



MADRID 2002

Manual para Analistas y programadores IBM AS / 400

1. ¿Que es el SQL?
2. Consultas básicas
3. Consultas avanzadas
4. Funciones
5. Agrupamientos
6. Consultas sobre varias tablas
7. Actualización
8. Borrar
9. Insertar



¿Que es el SQL?

El SQL (Estructure Query Language) es un lenguaje de programación que nos permite trabajar con bases de datos tanto a nivel usuario como a nivel administrador , permitiéndonos acceder a la información que contienen. El SQL se puede utilizar desde una terminal o puede estar embebido dentro de un programa el cual puede haber sido desarrollado en lenguajes como RPG , COBOL , FORTRAN , VISUAL BASIC , etc. El structure query lenguaje se ha divulgado en los últimos quince años pudiendo encontrar aplicaciones del mismo tanto en ordenadores personales , redes de PC , mainframe y Host. Nos permite realizar consultas y tareas de administración en forma amigable y con poco esfuerzo. Consultar , actualizar , crear y borrar registros como así también tablas, es una tarea sencilla con SQL. Como todo lenguaje tiene su sintaxis y semántica la cual desarrollaremos siguiendo las especificaciones de IBM. Cabe aclarar que el SQL se encuentra estandarizado por el ANSI (Instituto Americano de Normas) y por la ISO (International Standards Organization). Este lenguaje esta compuesto por:

- Palabras predefinidas
- Nombres de objetos
 - Tablas
 - Campos
- Constantes
- Signos

Con ellos construiremos las sentencias para acceder, actualizar , insertar y borrar información de las bases de datos. Dentro de las sentencias que utiliza SQL las podemos clasificar según su objetivo:



- Las sentencias que nos permiten trabajar con datos
 - **SELECT** : Nos permite realizar consultas sobre una tabla
 - **INSERT** : Con esta sentencia insertamos registros a una tabla
 - **UPDATE** : Nos permite actualizar , modificar datos dentro de una tabla
 - **DELETE** : Con esta sentencia borramos registros de una tabla.

- Las sentencias que nos permiten definir datos
 - **CREATE** : Con esta sentencia podemos crear tablas.
 - **DROP** : Nos permite destruir tablas.

- Las sentencias de control de datos
 - **GRANT** : Nos permite otorgar autorización a usuarios para que puedan acceder a una tabla.
 - **REVOKE** : Con esta sentencia podemos revocar una autorización de acceso a una tabla.



Consultas básicas en SQL

Anteriormente expresamos que la sentencia **SQL** que nos permite realizar consultas en :

SELECT

Supongamos que tenemos un tabla definida según la siguiente definición llamada **SECTOR**

CODS	5	N	0
SECD	30	A	

Los registros de este archivo contienen la siguiente información sobre los rubros de acciones en donde tiene inversiones una compañía petrolera.

CODS	SECD
10	SECTOR TECNOLOGÍAS
20	SECTOR CONSUMO
30	SECTOR FARMACEUTICO

Vamos a realizar una consulta para poder visualizar los registros de esta tabla.

La sentencia que escribiremos en la línea de comandos del SQL será la siguiente:

SELECT * FROM SECTOR



Luego de ejecutar esta sentencia observaremos en la pantalla el siguiente resultado.

CODS	SECD
10	SECTOR TECNOLOGÍAS
20	SECTOR CONSUMO
30	SECTOR FARMACEUTICO

Estamos visualizando todos los registros de este archivo y todos sus campos. La semántica básica de la sentencia SELECT es la siguiente

SELECT (campos a visualizar) FROM nombre del archivo

Cuando queremos ver todos los campos que se encuentran definidos en la tabla utilizamos el signo * y cuando queremos ver algún o algunos campos debemos enumerar sus nombres según la DDS.

Supongamos que queremos ver solamente el contenido del campo CODS, entonces escribiremos la siguiente sentencia:

SELECT CODS FROM SECTOR

Podemos decir que tanto **SELECT** como **FROM** son palabras predefinidas y CODS y SECTOR son nombres de objetos, en este caso el primero es el nombre de un campo y el segundo un nombre de archivo. Entonces para poder generalizar la interpretación de la sentencia SELECT y su sintaxis podemos decir :

SELECT cam01, camp02, camp03, campXX FROM nombre del archivo

Observamos que después de SELECT o escribimos * para visualizar todos los campos o escribimos una lista de campos que queremos visualizar en nuestra consulta.



Vamos consultar nuestra pequeña base de datos y queremos visualizar solamente aquellos sectores cuyo código sea superior a 10.

SELECT * FROM SECTOR WHERE CODS > 10

Hemos condicionado nuestra consulta con la palabra predefinida WHERE diciéndole que solamente deseamos visualizar los registros cuyo código de sector sean mayores a 10 y para eso también hemos utilizado la constante numérica 10 y el signo matemático de mayor. Al ejecutar esta consulta observaremos en nuestra pantalla los siguientes datos:

```

CODS  SECD
20    SECTOR CONSUMO
30    SECTOR FARMACEUTICO

```

Las constantes alfanuméricas se escriben entre comillas y se las aplica a campos alfanuméricos y las constantes numéricas sin comillas y se las aplica a campos numéricos.

A continuación tenemos definida una nueva tabla llamada **ACCION** según los siguientes datos.

CODAT	5	N	0
CODST	5	N	0
DESAT	35	A	
PAIS	3	A	



Ahora trabajaremos sobre ella y realizaremos nuevas consultas. El contenido de la misma es el siguiente :

CODAT	CODST	DESAT	PAIS
10	10	BBVA TELECOMUNICACIONES	ES
15	10	RENTA 4 TECNOLOGIA	ES
45	30	BANKPYME MULTISALUD	ES
70	10	BCH INTERNET	ES
25	10	CARMINAGNAC TECHNOLOGIES	FR
35	30	BBL INVT. FOOD & DEVE	BL
20	20	BAWESTO SECTOR CONSUMO	ES
30	20	FONDOESPACE FINANCIERO	ES
50	20	FRAMLINGTON FINANCIAL	UK
40	10	FINSBURY TECHNOLOGY TR	UK

SELECT * FROM ACCION WHERE CODS = 30

Luego de ejecutar esta sentencia observaremos por pantalla los siguientes datos

CODAT	CODST	DESAT	PAIS
45	30	BANKPYME MULTISALUD	ES
35	30	BBL INVT. FOOD & DEVE	BL

SELECT CODAT, DESAT, PAIS FROM ACCION WHERE PAIS = 'ES'

Observamos que queremos realizar una consulta que nos muestre las acciones de empresas españolas y para eso hemos condicionado la búsqueda por medio del campo **PAIS = 'ES'** . Considérese que **ES** es una constante alfanumérica y por lo tanto debe ir entre comillas.



CODAT	DESAT	PAIS
10	BBVA TELECOMUNICACIONES	ES
15	RENTA 4 TECNOLOGIA	ES
45	BANKPYME MULTISALUD	ES
70	BCH INTERNET	ES
20	BAWESTO SECTOR CONSUMO	ES
30	FONDOESPACE FINANCIERO	ES

Realizaremos otra consulta en base a la anterior pero con operadores lógicos. Queremos seleccionar las acciones españolas del sector tecnológico.

```
SELECT CODAT, DESAT FROM ACCION WHERE PAIS = 'ES' AND CODS = 10
```

El resultado será el siguiente:

CODAT	DESAT
10	BBVA TELECOMUNICACIONES
15	RENTA 4 TECNOLOGIA
70	BCH INTERNET
40	FINSBURY TECHNOLOGY TE

Ahora queremos visualizar todos los registros del archivo que no sean acciones de empresas de España.

```
SELECT CODAT , DESAT FROM ACCION WHERE PAIS <> 'ES'
```

Observemos el operador de comparación <> esto quiere decir que deseamos que se cumpla la condición campo PAIS distinto de 'ES'

(España). El resultado que obtendremos en nuestra pantalla será siguiente :



CODAT	DESAT
25	CARMINAGNAC TECHNOLOGIES
35	BBL INVT. FOOD & DEVE
50	FRAMLINGTON FINANCIAL
40	FINSBURY TECHNOLOGY TR

Ahora trabajaremos sobre otra tabla llamada **MOVIA** movimiento de acciones en donde se encuentran almacenados las operaciones de compra y venta de la cartera de la empresa. Esta tabla esta definida según la siguiente DDS:

CODAA	5	N	0
CODSS	5	N	0
FECT	8	N	0
PCU	9	N	2
CANC	9	N	0
IMPC	11	N	2
PVU	9	N	2
CANV	9	N	0
IMPV	11	N	2
CC	9	N	2
CV	9	N	2
EST	1	A	



Los datos en ella son los siguientes:

CODAA	CODS	FECT	PCU	CANC	IMPC	PVU	CANV	IMPV	CC	CV	EST
10	10	20020424	10	1000	9880	12	1000	11860	120	140	V
50	20	20020428	15	500	7410	16	500	7890	90	110	V
10	10	20020428	11	1000	10860	13	1000	12830	140	170	V
70	10	20020428	8	3000	23730				270		C
35	30	20020428	12	1000	11820	14	1000	13780	180	220	V
45	30	20020428	9	1000	8860				140		C
10	10	20020505	11	1000	10860	13	1000	12830	140	170	V
20	20	20020505	6	2000	11870				130		C
70	10	20020510	9	1000	8920				80		C
20	20	20020510	6	1000	5935				65		C

Se nos pide realizar una consulta para visualizar las acciones de BBVA Telecomunicaciones que se han vendido

SELECT * FROM MOVIA WHERE EST = ' V ' AND CODA=10

En resultado será el siguiente:

CODAA	COD	FECT	PCU	CANC	IMPC	PVU	CANV	IMPV	CC	CV	EST
	S										
10	10	20020424	10	1000	9880	12	1000	11860	120	140	V
10	10	20020428	11	1000	10860	13	1000	12830	140	170	V
10	10	20020210	11	1000	10860	13	1000	12830	140	170	V

Obsérvese el uso del operador lógico **AND**. Ahora queremos visualizar todos los movimientos de acciones código 50 y 35 mostrando código de acción, precio unitario en la compra, importe de compra, precio unitario de venta e importe de venta:

SELECT CODAA, PCU, IMPC, PVU, IMPV FROM MOVIA WHERE CODA IN(50, 35)

En resultado será el siguiente:



CODAA	PCU	IMPC	PVU	IMPV
50	15	7410	16	7890
35	12	11820	14	13780

Obsérvese que hemos utilizado en predicado (**NOT**) **IN** este nos permite listar aquellos registros cuyo campo **CODAA** tenga los valores 50 o 35.

Ahora realizaremos una consulta de todas las acciones que pertenezcan al sector consumo y farmacéutico. Nuestra sentencia **SQL** será:

SELECT * FROM MOVIA WHERE CODSS IN(20, 30)

Visualizaremos en nuestro ordenador la siguiente información:

CODAA	CODSS	FECT	PC U	CANC	IMPC	PVU	CANV	IMPV	CC	CV	EST
50	20	20020428	15	500	7410	16	500	7890	90	110	V
35	30	20020428	12	1000	11820	14	1000	13780	180	220	V
45	30	20020428	9	1000	8860				140		C
20	20	20020505	6	2000	11870				130		C
20	20	20020510	6	1000	5935				65		C

Ahora queremos visualizar aquellas acciones cuyos códigos no sean **10** y **50** entonces nuestra sentencia será:

SELECT * FROM MOVIA WHERE CODSS NOT IN(10, 30) ORDER BY IMPC

CODAA	CODSS	FECT	PC U	CANC	IMPC	PVU	CANV	IMPV	CC	CV	EST
20	20	20020510	6	1000	5935				65		C
20	20	20020505	6	2000	11870				130		C

En la siguiente consulta deseamos visualizar todos los movimientos de acciones anteriores al mes de mayo por lo que nuestra sentencia **SQL** será:



SELECT * FROM MOVIA WHERE FECT < 20020501

CODAA	CODS	FECT	PCU	CANC	IMPC	PVU	CANV	IMPV	CC	CV	EST
	S										
10	10	20020424	10	1000	9880	12	1000	11860	120	140	V
50	20	20020428	15	500	7500	16	500	7890	90	110	V
10	10	20020428	11	1000	10860	13	1000	12830	140	170	V
70	10	20020428	8	3000	23730				270		C
35	30	20020428	12	1000	11820	14	1000	13780	180	220	V
45	30	20020428	9	1000	8860				140		C

Queremos realizar la misma consulta pero que la información se genere ordenada por código de sector

**SELECT * FROM MOVIA WHERE FECT < 20020501 ORDER BY
CODS**

CODAA	CODS	FECT	PCU	CANC	IMPC	PVU	CANV	IMPV	CC	CV	EST
10	10	20020424	10	1000	9880	12	1000	11860	120	140	V
10	10	20020428	11	1000	10860	13	1000	12830	140	170	V
70	10	20020428	8	3000	23730				270		C
50	20	20020428	15	500	7410	16	500	7890	90	110	V
35	30	20020428	12	1000	11820	14	1000	13780	180	220	V
45	30	20020428	9	1000	8860				140		C

Con el predicado **ORDEN BY** podemos ordenar la información a visualizar por el campo que sea mas adecuado para la presentación de la misma.



Consultas avanzadas

Hemos llegado a un punto donde realizares consultas avanzadas

Queremos calcular la Rentabilidad luego de impuesto de los movimientos de acciones vendidas en base a la información que se encuentra en la tabla **MOVIA**.

Definamos la formula de rentabilidad en la compra y venta de acciones considerando que sobre el importe que resulta de restar el importe de venta menos el de compra se tributa el 10 % de impuesto. Los importes de venta y compra son netos de comisiones.

$$R = (((IMPV - IMPC) * 0.90) / IMPC) * 100$$

Aclaremos con un ejemplo numérico en base al primer registro de la tabla:

$$R = (((11860 - 9880) * 0.90) / 9880 + CC * 100 = 17,86$$

Nuestra sentencia **SQL** será la siguiente

```
SELECT CODAA, IMPC, IMPV, ((( IMPV - IMPC) * 0.90 ) / IMPC +  
CC ) * 100 FROM MOVIA WHERE EST= 'V' ORDER BY CODA ,  
CODS
```

El resultado será el siguiente

CODAA	IMPC	IMPV	COL-4
10	9880	11860	18,03
10	10860	12830	15,32
10	10860	12830	15,32
35	11820	13780	12,92
50	7410	7890	4,82



Obsérvese el uso del predicado **ORDER BY** para ordenar la información por un campo o campos determinados

Como vemos hemos seleccionado las acciones vendidas por medio del campo **EST** para garantizarnos que al efectuar los cálculos el campo importe de venta contuviese valores distintos de cero pero también podríamos usar el predicado **IS (NOT) NULL**.

```
SELECT CODAA, IMPC, IMPV, ((( IMPV - IMPC ) * 0.90 ) / IMPC + CC)  
* 100 FROM MOVIA WHERE IMPV IS NOT NULL ORDER BY CODA
```

EL resultado seria el mismo.

Ahora tenemos que seleccionar las acciones vendidas entre el 20/04/2002 y el 25/04/2002 y calcular su rentabilidad.

```
SELECT CODAA, IMPC, IMPV, ((( IMPV - IMPC ) * 0.90 ) / IMPC + CC)  
* 100 FROM MOVIA WHERE IMPV IS NOT NULL AND FECT  
BETWEEN 20020420 AND 20020425 ORDER BY CODA
```

El resultado es el siguiente:

CODAA	IMPC	IMPV	COL-4
10	9880	11860	18,03

Con el predicado **(NOT) BETWEEN** podemos realizar una selección entre rangos de un campo numérico. Aclaremos este concepto con un ejemplo.

Nos piden Seleccionar todos los movimientos de la tabla que no estén comprendidos entre las siguientes fechas:

28/04/2002 Y 05/05/2002

```
SELECT CODAA, IMPC, IMPV, CC, CV FROM MOVIA WHERE FECT  
NOT BETWEEN 20020428 AND 20020505 ORDER BY IMPC
```

Obtendremos luego de ejecutar la sentencia la siguiente información:



CODAA	IMPC	IMPV	CC	CV
20	5935		65	
70	8920		80	
10	9880	11860	120	140

Recordemos la tabla **ACCION** y supongamos que queremos seleccionar todas aquellos registros cuyo nombre de acción comiencen con ' **BB** '

CODAT	CODST	DESAT	PAIS
10	10	BBVA TELECOMUNICACIONES	ES
15	10	RENTA 4 TECNOLOGIA	ES
45	30	BANKPYME MULTISALUD	ES
70	10	BCH INTERNET	ES
25	10	CARMINAGNAC TECHNOLOGIES	FR
35	30	BBL INVT. FOOD & DEVE	BL
20	20	BAWESTO SECTOR CONSUMO	ES
30	20	FONDOESPACE FINANCIERO	ES
50	20	FRAMLINGTON FINANCIAL	UK
40	10	FINSBURY TECHNOLOGY TR	UK

SELECT * FROM ACCION WHERE DESAT LIKE 'BB%' ORDER BY DESA

El resultado de aplicar el predicado (**NOT**) **LIKE** Será el siguiente

CODA	CODS	DESA	PAIS
35	30	BBL INVT. FOOD & DEVE	BL
10	10	BBVA TELECOMUNICACIONES	ES



Seleccionar todos los nombres de acciones que contengan ' TEC ' dentro del campo **DESA**

```
SELECT * FROM ACCION WHERE DESAT LIKE '%TEC%' ORDER BY DESAT
```

CODAT	CODST	DESAT	PAIS
10	10	BBVA TELECOMUNICACIONES	ES
25	10	CARMINAGNAC TECHNOLOGIES	FR
40	10	FINSBURY TECHNOLOGY TR	UK
15	10	RENTA 4 TECNOLOGIA	ES

Ahora queremos visualizar las acciones cuyo nombre finalice en ES Ordenada por nombre

```
SELECT * FROM ACCION WHERE DESAT LIKE '%ES' ORDER BY DESAT
```

CODAT	CODST	DESAT	PAIS
10	10	BBVA TELECOMUNICACIONES	ES
25	10	CARMINAGNAC TECHNOLOGIES	FR



Funciones

Las funciones SQL nos permiten realizar operaciones con datos contenidos en las tablas.

Función LENGHT

Con esta función podemos conocer la longitud de los datos contenidos en un determinado campo.

Queremos visualizar aquellos nombre de acciones que tengan 12 caracteres

CODAT	CODST	DESAT	PAIS
10	10	BBVA TELECOMUNICACIONES	ES
15	10	RENTA 4 TECNOLOGIA	ES
45	30	BANKPYME MULTISALUD	ES
70	10	BCH INTERNET	ES
25	10	CARMINAGNAC TECHNOLOGIES	FR
35	30	BBL INVT. FOOD & DEVE	BL
20	20	BAWESTO SECTOR CONSUMO	ES
30	20	FONDOESPACE FINANCIERO	ES
50	20	FRAMLINGTON FINANCIAL	UK
40	10	FINSBURY TECHNOLOGY TR	UK

SELECT DESAT FROM ACCION WHERE LENGHT (DESAT) = 12

En resultado será el siguiente

CODAT	CODST	DESAT	PAIS
70	10	BCH INTERNET	ES



Ahora queremos visualizar aquellos nombres de acciones que tengan mas de 19 caracteres

```
SELECT DESAT FROM ACCION WHERE LENGHT(DESAT) > 19  
ORDER BY DESAT
```

Obtendremos la siguiente lista

CODAT	CODST	DESAT	PAIS
10	10	BBVA TELECOMUNICACIONES	ES
25	10	CARMINAGNAC TECHNOLOGIES	FR
35	30	BBL INVT. FOOD & DEVE	BL
20	20	BAWESTO SECTOR CONSUMO	ES
30	20	FONDOESPACE FINANCIERO	ES
50	20	FRAMLINGTON FINANCIAL	UK
40	10	FINSBURY TECHNOLOGY TR	UK

FUNCION SUBSTR

Al aplicar esta función a un campo alfanumérico obtendremos una subcadena del mismo cuyo comienzo y longitud debemos definir en los argumentos de la función:

Queremos visualizar el campo DESA a partir del tercer carácter y con una longitud de siete caracteres, entonces nuestra búsqueda se expresara de la siguiente forma

```
SELECT CODAT, SUBSTR ( DESAT , 3 , 7 ) FROM ACCION
```



CODAT	DESAT
10	VA TELE
25	RMINAGN
35	L INVT.
20	WESTO S
30	NDOESP
50	AMLING
40	FINSBURY TECHNOLOGY TR

Ahora necesitamos visualizar el campo **DESA** en sus primeros **10** caracteres de aquellos nombres que tengan mas de 15 caracteres.

SELECT CODAT, SUBSTR(DESAT, 1, 10) WHERE LENGHT (DESAT) > 15 ORDER BY DESAT

CODAT	DESAT
20	BAWESTO SE
35	BBL INVT. FO
10	BBVA TELEC
25	CARMINAGNA
40	FINSBURY T
30	FONDOESPAC
50	FRAMLINGTO

Función SUM

Con esta función podemos sumar los campos numéricos dentro de una tabla. Supongamos que necesitamos saber



**SELECT SUM (IMPV), SUM(IMPC), SUM(CV), SUM(CC), (SUM (IMPV)
- SUM(IMPC) FROM MOVIA WHERE EST = 'V'**

CODAA	CODSS	FECT	PC	CANC	IMPC	PVU	CANV	IMPV	CC	CV	EST
			U								
10	10	20020424	10	1000	9880	12	1000	11860	120	140	V
50	20	20020428	15	500	7410	16	500	7890	90	110	V
10	10	20020428	11	1000	10860	13	1000	12830	140	170	V
70	10	20020428	8	3000	23730				270		C
35	30	20020428	12	1000	11820	14	1000	13780	180	220	V
45	30	20020428	9	1000	8860				140		C
10	10	20020505	11	1000	10860	13	1000	12830	140	170	V
20	20	20020505	6	2000	11870				130		C
70	10	20020510	9	1000	8920				80		C
20	20	20020510	6	1000	5935				65		C

Luego de ejecutar la sentencia visualizaremos en nuestra pantalla la siguiente información

COL-1	COL-2	COL-3	COL-4	COL-5
59860	50830	810	670	8360

Función MAX

Nos permite conocer el valor máximo de un campo numérico dentro de una tabla

Supongamos que queremos conocer los máximos de importe de compra, importe de venta, comisión de compra y comisión de venta, entonces nuestra sentencia SQL será:



```
SELECT MAX(IMPC) , MAX( IMPV) , MAX (CC) , MAX(CV) FROM  
MOVIA WHERE EST='V'
```

COL-1	COL-2	COL-3	COL-4
11860	13780	180	220

Función MIN

Nos permite conocer el valor minimo de un campo numérico dentro de una tabla

Supongamos que queremos conocer los valores mininos de importe de compra , importe de venta, comisión de compra y comisión de venta, de las acciones vendidas, entonces nuestra sentencia SQL será:

```
SELECT MIN(IMPC) , MIN( IMPV) , MIN (CC) , MIN(CV) FROM MOVIA  
WHERE EST='V'
```

COL-1	COL-2	COL-3	COL-4
7410	7890	90	110



Función AVG

Nos permite conocer el valor promedio de un campo numérico dentro de una tabla

Supongamos que queremos conocer los valores promedio de importe de compra , importe de venta, comisión de compra y comisión de venta, entonces nuestra sentencia SQL será:

```
SELECT AVG(IMPC) , AVG( IMPV) , AVG (CC) , AVG(CV) FROM  
MOVIA WHERE EST='V'
```

COL-1	COL-2	COL-3	COL-4
10166	11838	134	162

Función COUNT

Nos permite conocer la cantidad de registros que tiene una tabla o un subconjunto de la misma

Por ejemplo deseamos conocer cuantos registros de acciones sin vender tenemos en nuestra tabla **MOVIA**

```
SELECT COUNT (*) FROM MOVIA WHERE EST <> 'V'
```



Ahora deseamos calcular el promedio de comisiones de compras sin usar la función **AVG** entonces nuestra sentencia **SQL** será :

```
SELECT SUM (CC) , COUNT (*), CC/COUNT(*) FROM MOVIA WHERE  
EST <> 'C'
```

COL-1	COL-2	COL-3
670	5	134



Agrupamientos

El SQL nos permite formar grupos de filas según un determinado criterio por intermedio del predicado **GROUP BY**.

Deseamos conocer el importe promedio de compras y el importe promedio de ventas para cada sector de acciones vendidas según los datos de nuestra tabla

CODAA	CODS	FECT	PCU	CANC	IMPC	PVU	CANV	IMPV	CC	CV	EST
10	10	20020424	10	1000	9880	12	1000	11860	120	140	V
50	20	20020428	15	500	7410	16	500	7890	90	110	V
10	10	20020428	11	1000	10860	13	1000	12830	140	170	V
70	10	20020428	8	3000	23730				270		C
35	30	20020428	12	1000	11820	14	1000	13780	180	220	V
45	30	20020428	9	1000	8860				140		C
10	10	20020505	11	1000	10860	13	1000	12830	140	170	V
20	20	20020505	6	2000	11870				130		C
70	10	20020510	9	1000	8920				80		C
20	20	20020510	6	1000	5935				65		C

```
SELECT CODAA , AVG (IMPC), AVG (IMPV) FROM MOVIA WHERE
EST = 'V' GROUP BY CODAA ORDER BY CODAA
```

Observaremos en pantalla la siguiente información

CODAA	Col-1	Col-2
10	10533,3333	12506,6667
35	11820	13780
50	7410	7890



Cuando agrupamos muchas veces no deseamos considerar determinados grupos por lo que para excluirlos utilizaremos el predicado **HAVING**

```
SELECT CODAA , AVG (IMPC), AVG (IMPV) FROM MOVIA WHERE  
EST = 'V' GROUP BY CODAA HAVING AVG (IMPC) > 8000 ORDER BY  
COD
```

El resultado sera

CODAA	Col-1	Col-2
10	10533,3333	12506,6667
35	11820	13780



Consultas sobre mas de una tabla

Hasta el momento hemos trabajado con tablas en forma independiente pero supongamos que queremos visualizar la información del movimiento de acciones (**MOVIA**) y que nos aparezca en pantalla el nombre de la empresa a la cual pertenece la acción (**ACCION**) . Para ello SQL nos brinda una manera de realizar un JOIN entre bases de datos relacionales.

```
SELECT CODAA, DESA , IMPC , IMPV , FROM ACCION, MOVIA  
WHERE CODAT = CODAA ORDER BY DESA
```

CODAA	DESA	IMPC	IMPV
45	BANKPYME MULTISALUD	8860	
20	BAWESTO SECTOR CONSUMO	11870	
20	BAWESTO SECTOR CONSUMO	5935	
35	BBL INVT. FOOD & DEVE	11820	13780
10	BBVA TELECOMUNICACIONES	9880	11860
10	BBVA TELECOMUNICACIONES	10860	12830
10	BBVA TELECOMUNICACIONES	10860	12830
70	BCH INTERNET	23730	
70	BCH INTERNET	8920	
50	FRAMLINGTON FINANCIAL	7410	7890



Actualización

En muchas ocasiones deberemos actualizar las tablas ya sea un registro o varios registros. El lenguaje SQL nos brinda la sentencia UPDATE la que permite actualizar el contenido de los campos de un registro existente.

CODAA	CODS	FECT	PCU	CANC	IMPC	PVU	CANV	IMPV	CC	CV	EST
10	10	20020424	10	1000	9880	12	1000	11860	120	140	V
50	20	20020428	15	500	7410	16	500	7890	90	110	V
10	10	20020428	11	1000	10860	13	1000	12830	140	170	V
70	10	20020428	8	3000	23730				270		C
35	30	20020428	12	1000	11820	14	1000	13780	180	220	V
45	30	20020428	9	1000	8860				140		C
10	10	20020505	11	1000	10860	13	1000	12830	140	170	V
20	20	20020505	6	2000	11870				130		C
70	10	20020510	9	1000	8920				80		C
20	20	20020510	6	1000	5935				65		C

Deseamos actualizar la información de la acciones código 20 grabando los datos correspondiente a importe unitario de venta , importe de venta, comisión de venta , cantidad vendida y estado igual a 'V'

**UPDATE MOVIA SET PVU=8 , CV = 150, IMPV= 15850, EST='V',
CANV=2000 WHERE CODAA=20 AND FECT = 20020505**

CODAA	CODS	FECT	PCU	CANC	IMPC	PVU	CANV	IMPV	CC	CV	EST
10	10	20020424	10	1000	9880	12	1000	11860	120	140	V
50	20	20020428	15	500	7410	16	500	7890	90	110	V
10	10	20020428	11	1000	10860	13	1000	12830	140	170	V
70	10	20020428	8	3000	23730				270		C
35	30	20020428	12	1000	11820	14	1000	13780	180	220	V
45	30	20020428	9	1000	8860				140		C
10	10	20020505	11	1000	10860	13	1000	12830	140	170	V
20	20	20020505	6	2000	11870	8	2000	15850	130	150	V
70	10	20020510	9	1000	8920				80		C
20	20	20020510	6	1000	5935				65		C



Ahora debemos actualizar el registro cuya fecha de transacción es 10/05/2002

**UPDATE MOVIA SET PVU=8 , CV = 80, IMPV= 7920, EST='V',
CANV=1000 WHERE CODAA=20 AND FECT = 20020515**

CODAA	COD	FECT	PCU	CANC	IMPC	PVU	CANV	IMPV	CC	CV	EST
	S										
10	10	20020424	10	1000	9880	12	1000	11860	120	140	V
50	20	20020428	15	500	7410	16	500	7890	90	110	V
10	10	20020428	11	1000	10860	13	1000	12830	140	170	V
70	10	20020428	8	3000	23730				270		C
35	30	20020428	12	1000	11820	14	1000	13780	180	220	V
45	30	20020428	9	1000	8860				140		C
10	10	20020505	11	1000	10860	13	1000	12830	140	170	V
20	20	20020505	6	2000	11870	8	2000	15850	130	150	V
70	10	20020510	9	1000	8920				80		C
20	20	20020510	6	1000	5935	8	1000	7920	65	80	V



Borrar

La operación de borrar uno , varios o todos los registros de una tabla esta contemplada por la sentencia DELETE.

Supongamos que queremos borrar de nuestra tabla ACCION aquellas cuyo código de país sea UK

CODA	CODS	DESA	PAIS
10	10	BBVA TELECOMUNICACIONES	ES
15	10	RENTA 4 TECNOLOGIA	ES
45	30	BANKPYME MULTISALUD	ES
70	10	BCH INTERNET	ES
25	10	CARMINAGNAC TECHNOLOGIES	FR
35	30	BBL INVT. FOOD & DEVE	BL
20	20	BAWESTO SECTOR CONSUMO	ES
30	20	FONDOESPACE FINANCIERO	ES
50	20	FRAMLINGTON FINANCIAL	UK
40	10	FINSBURY TECHNOLOGY TR	UK

DELETE FROM ACCION WHERE PAIS = 'UK'

El resultado luego de la ejecución de esta sentencia es la siguiente

CODA	CODS	DESA	PAIS
10	10	BBVA TELECOMUNICACIONES	ES
15	10	RENTA 4 TECNOLOGIA	ES
45	30	BANKPYME MULTISALUD	ES
70	10	BCH INTERNET	ES
25	10	CARMINAGNAC TECHNOLOGIES	FR
35	30	BBL INVT. FOOD & DEVE	BL
20	20	BAWESTO SECTOR CONSUMO	ES
30	20	FONDOESPACE FINANCIERO	ES



Ahora supongamos que nos piden borrar todos los movimientos de acciones vendidas cuya fecha de transacción es anterior a mayo del 2002.

CODAA	COD	FECT	PCU	CANC	IMPC	PVU	CANV	IMPV	CC	CV	EST
	SS										
10	10	20020424	10	1000	9880	12	1000	11860	120	140	V
50	20	20020428	15	500	7410	16	500	7890	90	110	V
10	10	20020428	11	1000	10860	13	1000	12830	140	170	V
70	10	20020428	8	3000	23730				270		C
35	30	20020428	12	1000	11820	14	1000	13780	180	220	V
45	30	20020428	9	1000	8860				140		C
10	10	20020505	11	1000	10860	13	1000	12830	140	170	V
20	20	20020505	6	2000	11870				130		C
70	10	20020510	9	1000	8920				80		C
20	20	20020510	6	1000	5935				65		C

DELETE FROM MOVIA WHERE EST = 'V' AND FECT < 20020501

El resultado que obtenemos es el siguiente

CODAA	COD	FECT	PCU	CANC	IMPC	PVU	CANV	IMPV	CC	CV	EST
	SS										
70	10	20020428	8	3000	23730				270		C
45	30	20020428	9	1000	8860				140		C
10	10	20020505	11	1000	10860	13	1000	12830	140	170	V
20	20	20020505	6	2000	11870				130		C
70	10	20020510	9	1000	8920				80		C
20	20	20020510	6	1000	5935				65		C



Insertar

La manera de insertar un registro en una tabla esta contemplada por la sentencia **INSERT INTO**

Nos había quedado la tabla **MOVIA** con los siguientes datos luego de realizar la ultima sentencia **DELETE**

CODAA	COD	FECT	PCU	CANC	IMPC	PVU	CANV	IMPV	CC	CV	EST
	SS										
70	10	20020428	8	3000	23730				270		C
45	30	20020428	9	1000	8860				140		C
10	10	20020505	11	1000	10860	13	1000	12830	140	170	V
20	20	20020505	6	2000	11870				130		C
70	10	20020510	9	1000	8920				80		C
20	20	20020510	6	1000	5935				65		C

Deseamos agregar los datos correspondientes a una compra de acciones del 10/05/2002.

INSERT INTO MOVIA VALUES (70, 20, 20020610, 10, 1000, 9800, 0, 0, 0, 200, 0, 'C')

En resultado es el siguiente:

CODAA	COD	FECT	PCU	CANC	IMPC	PVU	CANV	IMPV	CC	CV	EST
	S										
70	10	20020428	8	3000	23730				270		C
45	30	20020428	9	1000	8860				140		C
10	10	20020505	11	1000	10860	13	1000	12830	140	170	V
20	20	20020505	6	2000	11870				130		C
70	10	20020510	9	1000	8920				80		C
20	20	20020510	6	1000	5935				65		C
70	20	20020610	10	1000	9800				200		C