## 5 - Recuperar algunos campos (select)

Hemos aprendido cómo ver todos los registros de una tabla, empleando la instrucción "select".

La sintaxis básica y general es la siguiente:

```
select * from NOMBRETABLA;
```

El asterisco (\*) indica que se seleccionan todos los campos de la tabla.

Podemos especificar el nombre de los campos que queremos ver separándolos por comas:

```
select titulo, autor from libros;
```

La lista de campos luego del "select" selecciona los datos correspondientes a los campos nombrados. En el ejemplo anterior seleccionamos los campos "titulo" y "autor" de la tabla "libros", mostrando todos los registros. Los datos aparecen ordenados según la lista de selección, en dicha lista los nombres de los campos se separan con comas.

## Servidor de SQL Server instalado en forma local.

Ingresemos el siguiente lote de comandos en el SQL Server Management Studio:

```
-- borramos la tabla libros si ya existe
if object id('libros') is not null
 drop table libros;
-- creamos la tabla libros
create table libros(
 titulo varchar(40),
 autor varchar(30),
 editorial varchar(15),
 precio float,
 cantidad integer
);
-- llamamos al procedimiento almacenado sp columns para conocer los campos
-- de la tabla libros
exec sp columns libros;
-- insertamos varias filas en la tabla libros
insert into libros (titulo,autor,editorial,precio,cantidad)
  values ('El aleph', 'Borges', 'Emece', 25.50, 100);
insert into libros (titulo,autor,editorial,precio,cantidad)
 values ('Alicia en el pais de las maravillas', 'Lewis Carroll', 'Atlantida', 10, 200);
insert into libros (titulo,autor,editorial,precio,cantidad)
 values ('Matematica estas ahi', 'Paenza', 'Siglo XXI', 18.8, 200);
-- recuperamos todas los datos de la tabla libros
select * from libros;
```

```
-- recuperamos solo el titulo, autor y editorial de la tabla libros select titulo, autor, editorial from libros;
```

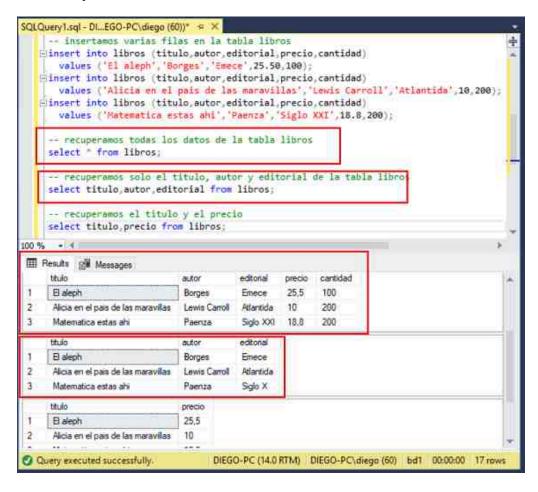
```
-- recuperamos el titulo y el precio
select titulo,precio from libros;
```

```
-- recuperamos la editorial y la cantidad
select editorial,cantidad from libros;
```

Para disponer un comentario y que no lo tenga en cuenta el servidor SQL Server debemos anteceder los caracteres --

Es bueno comentar los algoritmos que implementamos sobre todo aquellos que sean complejos de entender.

Cuando ejecutamos este lote de comandos SQL tenemos como resultado:



## Comando "go" del "SQL Server Management Studio"

Cuando tenemos un lote de comandos SQL y necesitamos particionarlo en distintos sublotes debemos utilizar el comando "go", no es un comando SQL sino un comando propio del "SQL Server Management Studio":

```
-- borramos la tabla libros si ya existe
if object_id('libros') is not null
  drop table libros;
```

-- creamos la tabla libros

```
create table libros(
  titulo varchar(40),
  autor varchar(30),
  editorial varchar(15),
  precio float,
  cantidad integer
);
go
-- llamamos al procedimiento almacenado sp columns para conocer los campos
-- de la tabla libros
exec sp columns libros;
-- insertamos varias filas en la tabla libros
insert into libros (titulo,autor,editorial,precio,cantidad)
  values ('El aleph', 'Borges', 'Emece', 25.50, 100);
insert into libros (titulo,autor,editorial,precio,cantidad)
  values ('Alicia en el pais de las maravillas', 'Lewis Carroll', 'Atlantida', 10, 200);
insert into libros (titulo,autor,editorial,precio,cantidad)
  values ('Matematica estas ahi', 'Paenza', 'Siglo XXI', 18.8, 200);
-- recuperamos todas los datos de la tabla libros
select * from libros;
-- recuperamos solo el titulo, autor y editorial de la tabla libros
select titulo, autor, editorial from libros;
-- recuperamos el titulo y el precio
select titulo, precio from libros;
-- recuperamos la editorial y la cantidad
select editorial, cantidad from libros;
```

No disponemos punto y coma después de "go". Logramos que primer se ejecuten en la primera conexión con el servidor SQL Server los dos primeros comandos y en la segunda conexión el resto de los comandos.