**Relacionando datos desde múltiples tablas**

Hasta ahora todas las instrucciones SELECT que vimos usaban una sola tabla en la cláusula FROM. Generalmente, como los datos están almacenados en varias tablas, es necesario juntar (hacer un join ) entre dos o más tablas.

Existen dos formas de juntar dos tablas:

1. **Cláusula WHERE. Hasta ANSI SQL-89.**

Podemos juntar dos tablas especificando explícitamente una condición en la cláusula WHERE.

Los nombres de columnas en común deben cualificarse con el nombre de la tabla a la cual pertenecen.

**Obtener el código de vendedor, apellido, nombre y bono de todos los vendedores. 17**

SELECT Sales.SalesPerson.BusinessEntityID, Person.Person.LastName,

Person.Person.FirstName, Sales.SalesPerson.Bonus

FROM Sales.SalesPerson , Person.Person

WHERE Sales.SalesPerson.BusinessEntityID= Person.Person. BusinessEntityID;

Para reducir código y consumir menos memoria pueden utilizarse nombres de alias para las tablas de la cláusula FROM. Se recomienda que sean cortos pero significativos.

Si se usan alias no puede cualificarse con el nombre de la tabla; y los alias solo tienen efecto en la instrucción SELECT en que son definidos.

SELECT s.BusinessEntityID, p.LastName, p.FirstName, s.Bonus

FROM Sales.SalesPerson s, Person.Person p

WHERE s.BusinessEntityID= p. BusinessEntityID;

Si no se especifica condición en la cláusula WHERE se obtendrá el producto cartesiano formado por todas las combinaciones posibles entre las filas de ambas tablas.

SELECT \* FROM Production.Product p, Production.ProductSubcategory s;

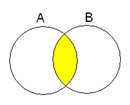
504 x 37 = 18648

1. **JOIN interno ( INNER JOIN ).**

Especifica un join entre dos tablas con una condición de join.

Solo se incluyen las filas que satisfacen con la condición de join.

**tabla1** [INNER] JOIN tabla2 ON **condición**



**Obtener el código de producto, número y nombre de subcategoría de todos los productos. 295**

SELECT p.ProductID, p.ProductNumber, s.Name

FROM Production.Product p JOIN Production.ProductSubcategory s

ON p.ProductSubcategoryID= s.ProductSubcategoryID;

1. **JOIN externo ( OUTER JOIN ).**

Hay 3 tipos:

1. **LEFT OUTER JOIN.**

Especifica un join entre dos tablas con condición de join.

Se devuelven también las filas de la primer tabla ( izquierda ), que no satisfacen la condición de join, juntándolas con filas con valores nulos correspondientes a la segunda tabla ( derecha ).

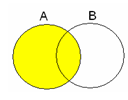


tabla1 LEFT [OUTER] JOIN tabla2 ON condición

**Obtener el código de vendedor y el código y nombre del territorio de venta asignado de todos los vendedores. 17**

SELECT s.BusinessEntityID, s.TerritoryID, t.Name

FROM Sales.SalesPerson s LEFT JOIN Sales.SalesTerritory t

ON s.TerritoryID=t. TerritoryID;

1. **RIGHT OUTER JOIN.**

Especifica un join entre dos tablas con condición de join.

Se devuelven también las filas de la segunda tabla ( derecha ), que no satisfacen la condición de join, juntándolas con filas con valores nulos correspondientes a la primer tabla ( izquierda ).

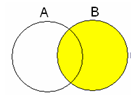


tabla1 RIGHT [OUTER] JOIN tabla2 ON condición

SELECT s.BusinessEntityID, s.TerritoryID, t.Name

FROM Sales.SalesPerson s RIGHT JOIN Sales.SalesTerritory t

ON s.TerritoryID=t. TerritoryID; 14 🡪 15

1. **FULL OUTER JOIN.**

Especifica un join entre dos tablas con condición de join.

Se devuelven también las filas de la primer tabla (izquierda ), que no satisfacen la condición de join, juntándolas con filas con valores nulos correspondientes a la segunda tabla ( derecha ).

Se devuelven también las filas de la segunda tabla (derecha ), que no satisfacen la condición de join, juntándolas con filas con valores nulos correspondientes a la primer tabla ( izquierda ).

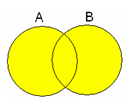


tabla1 FULL [OUTER] JOIN tabla2 ON condición

SELECT s.BusinessEntityID, s.TerritoryID, t.Name

FROM Sales.SalesPerson s FULL JOIN Sales.SalesTerritory t

ON s.TerritoryID=t. TerritoryID; 18

1. **Producto Cartesiano ( CROSS JOIN ).**

Especifica el producto cartesiano entre dos tablas sin condición de join.

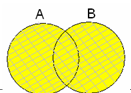


Tabla1 CROSS JOIN tabla2

SELECT \* FROM Production.Product p 504 x 37 = 18648

CROSS JOIN

Production.ProductSubcategory s;