



# PUNTOS POR MIL

Jonathan Alexander Garzon Soto 635601

## CODIGO

```
CREATE OR REPLACE FUNCTION calc_puntos (f_venta NUMBER)
RETURN NUMBER
IS
    resultado NUMBER;
    cien_mil NUMBER;

BEGIN
    IF f_venta > 99999 THEN
        SELECT TRUNC(f_venta/100000)*10 INTO cien_mil FROM DUAL;
        SELECT TRUNC(f_venta/1000)+cien_mil INTO resultado FROM DUAL;
        RETURN resultado;
    ELSE
        SELECT TRUNC(f_venta/1000) INTO resultado FROM DUAL;
        RETURN resultado;
    END IF;

END;
```

```
CREATE OR REPLACE FUNCTION F_DIAS(p_fecini DATE, p_fecfin DATE)
RETURN NUMBER
IS
    v_dias NUMBER := 0;

BEGIN
    SELECT TRUNC ( p_fecfin )
        - ADD_MONTHS (
            TRUNC ( p_fecini ),
            FLOOR ( MONTHS_BETWEEN ( TRUNC ( p_fecfin ), TRUNC ( p_fecini ) ) ) ) + 1
        INTO v_dias
    FROM DUAL;

    RETURN v_dias;
```

```

SELECT TRUNC ( p_fecfin ,

- ADD_MONTHS (

    TRUNC ( p_fecini ),

    FLOOR ( MONTHS_BETWEEN ( TRUNC ( p_fecfin ), TRUNC ( p_fecini ) ) ) ) + 1

    INTO v_dias

FROM DUAL;

RETURN v_dias;

EXCEPTION

    WHEN OTHERS

        THEN

            RETURN 0;

END F_DIAS;

```

```

CREATE OR REPLACE PROCEDURE puntos_por_mil (p_cedula_cliente IN NUMBER, p_valor_venta IN NUMBER)
IS
    validar_cedula NUMBER;

    puntos NUMBER;
    dias NUMBER;
    valor NUMBER;
    descuento NUMBER;
    cincuenta NUMBER;
    contador NUMBER;
    cal_puntos NUMBER;
    refe_factura NUMBER;
    total_puntos NUMBER;
    fecha DATE;

BEGIN

    SELECT COUNT(*) INTO validar_cedula FROM clientes WHERE pk_nu_cedula=p_cedula_cliente;

    IF validar_cedula < 1 THEN

        INSERT INTO clientes
        VALUES (p_cedula_cliente, 'cliente' , 'nuevo', 'sin dato', 'sin dato','31/12/1999');

    END IF;

```

```

VALUES (p_cedula_cliente, cliente , nuevo , sin dato , sin dato , 31/12/1999 );

INSERT INTO ventas
VALUES (refe.NEXTVAL, p_cedula_cliente, 'productos canasta familiar', p_valor_venta, SYSDATE);

ELSE

INSERT INTO ventas
VALUES (refe.NEXTVAL, p_cedula_cliente, 'productos canasta familiar', p_valor_venta, SYSDATE);

END IF;

SELECT COUNT(*) INTO contador FROM puntos_clientes
WHERE FK_NU_CEDULA = p_cedula_cliente;

SELECT PK_NU_REF INTO refe_factura
from (SELECT * FROM ventas WHERE fk_nu_cedula = p_cedula_cliente
ORDER BY PK_NU_REF DESC )
WHERE ROWNUM = 1 ;

IF contador > 0 THEN

SELECT DA_PUNTOS INTO fecha FROM puntos_clientes
WHERE fk_nu_cedula = p_cedula_cliente;

SELECT NU_PUNTOS INTO puntos FROM puntos_clientes
WHERE fk_nu_cedula = p_cedula_cliente;

```

```

SELECT F_DIAS(TO_DATE(fecha), SYSDATE) INTO dias FROM DUAL;

IF dias > 30 OR dias IS NOT NULL THEN

IF puntos > 1999 THEN
SELECT TRUNC(puntos/2000)*50 INTO cincuenta FROM DUAL;
SELECT puntos+cincuenta INTO valor FROM DUAL;
SELECT calc_puntos(p_valor_venta) INTO cal_puntos FROM DUAL;
SELECT TRUNC((puntos*1)/100) INTO descuento FROM DUAL;
total_puntos := (valor+cal_puntos+puntos)-descuento;
--listo para actualizar puntos
UPDATE puntos_clientes SET fk_nu_ref_venta = refe_factura, nu_puntos = total_puntos, nu_puntos_venci = descuento, da_puntos = SYSDATE
WHERE fk_nu_cedula = p_cedula_cliente;

ELSE IF puntos < 1999 THEN
SELECT calc_puntos(p_valor_venta) INTO cal_puntos FROM DUAL;
SELECT TRUNC((puntos*1)/100) INTO descuento FROM DUAL;
total_puntos := (cal_puntos+puntos)- descuento;
--listo para actualizar puntos
UPDATE puntos_clientes SET fk_nu_ref_venta = refe_factura, nu_puntos = total_puntos, nu_puntos_venci = descuento, da_puntos = SYSDATE
WHERE fk_nu_cedula = p_cedula_cliente;
ELSE
SELECT calc_puntos(p_valor_venta) INTO cal_puntos FROM DUAL;
total_puntos := puntos + cal_puntos;
--Insertar datos
UPDATE puntos_clientes SET fk_nu_ref_venta = refe_factura, nu_puntos = total_puntos, nu_puntos_venci = descuento, da_puntos = SYSDATE
WHERE fk_nu_cedula = p_cedula_cliente;
END IF;

```

```

ELSE
    IF puntos > 1999 THEN
        SELECT TRUNC(puntos/2000)*50 INTO cincuenta FROM DUAL;
        SELECT puntos+cincuenta INTO valor FROM DUAL;
        SELECT calc_puntos(p_valor_venta) INTO cal_puntos FROM DUAL;
        total_puntos := valor+cal_puntos+puntos;
        --listo para actualizar puntos
        UPDATE puntos_clientes SET fk_nu_ref_venta = refe_factura, nu_puntos = total_puntos, da_puntos = SYSDATE
        WHERE fk_nu_cedula = p_cedula_cliente;
    ELSE
        SELECT calc_puntos(p_valor_venta) INTO cal_puntos FROM DUAL;
        total_puntos :=cal_puntos+puntos;
        --listo para actualizar puntos
        UPDATE puntos_clientes SET fk_nu_ref_venta = refe_factura, nu_puntos = total_puntos, da_puntos = SYSDATE
        WHERE fk_nu_cedula = p_cedula_cliente;
    END IF;

END IF;

ELSE
    SELECT calc_puntos(p_valor_venta) INTO cal_puntos FROM DUAL;
    total_puntos := cal_puntos;
    --Insertar datos
    INSERT INTO puntos_clientes
    VALUES (p_cedula_cliente, refe_factura, total_puntos, 0, SYSDATE);
END IF;

```

## PRUEBAS DE EJECUCION

```
EXECUTE puntos_por_mil (105440178,10000);
```

Salida de Script x Resultado de la Consulta x

Tarea terminada en 0,112 segundos

Procedure PUNTOS\_POR\_MIL compilado

Procedure PUNTOS\_POR\_MIL compilado

Procedimiento PL/SQL terminado correctamente.

```
select * from clientes,  
select * from ventas;  
select * from puntos_clientes;
```

Salida de Script x Resultado de la Consulta x

Todas las Filas Recuperadas: 1 en 0,005 segundos

	PK_NU_REF	FK_NU_CEDULA	VAR_DESCRIPCION	NU_VALOR	DA_VENTA
1	1000	105440178	productos canasta familiar	10000	01/05/20

```
select * from puntos_clientes;
```

Salida de Script x Resultado de la Consulta x

Todas las Filas Recuperadas: 1 en 0,01 segundos

	FK_NU_CEDULA	FK_NU_R...	NU_PUNTOS	NU_PUNT...	DA_PUNTOS
1	105440178	1000	10	0	01/05/20

```
select * from ventas;
select * from puntos_clientes;
```

Salida de Script x Resultado de la Consulta x					
SQL   Todas las Filas Recuperadas: 2 en 0,006 segundos					
PK_NU_REF	FK_NU_CEDULA	VAR_DESCRIPCION		NU_VALOR	DA_VENTA
1	1000	105440178 productos canasta familiar		10000	01/05/20
2	1002	105440178 productos canasta familiar		100000	01/05/20

```
select * from ventas;
select * from puntos_clientes;
```

Salida de Script x Resultado de la Consulta x					
SQL   Todas las Filas Recuperadas: 1 en 0,006 segundos					
FK_NU_CEDULA	FK_NU_R...	NU_PUNTOS	NU_PUNT...	DA_PUNTOS	
1	105440178	1002	120	0 01/05/20	

```
select * from ventas;
select * from puntos_clientes;

INSERT INTO VENTAS
VALUES (refe.nextval, 105430168, 'canasta familiar', 200000, '01/04/20' );

INSERT INTO puntos_clientes
VALUES (105430168, 1003, 220, 0, '01/04/20')
```

Salida de Script x Resultado de la Consulta x					
SQL   Todas las Filas Recuperadas: 2 en 0,003 segundos					
FK_NU_CEDULA	FK_NU_R...	NU_PUNTOS	NU_PUNT...	DA_PUNTOS	
1	105440178	1002	120	0 01/05/20	
2	105430168	1004	328	2 01/05/20	