

**CREATE OR REPLACE** PACKAGE paquete\_modificacion **AS**

```
PROCEDURE sp_insert_product (  
    categoria    VARCHAR2,  
    l_name       VARCHAR2,  
    l_descripcion VARCHAR2,  
    precio       NUMBER,  
    pyme         NUMBER  
);
```

```
PROCEDURE create_order (  
    id_cliente NUMBER  
);
```

```
PROCEDURE create_line (  
    orden    NUMBER,  
    nombre   VARCHAR2,  
    cantidad NUMBER  
);
```

```
PROCEDURE sp_cancel_order (  
    order_id NUMBER  
);
```

```
PROCEDURE sp_edit_product (  
    product NUMBER,  
    n_desc   VARCHAR2,  
    n_mont   NUMBER,  
    opc      NUMBER  
);
```

**END** paquete\_modificacion;

/

**CREATE OR REPLACE** PACKAGE BODY paquete\_modificacion **AS**

```
PROCEDURE sp_insert_product (  
    categoria    VARCHAR2,  
    l_name       VARCHAR2,  
    l_descripcion VARCHAR2,  
    precio       NUMBER,  
    pyme         NUMBER  
) AS
```

```
    cat_id NUMBER;  
    aux    NUMBER;
```

**BEGIN**

**SELECT**

**COUNT**(\*)

**INTO** aux

**FROM**

categoria ct

**WHERE**

ct.nombre\_categoria = categoria;

**IF** aux = 0 **THEN**

**INSERT INTO** categoria ( nombre\_categoria ) **VALUES** ( categoria );

**SELECT**

ct.categoria\_id

**INTO** cat\_id

**FROM**

categoria ct

**WHERE**

ct.nombre\_categoria = categoria;

**ELSE**

**SELECT**

```

        ct.categoria_id
    INTO cat_id
    FROM
        categoria ct
    WHERE
        ct.nombre_categoria = categoria;

END IF;

INSERT INTO producto (
    nombre_producto,
    descripcion,
    precio_unitario,
    categoria_id,
    pyme_id
) VALUES (
    l_name,
    l_descripcion,
    precio,
    cat_id,
    pyme
);

END sp_insert_product;

PROCEDURE create_order (
    id_cliente NUMBER
) AS
    fecha DATE;
BEGIN
    INSERT INTO orden (
        monto_total,
        estado,
        fecha_orden,
        cliente_id
    ) VALUES (
        0,
        1,
        current_date,
        id_cliente
    );

END create_order;

PROCEDURE create_line (
    orden    NUMBER,
    nombre   VARCHAR2,
    cantidad NUMBER
) AS
    cash    NUMBER(10, 2);
    product NUMBER;
BEGIN
    SELECT
        search_product(nombre)
    INTO product
    FROM
        dual;

    IF product != 0 THEN
        SELECT
            ( pr.precio_unitario * cantidad )
        INTO cash
        FROM
            producto pr
        WHERE

```

```

        product = pr.producto_id;

    INSERT INTO linea_de_producto (
        cantidad,
        monto,
        orden_id,
        producto_id
    ) VALUES (
        cantidad,
        cash,
        orden,
        product
    );

END IF;

END create_line;

PROCEDURE sp_cancel_order (
    order_id NUMBER
) AS
BEGIN
    UPDATE orden od
    SET
        estado = 0
    WHERE
        order_id = od.orden_id;

END sp_cancel_order;

PROCEDURE sp_edit_product (
    product NUMBER,
    n_desc VARCHAR2,
    n_mont NUMBER,
    opc NUMBER
) AS
    old_mont NUMBER;
BEGIN
    IF opc = 1 THEN
        UPDATE producto pr
        SET
            nombre_producto = n_desc
        WHERE
            product = pr.producto_id;

    END IF;

    IF opc = 2 THEN
        UPDATE producto pr
        SET
            descripcion = n_desc
        WHERE
            product = pr.producto_id;

    END IF;

    IF opc = 3 THEN
        UPDATE producto pr
        SET
            pr.precio_unitario = n_mont
        WHERE
            product = pr.producto_id;

    END IF;

```

**END** sp\_edit\_product;

**END** paquete\_modificacion;

/

**CREATE OR REPLACE** PACKAGE paquete\_consulta **AS**

**FUNCTION** search\_product (  
    descp VARCHAR2  
) **RETURN** NUMBER;

**FUNCTION** fc\_valid\_cliente (  
    new\_cedula        VARCHAR2,  
    new\_apodo          VARCHAR2,  
    new\_telefono\_cliente VARCHAR2  
) **RETURN** NUMBER;

**FUNCTION** fc\_valid\_pyme (  
    new\_cedula        VARCHAR2,  
    new\_email          VARCHAR2,  
    new\_telefono\_cliente VARCHAR2  
) **RETURN** NUMBER;

**FUNCTION** fc\_cliente\_login (  
    ident VARCHAR2,  
    pass  VARCHAR2  
) **RETURN** NUMBER;

**FUNCTION** fc\_pyme\_login (  
    ident VARCHAR2,  
    pass  VARCHAR2  
) **RETURN** NUMBER;

**PROCEDURE** sp\_select\_orders (  
    cliente  NUMBER,  
    condition NUMBER  
)  
;

**PROCEDURE** sp\_view\_sales (  
    pyme NUMBER,  
    mes  NUMBER  
)  
;

**END** paquete\_consulta;

/

**CREATE OR REPLACE** PACKAGE BODY paquete\_consulta **AS**

**FUNCTION** search\_product (  
    descp VARCHAR2  
) **RETURN** NUMBER **IS**  
    aux  NUMBER;  
    pr\_id NUMBER;  
**BEGIN**  
    **SELECT**  
        COUNT(\*)  
    **INTO** aux  
    **FROM**  
        producto pr  
    **WHERE**  
        descp = pr.nombre\_producto  
        **OR** pr.descripcion **LIKE** concat('%', concat(descp, '%'));  
  
    **IF** aux > 1 **THEN**  
        **RETURN** 0;  
    **END IF**;

```

IF aux = 1 THEN
    SELECT
        producto_id
    INTO pr_id
    FROM
        producto pr
    WHERE
        descp = pr.nombre_producto
        OR pr.descripcion LIKE concat('%', concat(descp, '%'));

    RETURN pr_id;
END IF;

RETURN 0;
END search_product;

FUNCTION fc_valid_cliente (
    new_cedula          VARCHAR2,
    new_apodo            VARCHAR2,
    new_telefono_cliente VARCHAR2
) RETURN NUMBER IS
    apodex NUMBER;
    cedulex NUMBER;
    numberx NUMBER;
BEGIN
    SELECT
        COUNT(*)
    INTO apodex
    FROM
        cliente
    WHERE
        new_apodo = apodo;

    SELECT
        COUNT(*)
    INTO cedulex
    FROM
        cliente
    WHERE
        new_cedula = cedula;

    SELECT
        COUNT(*)
    INTO numberx
    FROM
        cliente
    WHERE
        new_telefono_cliente = telefono_cliente;

    IF
        apodex = 0
        AND cedulex = 0
        AND numberx = 0
    THEN
        RETURN 0;
    ELSE
        IF apodex != 0 THEN
            RETURN 1;
        END IF;
        IF cedulex != 0 THEN
            RETURN 2;
        END IF;
        IF numberx != 0 THEN
            RETURN 3;
        END IF;

```

**END IF;**

**END** fc\_valid\_cliente;

**FUNCTION** fc\_valid\_pyme (  
    new\_cedula        VARCHAR2,  
    new\_email          VARCHAR2,  
    new\_telefono\_cliente VARCHAR2

) **RETURN** NUMBER **IS**

    emailx NUMBER;  
    cedulex NUMBER;  
    numberx NUMBER;

**BEGIN**

**SELECT**

        COUNT(\*)

**INTO** emailx

**FROM**

        pyme

**WHERE**

        new\_email = email;

**SELECT**

        COUNT(\*)

**INTO** cedulex

**FROM**

        pyme

**WHERE**

        new\_cedula = cedula\_juridica;

**SELECT**

        COUNT(\*)

**INTO** numberx

**FROM**

        pyme

**WHERE**

        new\_telefono\_cliente = telefono\_pyme;

**IF**

        emailx = 0

**AND** cedulex = 0

**AND** numberx = 0

**THEN**

**RETURN** 0;

**ELSE**

**IF** emailx != 0 **THEN**

**RETURN** 1;

**END IF;**

**IF** cedulex != 0 **THEN**

**RETURN** 2;

**END IF;**

**IF** numberx != 0 **THEN**

**RETURN** 3;

**END IF;**

**END IF;**

**END** fc\_valid\_pyme;

**FUNCTION** fc\_cliente\_login (  
    ident VARCHAR2,  
    pass VARCHAR2

) **RETURN** NUMBER **IS**  
    resultado NUMBER;  
    c\_id        NUMBER;  
    new\_pass    **BLOB**;

**BEGIN**

```

new_pass := to_blob(utl_raw.cast_to_raw(pass));
SELECT
    COUNT(*)
INTO resultado
FROM
    cliente cl
WHERE
    ( ident = cl.apodo
    OR ident = cl.telefono_cliente
    OR ident = cl.cedula )
    AND dbms_lob.compare(new_pass, cl.contraseña) = 0;

IF resultado = 1 THEN
    SELECT
        cliente_id
    INTO c_id
    FROM
        cliente cl
    WHERE
        ( ident = cl.apodo
        OR ident = cl.telefono_cliente
        OR ident = cl.cedula )
        AND dbms_lob.compare(new_pass, cl.contraseña) = 0;

    RETURN c_id;
END IF;

RETURN 0;
END fc_cliente_login;

FUNCTION fc_pyme_login (
    ident VARCHAR2,
    pass VARCHAR2
) RETURN NUMBER IS
    resultado NUMBER;
    p_id NUMBER;
    new_pass BLOB;
BEGIN
    new_pass := to_blob(utl_raw.cast_to_raw(pass));
    SELECT
        COUNT(*)
    INTO resultado
    FROM
        pyme py
    WHERE
        ( ident = py.cedula_juridica
        OR ident = py.telefono_pyme
        OR ident = py.email )
        AND dbms_lob.compare(new_pass, py.clave) = 0;

    IF resultado = 1 THEN
        SELECT
            pyme_id
        INTO p_id
        FROM
            pyme py
        WHERE
            ( ident = py.cedula_juridica
            OR ident = py.telefono_pyme
            OR ident = py.email )
            AND dbms_lob.compare(new_pass, py.clave) = 0;

        RETURN p_id;
    END IF;

```

```

RETURN 0;
END fc_pyme_login;

PROCEDURE sp_select_orders (
    cliente  NUMBER,
    condition NUMBER
) AS
    c1 SYS_REFCURSOR;
BEGIN
    IF condition = 0 THEN
        OPEN c1 FOR SELECT
            concat('Orden#', ord.orden_id) orden,
            ord.monto_total      monto,
            ord.fecha_orden      fecha,
            estado
        FROM
            orden ord
        WHERE
            ord.cliente_id = cliente;

        dbms_sql.return_result(c1);
    END IF;

    IF condition = 1 THEN
        OPEN c1 FOR SELECT
            concat('Orden#', ord.orden_id) orden,
            ord.monto_total      monto,
            ord.fecha_orden      fecha,
            estado
        FROM
            orden ord
        WHERE
            ord.cliente_id = cliente
            AND ord.estado = 0;

        dbms_sql.return_result(c1);
    END IF;

    IF condition = 2 THEN
        OPEN c1 FOR SELECT
            concat('Orden#', ord.orden_id) orden,
            ord.monto_total      monto,
            ord.fecha_orden      fecha,
            estado
        FROM
            orden ord
        WHERE
            ord.cliente_id = cliente
            AND ord.estado = 1;

        dbms_sql.return_result(c1);
    END IF;
END sp_select_orders;

```

```

PROCEDURE sp_view_sales (
    pyme NUMBER,
    mes  NUMBER
) AS
    c1 SYS_REFCURSOR;
    articulos_vendidos NUMBER;
    pmonto_total      NUMBER;
    name_prod         VARCHAR2(55);
    mesesito          NUMBER;

```



**BEGIN**

**OPEN** c1 **FOR SELECT**

```
pr.nombre_producto      porducto,
SUM(cantidad)           cantidad,
SUM(pr.precio_unitario * cantidad) montototal,
mes                     mes
```

**INTO**

```
name_prod,
articulos_vendidos,
pmonto_total,
mesesito
```

**FROM**

```
linea_de_producto ldp
```

```
INNER JOIN producto pr ON ldp.producto_id = pr.producto_id
```

```
INNER JOIN orden ord ON ldp.orden_id = ord.orden_id
```

**WHERE**

```
pr.pyme_id = pyme
```

```
AND mes = EXTRACT(MONTH FROM ord.fecha_orden)
```

```
AND ord.estado = 1
```

**GROUP BY**

```
pr.nombre_producto;
```

```
dbms_sql.return_result(c1);
```

**END** sp\_view\_sales;

**END** paquete\_consulta;

/