

## 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 um *script* associado à conexão **user\_dev**.
- 2) Digite o seguinte comando, para criar o cabeçalho do pacote e execute:

```
CREATE OR REPLACE PACKAGE CLIENTE_PAC
IS
PROCEDURE INCLUIR_CLIENTE (
    p_id in cliente.id%type,
    p_razao_social in cliente.razao_social%type,
    p_CNPJ cliente.CNPJ%type ,
    p_segmercado_id cliente.segmercado_id%type,
    p_faturamento_previsto cliente.faturamento_previsto%type
);
END;
```

- 3) Crie o corpo do pacote. Para isso, digite e execute o comando:

```
CREATE OR REPLACE PACKAGE BODY CLIENTE_PAC
IS
PROCEDURE INCLUIR_CLIENTE (
    p_id in cliente.id%type,
    p_razao_social in cliente.razao_social%type,
    p_CNPJ cliente.CNPJ%type ,
    p_segmercado_id cliente.segmercado_id%type,
    p_faturamento_previsto cliente.faturamento_previsto%type
)
IS
    v_categoria cliente.categoria%type;
    v_CNPJ cliente.cnpj%type := p_CNPJ;
    v_codigo_erro number(5);
    v_mensagem_erro varchar2(200);
    v_dummy number;
    v_verifica_segmento boolean;
    e_segmento exception;
BEGIN
    v_verifica_segmento := verifica_segmento_mercado(p_segmercado_id);
    IF v_verifica_segmento = false THEN
        RAISE e_segmento;
    END IF;
    v_categoria := obter_categoria_cliente(p_faturamento_previsto);
    format_cnpj (v_cnpj);
    INSERT INTO cliente
        VALUES (p_id, UPPER(p_razao_social), v_CNPJ, p_segmercado_id,
                SYSDATE, p_faturamento_previsto, v_categoria);
    COMMIT;
EXCEPTION
    WHEN dup_val_on_index then
```

```

        raise_application_error(-20010,'Cliente já cadastrado');
WHEN e_segmento THEN
        raise_application_error (-20011,'Segmento de mercado inexistente');
WHEN OTHERS THEN
        v_codigo_erro := sqlcode;
        v_mensagem_erro := sqlerrm;
        raise_application_error (-20000,to_char(v_codigo_erro)||v_mensagem_erro);
END;
END;

```

4) Para dar permissão ao usuário **user\_app** para que ele visualize o pacote, digite e execute:

```
GRANT EXECUTE ON CLIENTE_PAC TO user_app;
```

5) Para criar um sinônimo do pacote, use:

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

6) Crie um *script* associado à conexão **user\_app**.

7) Inclua o primeiro cliente com o pacote:

```
EXECUTE user_dev.CLIENTE_PAC.INCLUIR_CLIENTE(10, 'PRIMEIRO CLIENTE INCLUIDO POR user_app VIA PACKAGE')
```

8) Visualize os resultados obtidos. Digite e execute:

```
SELECT * FROM CLIENTE;
```

ID	RAZAO_SOCIAL	CNPJ
1	1 SUPERMERCADO XYZ	12/345
2	2 SUPERMERCADO IJK	67/890
3	3 SUPERMERCADO IJK	89/012
4	4 FARMACIA AXZ	12/378
5	5 PRIMEIRO CLIENTE INCLUIDO POR USER_APP	23/456

9) Crie um *script* associado à conexão **user\_dev**.

10) Inclua outras *procedures* no cabeçalho do pacote. Digite e execute:

```

CREATE OR REPLACE PACKAGE CLIENTE_PAC
IS
PROCEDURE INCLUIR_CLIENTE (
    p_id IN cliente.id%type,

```

```

p_razao_social in cliente.razao_social%type,
p_CNPJ cliente.CNPJ%type ,
p_segmercado_id cliente.segmercado_id%type,
p_faturamento_previsto cliente.faturamento_previsto%type
);
PROCEDURE ATUALIZAR_CLI_SEG_MERCADO (
    p_id cliente.id%type,
    p_segmercado_id cliente.segmercado_id%type
);
PROCEDURE ATUALIZAR_FATURAMENTO_PREVISTO (
    p_id in cliente.id%type,
    p_faturamento_previsto in cliente.faturamento_previsto%type
);
PROCEDURE EXCLUIR_CLIENTE (
    p_id in cliente.id%type
);
END;

```

11) Repita o processo para criar o corpo:

```

create or replace PACKAGE BODY CLIENTE_PAC
IS
PROCEDURE INCLUIR_CLIENTE (
    p_id in cliente.id%type,
    p_razao_social in cliente.razao_social%type,
    p_CNPJ cliente.CNPJ%type ,
    p_segmercado_id cliente.segmercado_id%type,
    p_faturamento_previsto cliente.faturamento_previsto%type
)
IS
    v_categoria cliente.categoria%type;
    v_CNPJ cliente.cnpj%type := p_CNPJ;
    v_codigo_erro number(5);
    v_mensagem_erro varchar2(200);
    v_dummy number;
    v_verifica_segmento boolean;
    e_segmento exception;
BEGIN
    v_verifica_segmento := verifica_segmento_mercado(p_segmercado_id);
    IF v_verifica_segmento = false THEN
        RAISE e_segmento;
    END IF;
    v_categoria := obter_categoria_cliente(p_faturamento_previsto);
    format_cnpj (v_cnpj);
    INSERT INTO cliente
        VALUES (p_id, UPPER(p_razao_social), v_CNPJ, p_segmercado_id,
                SYSDATE, p_faturamento_previsto, v_categoria);
    COMMIT;
EXCEPTION
    WHEN dup_val_on_index then
        raise_application_error(-20010,'Cliente já cadastrado');
    WHEN e_segmento then
        raise_application_error (-20011,'Segmento de mercado inexistente');
    WHEN OTHERS then
        v_codigo_erro := sqlcode;
        v_mensagem_erro := sqlerrm;
        raise_application_error (-20000 to char(v_codigo_erro||v_mensagem_erro));

```

```
  RAISE_APPLICATION_ERROR(-20000, 'O cliente não possui segmento de mercado definido');
END;

PROCEDURE ATUALIZAR_CLI_SEG_MERCADO (
    p_id cliente.id%type,
    p_segmercado_id cliente.segmercado_id%type
)
IS
    e_fk exception;
    pragma exception_init(e_fk, -2291);
    e_no_update exception;
BEGIN
    UPDATE cliente
        SET segmercado_id = p_segmercado_id
        WHERE id = p_id;
    IF SQL%NOTFOUND then
        RAISE e_no_update;
    END IF;
    COMMIT;
EXCEPTION
    WHEN e_fk then
        RAISE_APPLICATION_ERROR(-20001, 'Segmento de Mercado Inexistente');
    WHEN e_no_update then
        RAISE_APPLICATION_ERROR(-20002, 'Cliente Inexistente');
END;

PROCEDURE ATUALIZAR_FATURAMENTO_PREVISTO (
    p_id in cliente.id%type,
    p_faturamento_previsto in cliente.faturamento_previsto%type
)
IS
    v_categoria cliente.categoria%type;
    e_error_id exception;
BEGIN
    v_categoria := obter_categoria_cliente(p_faturamento_previsto);
    UPDATE cliente
        SET categoria = v_categoria,
            faturamento_previsto = p_faturamento_previsto
        WHERE id = p_id;
    IF SQL%NOTFOUND THEN
        RAISE e_error_id;
    END IF;
    COMMIT;
EXCEPTION
    WHEN e_error_id then
        raise_application_error(-20010, 'Cliente inexistente');
END;

PROCEDURE EXCLUIR_CLIENTE (
    p_id in cliente.id%type
)
IS
    e_error_id exception;
BEGIN
    DELETE FROM cliente
        WHERE id = p_id;
    IF SQL%NOTFOUND THEN
        RAISE e_error_id;
    END IF;
    COMMIT;
EXCEPTION
    WHEN e_error_id then
        raise_application_error(-20010, 'Cliente inexistente');
```

```

WHEN E_CLIENTE_NAO_ENCONTRADO THEN
    raise_application_error(-20010,'Cliente inexistente');
END;

```

12) Crie um *script* associado à conexão **user\_app**.

13) Visualize a tabela executando:

```
SELECT * FROM CLIENTE;
```

14) Observe o seguinte resultado:

ID	RAZAO_SOCIAL	CNPJ	SEGMENTO_ID	DATA_INCLUSAO	FATURAMENTO_PREVISTO	CATEGORIA
1	1 SUPERMERCADO XYZ	12/345	5	15/10/19	150000	GRANDE
2	2 SUPERMERCADO IJK	67/890	1	15/10/19	90000	MEDIO GRAN
3	3 SUPERMERCADO IJK	89/012	3	15/10/19	80000	MEDIO GRAN
4	4 FARMACIA AXZ	12/378	3	15/10/19	80000	MEDIO GRAN
5	5 PRIMEIRO CLIENTE INCLUIDO POR USER_APP	23/456	2	15/10/19	100000	MEDIO GRAN
6	6 SEGUNDO CLIENTE INCLUIDO POR USER_APP	23/456	2	15/10/19	100000	MEDIO GRAN
7	10 PRIMEIRO CLIENTE INCLUIDO POR USER_APP VIA PACKAGE	45/678	2	16/10/19	120000	GRANDE
8	11 SEGUNDO CLIENTE INCLUIDO POR USER_APP VIA PACKAGE	24/534	2	16/10/19	130000	GRANDE

15) Digite e execute o código abaixo para excluir o cliente 10:

```
EXECUTE CLIENTE_PAC.EXCLUIR_CLIENTE(10);
```

16) Visualize novamente os resultados. Digite e execute:

```
SELECT * FROM CLIENTE;
```

ID	RAZAO_SOCIAL	CNPJ	SEGMENTO_ID	DATA_INCLUSAO	FATURAMENTO_PREVISTO	CATEGORIA
1	1 SUPERMERCADO XYZ	12/345	5	15/10/19	150000	GRANDE
2	2 SUPERMERCADO IJK	67/890	1	15/10/19	90000	MEDIO GRAN

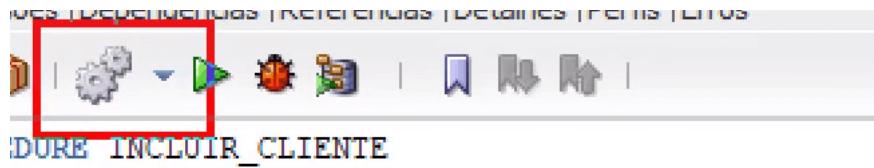
17) Clique sobre o cabeçalho do pacote, através da árvore de objetos do SQL Developer. Haverá uma área de edição de texto com seu conteúdo, e sem a necessidade de copiar e colar em uma área de edição do SQL Developer. Insira a seguinte *procedure* no cabeçalho do pacote:

```

PROCEDURE INCLUIR_CLIENTE (
    p_id IN cliente.id%type,
    p_razao_social IN cliente.razao_social%type,
    p_segmento_id IN cliente.segmento_id%type
);

```

18) Clique no botão indicado abaixo e selecione a opção **Compilar**:



19) Existe uma inconsistência entre o cabeçalho e o corpo do pacote. O próximo passo é complementar o corpo. Para isso, clique sobre **CLIENTE\_PAC Body** para abrir a área de edição do corpo do pacote.

20) Acrescente ao código do corpo:

```

PROCEDURE INCLUIR_CLIENTE (
    p_id in cliente.id%type,
    p_razao_social in cliente.razao_social%type,
    p_segmercado_id cliente.segmercado_id%type
)
IS
    v_codigo_erro number(5);
    v_mensagem_erro varchar2(200);
    v_dummy number;
    v_verifica_segmento boolean;
    e_segmento exception;
BEGIN
    v_verifica_segmento := verifica_segmento_mercado(p_segmercado_id);
    IF v_verifica_segmento = false THEN
        RAISE e_segmento;
    END IF;
    INSERT INTO cliente (ID, RAZAO_SOCIAL, SEGMERCADO_ID, DATA_INCLUSAO)
        VALUES (p_id, UPPER(p_razao_social), p_segmercado_id, SYSDATE);
    COMMIT;
EXCEPTION
    WHEN dup_val_on_index then
        raise_application_error(-20010,'Cliente já cadastrado');
    WHEN e_segmento then
        raise_application_error (-20011,'Segmento de mercado inexistente');
    WHEN OTHERS then
        v_codigo_erro := sqlcode;
        v_mensagem_erro := sqlerrm;
        raise_application_error (-20000,to_char(v_codigo_erro)||v_mensagem_erro);
END;

```

21) Crie um *script* associado à conexão **user\_app**.

22) Digite e execute o seguinte código para incluir um cliente:

```
EXECUTE CLIENTE_PAC.INCLUIR_CLIENTE(15, 'INCLUIR CLIENTE COM 5 PARAMETROS', '99999', 2, 90000);
```

23) Observe que o cliente já está incluído:

24) Por fim, digite e execute o código para permitir a inclusão de clientes:

```
EXECUTE CLIENTE_PAC.INCLUIR_CLIENTE(16, 'INCLUIR CLIENTE COM 3 PARAMETROS', 2);
```

25) Visualize novamente os resultados. A tabela estará assim: