

RM87771 - Bruna da Silva Moura
RM87138 - Filipe Marchetti Giusti Holanda
RM88466 - Paulo Augusto Grillo
RM88838 - Thais Brasil Lenhard
RM86992 - Wellington dos Santos Ramos

Supermercado - Cap 10 – PL/SQL (Procedimentos, Gatilhos Mágicos e Testes)

Procedimentos

- Renovar Senha e Enviar E-mail:

```
CREATE OR REPLACE PROCEDURE RM87138.prc_resetar_senha(p_id_usuario IN
NUMBER) AS
    PRAGMA AUTONOMOUS_TRANSACTION;

    -- Variáveis
    v_email varchar2(100);

BEGIN

    -- Capturar a informação e-mail do usuário
    SELECT
        DS_EMAIL INTO v_email
    FROM
        RM87138.TB_SPM_USUARIO
    WHERE
        ID_USUARIO = p_id_usuario;

    -- Incluir na tabela de controle Email com status "Disparar"
    -- Obs.: Existe um JOB que pegará todos registros que estão na fila com
    status "Disparar" e serão disparados
    INSERT INTO TB_SPM_CONTROLE_EMAIL(ID_CONTROLE_EMAIL, DS_EMAIL,
STATUS_ID, DH_CRIACAO)
    VALUES(SEQ_SPM_CONTROLE_EMAIL.NEXTVAL, v_email, 3, SYSDATE);
    COMMIT;

END;
```

- Usuário Ausente:

```
CREATE OR REPLACE PROCEDURE RM87138.prc_usuario_ausente(p_id_usuario IN
NUMBER) AS
    PRAGMA AUTONOMOUS_TRANSACTION;

BEGIN

    -- Mudar o status para "Desativado" do usuário
    UPDATE
        RM87138.TB_SPM_USUARIO
    SET
        ID_STATUS = 6 -- DESATIVADO
    WHERE
        ID_USUARIO = p_id_usuario;
    COMMIT;

END;
```

- Usuário Morto:

```
CREATE OR REPLACE PROCEDURE RM87138.prc_usuario_morto(p_id_usuario IN
NUMBER) AS
    PRAGMA AUTONOMOUS_TRANSACTION;

BEGIN

    -- Excluir todos registros de Carrinho relacionado através pelo
    parametro de id usuario
    DELETE FROM
        RM87138.TB_SPM_CARRINHO
    WHERE
        ID_USUARIO = p_id_usuario;
    COMMIT;

    -- Excluir todos registros de Endereco relacionado através pelo
    parametro de id usuario
    DELETE FROM
        RM87138.TB_SPM_ENDERECO
    WHERE
        ID_USUARIO = p_id_usuario;
    COMMIT;

    -- Excluir todos registros de Telefone relacionado através pelo
    parametro de id usuario
    DELETE FROM
        RM87138.TB_SPM_TELEFONE
    WHERE
        ID_USUARIO = p_id_usuario;
    COMMIT;

    -- Excluir todos registros de Usuario relacionado através pelo parametro
    de id usuario
    DELETE FROM
        RM87138.TB_SPM_USUARIO
    WHERE
        ID_USUARIO = p_id_usuario;
    COMMIT;

END;
```

- Adicionar produto(s) no Carrinho:

- Manter os salvos para dar continuidade outra hora;

```
CREATE OR REPLACE PROCEDURE
RM87138.prc_produto_carrinho_adicionar(p_cd_pedido IN VARCHAR2, p_id_usuario
IN NUMBER, p_id_produto IN NUMBER, p_quantidade IN NUMBER) AS
    PRAGMA AUTONOMOUS_TRANSACTION;

BEGIN
    -- Incluir produto na tabela de Carrinho com status "Temporário"
    INSERT INTO RM87138.TB_SPM_CARRINHO
    (ID_CARRINHO, ID_PRODUTO, ID_STATUS, ID_USUARIO, CD_PEDIDO, NR_QUANTIDADE)
    VALUES (SEQ_SPM_CARRINHO.NEXTVAL, p_id_produto, 2, p_id_usuario,
    p_cd_pedido, p_quantidade);
    COMMIT;

END;
```

- Movimentar registros (quantidade e produto) para seguir próxima etapa;

```
CREATE OR REPLACE PROCEDURE
RM87138.prc_produto_carrinho_pagamento(p_cd_pedido IN VARCHAR2) AS
    PRAGMA AUTONOMOUS_TRANSACTION;

BEGIN
    -- Mudar o status para "Em andamento" de todos registros relacionado ao
    código do pedido
    UPDATE RM87138.TB_SPM_CARRINHO
    SET ID_STATUS = 1 -- EM ANDAMENTO
    WHERE CD_PEDIDO = p_cd_pedido;
    COMMIT;

END;
```

- Controle de Estoque:

```
CREATE OR REPLACE PROCEDURE RM87138.prc_produto_controle_estoque(p_cd_pedido
IN VARCHAR2) AS
    PRAGMA AUTONOMOUS_TRANSACTION;

    -- Buscar todos registros do carrinho através pelo parametro código de
    Pedido com status "Em andamento"
    CURSOR curCarrinho IS
        SELECT
            tsc.ID_PRODUTO,
            tsc.NR_QUANTIDADE
        FROM
            TB_SPM_CARRINHO tsc
        WHERE
            tsc.CD_PEDIDO = p_cd_pedido
            AND tsc.ID_STATUS = 1; -- EM ANDAMENTO

    -- Variáveis
    v_ID_PRODUTO varchar2(100);
    v_NR_QUANTIDADE NUMBER(19,0);
    v_CARRINHO_OK NUMBER(1,0);
BEGIN

    v_CARRINHO_OK := 0;

    -- Abrir cursor da consulta
    OPEN curCarrinho;

    -- Percorrer todos registros, conforme da consulta
    LOOP
        FETCH curCarrinho INTO v_ID_PRODUTO, v_NR_QUANTIDADE;
        EXIT WHEN curCarrinho%NOTFOUND;
        -- Alterar a quantidade de estoque do produto
        UPDATE
            RM87138.TB_SPM_PRODUTO
        SET
            NR_QTD_ESTOQUE = NR_QTD_ESTOQUE - v_NR_QUANTIDADE
        WHERE
            ID_PRODUTO = v_ID_PRODUTO;
        COMMIT;
        v_CARRINHO_OK := 1;
    END LOOP;

    -- Fechar cursor da consulta
    CLOSE curCarrinho;

    -- Verifica se realmente foi todo processado e/ evitar da atualização
    sem produto(s) no carrinho
    IF v_CARRINHO_OK = 1 THEN

        -- Mudar o status para "Pago", ou seja, foi processado e não há
        necessidade de entrar novamente no processo, caso seja chamado com mesmo
        pedido
        UPDATE
            RM87138.TB_SPM_CARRINHO
        SET
            ID_STATUS = 7 -- Pago
        WHERE
            CD_PEDIDO = p_cd_pedido;
        COMMIT;
    END IF;
END;
```

Gatilhos Mágicos

- Controle de Estoque:

```
CREATE OR REPLACE TRIGGER RM87138.tri_controle_estoque
BEFORE INSERT ON TB_SPM_CAIXA
FOR EACH ROW
BEGIN

    -- Após inserção na tabela TB_SPM_CAIXA com status = "Pago" (id_status:
    7), executar procedimento abaixo:
    IF INSERTING AND :NEW.ID_STATUS = 7 THEN
        RM87138.prc_produto_controle_estoque(:NEW.CD_PEDIDO);
    END IF;

END tri_controle_estoque;
```

- Renovar Senha e Enviar E-mail:

```
CREATE OR REPLACE TRIGGER RM87138.tri_resetar_senha
BEFORE UPDATE ON TB_SPM_USUARIO
FOR EACH ROW
BEGIN

    -- Após atualização na tabela TB_SPM_USUARIO com status = "Resetar
    senha" (id_status: 9), executar procedimento abaixo:
    IF UPDATING AND :NEW.ID_STATUS = 9 THEN
        -- Resetar a senha de forma aleatória do usuário
        :NEW.DS_SENHA := TO_CHAR(sysdate, 'SSSSS');
        -- Setar status para "Em andamento resete da senha"
        :NEW.ID_STATUS := 10;
        RM87138.prc_resetar_senha(:NEW.ID_USUARIO);
    END IF;

END tri_resetar_senha;
```

- Usuário Ausente:

```
CREATE OR REPLACE TRIGGER RM87138.tri_usuario_ausente
AFTER INSERT OR UPDATE OR DELETE ON TB_SPM_USUARIO

DECLARE

    -- Buscar todos usuários que não houve acesso a mais de 6 meses
    CURSOR curUsuario IS
        SELECT
            tsu.ID_USUARIO
        FROM
            TB_SPM_USUARIO tsu
        WHERE
            MONTHS_BETWEEN(SYSDATE, tsu.DH_ULTIMO_ACESSO) >= 6
            AND tsu.ID_STATUS <> 6; -- Desativado

    -- Variáveis
    v_ID_USUARIO NUMBER(19,0);
BEGIN

    -- Abrir cursor da consulta
    OPEN curUsuario;

    -- Percorrer todos registros, conforme da consulta
    LOOP
        FETCH curUsuario INTO v_ID_USUARIO;
        EXIT WHEN curUsuario%NOTFOUND;
        -- Alterar status para "Desativado" (código: 6)
        RM87138.prc_usuario_ausente(v_ID_USUARIO);
    END LOOP;

    -- Fechar cursor da consulta
    CLOSE curUsuario;

END tri_usuario_ausente;
```

- Usuário Morto:

```
CREATE OR REPLACE TRIGGER RM87138.tri_usuario_morto
AFTER INSERT OR UPDATE OR DELETE ON TB_SPM_USUARIO

DECLARE

    -- Buscar todos usuários que não houve acesso a mais de 6 meses
    CURSOR curUsuario IS
        SELECT
            tsu.ID_USUARIO
        FROM
            TB_SPM_USUARIO tsu
        WHERE
            MONTHS_BETWEEN(SYSDATE, tsu.DH_ULTIMO_ACESSO) >= 12
            AND tsu.ID_STATUS = 6; -- Desativado

    -- Variáveis
    v_ID_USUARIO NUMBER(19,0);
BEGIN

    -- Abrir cursor da consulta
    OPEN curUsuario;

    -- Percorrer todos registros, conforme da consulta
    LOOP
        FETCH curUsuario INTO v_ID_USUARIO;
        EXIT WHEN curUsuario%NOTFOUND;
        -- Remover todos registros relacionado ao usuário
        RM87138.prc_usuario_morto(v_ID_USUARIO);
    END LOOP;

    -- Fechar cursor da consulta
    CLOSE curUsuario;

END tri_usuario_morto;
```

Testes

```
-- ### Resetar a Senha e Envia E-mail ### --

-- Efetuar atualização do status para "Resetar senha" (código: 9)
UPDATE RM87138.TB_SPM_USUARIO SET ID_STATUS = 9 WHERE ID_USUARIO = 1;

-- Verificar se foi atualizado a senha e status setado para "Em andamento
resete da senha" (código: 10)
SELECT * FROM RM87138.TB_SPM_USUARIO WHERE ID_USUARIO = 1;

-- Verificar se foi incluído novo registro com status "E-mail - Disparar"
(código: 3)
SELECT * FROM RM87138.TB_SPM_CONTROLE_EMAIL;
```

```
-- ### Controle de Estoque ### --

-- Inclusão dos produtos
CALL RM87138.prc_produto_carrinho_adicionar('00001', 1, 1, 5);
CALL RM87138.prc_produto_carrinho_adicionar('00001', 1, 2, 5);
CALL RM87138.prc_produto_carrinho_adicionar('00001', 1, 3, 5);

-- Cenário: Durante navegação do usuário até pagamento será chamado
-- Força status de "Temporário" (código: 2) para "Em andamento" (código: 1)
CALL RM87138.prc_produto_carrinho_pagamento('00001');

-- Verificar se foi setado para status "Em andamento" (código: 1)
SELECT * FROM RM87138.TB_SPM_CARRINHO;

-- Incluir pagamento efetuado do código de pedido: 00001
INSERT INTO RM87138.TB_SPM_CAIXA (ID_CAIXA, ID_STATUS, CD_PEDIDO,
DH_CRIACAO)
VALUES (SEQ_SPM_CAIXA.NEXTVAL, 7, '00001', SYSDATE);

-- Verificar se foi setado o status para "Pago" (código: 7)
SELECT * FROM RM87138.TB_SPM_CAIXA;

-- Verificar se foi subtraído o nr quantidade de estoque do(s) produto(s)
em relação a quantidade do carrinho do pedido
SELECT * FROM RM87138.TB_SPM_PRODUTO;
```

```
-- ### Usuario Ausente ### --

-- Fazer ação qualquer (Insert ou Update ou Delete, pois existe um usuário
(id_usuario = 2) que não acessou há mais de 6 meses
UPDATE RM87138.TB_SPM_USUARIO SET ID_STATUS = 5 WHERE ID_USUARIO = 1;

-- Cenário de teste - Usuário que não acessou há mais de 6 meses
-- Obs.: Ao tentar inserir com condição +6 meses sem uso será "Desativado"
-- E outros registros que estiverem dentro na condição +6 meses sem
uso tb serão trocados de status para "Desativado"
INSERT INTO RM87138.TB_SPM_USUARIO (ID_USUARIO, NM_NOME, NM_SOBRENOME,
DS_LOGIN, DS_SENHA, DS_EMAIL, ID_GRUPO, ID_STATUS, DH_CRIACAO,
DH_ULTIMO_ACESSO)
VALUES (SEQ_SPM_USUARIO.NEXTVAL, 'Fulano 1', 'Silva', 'fulano1silva',
'1234', 'fulano1silva@email.com', 1, 5, SYSDATE, SYSDATE);

-- Verificar se foi setado o status para "Desativado" (código: 6)
SELECT * FROM RM87138.TB_SPM_USUARIO WHERE ID_USUARIO = 2;
-- ou
SELECT * FROM RM87138.TB_SPM_USUARIO tsu WHERE NM_NOME = 'Fulano 1';
```

```
-- ### Usuario Morto ### --

-- Cenário de teste - Usuário que não acessou há mais de 12 meses e com
status "Desativado" (código: 6)
-- Obs.: Ao tentar inserir com condição +12 meses sem uso + status
"Desativado" será removido imediato
-- E outros registros que estiverem dentro na condição +12 meses sem
uso tb serão removidos
INSERT INTO RM87138.TB_SPM_USUARIO (ID_USUARIO, NM_NOME, NM_SOBRENOME,
DS_LOGIN, DS_SENHA, DS_EMAIL, ID_GRUPO, ID_STATUS, DH_CRIACAO,
DH_ULTIMO_ACESSO)
VALUES (SEQ_SPM_USUARIO.NEXTVAL, 'Bruno', 'Silva', 'brunosilva', '1234',
'bruno@email.com', 1, 6, SYSDATE, TO_DATE('01/01/2020', 'DD/MM/YYYY'));

-- Verificar se existe registrado
SELECT * FROM RM87138.TB_SPM_USUARIO tsu WHERE NM_NOME = 'Bruno'
```