

2. CREAR TABLAS

Una base de datos almacena su información en tablas.

Una tabla es una estructura de datos que organiza los datos en columnas y filas; cada columna es un campo (o atributo) y cada fila, un registro. La intersección de una columna con una fila, contiene un dato específico, un solo valor.

Cada registro contiene un dato por cada columna de la tabla.

Cada campo (columna) debe tener un nombre. El nombre del campo hace referencia a la información que almacenará.

Cada campo (columna) también debe definir el tipo de dato que almacenará.

Las tablas forman parte de una base de datos.

Nosotros trabajaremos con la base de datos llamada bd1.

Para ver las tablas existentes creadas por los usuarios en una base de datos usamos el procedimiento almacenado "sp_tables @table_owner='dbo';":

```
exec sp_tables @table_owner='dbo';
```

El parámetro @table_owner='dbo' indica que solo muestre las tablas de usuarios y no las que crea el SQL Server para administración interna.

Finalizamos cada comando con un punto y coma.

Al crear una tabla debemos resolver qué campos (columnas) tendrá y que tipo de datos almacenarán cada uno de ellos, es decir, su estructura.

La sintaxis básica y general para crear una tabla es la siguiente:

```
create table NOMBRETABLA(  
    NOMBRECAMPO1 TIPODEDATO,  
    ...  
    NOMBRECAMPON TIPODEDATO  
);
```

La tabla debe ser definida con un nombre que la identifique y con el cual accederemos a ella.

Creamos una tabla llamada "usuarios" y entre paréntesis definimos los campos y sus tipos:

```
create table usuarios (  
    nombre varchar(30),  
    clave varchar(10)  
);
```

Cada campo con su tipo debe separarse con comas de los siguientes, excepto el último.

Cuando se crea una tabla debemos indicar su nombre y definir al menos un campo con su tipo de dato. En esta tabla "usuarios" definimos 2 campos:

- nombre: que contendrá una cadena de caracteres de 30 caracteres de longitud, que almacenará el nombre de usuario y
- clave: otra cadena de caracteres de 10 de longitud, que guardará la clave de cada usuario.

Cada usuario ocupará un registro de esta tabla, con su respectivo nombre y clave.

Para nombres de tablas, se puede utilizar cualquier caracter permitido para nombres de directorios, el primero debe ser un caracter alfabético y no puede contener espacios. La longitud máxima es de 128 caracteres.

Si intentamos crear una tabla con un nombre ya existente (existe otra tabla con ese nombre), mostrará un mensaje indicando que ya hay un objeto llamado 'usuarios' en la base de datos y la sentencia no se ejecutará. Esto es muy importante ya que cuando haga los ejercicios en este sitio puede haber otra persona que haya creado una tabla con el nombre que usted especifique.

Para ver la estructura de una tabla usamos el procedimiento almacenado "sp_columns" junto al nombre de la tabla:

```
exec sp_columns usuarios;
```

aparece mucha información que no analizaremos en detalle, como el nombre de la tabla, su propietario, los campos, el tipo de dato de cada campo, su longitud, etc.:

...COLUMN_NAME	TYPE_NAME	LENGHT
nombre	varchar	30
clave	varchar	10

Para eliminar una tabla usamos "drop table" junto al nombre de la tabla a eliminar:

```
drop table usuarios;
```

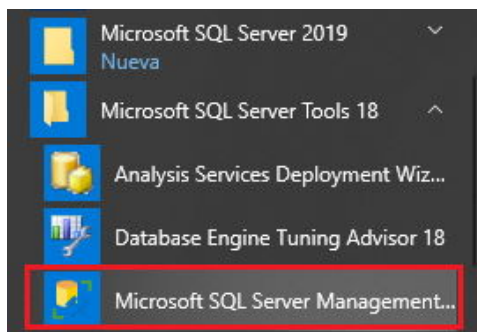
Si intentamos eliminar una tabla que no existe, aparece un mensaje de error indicando tal situación y la sentencia no se ejecuta. Para evitar este mensaje podemos agregar a la instrucción lo siguiente:

```
if object_id('usuarios') is not null  
drop table usuarios;
```

En la sentencia precedente especificamos que elimine la tabla "usuarios" si existe.

Servidor de SQL Server instalado en forma local.

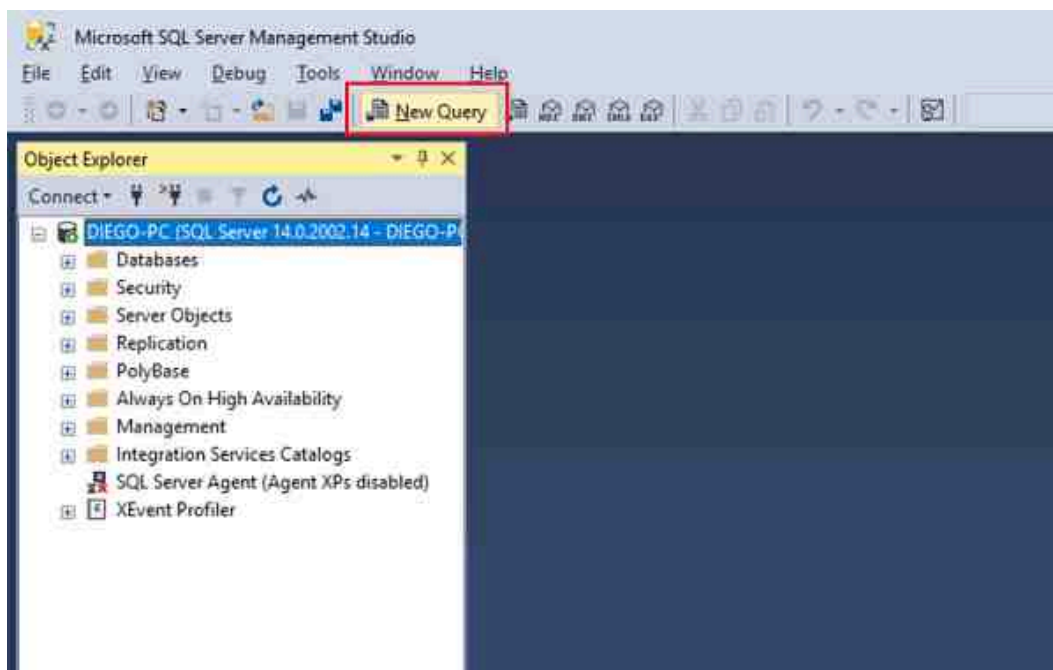
Si ha instalado el servidor de SQL Server en forma local en su computadora debe ingresar al SQL Server Management Studio:



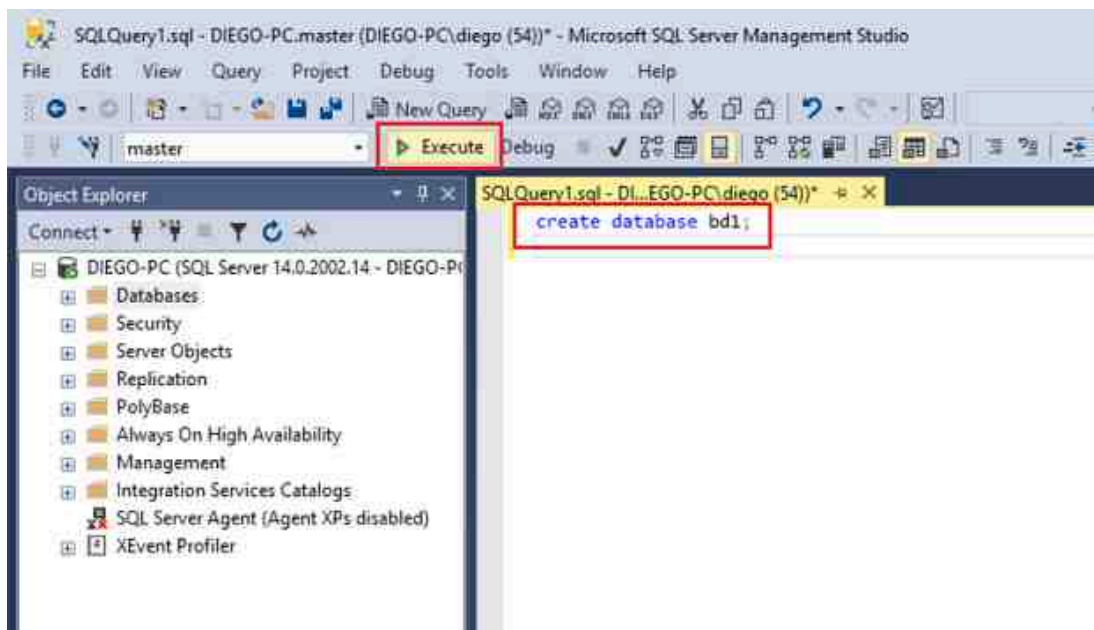
Debemos crear por única vez la base de datos donde haremos todos los ejercicios de este curso de SQL Server, para esto utilizamos el comando SQL:

```
create database bd1;
```

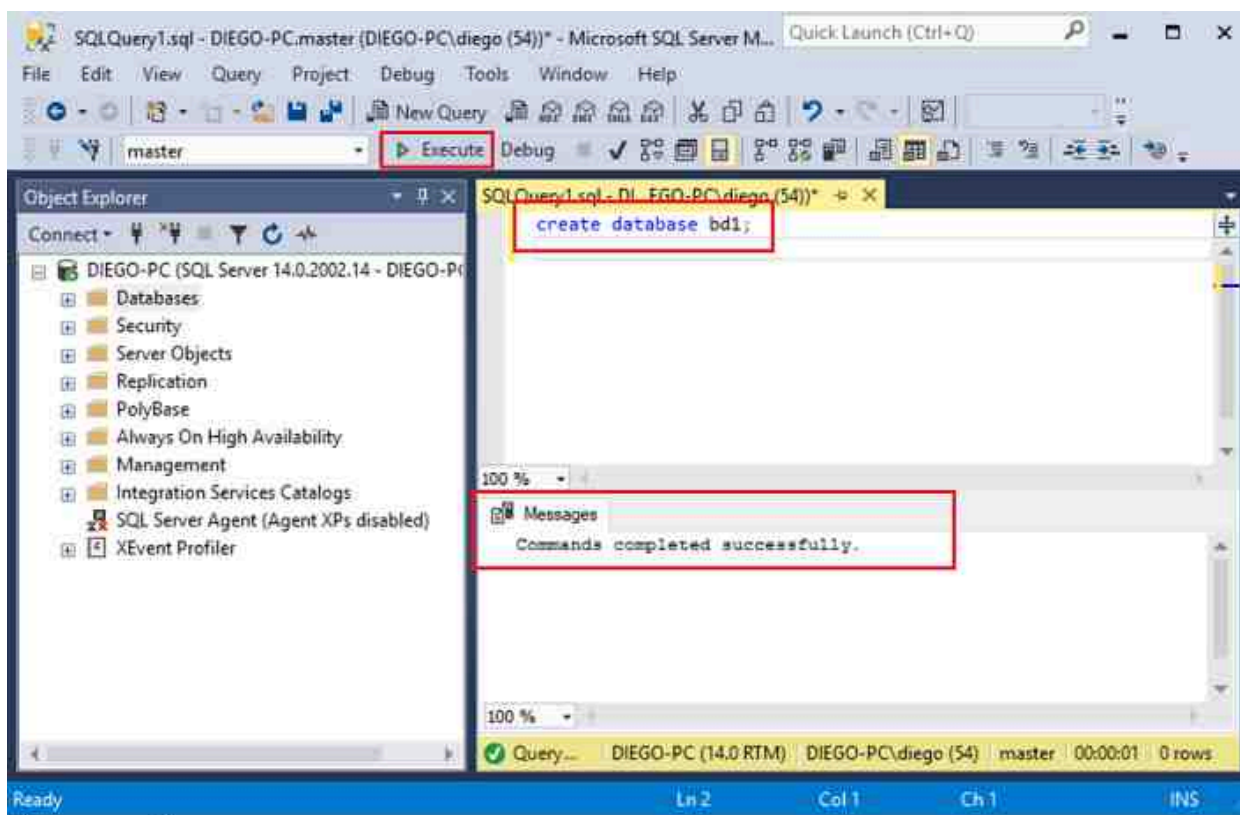
Para poder ejecutar el comando 'create database' debemos abrir una ventana mediante la opción "New Query":



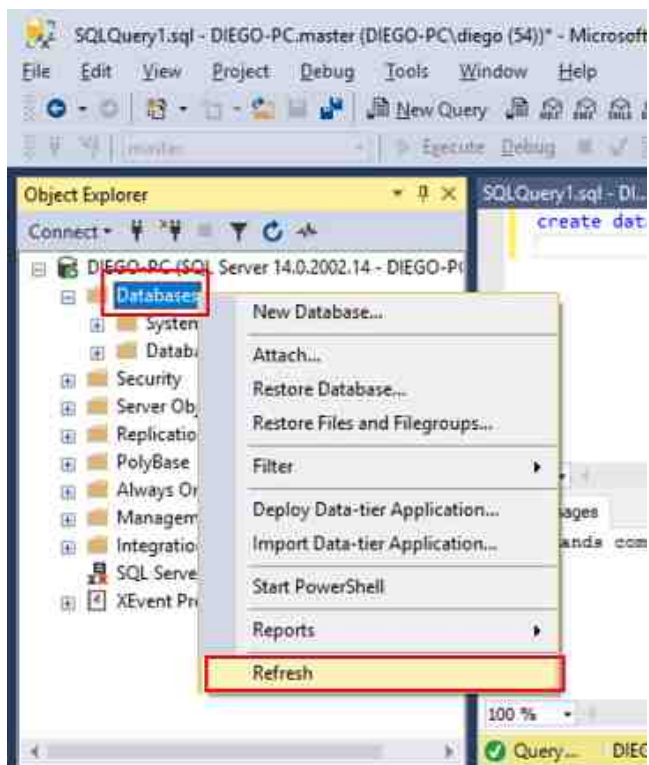
Ahora si creamos la base de datos llamada 'bd1':



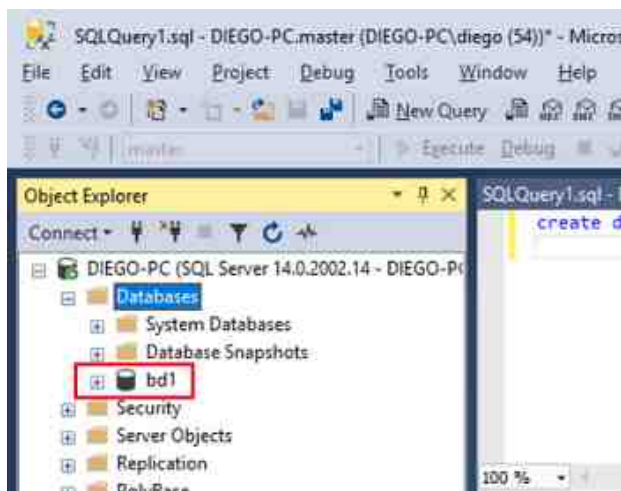
Para que se ejecute el comando 'create database' que acabamos de escribir debemos presionar el botón "Execute", en el caso que no haya errores se nos informará que la base de datos 'bd1' se ha creado en forma correcta:



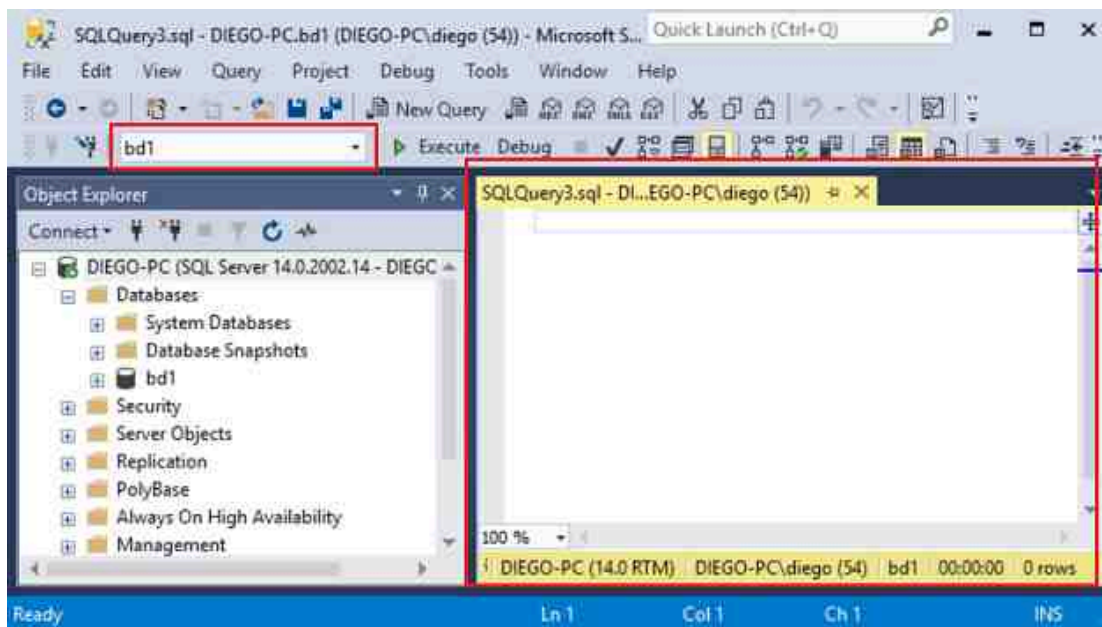
En el lado izquierdo del SSMS aparece el 'Object Explorer' donde podemos ver todas las bases de datos existentes en nuestro servidor local de SQL Server, para que se actualice debemos presionar el botón derecho del mouse sobre el texto 'Databases' y seleccionar la opción 'Refresh'



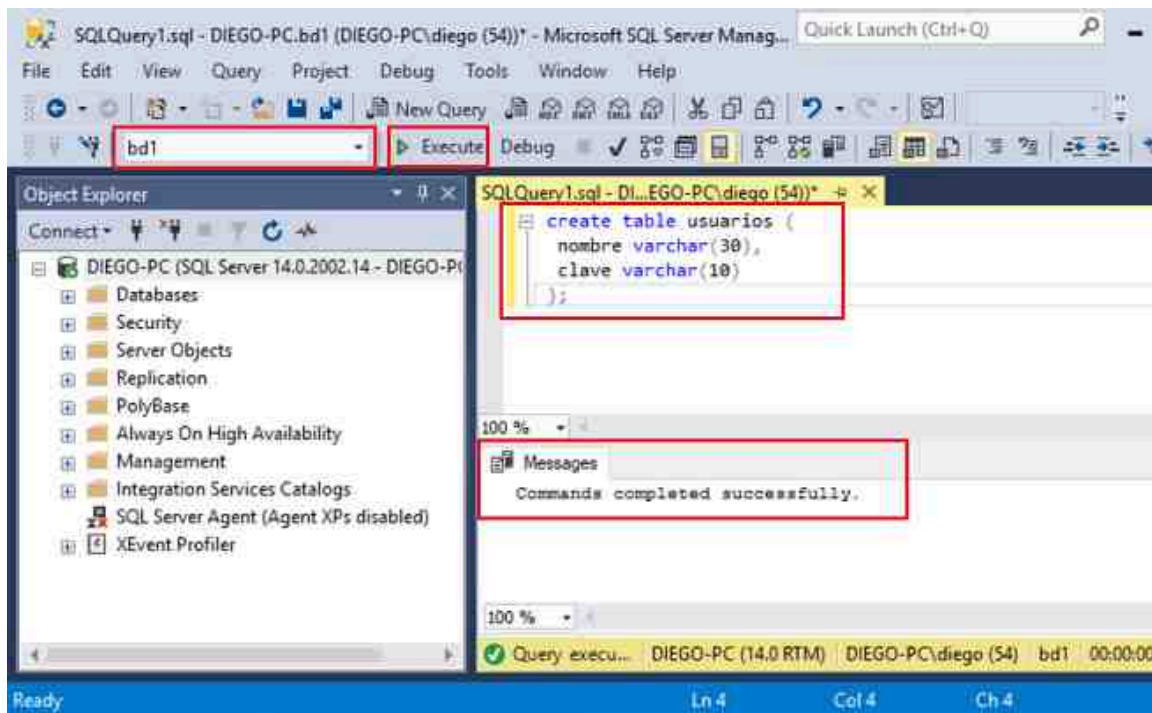
Luego de refrescar la opción 'Databases' podemos ver que aparece la base de datos 'bd1' que se ha creado en nuestro servidor de base de datos de SQL Server:



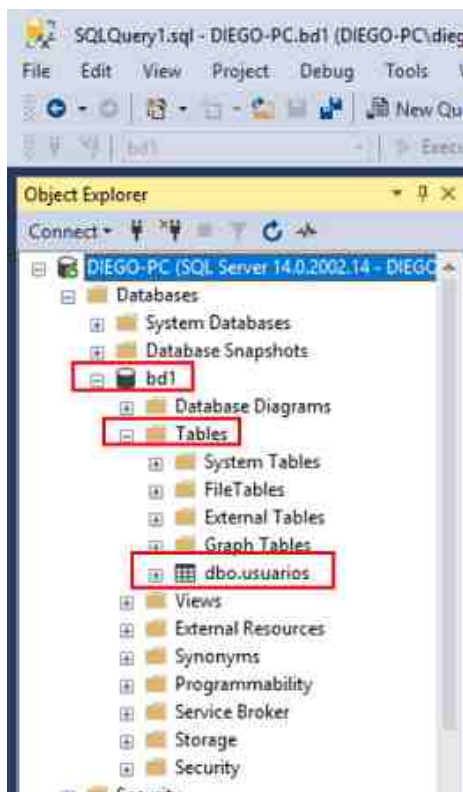
Debemos seleccionar esta base de datos 'bd1' para poder ejecutar comandos SQL que afecten dicha base de datos. Cerramos la ventana Query donde ejecutamos el create database y nuevamente elijamos la opción 'New Query', ahora debemos seleccionar la base de datos 'bd1':



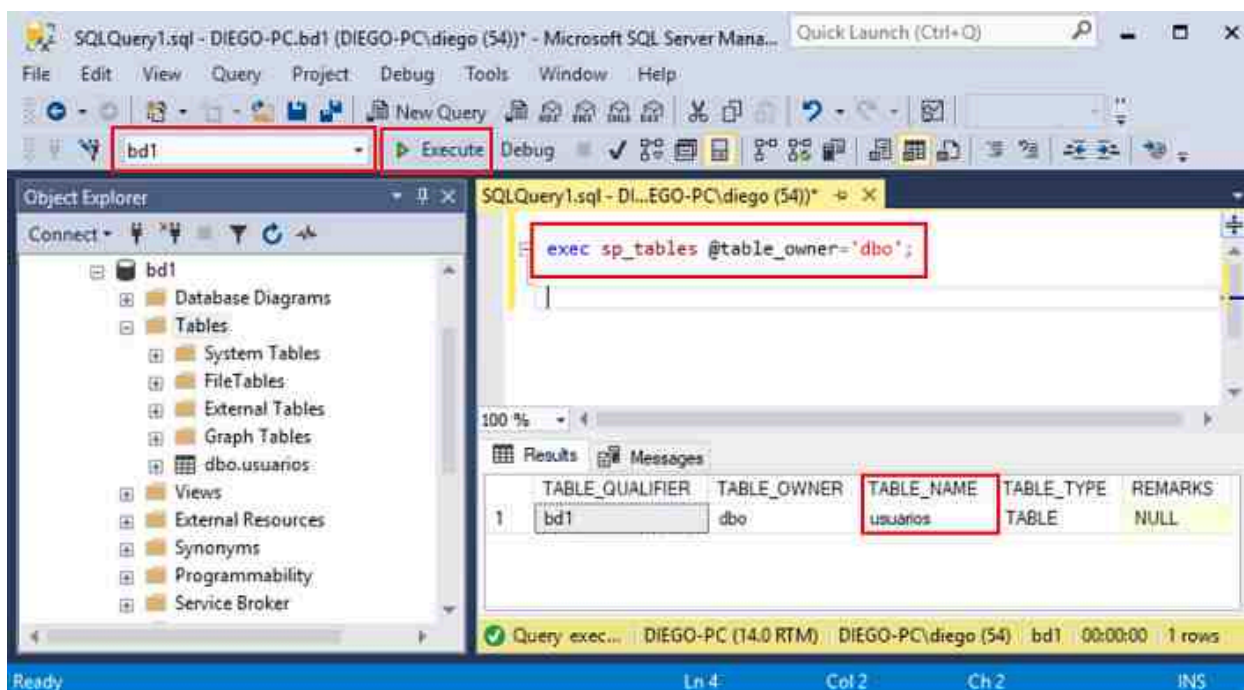
Creemos ahora la tabla 'usuarios' en la base de datos 'bd1':



Si refrescamos el "Object Explorer" podemos ver ahora la base de datos 'bd1' tiene una tabla llamada 'usuarios':



Probemos ahora ejecutar el procedimiento almacenado 'sp_tables' que nos muestra todas las tablas que hemos creado (el resultado es idéntico al que nos muestra el Object Explorer):



El programa SQL Server Management Studio nos ayuda en muchas situaciones para evitar tener que llamar a comandos SQL o procedimientos almacenados como sp_tables (es más fácil ver el Object Explorer directamente para conocer las tablas contenidas en una base de datos)

Mediante el SQL Server Management Studio podemos ejecutar un lote de comandos SQL de una sola vez, probemos de copiar el siguiente bloque de comandos SQL y ejecutarlos:

```
if object_id('usuarios') is not null
    drop table usuarios;
```

```

create table usuarios (
    nombre varchar(30),
    clave varchar(10)
);

exec sp_tables @table_owner='dbo';

exec sp_columns usuarios;

drop table usuarios;

exec sp_tables @table_owner='dbo';

```

Podemos ver que todos los comandos SQL se ejecutan en forma secuencial:

The screenshot shows the Microsoft SQL Server Enterprise Manager interface. The left pane displays the Object Explorer with the database 'bd1' selected. The right pane shows a SQL query script with the following commands:

```

if object_id('usuarios') is not null
    drop table usuarios;

create table usuarios (
    nombre varchar(30),
    clave varchar(10)
);

exec sp_tables @table_owner='dbo';

exec sp_columns usuarios;

drop table usuarios;

exec sp_tables @table_owner='dbo';

```

The Results pane shows the output of the queries. The first query, 'exec sp_tables @table_owner='dbo'', returns a single row:

TABLE_QUALIFIER	TABLE_OWNER	TABLE_NAME	TABLE_TYPE	REMARKS
bd1	dbo	usuarios	TABLE	NULL

The second query, 'exec sp_columns usuarios;', returns two rows:

TABLE_QUALIFIER	TABLE_OWNER	TABLE_NAME	COLUMN_NAME	DATA_TYPE
bd1	dbo	usuarios	nombre	12
bd1	dbo	usuarios	clave	12

The third query, 'drop table usuarios;', returns no results. The fourth query, 'exec sp_tables @table_owner='dbo'', returns no results.