

# Practicar con la inserción de registros

Cuando tenemos que insertar registros lo primero que nos tenemos que preguntar es si la tabla donde hay que insertar las filas existe o se tiene que crear en la propia instrucción:

Si existe → utilizamos INSERT INTO

Si hay que crearla → utilizamos SELECT ... INTO

Con el SELECT INTO lo mejor es escribir la SELECT que permite obtener las filas a insertar y luego añadir la cláusula INTO delante de FROM.

Hay que tener en cuenta que la nueva tabla hereda como nombres de columnas los encabezados de la SELECT, habrá que tenerlo en cuenta a la hora de rellenar la lista de selección de la SELECT.

1.- Crear una nueva tabla que se llamará Este con las oficinas del Este. Como no nos dicen nada de los nombres de las columnas de la nueva tabla asumiremos que serán los mismos que en la tabla Oficinas:

Escribimos la SELECT que me permite obtener las oficinas del Este:

```
SELECT *
FROM Oficinas
WHERE ciudad = 'Este'
```

Y a sólo nos falta añadir el INTO para que se cree la nueva tabla con el resultado:

```
SELECT *
INTO Este
FROM Oficinas
WHERE ciudad = 'Este'
```

2.- Obtener los pedidos del representante Juan Victor indicando para cada pedido los datos del pedido así como el precio unitario del producto que se ha vendido en el pedido, y un campo que se llamará *tipo* con el valor *prim*. Guardar el resultado en una nueva tabla que se llamará PedidosEspeciales

Escribimos la SELECT que me permite obtener los datos requeridos, en este caso necesitamos dos tabla pedidos y empleados para saber el nombre del representante del pedido, pero como en la nueva tabla no hay que guardar ningún dato de la tabla Empleados, podemos redactar la SELECT con un JOIN o con una subconsulta:

```
SELECT
FROM Pedidos INNER JOIN Empleados ON rep=numemp
WHERE nombre = 'Juan Victor'
```

O bien:

```
SELECT
FROM Pedidos
WHERE rep = ANY (SELECT numemp FROM Empleados WHERE nombre = 'Juan Victor')
```

Esta subconsulta es un ejemplo, podríamos utilizar un IN o un EXISTS.

Ahora nos falta determinar la lista de selección. ¿Qué campos queremos guardar en la nueva tabla? Los campos de la tabla Pedidos, el precio unitario y el campo tipo, pues esa es la lista de selección que necesitamos:

```
SELECT pedidos.*, ROUND(importe/cant,2) AS PrecioUni, 'prim' AS tipo
FROM Pedidos INNER JOIN Empleados WHERE rep=numemp
WHERE nombre = 'Juan Victor '
```

Y a sólo nos falta añadir el INTO para que se cree la nueva tabla con el resultado:

```
SELECT pedidos.*, ROUND(importe/cant,2) AS PrecioUni, 'prim' AS tipo
INTO PedidosEspeciales
FROM Pedidos INNER JOIN Empleados ON rep=numemp
WHERE nombre = 'Juan Victor'
```

Con los alias de campo asigno los nombres de la columnas de la nueva tabla.

### 3.- Añadir a la tabla Empleados los empleados:

Antonio Lopez, 45 años, asignado a la oficina 12 con jefe 105, ventas y cuota a cero.

Luis Viver, 30 años, con ventas cero y cuota 1.000€

En los dos casos la fecha de contrato será hoy.

En este caso la tabla existe, luego es un INSERT INTO, los valores a introducir son fijos, luego utilizaremos la cláusula VALUES, con VALUES sólo se puede insertar una fila cada vez, luego tengo que redactar dos INSERTs:

Para el primer empleado a insertar:

```
INSERT INTO Empleados
    (numemp, nombre, edad, oficina, jefe, ventas, cuota, contrato)
VALUES (115, 'Antonio Lopez', 45, 12, 105, 0, 0, GETDATE())
```

Para el segundo:

```
INSERT INTO Empleados
    (numemp, nombre, edad, ventas, cuota, contrato)
VALUES (116, 'Luis Viver', 30, 0, 1000, GETDATE())
```

La lista de columnas antes del VALUES es opcional pero muy recomendable, de esta forma nos aseguramos que asignamos los valores al campo que toca.

Los valores pueden ser valores fijos o expresiones como con la función GETDATE().

Aunque no nos hayan dicho nada en el enunciado, es obligatorio asignar a la clave principal un valor, para ello por ejemplo podemos ir a la tabla, ver cuál es el último valor y asignar el siguiente, el único requisito es que no exista ya el valor en otra fila.

Si la clave principal es una columna de identidad, en este caso no se puede asignar un valor a la clave, el sistema se lo asignará automáticamente.

### 4.- Añadir a la tabla Pedidosespeciales creada en el ejercicio 1 los pedidos del empleado 108 y rellenando el campo tipo con el valor *segundo*.

Aquí los valores a insertar están en una tabla luego utilizaremos la sintaxis del INSERT INTO con SELECT.

¿Qué valores hay que insertar? La respuesta me da la SELECT:

```
SELECT *, ROUND(importe/cant,2), 'segundo'
FROM Pedidos WHERE rep= 108
```

En este caso no se necesita la tabla Empleados porque nos dan el código del representante de los pedidos a insertar en vez del nombre.

Los alias de campo ya no son necesarios porque el resultado de la SELECT no se visualiza ni se hace nada con los encabezados de columnas, la SELECT sólo nos sirve para generar los valores a insertar.

Ahora el resultado de la SELECT lo insertamos en la tabla PedidosEspeciales:

```
INSERT INTO PedidosEspeciales SELECT *, ROUND(importe/cant,2), 'segundo'
FROM Pedidos WHERE rep= 108
```

En este caso no he puesto la lista de columnas para acortar, como acabamos de crear la nueva tabla y no hemos cambiado el orden de las columnas de la tabla Pedidos sabemos que los valores se van a insertar en las columnas correctas.

Pero si quisiéramos que la instrucción sirviese en cualquier momento y previendo que el orden de columnas pudiese cambiar, habría que incluir la lista de columnas y detallar las columnas en la SELECT:

```
INSERT INTO PedidosEspeciales (codigo, numpedido, fechapedido, clie, rep,
fab, producto, cant, importe, preciouni, tipo)
SELECT codigo, numpedido, fechapedido, clie, rep, fab, producto, cant,
importe, ROUND(importe/cant,2), 'segundo'
FROM Pedidos WHERE rep= 108
```