

1. Desarrolle la función F_OBTENER_SALARIO, la cual, dada la cédula de un empleado, devuelve su salario actual (recuerde que el salario actual está determinado por la posición vigente del empleado):

Posteriormente, seleccione cédula, nombre y apellido y salario de la tabla B_EMPLEADOS. El salario deberá ser obtenido a través de la función creada.
2. Dropee la función anterior, y realice el mismo SELECT utilizando la función en la cláusula **WITH**:
3. Cree el procedimiento P_GENERAR_PLAN que recibe como parámetro un ID de la tabla VENTAS. El procedimiento generará el plan de pago (cuotas) a pagar. La generación se hará sólo si el PLAZO es mayor a 1. Genere con el siguiente criterio:
 - Si plazo = 30, 1 sola cuota con 1,5% de interés
 - Si plazo = 60, 2 cuotas con 3% de interés sobre el monto total de venta
 - Si plazo = 90, 3 cuotas con 5% de interés sobre el monto total de venta.
 Los vencimientos serán mensuales a partir de la fecha de venta.
4. Cree el procedimiento P_AUMENTAR_SALARIOS que recibe como parámetro el porcentaje de aumento, y lo aplica sobre la tabla de categorías.
5. Cree la función F_CALCULAR_SALDO que reciba como parámetro un código de cuenta contable, y devuelva como resultado el SALDO de la cuenta. El saldo de la cuenta se obtiene con la suma de todos los débitos – la suma de todos los créditos del MAYOR hasta la fecha.
6. Cree el procedimiento P_ULTIMA_COMPRA, que tendrá como función recorrer la tabla de artículos y actualizar la tabla con la fecha de la última compra, si es que tiene registros de compra. Realice esta operación con un cursor FOR UPDATE.
7. Una de las propiedades de las Funciones es la Recursividad.
 - Un típico ejemplo de recursividad puede verse en la función para cálculo de Factorial. Desarrolle una función denominada F_FACTORIAL que recibirá como parámetro un número positivo, y devolverá el factorial. Recuerde que la fórmula para cálculo de factorial es $N! = N \cdot (N-1)!$
 - Escriba la función F_FIBONACCI que calcule el valor de la serie de Fibonacci de una posición dada. Como sabrá, la Serie de Fibonacci se muestra así:

Serie de Fibonacci	1	1	2	3	5	8	13	21	34	55	89
Posición	0	1	2	3	4	5	6	7	8	9	10

Por lo tanto Fibonacci(3) = 3;
Fibonacci(4) = 5;

$$\text{Fibonacci}(5) = 8 ;$$

Es decir, cada posición es igual a la suma de las 2 posiciones anteriores, con excepción de las posiciones 0 y 1.

8. Escriba el procedimiento P_DETALLE_VENTA que recibirá como parámetros un ID de venta, y el ID de un artículo, y la cantidad. El procedimiento deberá:

- Contar con la función interna F_VERIFICAR_ARTÍCULO, que recibe como parámetro el artículo y la cantidad. La función verificará si el artículo existe, y si la cantidad solicitada no supera al stock. Si ambas condiciones se cumplen, retornará el **precio** del artículo. En caso contrario, retornará -1.

Desarrollo del cuerpo principal:

- Verificar si existe el id de venta. Si no existe, mostrará el mensaje “La venta no existe”, y terminará el procedimiento.
- Verificar si existe el artículo y obtener su precio con la función F_VERIFICAR_ARTÍCULO. Si la función devuelve -1, dará un mensaje de error: “El artículo no existe o no cuenta con stock suficiente” y terminará el procedimiento.
- Insertar el articulo con el ID de venta, el id de artículo y la cantidad recibida por parámetro, y el precio obtenido en la función. Tenga en cuenta que el numero_item deberá ser el último número de ítem de la venta +1, por lo que deberá recuperar también dicho valor.
- Finalmente actualizará la cabecera de la venta, haciendo que $\text{monto_total} = \text{monto_total} + (\text{precio} * \text{cantidad del artículo que se acaba de insertar})$ y el stock del artículo, haciendo que $\text{stock_actual} = \text{stock_actual} - \text{cantidad (de la venta)}$.

9. El Dpto. Comercial desea asignar calificaciones a los clientes teniendo en cuenta el monto de sus operaciones en compras (contado y crédito) realizadas hasta el momento, los rangos se definen de la siguiente manera:

- Calificación 'A': para operaciones mayores o iguales a 15.000.000.
- Calificación 'B': para operaciones mayor o iguales a 8.000.000 y menores a 15.000.000.
- Calificación 'C': para operaciones menores a 8.000.000.

Las compras a créditos cuyas cuotas tienen atraso en el pago no deben ser tenidas en cuenta en cálculo del total de operaciones.

Se pide que desarrolle la función F_OBTENER_CALIFICACION que reciba como parámetro el id de un cliente y retorne la calificación del mismo según las especificaciones mencionadas anteriormente.

10. Antes de realizar una venta a crédito se desea verificar el saldo que dispone el cliente para este tipo de operaciones, la fórmula para obtener dicho monto es:

Límite de crédito - Saldo de todas las cuotas (vencidas o no) de ventas a crédito que aún no fueron pagadas.

El límite de crédito se establece según la calificación del cliente, los valores son los siguientes:

Calificación A: 22.000.000

Calificación B: 18.000.000

Calificación C: 10.000.000

Se pide que desarrolle la función `F_OBTENER_SALDO_CRED`, la misma recibe por parámetro el id del cliente y retorna el saldo que dispone para ventas a crédito.

La calificación del cliente puede obtenerla invocando la función `F_OBTENER_CALIFICACION` desarrollada en el ejercicio anterior.