



UNIVERSIDAD
DE MÁLAGA

Departamento de Lenguajes
y Ciencias de la Computación
UNIVERSIDAD DE MÁLAGA

Gestión de la Información

Práctica 1

SQL - Server

SQL Server 2008 es la base datos cliente-servidor de Microsoft.

Su lenguaje de programación primario es T-SQL.

SQL Server es más que un simple motor de base de datos ya que:

- Conectarse a SQL Server es sencillo ya que dispone de un amplio abanico de opciones de conexión y una amplia variedad de tecnologías para importar y exportar datos, tales como Tabular Data Stream (TDS), XML, Integration Services, bulk copy, SQL Native Connectivity, OLE DB, ODBC y consultas distribuidas con un coordinador de transacciones distribuidas.
- El motor trabaja bien con diferentes datos (XML, datos vectoriales, palabras dentro de textos y datos binarios grandes “blob data”).
- SQL Server tiene un amplio conjunto de componentes y herramientas para trabajar con datos multidimensionales y hacer minería de datos.
- SQL Server incluye un amplio conjunto de herramientas para la realización de informes.
- SQL Server ofrece un impresionante nivel de detalle de diagnóstico con el estudio de rendimiento “Performance Studio”, “SQL Trace/Profiler”, y el manejo de vistas y funciones de la base de datos..
- SQL Server incluye diferentes opciones para una alta disponibilidad con diferentes grados de latencia, rendimiento, número de several options for high availability with varying degrees of latency, performance, número de nueves, distancia física y sincronización.
- SQL Server puede ser manejado de forma declarativa mediante Policy-Based Management.
- SQL Server’s Management Studio es una interfaz de usuario Madura y útil tanto para el desarrollador como para el administrador de bases de datos.
- SQL Server está disponible en diferentes ediciones tanto para 32-bit como para 64-bit, siendo la versión Express, totalmente gratuita.

Vamos a crear una base de datos en SQL-Server, con un conjunto de tablas y relaciones. Para ello podemos usar:

- SQL Server Management Studio
- Sqlcmd
- Microsoft Visual Studio 2010

SQL Server Management Studio

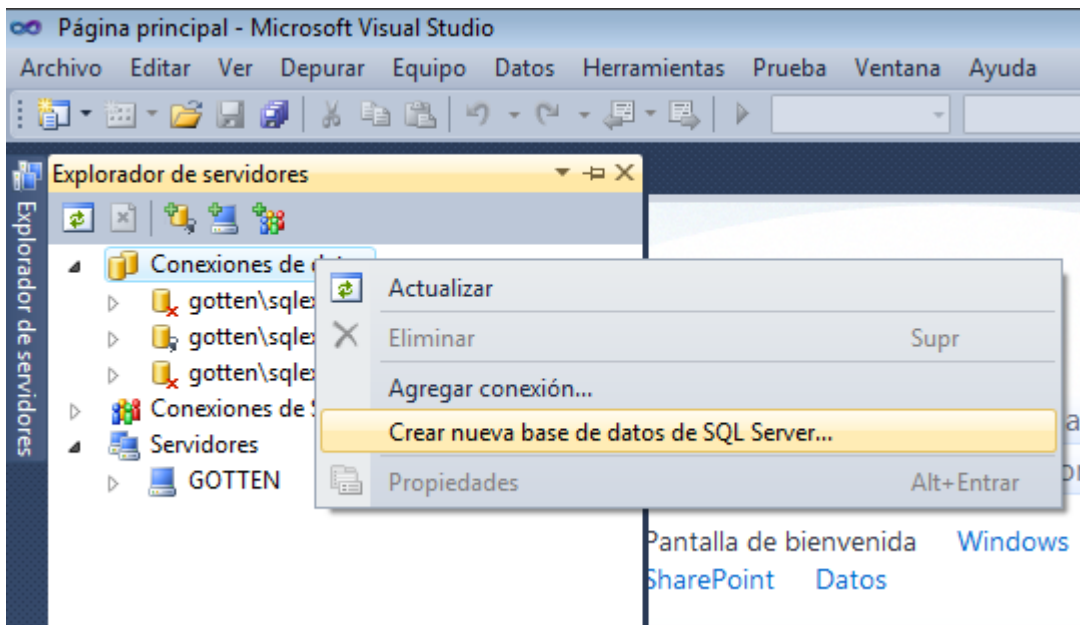
Hay 2 formas de conectarse al Servidor de SQL-Server:

- Con autenticación de SQL-Server
- con autenticación de Windows (usaremos esta última).

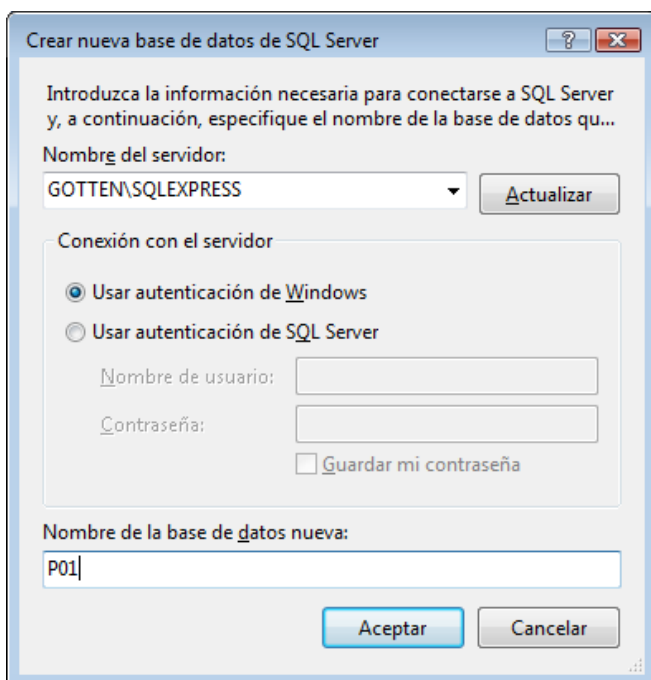


Microsoft Visual Studio 2010

Para crear una base de datos nueva en SQL-Server usando Microsoft Visual Studio 2010, debemos ir al explorador de servidores y con el botón derecho seleccionar “Crear nueva base de datos de SQL Server”.



Posteriormente introduciremos el nombre del servidor y de la base de datos



SQLCMD

Para usar el intérprete de comandos de SQL debemos ejecutar `Sqlcmd -S Nombre_instancia`. Donde el nombre de la instancia es la máquina\SQLEXPRESS :

Ejemplo para crear una base de datos en sqlcmd:

```
C:\Users\pastrana>sqlcmd -S GOTTEN\SQLEXPRESS
```

```
1> create database P01
```

```
2> GO
```

```
1>
```

Ejercicio 1. Crear la Base de datos P01 usando sqlcmd

Creación de Tablas

Para cada columna de una tabla hay que expresar su nombre, su tipo y si permite valor NULL.

TIPOS DE DATOS

TIPO	TAMAÑO	DESCRIPCIÓN
Enteros (números enteros positivos y negativos)		
bigint	8 bytes	De - 9223372036854775808 a +9223372036854775807
int	4 bytes	De - 2,147,483,648 a +2,147,483,647
smallint	2 bytes	De - 32,768 a +32,767
tinyint	1 byte	De 0 a 255
bit	1 bit	0 ó 1
Números Exactos		
decimal, numeric	5 - 17 bytes	De (- 10 ³⁸ +1) a (10 ³⁸ -1) Se declaran con precisión (nº total de dígitos) y escala (nº total de decimales).
money	8 bytes	De - 922,337,203,685,477.5808 a +922,337,203,685,477.5807 con unidad monetaria
smallmoney	4 bytes	De - 214,748.3648 a +214,748.3647 con unidad monetaria
Números Aproximados		
float	4 bytes or 8 bytes	De -3.40E-38 a 3.40E+38 De -1.79E+308 a 1.79E+308
real	4 bytes	4 - byte float.
Fecha y Hora		
datetime	8 bytes	Desde 01/01/1753 a 31/12/9999 con una precisión de 3.33 ms
datetime2	8 bytes	Desde 01/01/0001 a 31/12/9999 con una precisión de 100 ns
smalldatetime	4 bytes	Desde 01/01/1900 a 31/12/2079 con una precisión de 1 minuto
date	3 bytes	Desde 01/01/0001 a 31/12/9999 con una precisión de 1 día
time	5 bytes	Horas con una precisión de 100ns
datetimeoffset	10 bytes	Desde 01/01/0001 a 31/12/9999 la precisión en función del HW
Caracteres		
char	1 byte por carácter	Entre 1 y 8000 caracteres
Varchar,	1 byte por carácter	Idéntico al char pero de longitud variable.
text	1 byte por carácter	Similar a varchar pero admite hasta 2 billones de caracteres.
nchar	2 bytes por carácter	Equivalente al char
nvarchar	2 bytes por carácter	Equivalente a varchar
ntext	2 bytes por carácter	Equivalente a text
Tipos Binarios		
binary	1 - 8000 bytes	Longitud fija
varbinary	hasta 2,147,483,647 bytes	Longitud Variable
image	hasta 2,147,483,647 bytes	similar a varbinary

Ejercicio 2. Crear una tabla usando sqlcmd llamada estudiante con 3 atributos: ID(int), nombre(varchar(40)) y nota (int).

```
1> USE P01
2> go
1> create table estudiante( ID int, nombre varchar(40), nota int )
2> go
```

A continuación vamos a cargar el Visual Studio 2010 y seguiremos trabajando a partir de allí. Para ello habrá de crear una nueva conexión de base de datos SQLSERVER.

Ejercicio 3. Conectarse a la base de datos y modificar la tabla anterior para que ID sea la clave de la tabla.

Ejercicio 4. Crear el diagrama de bases de datos, útil para trabajar y programar la BD y las aplicaciones. Poner vista de tablas estándar.

Ejercicio 5. Sobre el diagrama crear la tabla pruebas con 4 atributos: ID_PRUEBA(int), ID_ALUMNO (int), nombre_prueba(varchar(40)) y nota_prueba (real). Establecer ID_PRUEBA como clave de la tabla. Hacer que ID_PRUEBA sea autoincremental de 1 en 1 unidad. Usar "Especificación de Identidad".

Ejercicio 6. Sobre el diagrama establecer la relación entre ambas tablas de tal forma que 1 alumno puede tener muchas pruebas. Exigir integridad referencial de inserción, actualización y borrado. Usar "Especificación de INSERT y UPDATE"

Ejercicio 7. Utilizando las instrucciones de manipulación básicas insertar datos en ambas tablas, modificarlos y borrarlos. Tratar de violar las restricciones impuestas como insertar otro alumno con el mismo ID, o asignar una prueba a un ID_ALUMNO que no se encuentra en la tabla alumnos, al insertar no rellenar un campo con restricción NOT NULL, etc. Por ejemplo:

```
insert into estudiante values(null,"enull",null)
insert into estudiante values(1,"e1",null)
insert into estudiante values(1,"e11",null)
insert into estudiante values(2,"e2",null)
insert into Pruebas (ID_ALUMNO ,nombre_prueba, nota_prueba) values(1,"p1",7)
insert into Pruebas (ID_ALUMNO ,nombre_prueba, nota_prueba) values(7,"p1",null)
insert into Pruebas (ID_ALUMNO ,nombre_prueba, nota_prueba) values(1,"p2",9)
select * from Estudiante
insert into estudiante values(3,"e3",null)
insert into Pruebas (ID_ALUMNO,nombre_prueba,nota_prueba) values(3,"pkk",null)
select * from Estudiante
select * from Pruebas
delete from estudiante where ID=3
select * from Estudiante
select * from Pruebas
```

Ejercicio 8. Consultas sencillas:

- Cree una consulta que devuelva los alumnos cuyo nombre incluya la letra "e".
- Cree una consulta que devuelva las pruebas con una nota mayor o igual a 5.
- Cree una consulta que devuelva las pruebas con una nota entre 7 y 9.

Ejercicio 9. Consultas de 2 tablas:

- Cree una consulta que devuelva el nombre del alumno, el nombre de la prueba y la nota de la prueba de todas las pruebas del alumno con ID 1.
- Cree una consulta que devuelva el nombre del alumno, el nombre de la prueba y la nota de la prueba de todas las pruebas con nota mayor o igual a 5.

Ejercicio 10. Actualizaciones y Borrados:

- Actualice la nota de todas las pruebas sumándoles 2 puntos.
- Actualice la nota de la prueba a 10 para todas las pruebas cuya nota sea mayor que 10.
- Borre de la tabla alumnos el alumno con ID=1 y luego muestre las pruebas del alumno con ID_ALUMNO=1