

# Operaciones básicas de manipulación de datos en SQL

El lenguaje de manipulación de datos (Data Manipulation Language, o DML) es un lenguaje que se utiliza para la definición del nivel externo y las operaciones de manipulación de la información.

La manipulación de los datos consiste en la realización de operaciones de inserción, borrado, modificación y consulta de la información almacenada en la base de datos. La inserción y el borrado son el resultado de añadir nueva información a la ya que se encontraba almacenada o eliminarla de nuestra base de datos, tomando en cuenta las restricciones marcadas por el DDL y las relaciones entre la nueva información y la antigua. La modificación nos permite alterar esta información, y la consulta nos permite el acceso a la información almacenada en la base de datos siguiendo criterios específicos.

En general a las operaciones básicas de manipulación de datos que podemos realizar con SQL se les denomina operaciones CRUD (de Create, Read, Update and Delete, o sea, Crear, Leer, Actualizar y Borrar, sería CLAB en español, pero no se usa). Lo verás utilizado de esta manera en muchos sitios, así que apréndete ese acrónimo.

Hay cuatro instrucciones para realizar estas tareas:

- **INSERT:** Inserta filas en una tabla. Se corresponde con la “C” de CRUD.
- **SELECT:** muestra información sobre los datos almacenados en la base de datos. Dicha información puede pertenecer a una o varias tablas. Es la “R”.
- **UPDATE:** Actualiza información de una tabla. Es, obviamente, la “U”.
- **DELETE:** Borra filas de una tabla. Se corresponde con la “D”.

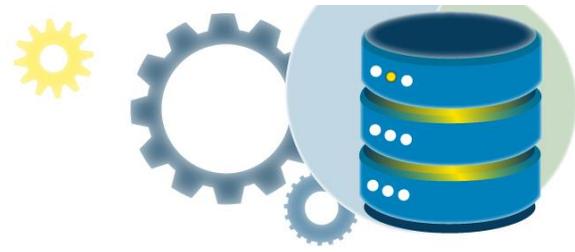
En este documento usted encontrará:

**INSERTAR DATOS: SENTENCIA INSERT**

**ACTUALIZAR DATOS: SENTENCIA UPDATE**

**BORRAR DATOS: SENTENCIA DELETE**

**CONSULTAR DATOS: SENTENCIA SELECT**



## INSERTAR DATOS: SENTENCIA INSERT

La sentencia **INSERT INTO** se utiliza para insertar nuevos registros en una tabla.

### *Sintaxis INSERT INTO*

Es posible escribir la sentencia **INSERT INTO** de dos formas:

1. Especifique los nombres de las columnas y los valores que se insertarán:

```
INSERT INTO nombre_tabla (column1, column2, column3, ...)  
VALUES (value1, value2, value3, ...);
```

2. Si agrega valores para todas las columnas de la tabla, no es necesario que especifique los nombres de las columnas en la consulta SQL. Sin embargo, asegúrese de que el orden de los valores esté en el mismo orden que las columnas de la tabla. Aquí, la sintaxis **INSERT INTO** sería la siguiente:

```
INSERT INTO nombre_tabla  
VALUES (value1, value2, value3, ...);
```

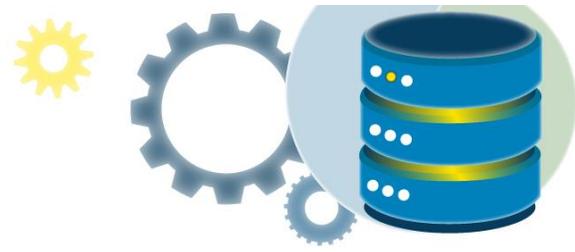
### *Base de datos de demostración*

A continuación, se muestra una selección de la tabla "Clientes" en la base de datos de muestra de Northwind:

CustomerID	CustomerName	ContactName	Address	City	PostalCode	Country
89	White Clover Markets	Karl Jablonski	305 - 14th Ave. S. Suite 3B	Seattle	98128	USA
90	Wilman Kala	Matti Karttunen	Keskuskatu 45	Helsinki	21240	Finland
91	Wolski	Zbyszek	ul. Filtrowa 68	Walla	01-012	Poland

La siguiente instrucción SQL inserta un nuevo registro en la tabla "Clientes":

```
INSERT INTO Clientes (CustomerName, ContactName, Address, City,  
PostalCode, Country)
```



```
VALUES('Cardinal', 'Tom B. Erichsen', 'Skagen  
21', 'Stavanger', '4006', 'Norway');
```

La selección de la tabla "Clientes" ahora se verá así:

CustomerID	CustomerName	ContactName	Address	City	PostalCode	Country
89	White Clover Markets	Karl Jablonski	305 - 14th Ave. S. Suite 3B	Seattle	98128	USA
90	Wilman Kala	Matti Karttunen	Keskuskatu 45	Helsinki	21240	Finland
91	Wolski	Zbyszek	ul. Filtrowa 68	Walla	01-012	Poland
92	Cardinal	Tom B. Erichsen	Skagen 21	Stavanger	4006	Norway

¿Notó que no insertamos ningún número en el campo CustomerID?

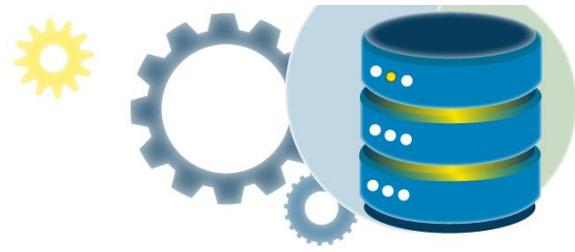
La columna CustomerID es un campo de incremento automático y se generará automáticamente cuando se inserte un nuevo registro en la tabla.

*Insertar datos solo en columnas especificadas*

También es posible insertar solo datos en columnas específicas.

La siguiente instrucción SQL insertará un nuevo registro, pero solo insertará datos en las columnas "CustomerName", "City" y "Country" (CustomerID se actualizará automáticamente):

```
INSERT INTO Clientes (CustomerName, City, Country)  
VALUES ('Cardinal', 'Stavanger', 'Norway');
```



La selección de la tabla "Clientes" ahora se verá así:

CustomerID	CustomerName	ContactName	Address	City	PostalCode	Country
89	White Clover Markets	Karl Jablonski	305 - 14th Ave. S. Suite 3B	Seattle	98128	USA
90	Wilman Kala	Matti Karttunen	Keskuskatu 45	Helsinki	21240	Finland
91	Wolski	Zbyszek	ul. Filtrowa 68	Walla	01-012	Poland
92	Cardinal	null	null	Stavanger	null	Norway

## ACTUALIZAR DATOS: SENTENCIA UPDATE

La sentencia **UPDATE** se utiliza para modificar los registros existentes en una tabla.

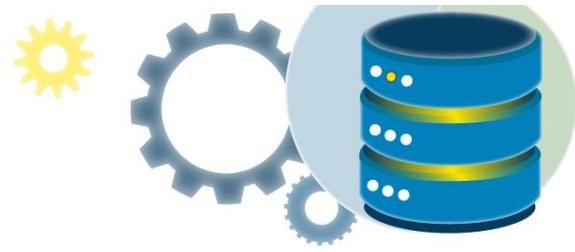
### *Sintaxis UPDATE INTO*

```
UPDATE nombre_tabla  
SET column1 = value1, column2 = value2, ...  
WHERE condition;
```

**Nota:** ¡Tenga cuidado al actualizar registros en una tabla! Note la cláusula **WHERE** en la sentencia **UPDATE**. La cláusula **WHERE** especifica qué registros deben actualizarse. Si omite la cláusula **WHERE**, se actualizarán todos los registros de la tabla.

### *Base de datos de demostración*

A continuación, se muestra una selección de la tabla "Clientes" en la base de datos de muestra de Northwind:



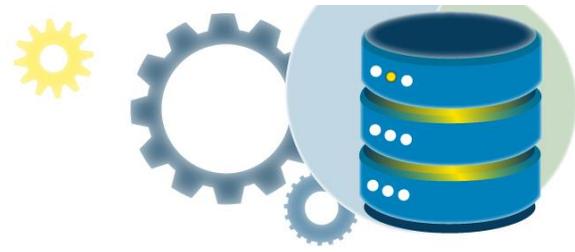
CustomerID	CustomerName	ContactName	Address	City	PostalCode	Country
1	Alfreds Futterkiste	Maria Anders	Obere Str. 57	Berlin	12209	Germany
2	Ana Trujillo Emparedados y helados	Ana Trujillo	Avda. de la Constitución 2222	México D.F.	05021	Mexico
3	Antonio Moreno Taquería	Antonio Moreno	Mataderos 2312	México D.F.	05023	Mexico
4	Around the Horn	Thomas Hardy	120 Hanover Sq.	London	WA1 1DP	UK
5	Berglunds snabbköp	Christina Berglund	Berguvsvägen 8	Luleå	S-958 22	Sweden

La siguiente declaración SQL actualiza el primer cliente (CustomerID = 1) con una nueva persona de contacto *y* una nueva ciudad.

```
UPDATE Clientes
SET ContactName = 'Alfred Schmidt', City= 'Frankfurt'
WHERE CustomerID = 1;
```

La selección de la tabla "Clientes" ahora se verá así:

CustomerID	CustomerName	ContactName	Address	City	PostalCode	Country
1	Alfreds Futterkiste	Alfred Schmidt	Obere Str. 57	Frankfurt	12209	Germany
2	Ana Trujillo Emparedados y helados	Ana Trujillo	Avda. de la Constitución 2222	México D.F.	05021	Mexico
3	Antonio Moreno Taquería	Antonio Moreno	Mataderos 2312	México D.F.	05023	Mexico
4	Around the Horn	Thomas Hardy	120 Hanover Sq.	London	WA1 1DP	UK
5	Berglunds snabbköp	Christina Berglund	Berguvsvägen 8	Luleå	S-958 22	Sweden



### Actualizar varios registros

La cláusula **WHERE** determinará cuántos registros se actualizarán. La siguiente instrucción SQL actualizará el ContactName a “Juan” para todos los registros donde el país es “México”:

```
UPDATE Clientes  
SET ContactName='Juan'  
WHERE Country='Mexico';
```

La tabla Clientes ahora se verá así:

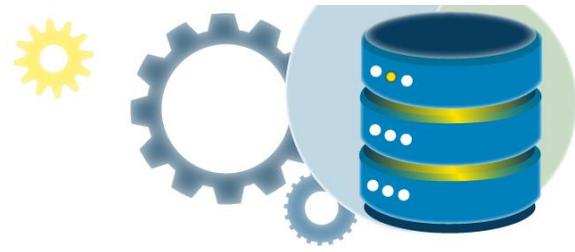
CustomerID	CustomerName	ContactName	Address	City	PostalCode	Country
1	Alfreds Futterkiste	Alfred Schmidt	Obere Str. 57	Frankfurt	12209	Germany
2	Ana Trujillo Emparedados y helados	Juan	Avda. de la Constitución 2222	México D.F.	05021	Mexico
3	Antonio Moreno Taquería	Juan	Mataderos 2312	México D.F.	05023	Mexico
4	Around the Horn	Thomas Hardy	120 Hanover Sq.	London	WA1 1DP	UK
5	Berglunds snabbköp	Christina Berglund	Berguvsvägen 8	Luleå	S-958 22	Sweden

### BORRAR DATOS: SENTENCIA DELETE

La sentencia **DELETE** se utiliza para eliminar los registros existentes en una tabla.

#### Sintaxis DELETE

```
DELETE FROM nombre_tabla WHERE condition;
```



**Nota:** ¡Tenga cuidado al eliminar registros en una tabla! Note la cláusula **WHERE** en la sentencia **DELETE**. La cláusula **WHERE** especifica qué registros deben eliminarse. Si omite la cláusula **WHERE**, se eliminarán todos los registros de la tabla.

### *Base de datos de demostración*

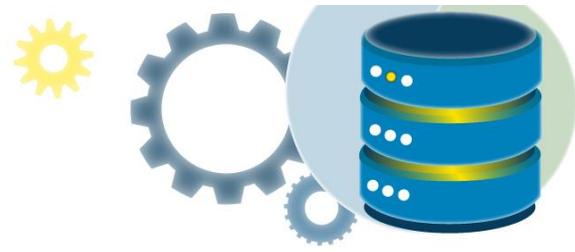
A continuación, se muestra una selección de la tabla "Clientes" en la base de datos de muestra de Northwind:

CustomerID	CustomerName	ContactName	Address	City	PostalCode	Country
1	Alfreds Futterkiste	Maria Anders	Obere Str. 57	Berlin	12209	Germany
2	Ana Trujillo Emparedados y helados	Ana Trujillo	Avda. de la Constitución 2222	México D.F.	05021	Mexico
3	Antonio Moreno Taquería	Antonio Moreno	Mataderos 2312	México D.F.	05023	Mexico
4	Around the Horn	Thomas Hardy	120 Hanover Sq.	London	WA1 1DP	UK
5	Berglunds snabbköp	Christina Berglund	Berguvsvägen 8	Luleå	S-958 22	Sweden

La siguiente instrucción SQL elimina el cliente "Alfreds Futterkiste" de la tabla "Clientes":

```
DELETE FROM Clientes WHERE CustomerName='Alfreds Futterkiste';
```

CustomerID	CustomerName	ContactName	Address	City	PostalCode	Country
2	Ana Trujillo Emparedados y helados	Ana Trujillo	Avda. de la Constitución 2222	México D.F.	05021	Mexico
3	Antonio Moreno Taquería	Antonio Moreno	Mataderos 2312	México D.F.	05023	Mexico
4	Around the Horn	Thomas Hardy	120 Hanover Sq.	London	WA1 1DP	UK
5	Berglunds snabbköp	Christina Berglund	Berguvsvägen 8	Luleå	S-958 22	Sweden



### *Eliminar todos los registros*

Es posible eliminar todas las filas de una tabla sin eliminar la tabla. Esto significa que la estructura de la tabla, los atributos y los índices estarán intactos:

```
DELETE FROM nombre_tabla;
```

La siguiente instrucción SQL elimina todas las filas de la tabla "Clientes", sin eliminar la tabla:

```
DELETE FROM Clientes;
```

### SENTENCIA TRUNCATE TABLE

Quita todas las filas de una tabla o las particiones especificadas de una tabla, sin registrar las eliminaciones individuales de filas. **TRUNCATE TABLE** es similar a la instrucción **DELETE** sin una cláusula **WHERE**; no obstante, **TRUNCATE TABLE** es más rápida y utiliza menos recursos de registros de transacciones y de sistema.

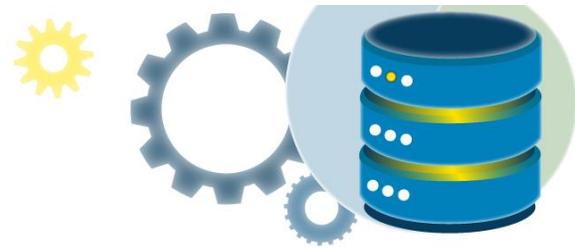
```
TRUNCATE TABLE nombre_tabla;
```

### CONSULTAR DATOS: SENTENCIA SELECT

La sentencia **SELECT** se utiliza para seleccionar datos de una base de datos. Los datos devueltos se almacenan en una tabla de resultados, denominada conjunto de resultados.

### *Sintaxis SELECT*

```
SELECT {*/campos} FROM nombre_tabla {WHERE condition};
```



La siguiente instrucción SQL selecciona las columnas "CustomerName" y "City" de la tabla "Clientes":

```
SELECT CustomerName, City FROM Clientes;
```

Para seleccionar todos los registros de una tabla en lugar de colocar los nombres de los campos, digitamos el símbolo \*. Por ejemplo, si deseamos obtener todos los registros de la tabla Clientes, usamos la siguiente instrucción:

```
SELECT * FROM Clientes;
```

En los próximos recursos didácticos, veremos más detalles de la sentencia **SELECT**, como realizar consultas simples y complejas.

## Ejercicio propuesto

En continuación con el ejercicio de la base de datos hospital, ingrese los siguientes datos:

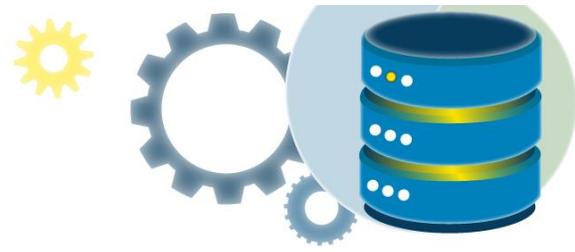
### Tabla Especialidades

- Maternidad
- Cardiología
- Dermatología
- Cirugía
- Otorrino
- Ortopedia
- Oncología

### Tabla Provincias

- San José
- Alajuela
- Heredia

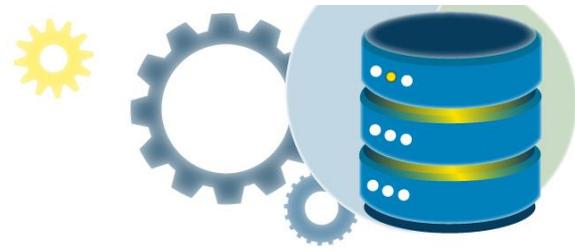




- Cartago
- Guanacaste
- Puntarenas
- Limón

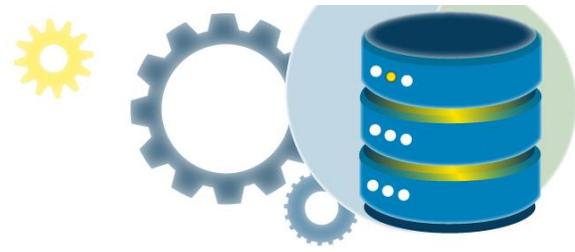
### Tabla Cantones

- Provincia San José
  - San José
  - Escazú
  - Desamparados
- Provincia Alajuela
  - Alajuela
  - Atenas
  - Grecia
- Provincia Heredia
  - Heredia
  - Barva
  - Santo Domingo
- Provincia Cartago
  - Cartago
  - Paraíso
  - La Unión
- Provincia Guanacaste
  - Liberia
  - Nicoya
  - Santa Cruz
- Provincia Puntarenas
  - Puntarenas
  - Esparza
  - Buenos Aires
- Provincia Limón
  - Limón
  - Pococí
  - Siquirres



## Tabla Distritos

- Provincia San José
  - Cantón San José
    - San José
  - Cantón Escazú
    - Escazú
  - Cantón Desamparados
    - Desamparados
- Provincia Alajuela
  - Cantón Alajuela
    - Alajuela
  - Cantón Atenas
    - Atenas
  - Cantón Grecia
    - Grecia
- Provincia Heredia
  - Cantón Heredia
    - Heredia
  - Cantón Barva
    - Barva
  - Cantón Santo Domingo
    - Santo Domingo
- Provincia Cartago
  - Cantón Cartago
    - Cartago
  - Cantón Paraíso
    - Paraíso
  - Cantón La Unión
    - La Unión
- Provincia Guanacaste
  - Cantón Liberia
    - Liberia
  - Cantón Nicoya
    - Nicoya
  - Cantón Santa Cruz
    - Santa Cruz



- Provincia Puntarenas
  - Cantón Puntarenas
    - Puntarenas
  - Cantón Esparza
    - Esparza
  - Cantón Buenos Aires
    - Buenos Aires
- Provincia Limón
  - Cantón Limón
    - Limón
  - Cantón Pococí
    - Pococí
  - Cantón Siquirres
    - Siquirres

Tabla Médicos

CodigoMedico	Nombre	PrimerApellido	SegundoApellido	Especialidad	Cargo	Descripcion
123456	Juan	Porras	Fernández	Maternidad	Jefatura servicio	NULL
789456	María	Carrasco	González	Cardiología	Jefatura servicio	NULL
852963	Fernanda	Mora	Chavarría	Dermatología	Jefatura servicio	NULL
741258	Francisco	Piedra	Moreno	Cirugía	Jefatura servicio	NULL
357951	Karla	Chavarría	Méndez	Otorrino	Jefatura servicio	NULL
951852	Luis Carlo	Arias	González	Ortopedia	Jefatura servicio	NULL
321951	Cinthy	Castro	Pérez	Oncología	Jefatura servicio	NULL
789852	Josefina	Skliar	Johnson	Maternidad	Médico planta	NULL
123654	Fernando	Castro	Madriz	Oncología	Médico planta	NULL
NULL	NULL	NULL	NULL	NULL	NULL	NULL



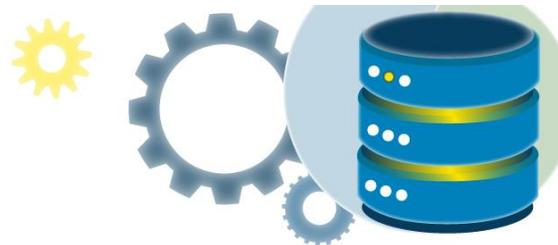
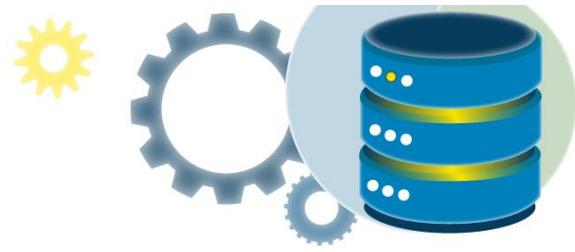


Tabla Pacientes

IDSeguroSocial	OrdenPatronal	Nombre	PrimerApellido	SegundoApellido	FechaNacimiento	Domicilio	Telefono	Provincia	Cantón	Distrito	Observaciones
123456789123456	12345678932165498	Pedro	Quirós	López	17/02/75	25 m norte del palo de aguacate	88889695	San José	San José	San José	NULL
123456789321654	12345678912345678	Carolina	Fernández	Aguero	01/12/51	100 m oeste y 200 m norte de la Guardia Rural	88887452	San José	Escazú	Escazú	NULL
123456789963258	12345678912345678	María	Aguero	Arias	10/11/32	100 m oeste y 200 m norte de la Guardia Rural	88887452	San José	Escazú	Escazú	NULL
852147963123658	87452369854789652	Victoria	González	Arias	24/11/83	Nicoya	88884521	Guanacaste	Nicoya	Nicoya	NULL
987654321147852	96325874132145698	Pablo	Chavarria	Piedra	05/05/05	Contiguo a la Escuela	24516398	Limón	Siquirres	Siquirres	NULL

Tabla Ingresos

IDIngreso	FechaIngreso	NumeroPlanta	NumeroCama	Paciente	Médico	Diagnostico	CostoTratamiento
1	1/1/2019	2	24	Carolina Fernández Aguero	Josefina Skliar Johnson	Cesárea	₡ 1 500 000,00
2	14/2/2021	1	5	Pedro Quirós López	Luis Carlo Arias González	Quebradura	₡ 300 000,00
3	15/3/2021	3	10	Pedro Quirós López	Karla Chavarría Méndez	Sinusitis	₡ 100 000,00



## Solución

Inserción datos tabla Especialidades

```
USE [Hospital]
GO
```

```
INSERT INTO [dbo].[Especialidades]([NombreEspecialidad]) VALUES ('Maternidad');
INSERT INTO [dbo].[Especialidades]([NombreEspecialidad]) VALUES ('Cardiología');
INSERT INTO [dbo].[Especialidades]([NombreEspecialidad]) VALUES ('Dermatología');
INSERT INTO [dbo].[Especialidades]([NombreEspecialidad]) VALUES ('Cirugía');
INSERT INTO [dbo].[Especialidades]([NombreEspecialidad]) VALUES ('Otorrino');
INSERT INTO [dbo].[Especialidades]([NombreEspecialidad]) VALUES ('Ortopedia');
INSERT INTO [dbo].[Especialidades]([NombreEspecialidad]) VALUES ('Oncología');
```

```
GO
```

Inserción datos tabla Provincias

```
USE [Hospital]
GO
```

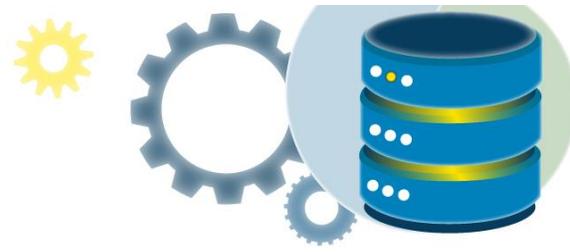
```
INSERT INTO [dbo].[Provincias]([NombreProvincia]) VALUES ('San José');
INSERT INTO [dbo].[Provincias]([NombreProvincia]) VALUES ('Alajuela');
INSERT INTO [dbo].[Provincias]([NombreProvincia]) VALUES ('Heredia');
INSERT INTO [dbo].[Provincias]([NombreProvincia]) VALUES ('Cartago');
INSERT INTO [dbo].[Provincias]([NombreProvincia]) VALUES ('Guanacaste');
INSERT INTO [dbo].[Provincias]([NombreProvincia]) VALUES ('Puntarenas');
INSERT INTO [dbo].[Provincias]([NombreProvincia]) VALUES ('Limón');
```

Inserción datos tabla Cantones

Se incluye una muestra de cantones

```
USE [Hospital]
GO
```

```
INSERT INTO [dbo].[Cantones]([IDProvincia],[NombreCanton]) VALUES (1,'San José');
INSERT INTO [dbo].[Cantones]([IDProvincia],[NombreCanton]) VALUES (1,'Escazú');
INSERT INTO [dbo].[Cantones]([IDProvincia],[NombreCanton]) VALUES (1,'Desamparados');
INSERT INTO [dbo].[Cantones]([IDProvincia],[NombreCanton]) VALUES (2,'Alajuela');
INSERT INTO [dbo].[Cantones]([IDProvincia],[NombreCanton]) VALUES (2,'Atenas');
INSERT INTO [dbo].[Cantones]([IDProvincia],[NombreCanton]) VALUES (2,'Grecia');
INSERT INTO [dbo].[Cantones]([IDProvincia],[NombreCanton]) VALUES (3,'Heredia');
```



```
INSERT INTO [dbo].[Cantones]([IDProvincia],[NombreCanton]) VALUES (3,'Barva');
INSERT INTO [dbo].[Cantones]([IDProvincia],[NombreCanton]) VALUES (3,'Santo Domingo');
INSERT INTO [dbo].[Cantones]([IDProvincia],[NombreCanton]) VALUES (4,'Cartago');
INSERT INTO [dbo].[Cantones]([IDProvincia],[NombreCanton]) VALUES (4,'Paraíso');
INSERT INTO [dbo].[Cantones]([IDProvincia],[NombreCanton]) VALUES (4,'La Unión');
INSERT INTO [dbo].[Cantones]([IDProvincia],[NombreCanton]) VALUES (5,'Liberia');
INSERT INTO [dbo].[Cantones]([IDProvincia],[NombreCanton]) VALUES (5,'Nicoya');
INSERT INTO [dbo].[Cantones]([IDProvincia],[NombreCanton]) VALUES (5,'Santa Cruz');
INSERT INTO [dbo].[Cantones]([IDProvincia],[NombreCanton]) VALUES (6,'Puntarenas');
INSERT INTO [dbo].[Cantones]([IDProvincia],[NombreCanton]) VALUES (6,'Esparza');
INSERT INTO [dbo].[Cantones]([IDProvincia],[NombreCanton]) VALUES (6,'Buenos Aires');
INSERT INTO [dbo].[Cantones]([IDProvincia],[NombreCanton]) VALUES (7,'Limón');
INSERT INTO [dbo].[Cantones]([IDProvincia],[NombreCanton]) VALUES (7,'Pococí');
INSERT INTO [dbo].[Cantones]([IDProvincia],[NombreCanton]) VALUES (7,'Siquirres');
```

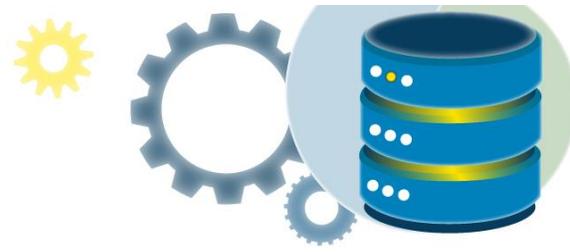
GO

Inserción datos tabla Distritos

Se incluye una muestra de distritos

```
USE [Hospital]
GO
```

```
INSERT INTO [dbo].[Distritos]([IDProvincia],[IDCanton],[NombreDistrito]) VALUES
(1,1,'San José');
INSERT INTO [dbo].[Distritos]([IDProvincia],[IDCanton],[NombreDistrito]) VALUES
(1,2,'Escazú');
INSERT INTO [dbo].[Distritos]([IDProvincia],[IDCanton],[NombreDistrito]) VALUES
(1,3,'Desamparados');
INSERT INTO [dbo].[Distritos]([IDProvincia],[IDCanton],[NombreDistrito]) VALUES
(2,4,'Alajuela');
INSERT INTO [dbo].[Distritos]([IDProvincia],[IDCanton],[NombreDistrito]) VALUES
(2,5,'Atenas');
INSERT INTO [dbo].[Distritos]([IDProvincia],[IDCanton],[NombreDistrito]) VALUES
(2,6,'Grecia');
INSERT INTO [dbo].[Distritos]([IDProvincia],[IDCanton],[NombreDistrito]) VALUES
(3,7,'Heredia');
INSERT INTO [dbo].[Distritos]([IDProvincia],[IDCanton],[NombreDistrito]) VALUES
(3,8,'Barva');
INSERT INTO [dbo].[Distritos]([IDProvincia],[IDCanton],[NombreDistrito]) VALUES
(3,9,'Santo Domingo');
INSERT INTO [dbo].[Distritos]([IDProvincia],[IDCanton],[NombreDistrito]) VALUES
(4,10,'Cartago');
```



```
INSERT INTO [dbo].[Distritos]([IDProvincia], [IDCanton],[NombreDistrito]) VALUES
(4,11,'Paráiso');
INSERT INTO [dbo].[Distritos]([IDProvincia], [IDCanton],[NombreDistrito]) VALUES
(4,12,'La Unión');
INSERT INTO [dbo].[Distritos]([IDProvincia], [IDCanton],[NombreDistrito]) VALUES
(5,13,'Liberia');
INSERT INTO [dbo].[Distritos]([IDProvincia], [IDCanton],[NombreDistrito]) VALUES
(5,14,'Nicoya');
INSERT INTO [dbo].[Distritos]([IDProvincia], [IDCanton],[NombreDistrito]) VALUES
(5,15,'Santa Cruz');
INSERT INTO [dbo].[Distritos]([IDProvincia], [IDCanton],[NombreDistrito]) VALUES
(6,16,'Puntarenas');
INSERT INTO [dbo].[Distritos]([IDProvincia], [IDCanton],[NombreDistrito]) VALUES
(6,17,'Esparza');
INSERT INTO [dbo].[Distritos]([IDProvincia], [IDCanton],[NombreDistrito]) VALUES
(6,18,'Buenos Aires');
INSERT INTO [dbo].[Distritos]([IDProvincia], [IDCanton],[NombreDistrito]) VALUES
(7,19,'Limón');
INSERT INTO [dbo].[Distritos]([IDProvincia], [IDCanton],[NombreDistrito]) VALUES
(7,20,'Pococí');
INSERT INTO [dbo].[Distritos]([IDProvincia], [IDCanton],[NombreDistrito]) VALUES
(7,21,'Siquirres');
```

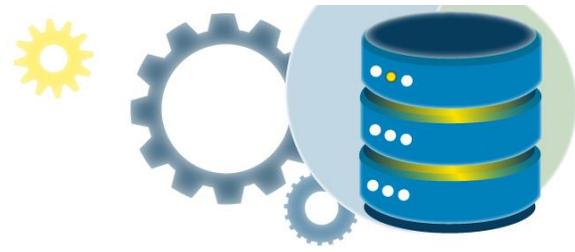
GO

Inserción datos tabla Médicos

```
USE [Hospital]
GO
```

```
INSERT INTO [dbo].[Medicos]
([CodigoMedico]
,[Nombre]
,[PrimerApellido]
,[SegundoApellido]
,[CodigoEspecialidad]
,[Cargo]
,[Observaciones])
VALUES
(<CodigoMedico, nchar(6),>
,<Nombre, varchar(50),>
,<PrimerApellido, varchar(30),>
,<SegundoApellido, varchar(30),>
,<CodigoEspecialidad, int,>
,<Cargo, varchar(30),>
,<Observaciones, varchar(max),>)
```

GO



*\*reemplace los valores por los indicados en la tabla de arriba*

Inserción datos tabla Pacientes

```
USE [Hospital]
GO
```

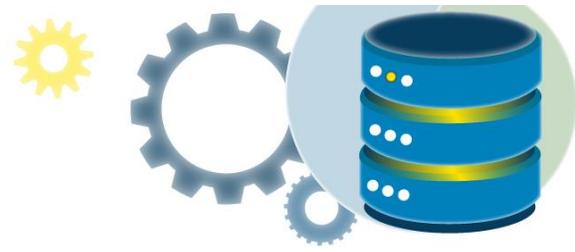
```
INSERT INTO [dbo].[Pacientes]
  ([IDSeguroSocial]
  ,[OrdenPatronal]
  ,[Nombre]
  ,[PrimerApellido]
  ,[SegundoApellido]
  ,[FechaNacimiento]
  ,[Domicilio]
  ,[Telefono]
  ,[IDProvincia]
  ,[IDCanton]
  ,[IDDistrito]
  ,[Observaciones])
VALUES
  (<IDSeguroSocial, nchar(15),>
  ,<OrdenPatronal, nchar(17),>
  ,<Nombre, nvarchar(30),>
  ,<PrimerApellido, nvarchar(50),>
  ,<SegundoApellido, nvarchar(50),>
  ,<FechaNacimiento, datetime,>
  ,<Domicilio, varchar(max),>
  ,<Telefono, nchar(8),>
  ,<IDProvincia, int,>
  ,<IDCanton, int,>
  ,<IDDistrito, int,>
  ,<Observaciones, varchar(max),>)
GO
```

*\*reemplace los valores por los indicados en la tabla de arriba*

Inserción datos tabla Ingresos

```
USE [Hospital]
GO
```

```
INSERT INTO [dbo].[Ingresos]
  ([FechaIngreso]
  ,[NumeroPlanta]
```



```
    , [NumeroCama]  
    , [IDPaciente]  
    , [IDMedico]  
    , [Diagnostico]  
    , [CostoTratamiento]  
    , [Observaciones])  
VALUES  
(<FechaIngreso, datetime,>  
 , <NumeroPlanta, int,>  
 , <NumeroCama, int,>  
 , <IDPaciente, nchar(15),>  
 , <IDMedico, nchar(6),>  
 , <Diagnostico, varchar(max),>  
 , <CostoTratamiento, float,>  
 , <Observaciones, varchar(max),>)  
GO
```

*\*reemplace los valores por los indicados en la tabla de arriba*