:

```
Erro a partir da linha : 2 no comando -
BEGIN APP_INCLUIR_CLIENTE (NULL, 'INCLUIDO POR USER_APP PARA TESTAR VALOR NULO', '23456', 1, 10000); END;
Relatório de erros -
ORA-20012: Preenchimento de campo obrigatório
ORA-06512: em "USER_DEV.CLIENTE_PAC", line 74
ORA-06512: em "USER_APP.APP_INCLUIR_CLIENTE", line 9
ORA-06512: em line 1
```

- 8) Crie uma nova área de edição associada ao usuário **user_app**. Crie uma *procedure* que irá editar dados da tabela de segmentos, com o código abaixo:

```
CREATE OR REPLACE PROCEDURE INCLUIR_SEGMENTO (
    p_ID IN SEGMERCADO.ID%type,
    p_DESCRICAO IN SEGMERCADO.DESCRICAO%type
)
IS
BEGIN
```

```

  INSERT INTO SEGMercado (ID, DESCRIÇÃO) VALUES (p_ID, p_DESCRIÇÃO);
END;

```

9) Ao executar o comando de criação, haverá um erro. Isso ocorre pois o usuário **user_app** não acessa a tabela `Segmercado`.

10) Em outra área de edição para o usuário **user_dev**, inclua e execute os comandos abaixo, para dar privilégios de acesso:

```

GRANT SELECT, INSERT ON SEGMercado TO user_app;
CREATE PUBLIC SYNONYM SEGMercado FOR user_dev.SEGMercado;
GRANT EXECUTE ON EXCEPTION_PAC TO PUBLIC;

```

11) Agora é possível voltar à área de edição do usuário **user_app** e executar, com sucesso, a criação da *procedure* **Incluir_Segmento**.

12) Execute a *procedure* que acabou de ser criada, dessa vez forçando a inclusão de um valor nulo:

```
EXECUTE INCLUIR_SEGMENTO (NULL, 'SEGMENTO INCLUIDO PELO USUÁRIO USER_APP');
```

13) Veja o erro de inclusão de nulo em um campo que não permite isso:

```

| Erro a partir da linha : 11 no comando -
| BEGIN INCLUIR_SEGMENTO (NULL, 'SEGMENTO INCLUIDO PELO USUÁRIO USER_APP'); END;
| Relatório de erros -
| ORA-01400: não é possível inserir NULL em ("USER_DEV"."SEGMercado"."ID")
| ORA-06512: em "USER_APP.INCLUIR_SEGMENTO", line 5
| ORA-06512: em line 1
| 01400. 00000 - "cannot insert NULL into (%s)"
| *Cause: An attempt was made to insert NULL into previously listed objects.
| *Action: These objects cannot accept NULL values.

```

14) Utilize o pacote de exceções para tratar o erro na *procedure* do usuário **user_app**. Use a exceção do pacote local. Edite o *script* de criação da *procedure* com o código abaixo:

```

CREATE OR REPLACE PROCEDURE INCLUIR_SEGMENTO (
  p_ID IN SEGMercado.ID%type,
  p_DESCRIÇÃO IN SEGMercado.DESCRIÇÃO%type
)
IS
BEGIN
  INSERT INTO SEGMercado (ID, DESCRIÇÃO) VALUES (p_ID, p_DESCRIÇÃO);
EXCEPTION
  WHEN EXCEPTION_PAC.e_null THEN
    raise_application_error (-20015, 'Campo do segmento com preenchimento obrigatório');
END;

```

15) Para que a exceção seja usada por um usuário externo, no *script* do usuário **user_dev**, execute:

```
CREATE PUBLIC SYNONYM EXCEPTION_PAC FOR user_dev.EXCEPTION_PAC;
```

16) Agora é possível executar a *procedure* novamente:

```
EXECUTE INCLUIR_SEGMENTO (NULL, 'SEGMENTO INCLUIDO PELO USUÁRIO USER_APP');
```

17) Observe o seguinte resultado:

```
Erro a partir da linha : 23 no comando -
BEGIN INCLUIR_SEGMENTO (NULL, 'SEGMENTO INCLUIDO PELO USUÁRIO USER_APP'); END;
Relatório de erros -
ORA-20015: Campo do segmento com preenchimento obrigatório
ORA-06512: em "USER_APP.INCLUIR_SEGMENTO", line 8
ORA-06512: em line 1
```



18) Faça o uso de constantes. Para isso, abra o pacote **Cliente_PAC**, na parte de cabeçalho. Inclua os comandos abaixo para a criação das constantes:

```
g_cli_lim_fatur_pequeno NUMBER(10) := 10000;
g_cli_lim_fatur_medio NUMBER(10) := 50000;
g_cli_lim_fatur_medio_grande NUMBER(10) := 100000;
```

19) Compile o cabeçalho do pacote **Cliente_PAC**.

20) No corpo do pacote, vá ao código da função **Obter_Categoria_Cliente** e modifique os testes da categoria para o uso das constantes:

```
IF p_faturamento_previsto <= g_cli_lim_fatur_pequeno THEN
    RETURN 'PEQUENO';
ELSIF p_faturamento_previsto <= g_cli_lim_fatur_medio THEN
    RETURN 'MEDIO';
ELSIF p_faturamento_previsto <= g_cli_lim_fatur_medio_grande THEN
    RETURN 'MEDIO GRANDE';
ELSE
    RETURN 'GRANDE';
END IF;
```

21) Crie um *script* para o usuário **user_app**. Execute a inclusão de um cliente, obtendo a categoria conforme os parâmetros das constantes incluídos no passo anterior:

```
EXECUTE APP_INCLUIR_CLIENTE (30, 'INCLUIDO USADO PACKAGE', '33343', 2, 100000);
```

22) Verifique a categoria do cliente incluído, executando a consulta a tabela de clientes:

```
SELECT * FROM CLIENTE;
```

ID	RAZAO_SOCIAL	CNPJ	SEGMENTO_ID	DATA_INCLUSAO	FATURAMENTO_PREVISTO	CATEGORIA
1	30 INCLUIDO USADO PACKAGE	33/343		21/8/19	100000	MEDIO GRANDE
2	1 SUPERMERCADO XYZ	12/345		5/15/19	150000	GRANDE
3	2 SUPERMERCADO IJK	67/890		1/15/19	90000	MEDIO GRANDE
4	3 SUPERMERCADO IJK	89/012		3/15/19	80000	MEDIO GRANDE
5	4 FARMACIA AXZ	12/378		3/15/19	80000	MEDIO GRANDE
6	5 PRIMEIRO CLIENTE INCLUIDO POR USER_APP	23/456		2/15/19	100000	MEDIO GRANDE
7	6 SEGUNDO CLIENTE INCLUIDO POR USER_APP	23/456		2/15/19	100000	MEDIO GRANDE

23) Com os dois comandos abaixo, o usuário **user_app** pode consultar o valor da constante:

```
SET SERVEROUTPUT ON
EXEC dbms_output.put_line(CLIENTE_PAC.g_cli_lim_fatur_medio_grande);
```

```
Procedimento PL/SQL concluido com sucesso.
100000
```

24) Volte ao código do pacote e altere o valor da constante:

```
g_cli_lim_fatur_medio_grande NUMBER(10) := 90000;
```

25) Compile o pacote (cabeçalho e corpo).

26) Voltando ao *script* do usuário **user_app**, inclua um novo cliente usando o valor do faturamento previsto igual ao outro:

```
EXECUTE APP_INCLUIR_CLIENTE (35, 'INCLUIDO USADO PACKAGE', '33343', 2, 100000);
```

27) Por fim, verifique a categoria dos dois clientes e observe que eles tiveram classificações diferentes:

```
SELECT * FROM CLIENTE;
```

ID	RAZAO_SOCIAL	CNPJ	SEGMENTO_ID	DATA_INCLUSAO	FATURAMENTO_PREVISTO	CATEGORIA
1	30 INCLUIDO USADO PACKAGE	33/343	2	18/10/19	100000	MÉDIO GRANDE
2	35 INCLUIDO USADO PACKAGE	33/343	2	18/10/19	100000	GRANDE
3	1 SUPERMERCADO XYZ	12/345	5	15/10/19	150000	GRANDE
4	2 SUPERMERCADO IJK	67/890	1	15/10/19	90000	MÉDIO GRANDE
5	3 SUPERMERCADO IJK	89/012	3	15/10/19	80000	MÉDIO GRANDE
6	4 FARMACIA AXZ	12/378	3	15/10/19	80000	MÉDIO GRANDE