

Apuntes BBDD Avanzado

JOIN (Inner, Left, Right, Full Outer)

Ejemplo (BBDD classicmodels) :

Dame todos los clientes y el importe de sus compras, tengan o no tengan compras.

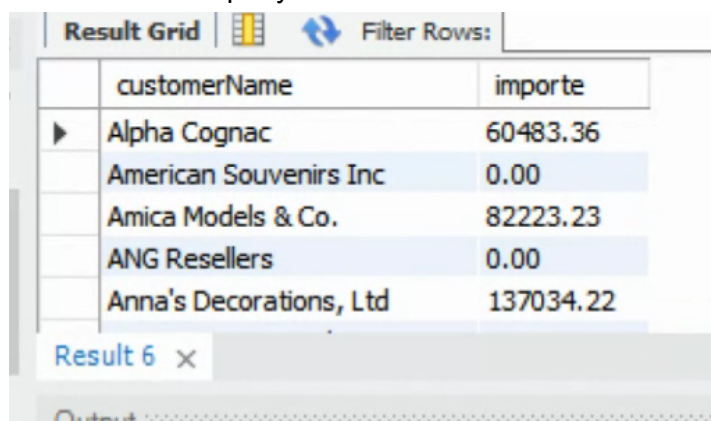
```
1 SELECT customers.customerName, SUM(orderdetails.quantityOrdered*orderdetails.priceEach) AS ImporteCompra
2 FROM orderdetails
3 JOIN orders
4 ON orders.orderNumber=orderdetails.orderNumber
5 RIGHT JOIN customers
6 ON orders.customerNumber=customers.customerNumber
7 GROUP BY customerName;
```

Cruzamos primero orderdetails y orders, hacemos un JOIN normal porque cada order (alvarán) va a tener seguro orderdetails (líneas de alvarán). Una vez las tenemos cruzadas, hacemos un RIGHT JOIN (porque la parte izquierda/LEFT es la que ya tenemos cruzada) con la tabla customers, usando la clave 'customerNumber' como referencia del cruzado (por ser la clave primaria y ajena), al final hacemos el GROUP BY para que nos de el listado entero, que si no lo hacemos nos daría solo el primer cliente.

COALESCE(cuando no es null, cuando es null)

```
select customers.customerName
      , coalesce(sum(orderdetails.priceEach*orderdetails.quantityOrdered),0) as importe
from orders
join orderdetails
  on orders.orderNumber = orderdetails.orderNumber
right join customers
  on orders.customerNumber = customers.customerNumber
group by 1
```

Resultado de la query:



The screenshot shows a 'Result Grid' window with a table containing two columns: 'customerName' and 'importe'. The table lists five customers with their respective purchase amounts. The first row is highlighted with a mouse cursor. Below the table, there is a tab labeled 'Result 6' and a partially visible 'Output' window.

customerName	importe
Alpha Cognac	60483.36
American Souvenirs Inc	0.00
Amica Models & Co.	82223.23
ANG Resellers	0.00
Anna's Decorations, Ltd	137034.22

Aquí Antonio usa el **COALESCE** para que si el sumatorio no es nulo, nos de el sumatorio, y si es nulo, en lugar de mostrar NULL nos muestre cero.

Ejercicio:

Familia del producto y suma de sus ventas:

```
1 SELECT products.productLine, SUM(orderdetails.quantityOrdered*orderdetails.priceEach) AS VentasTotal
2 FROM products
3 JOIN orderdetails
4 ON products.productCode=orderdetails.productCode
5 GROUP BY productLine;
```

Ejercicio:

Empleado, ProductLine(nombre del producto), Ventas

Todos los empleados y por cada uno todas las líneas de producto, tengan o no tengan ventas

*** Pista:

cruce sería employees (id) = customers(salesRepresentative)

customer (id) = orders (customerNumber)

order (orderNumber) = orderDetails(orderNumber)

orderDetails(productCode) = Products(productCode)