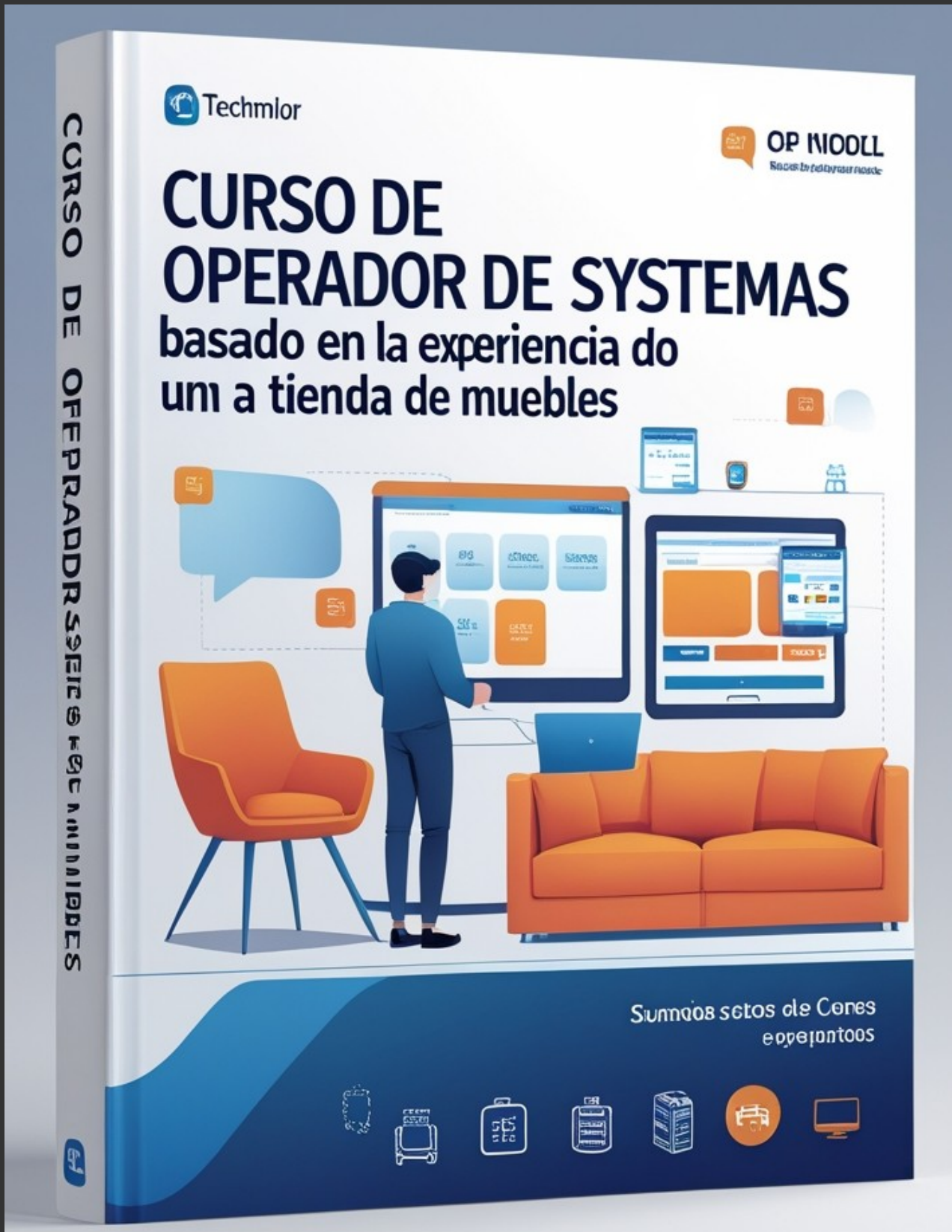


Actualización y Carga de Productos en el Sistema POS desde un Archivo CSV



¿Quieres aprender más sobre soporte técnico y otros temas?

¡Consigue el libro completo [aquí](#)!

Aprende a resolver problemas comunes y mucho más.

Objetivo:

El objetivo de este ejercicio es aprender a crear un archivo CSV con la información de los productos disponibles en una tienda de muebles, y luego importarlo a una base de datos en SQL Server 2008. El ejercicio no solo cubre la importación de nuevos productos, sino también la **actualización** de los productos existentes basados en su nombre (o código de barras si fuera necesario).

Este enfoque asegura que el inventario se mantenga actualizado, sin duplicar productos.

Paso 1: Crear el archivo CSV con los datos de productos

La tabla SQL Server tiene la siguiente estructura:

```
CREATE TABLE Productos (  
    ProductoID INT PRIMARY KEY IDENTITY,  
    Nombre NVARCHAR(100),  
    Precio DECIMAL(10,2),  
    Cantidad INT  
);
```

- **ProductoID** es una columna autoincrementable, por lo que no necesitas incluirla en el archivo CSV.
- Los demás campos son **Nombre**, **Precio** y **Cantidad**.

Paso 2: Crear el archivo CSV en una hoja de cálculo

A. Usando LibreOffice Calc o Excel:

1. Abre **LibreOffice Calc** o **Microsoft Excel**.
2. Crea una hoja de cálculo con los siguientes encabezados:

Nombre	Precio	Cantidad
Sofá Comfort	250.00	10
Mesa de comedor	120.00	5
Silla de oficina	80.00	15
Estante flotante	45.00	20
Lámpara de pie	35.00	30
Espejo rectangular	70.00	25
Silla ergonómica	100.00	12
Cojín decorativo	15.00	50
Mueble para TV	180.00	8
Lámpara de mesa	25.00	40
Silla plegable	50.00	18

¿Quieres aprender más sobre soporte técnico y otros temas?

¡Consigue el libro completo [aquí!](#)

Aprende a resolver problemas comunes y mucho más.

Nombre	Precio	Cantidad
Cama King Size	350.00	4
Sillón reclinable	200.00	6

3. **Nota importante:** No agregues la columna **ProductoID** en la hoja de cálculo, ya que se generará automáticamente en SQL Server al ser de tipo **IDENTITY**.

B. Guardar el archivo como CSV:

1. En **LibreOffice Calc** o **Microsoft Excel**, haz clic en **Archivo** y selecciona **Guardar como....**
2. En el cuadro de diálogo de guardar, selecciona el tipo de archivo **CSV (Comma Separated Values)**.
3. Asigna un nombre al archivo, por ejemplo: `productos.csv`.
4. Haz clic en **Guardar**.

Paso 3: Importar el archivo CSV a SQL Server 2008 y actualizar los productos existentes

Para **modificar los productos existentes** y **agregar los nuevos**, utilizaremos una combinación de comandos SQL para **actualizar** los productos existentes y **agregar** nuevos productos si no se encuentran en la tabla.

A. Importar el CSV como una tabla temporal

1. **Abrir SQL Server Management Studio (SSMS)** y conectarte a tu base de datos.
2. **Crear una tabla temporal** para almacenar los datos del archivo CSV.

```
CREATE TABLE #TempProductos (
    Nombre NVARCHAR(100),
    Precio DECIMAL(10,2),
    Cantidad INT
);
```

3. **Importar el archivo CSV** a la tabla temporal:
 - Haz clic derecho sobre la base de datos y selecciona **Tasks > Import Data....**
 - Selecciona el archivo **productos.csv** como origen de datos.
 - Mapea las columnas del archivo CSV con la tabla temporal **#TempProductos**.

B. Actualizar los productos existentes y agregar los nuevos

Una vez que los datos estén en la tabla temporal, puedes ejecutar un script para **actualizar los productos que ya existen** y **agregar los nuevos productos** si no existen.

```
-- Actualizar productos existentes
UPDATE p
SET
    p.Precio = t.Precio,
    p.Cantidad = t.Cantidad
```

¿Quieres aprender más sobre soporte técnico y otros temas?

¡Consigue el libro completo [aquí!](#)

Aprende a resolver problemas comunes y mucho más.

```
FROM Productos p
INNER JOIN #TempProductos t
    ON p.Nombre = t.Nombre;

-- Insertar productos nuevos
INSERT INTO Productos (Nombre, Precio, Cantidad)
SELECT t.Nombre, t.Precio, t.Cantidad
FROM #TempProductos t
LEFT JOIN Productos p
    ON t.Nombre = p.Nombre
WHERE p.ProductoID IS NULL;

-- Eliminar la tabla temporal
DROP TABLE #TempProductos;
```

Explicación del código:

1. **Actualizar productos existentes:** El primer bloque de código utiliza una **sentencia UPDATE** para modificar los productos existentes. Compara los productos de la tabla **Productos** con los productos en la tabla temporal **#TempProductos** basándose en el **Nombre** del producto. Si encuentra un producto con el mismo nombre, actualiza su **Precio** y **Cantidad**.
 2. **Agregar nuevos productos:** El segundo bloque de código utiliza una **sentencia INSERT** para agregar productos nuevos a la tabla **Productos**. Solo se insertan aquellos productos que **no existen previamente** en la tabla **Productos**. Esto se logra mediante un **LEFT JOIN** con la tabla **Productos**, y la condición **WHERE p.ProductoID IS NULL**, que asegura que solo los productos no encontrados en la tabla **Productos** sean insertados.
 3. **Eliminar la tabla temporal:** Después de realizar la actualización y la inserción, se elimina la tabla temporal **#TempProductos** con **DROP TABLE**.
-

Resumen del Proceso:

1. **Crear el archivo CSV** con los productos que deseas actualizar o agregar. Asegúrate de que el archivo CSV contenga solo las columnas **Nombre**, **Precio** y **Cantidad**.
2. **Importar el archivo CSV** a una **tabla temporal** en SQL Server utilizando el asistente de importación de datos de SSMS.
3. **Actualizar los productos existentes** en la tabla **Productos** utilizando una sentencia **UPDATE**.
4. **Agregar productos nuevos** que no existan en la tabla utilizando una sentencia **INSERT**.
5. **Eliminar la tabla temporal** después de realizar las actualizaciones.

¿Quieres aprender más sobre soporte técnico y otros temas?

¡Consigue el libro completo [aquí](#)!

Aprende a resolver problemas comunes y mucho más.

Errores Comunes al Importar un CSV a SQL Server 2008 Express y Cómo Solucionarlos

1. Error de Conversión de Tipo de Datos (String a INT o Decimal)

- **¿Qué pasa?** SQL Server intenta convertir datos de tipo `string` a tipos `INT` o `DECIMAL`, pero no puede hacerlo si los datos no son válidos. Por ejemplo:
 - Un campo que debería ser un número (como **Precio** o **Cantidad**) tiene texto o caracteres no válidos (como letras o comas en lugar de puntos).
- **Solución:**
 - Asegúrate de que los datos en tu archivo CSV estén correctamente formateados:
 - El **precio** debe ser un número con punto decimal (por ejemplo, 250 . 00).
 - La **cantidad** debe ser un número entero (por ejemplo, 10).
 - Si tienes valores como 250 , 00, cambia la coma por un punto: 250.00.
- **Pasos:**
 - Abre el archivo CSV en **Notepad++** o **Excel**.
 - Asegúrate de que los valores en **Precio** y **Cantidad** sean **números** correctos.
 - Guarda el archivo como **UTF-8** sin BOM.

2. Error de Delimitador Incorrecto (Fila y Columna)

- **¿Qué pasa?**
 - Los delimitadores no están bien configurados, lo que puede causar que SQL Server no reconozca los campos correctamente. Esto puede ocurrir si el archivo CSV no usa el delimitador de columna adecuado (una coma) o el delimitador de fila adecuado (CRLF para Windows).
- **Solución:**
 - **Delimitador de columna:** Debe ser **coma (,)**.
 - **Delimitador de fila:** En Windows debe ser **CRLF (\r\n)**.
- **Pasos:**
 - Si usas **Notepad++**, abre el archivo CSV y revisa que las filas terminen con **CRLF (\r\n)**.
 - Guarda el archivo como **UTF-8 sin BOM**.

3. Encabezados en el CSV (Nombre, Precio, Cantidad)

- **¿Qué pasa?** SQL Server no reconoce correctamente las columnas si el encabezado está mal configurado o si el asistente de importación no está configurado para saltarse la primera fila del archivo CSV.
- **Solución:**

¿Quieres aprender más sobre soporte técnico y otros temas?

¡Consigue el libro completo [aquí](#)!

Aprende a resolver problemas comunes y mucho más.

- Cuando importes, asegúrate de que el asistente de SQL Server reconozca que la **primera fila** es solo para los encabezados (como **Nombre**, **Precio**, **Cantidad**).
- Si usas **BULK INSERT**, asegúrate de que el parámetro **FIRSTROW = 2** esté configurado para que ignore la primera fila.

4. Error al Importar porque los Tipos de Datos No Coinciden

- **¿Qué pasa?** Si el archivo CSV tiene un valor incorrecto para alguna columna (por ejemplo, texto en una columna INT), SQL Server no puede hacer la importación.
- **Solución:**
 - Verifica que todos los valores en la columna **Precio** sean números, y que en la columna **Cantidad** no haya caracteres extraños.
- **Pasos:**
 - En el archivo CSV, asegúrate de que los valores sean solo números en los campos **Precio** y **Cantidad**. Si tienes algo como 35, 00, cámbialo por 35.00.

5. Error de Espacios Extraños o Caracteres Invisibles

- **¿Qué pasa?** A veces, el archivo CSV tiene espacios o caracteres invisibles que pueden interferir con la importación.
- **Solución:**
 - Abre el archivo CSV con un editor de texto avanzado como **Notepad++**.
 - Elimina cualquier espacio al final de las celdas o cualquier carácter extraño.
- **Pasos:**
 - En **Notepad++**, haz clic en **Ver > Mostrar caracteres invisibles** para ver si hay espacios extraños o saltos de línea innecesarios.

6. Problemas con el Formato del Archivo CSV en Linux vs Windows

- **¿Qué pasa?** Si trabajas en un sistema Linux, el archivo CSV puede usar un delimitador de fila diferente (LF en lugar de CRLF).
- **Solución:**
 - Asegúrate de que el archivo CSV tenga el formato correcto para **Windows**:
 - **Delimitador de columna:** coma , .
 - **Delimitador de fila:** CRLF (\r\n).
- **Pasos:**
 - Si el archivo fue creado en Linux, usa un editor de texto como **Notepad++** y guarda el archivo con el formato **CRLF**.

¿Quieres aprender más sobre soporte técnico y otros temas?

¡Consigue el libro completo [aquí!](#)

Aprende a resolver problemas comunes y mucho más.

Ejemplo de CSV Correcto para SQL Server 2008 Express

El archivo CSV debe verse así:

```
Nombre,Precio,Cantidad
Sofá Comfort,250.00,10
Mesa de comedor,120.00,5
Silla de oficina,80.00,15
Estante flotante,45.00,20
Lámpara de pie,35.00,30
Espejo rectangular,70.00,25
Silla ergonómica,100.00,12
Cojín decorativo,15.00,50
Mueble para TV,180.00,8
Lámpara de mesa,25.00,40
Silla plegable,50.00,18
Cama King Size,350.00,4
Sillón reclinable,200.00,6
```

Pasos para Importar el CSV Correctamente a SQL Server

1. Abre SQL Server Management Studio (SSMS) y conéctate a la base de datos.
2. Selecciona la base de datos de destino.
3. Usa el Asistente de Importación:
 - Haz clic derecho en la base de datos > **Tasks** > **Import Data**.
 - Selecciona **Flat File Source**, y luego selecciona el archivo CSV.
 - Configura el **delimitador de columna** como coma , y el **delimitador de fila** como CRLF (\r\n).
 - Verifica que los tipos de datos de las columnas coincidan con la tabla en SQL Server.
4. Ignorar la primera fila:
 - Si tu archivo CSV tiene encabezados, asegúrate de que el asistente esté configurado para **ignorar la primera fila**.

Solución de Errores Comunes

- Si encuentras **errores de conversión** o **valores incorrectos**, revisa cuidadosamente el archivo CSV para asegurarte de que solo haya números en las columnas de **Precio** y **Cantidad**.
- Si el archivo tiene **caracteres invisibles** o **espacios extraños**, usa un editor de texto como **Notepad++** para limpiarlo.

Resumen de Soluciones para los Errores

1. **Delimitador de columna:** Asegúrate de que sea una **coma ,**.

¿Quieres aprender más sobre soporte técnico y otros temas?

¡Consigue el libro completo [aquí!](#)

Aprende a resolver problemas comunes y mucho más.

2. **Delimitador de fila:** Verifica que el archivo CSV utilice **CRLF** (\r\n) si estás en Windows.
3. **Valores numéricos:** Asegúrate de que las columnas numéricas (como **Precio** y **Cantidad**) no tengan texto o comas.
4. **Encabezados:** Asegúrate de que SQL Server reconozca que la primera fila del CSV contiene encabezados.

¿Quieres aprender más sobre soporte técnico y otros temas?

¡Consigue el libro completo [aquí!](#)

Aprende a resolver problemas comunes y mucho más.