

Implementación de una base de datos SQL en microsoft azure y exploración de escenarios de consultas con SQL server management studio

Implementation of an SQL database in microsoft azure and exploration of query scenarios with SQL server management studio

Fabiola Mabel Montero González

Universidad de Panamá, Panamá

fabiola.monterog@up.ac.pa

<https://orcid.org/0000-0002-4681-9471>

Recibido: 31-10-2024, Aceptado: 1-1-2025

DOI <https://doi.org/10.48204/3072-9696.7411>

Resumen

Este artículo describe la implementación de una base de datos SQL en la plataforma en la nube Microsoft Azure y la posterior ejecución de consultas utilizando SQL Server Management Studio (SSMS). Se detalla la arquitectura adoptada, la selección del modelo relacional y las ventajas de Azure en cuanto a escalabilidad, disponibilidad y seguridad, que permiten una gestión de datos eficiente sin necesidad de infraestructura física. La metodología de investigación incluyó las etapas de selección de la plataforma, definición de la arquitectura, creación y configuración de la base de datos, así como la exploración de consultas avanzadas. En los resultados se exponen tres escenarios de análisis: identificación de los productos más vendidos aplicando el principio de Pareto (80/20), uso de los operadores PIVOT y UNPIVOT para el estudio de tendencias de ventas mensuales y una búsqueda que enumera los productos de cada orden de compra, incluyendo filtros por región. Estos escenarios demuestran la capacidad de SSMS para ejecutar consultas complejas y extraer información estratégica. En conjunto, la experiencia confirma que la combinación de Azure y SSMS constituye una solución sólida y

flexible para el almacenamiento, consulta y análisis de grandes volúmenes de datos en entornos empresariales y académicos.

Palabras clave: análisis de datos, base de datos, información

Abstract

This article describes the implementation of an SQL database on the Microsoft Azure cloud platform and the subsequent execution of queries using SQL Server Management Studio (SSMS). It details the adopted architecture, the selection of the relational model, and the advantages of Azure in terms of scalability, availability, and security, which enable efficient data management without the need for physical infrastructure. The research methodology included the stages of platform selection, architectural design, database creation and configuration, as well as the exploration of advanced queries. The results present three analysis scenarios: identification of best-selling products applying the Pareto principle (80/20), the use of PIVOT and UNPIVOT operators to study monthly sales trends, and a query that lists the products in each purchase order, including region-based filters. These scenarios demonstrate the ability of SSMS to execute complex queries and extract strategic information. Overall, the experience confirms that combining Azure and SSMS provides a robust and flexible solution for storing, querying, and analyzing large volumes of data in both business and academic environments.

Keywords: data analysis, database, information

Introducción

En la actualidad, el análisis de datos constituye un pilar esencial en los procesos de toma de decisiones en diversos sectores. Según Treviño et al. (2020), “el curso del mundo de los negocios ha tomado un viraje inesperado que apunta hacia la revolución del análisis de datos, su procesamiento y su transformación en

información útil capaz de hacer grandes aportaciones a la toma de decisiones dentro de las organizaciones” (p. 1065). En este contexto, la gestión eficiente de los datos se ha vuelto indispensable en el entorno tecnológico actual, donde el acceso, almacenamiento y análisis de grandes volúmenes de información son elementos clave para respaldar decisiones estratégicas y bien fundamentadas.

La implementación de bases de datos SQL en entornos en la nube se presenta como una alternativa eficaz frente a las demandas crecientes de escalabilidad, flexibilidad y rendimiento en la gestión de datos. Diversas plataformas de servicios en la nube ofrecen infraestructuras robustas que permiten almacenar y procesar grandes cantidades de información en ambientes seguros y de alto desempeño.

Estas soluciones requieren recursos significativos tanto en términos de infraestructura física como en aspectos relacionados con la seguridad y el uso eficiente de las redes. Por esta razón, las soluciones basadas en la nube han ganado popularidad entre las empresas, al ofrecer ventajas escalables y adaptables. Entre estas alternativas destaca el modelo de Plataforma como Servicio (PaaS). Según Arana et al. (2015), las soluciones PaaS consisten en plataformas de software cuya herramienta de desarrollo se encuentra alojada en la nube, y a las cuales se puede acceder directamente mediante un navegador web.

El modelo PaaS permite a las organizaciones reducir costos de infraestructura y concentrarse en el análisis de datos, sin preocuparse por la gestión de los servidores. En este ámbito, Microsoft Azure, una de las plataformas líderes del sector, ofrece más de 200 servicios en la nube (Microsoft, 2024). Parra (2022) afirma que “Microsoft Azure es uno de los cuatro principales proveedores de Nube a nivel mundial, junto con Amazon Web Services (AWS), Google Cloud y Huawei Cloud, pues tiene una cuarta parte de la cuota del mercado global” (p. 2).

En el ámbito del almacenamiento de datos, Azure ofrece servicios especializados para bases de datos SQL, que permiten a las organizaciones

desplegar y administrar entornos virtualizados de alta disponibilidad, rendimiento y seguridad (Microsoft, 2024). Las bases de datos, por su capacidad para organizar información de manera estructurada, permiten almacenar y gestionar grandes volúmenes de información, siendo esenciales en procesos como inserciones, consultas, actualizaciones y eliminaciones de datos (Valverde et al., 2019). Además, la disponibilidad de modelos tanto relacionales (SQL) como no relacionales (NoSQL) facilitan su adaptación a las necesidades específicas de cada organización.

Según Ramakrishnan y Gehrke (2003), las bases de datos relacionales estructuran la información en tablas interconectadas que utilizan SQL para realizar consultas eficientes. Este modelo emplea filas, columnas, claves primarias y foráneas para organizar y relacionar los datos.

Con el crecimiento exponencial en la generación y almacenamiento de datos, las empresas necesitan soluciones innovadoras que les permitan gestionar su información con eficacia. En este escenario, las bases de datos SQL alojadas en la nube representan una alternativa potente y rentable para cubrir estas demandas (West et al., 2019).

Asimismo, herramientas como SQL Server Management Studio (SSMS) resultan fundamentales para la administración de este tipo de bases de datos, al ofrecer funciones que facilitan la ejecución de consultas complejas, la creación y edición de estructuras de datos, la gestión de procesos de administración y la visualización de resultados. Su interfaz intuitiva permite agilizar el análisis y contribuir a una toma de decisiones más informada (Microsoft, 2024).

Este artículo tiene como finalidad ofrecer una descripción general del proceso de implementación de bases de datos SQL en la nube mediante Microsoft Azure, y explorar distintos escenarios de consulta y análisis de datos utilizando herramientas especializadas como SSMS, demostrando cómo estas tecnologías pueden optimizar tanto la gestión como el análisis de datos.

Materiales y métodos

La metodología empleada en este artículo sigue un enfoque de investigación acción, estructurada en cuatro etapas clave. La primera etapa consistió en una investigación inicial para la selección de la plataforma y las herramientas. La segunda etapa se centró en la definición del modelo arquitectónico más adecuado para la implementación. En la tercera etapa se procedió a la creación y configuración de la base de datos en Azure, seguida de su integración con SQL Server Management Studio (SSMS). Finalmente, en la cuarta etapa, se exploraron y ejecutaron consultas sobre la base de datos. Los materiales empleados incluyen la plataforma Azure y la herramienta SMSS. A continuación, se describen en detalle las etapas del proceso:

Primera etapa: selección de la plataforma y herramientas

La elección de Azure se fundamenta en varios criterios fundamentales:

- **Escalabilidad y flexibilidad:** Azure ofrece una infraestructura flexible y escalable que se ajusta a las necesidades cambiantes de almacenamiento y procesamiento de datos. Permite escalar vertical u horizontalmente, lo que facilita la gestión de conjuntos de datos de diferentes tamaños sin comprometer el rendimiento (Microsoft, 2024).
- **Disponibilidad y confiabilidad:** Azure garantiza un alto nivel de disponibilidad con controles avanzados de seguridad, redundancia de datos y garantías de tiempo de actividad, asegurando el acceso continuo a la base de datos y minimizando el riesgo de interrupciones (Microsoft, 2024).
- **Integración con herramientas:** Azure se integra fácilmente con herramientas como SQL Server Management Studio (SSMS), proporcionando un entorno óptimo para la administración y análisis de bases de datos SQL, entre otras.

La elección de SQL Server Management Studio (SSMS) como herramienta de gestión en el entorno de nube de Azure complementa eficazmente las capacidades de esta plataforma. Según Derfoufi (2024), SSMS es una herramienta integral de Microsoft para administrar y desarrollar bases de datos SQL. Esta plataforma facilita la ejecución de consultas avanzadas, gestión de estructuras de datos y visualización eficiente de resultados.

La selección de Azure y SSMS fue fundamental en esta primera etapa, ya que ambas garantizan una gestión eficiente de los datos desde el inicio del proceso de implementación hasta la fase de análisis.

Segunda etapa: definición del modelo arquitectónico

El modelo arquitectónico establece la interacción entre los componentes del sistema, tomando en cuenta diversos factores como el tipo de bases de datos, los lenguajes de consulta y los métodos de distribución de los datos, como el modelo peer-to-peer o el cliente/servidor.

Existen diferentes tipos de bases de datos, incluyendo los modelos jerárquicos, en red, relacionales y no relacionales. Sin embargo, en la actualidad, las bases de datos más utilizadas se dividen en dos grandes grupos: relacionales (SQL) y no relacionales (NoSQL) (Chingo & López, 2021).

El modelo relacional, propuesto por Edgar F. Codd en los años 70, ha sido el estándar en la gestión de datos estructurados debido a su capacidad para organizar la información en tablas relacionadas de forma lógica y coherente (Codd, 1970).

Según Bernal y Molina (2022), el acceso a la información “se realiza a través del Lenguaje de Consulta Estructurada (SQL), un lenguaje que permite tanto la recuperación como la gestión de datos estructurados, transformándolos en información” (p. 308).

Para esta implementación, se optó por el modelo de base de datos relacional debido a su amplio uso en la industria, su capacidad para manejar grandes volúmenes de datos y la flexibilidad que ofrece el lenguaje SQL para realizar análisis complejos y extraer información significativa (Valverde et al., 2019). Esta elección se complementa con la plataforma Microsoft Azure, que proporciona un entorno robusto para bases de datos relacionales, empleando SQL Server como motor principal.

Respecto al método de distribución de datos, se ha elegido un modelo cliente-servidor, que centraliza tanto la gestión como la seguridad de los datos. Azure facilita esta arquitectura, permitiendo que los clientes accedan a los datos mediante procesos que gestionan las consultas y la interfaz de usuario, mientras que el servidor en la nube se encarga de manejar las transacciones y la administración de los datos. En este sentido, García (2015) afirma que “la arquitectura cliente/servidor se divide en dos capas una la del cliente que implementa la interfaz y otra es donde se encuentra el sistema gestor de base de datos” (p. 68). Este enfoque garantiza un alto nivel de control, seguridad y además ofrece escalabilidad y flexibilidad al sistema.

Tercera etapa: creación, configuración y conexión de la base de datos

Para implementar la arquitectura, se utiliza la suscripción de prueba de Microsoft Azure, que ofrece 100,000 segundos de tiempo de procesamiento en núcleos virtuales por mes y hasta 32 GB de almacenamiento de datos por 12 meses. Esta opción permite acceder a una base de datos de uso general y comenzar a trabajar con Azure SQL Database. Esta versión emplea una plataforma como servicio (PaaS), completamente gestionada, con funcionalidades avanzadas de administración de bases de datos (Microsoft, 2024).

A continuación, se detalla el proceso paso a paso para la creación, configuración y conexión de la base de datos en Microsoft Azure.

Paso 1. Activación de la cuenta de microsoft azure

Desde el portal de Azure, en la sección “Comenzar a usar Azure”, se accede a la opción de prueba. Utilizando una cuenta de Outlook, se completa el registro mediante un formulario. Una vez activada la cuenta, se muestra la página principal del portal de Azure con la suscripción lista para crear la base de datos.

Paso 2. Creación y configuración de la base de datos sql en azure

Para crear la base de datos SQL, se accede al servicio desde el botón “Apply offer (Preview)”. Luego, se crea un grupo de recursos y se configura la base de datos con el nombre “myFreeDB” y al servidor “myfreesqldbserver”, siguiendo las convenciones establecidas por Azure.

La autenticación se realiza mediante una cuenta de Outlook, asignando permisos de administrador y creando una contraseña para el acceso. A continuación, se configuran las reglas del firewall para permitir que los servicios y recursos de Azure acceden al servidor, y se añade la dirección IP del cliente. Finalmente, se selecciona el conjunto de datos preexistente “*AdventureWorks*” en Azure como base para la creación de la base de datos, una opción que Azure ofrece para la implementación de bases de datos (Microsoft, 2024).

Paso 3. Conexión a la base de datos de azure desde sql server management studio (ssms)

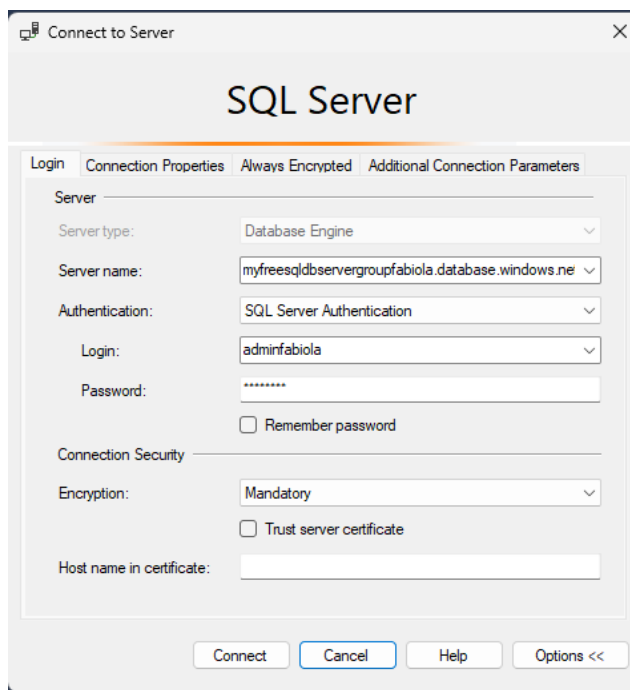
Una vez instalada la herramienta SSMS, se procede a conectarla a la base de datos de Azure. Al abrir el programa, se solicita ingresar los datos para la conexión. En el campo de Autenticación, se selecciona “Windows Authentication” y se ingresan las credenciales configuradas durante la creación de la base de datos en Azure.

Dentro de SSMS, se establece la conexión seleccionando “Connect-Database Engine”. En la ventana de conexión, se elige el servidor configurado en Azure, se selecciona el tipo de autenticación como “SQL_Server Authentication”, y

se ingresan la cuenta y contraseña creada desde Azure. Al hacer clic en “Connect”, se establece la conexión con el servidor, tal como se observa en la Figura 1.

Figura 1

Conexión a SQL Server desde SSMS

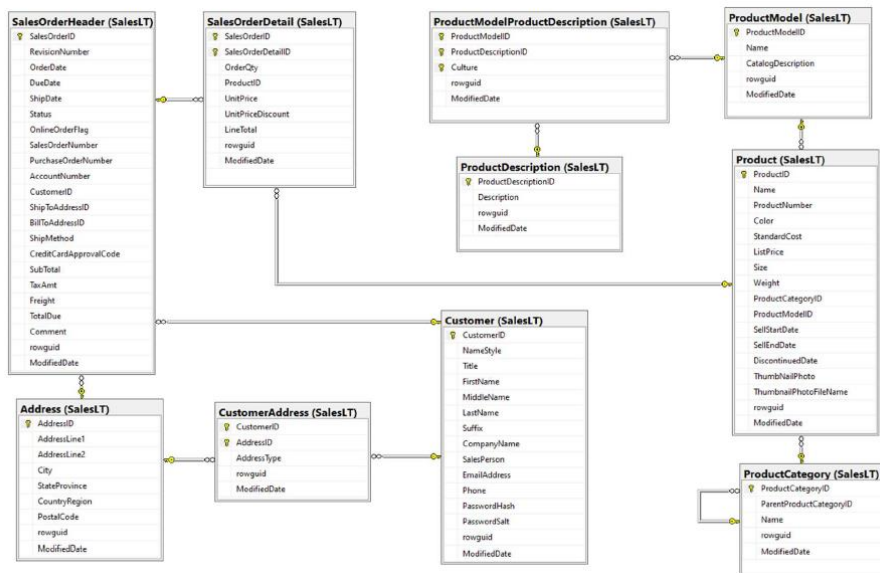


Una vez completada la conexión, se tiene acceso a los esquemas de la base de datos, incluyendo las tablas y los datos almacenados. Desde este entorno, es posible realizar consultas directamente a la base de datos en Azure, iniciando la interacción cliente-servidor.

En la Figura 2 se presenta el modelo relacional de la base de datos *AdventureWorks*. Esta base de datos simula un entorno de comercio minorista al recrear el escenario de una tienda ficticia de bicicletas llamada *Adventure Works*. Su finalidad es permitir la exploración y prueba de diversas funcionalidades de administración de datos en un contexto práctico y realista.

Figura 2

Modelo relacional de AdventureWorks



Cuarta etapa: exploración y ejecución de consultas

En esta etapa se desarrollan consultas basadas en diferentes escenarios, considerando la base de datos *AdventureWorks*. Una vez realizada la conexión a la base de datos en Azure desde SQL Server Management Studio (SSMS), se procede a la ejecución de consultas SQL diseñadas para extraer información relevante, aplicando operadores y funciones que satisfacen los requerimientos específicos del negocio.

Los detalles de estas consultas y los resultados obtenidos se presentarán en la sección de resultados, donde se examinarán las salidas generadas y su aplicación en contextos empresariales reales.

Resultados

A continuación, se presentan tres escenarios de consultas aplicadas en un contexto real de análisis de datos.

Escenario 1: extracción de distribución de pareto (80/20) en las ventas de productos

En el primer escenario de consulta, se realiza la extracción de la distribución de Pareto (80/20) de las ventas de los productos, una técnica establecida en el análisis de datos empresariales para identificar productos clave. Esta metodología permite a las organizaciones centrar sus recursos y estrategias en el 20% de productos que generan aproximadamente el 80% de las ventas, optimizando así el rendimiento y facilitando la toma de decisiones estratégicas. Al aplicar este enfoque en SQL, es posible identificar rápidamente aquellos productos de mayor rentabilidad, lo cual resulta fundamental para maximizar resultados en mercados competitivos. Plataformas tecnológicas líderes como Oracle implementan este análisis en entornos SQL para facilitar la automatización y simplificación de estas consultas (Lions, 2020).

La consulta SQL del primer escenario está estructurada en tres partes principales. La primera parte es una subconsulta para calcular las ventas por producto, que comienza con la declaración WITH, iniciando la subconsulta denominada SalesCTE. A partir de la tabla [SalesLT].[SalesOrderDetail], se extrae la columna ProductID y se genera una nueva columna con el alias TotalSales, que representa la suma de las unidades vendidas de cada producto por su precio unitario. Luego, se agrupan los valores por ProductID mediante un GROUP BY.

En la segunda parte, se realiza otra subconsulta para calcular los percentiles de ventas. Dentro del bloque WITH, se define una subconsulta adicional llamada TotalVentasSummary, en la que se seleccionan las columnas ProductID y TotalSales

previamente calculadas. Luego, se aplica la función `PERCENTILE_CONT` para obtener los percentiles 80 y 20 de las ventas, utilizando la cláusula `WITHIN GROUP (ORDER BY TotalSales)` para ordenar los datos antes de calcular los percentiles. Finalmente, la subconsulta toma los datos de `SalesCTE`, que contiene las ventas totales por producto.

En la tercera y última etapa, se construye la consulta principal, seleccionando las columnas `ProductID` y `TotalSales` de la subconsulta `TotalVentasSummary`, y filtrando para obtener solo aquellos productos cuyas ventas representan el 80% superior (P80) o el 20% inferior (P20) del total.

En la Figura 3 se muestra el código completo junto con el resultado de la consulta. Este resultado presenta todos los productos cuyas ventas totales representan el 80% de las ventas globales, es decir, los productos más vendidos.

Figura 3

Escenario 1 – Productos más vendidos

```
-- Definición de la primera CTE para calcular las ventas totales por producto
WITH SalesCTE AS (
    SELECT ProductID, SUM(OrderQty * UnitPrice) AS TotalSales
    FROM [SalesLT].[SalesOrderDetail]
    GROUP BY ProductID
),
-- Definición de la segunda CTE para resumir las ventas totales y calcular percentiles
TotalVentasSummary AS (
    SELECT
        ProductID,
        TotalSales,
        PERCENTILE_CONT(0.8) WITHIN GROUP (ORDER BY TotalSales) OVER() AS P80,
        PERCENTILE_CONT(0.2) WITHIN GROUP (ORDER BY TotalSales) OVER() AS P20
    FROM SalesCTE
)
-- Consulta principal para seleccionar productos en el 20% superior o inferior de ventas
SELECT
    ProductID,
    TotalSales
FROM
    TotalVentasSummary
WHERE
```

ProductID	TotalSales
42	972
43	779
44	966
45	954
46	793
47	794
48	781
49	974
50	784
51	973
52	780
53	967
54	957
55	782
56	783
57	969
58	976

Query executed successfully.

En este primer escenario, se muestra cómo una consulta SQL puede identificar los productos más vendidos de las ventas totales, aplicando el principio de Pareto (80/20). La consulta primero calcula el total de ventas por producto; luego, determina los percentiles 80 y 20 para clasificar las ventas; y, finalmente, aplica filtros para seleccionar únicamente los productos que cumplen con estos criterios. Este análisis permite a las empresas enfocar sus esfuerzos en los productos más rentables, optimizando la efectividad de sus estrategias comerciales.

Escenario 2: selección de datos de las órdenes de venta

En el segundo escenario se emplean los operadores relacionales PIVOT y UNPIVOT para transformar los datos en la tabla de una forma distinta, facilitando así su análisis. El operador PIVOT permite rotar datos de una columna en múltiples columnas de la salida, aplicando agregaciones que sintetizan la información relevante y permitiendo la visualización de relaciones entre los datos de forma clara. Esto es particularmente útil para analizar tendencias o patrones dentro de una tabla

de valores, ya que permite consolidar registros en una vista concisa. Por otro lado, UNPIVOT realiza la operación inversa, al transformar columnas en valores de fila, lo que facilita descomponer datos consolidados para un análisis más detallado y flexible. Según Microsoft Learn (2024), esta capacidad de alternar entre formatos de datos expande las opciones de análisis y brinda mayor comprensión en contextos donde la reestructuración de datos es esencial para obtener información significativa.

En este escenario, se extraen y organizan datos clave de la tabla SalesOrderDetail, que contiene los detalles de las órdenes de venta, con el objetivo de obtener un resumen de las ventas mensuales por producto. La consulta utiliza primero el operador PIVOT de SQL para agrupar y mostrar las ventas mensuales totales por producto. A continuación, estos resultados se desagrupan con el operador UNPIVOT, permitiendo así visualizar las ventas mensuales individuales de cada producto de manera detallada. En la Figura 4, se puede observar la consulta utilizando ambas funciones.

Figura 4

Escenario 2 – Detalle de las órdenes de venta

```

WITH SalesData AS (
    SELECT
        p.ProductID,
        p.Name AS ProductName,
        DATENAME(month, soh.OrderDate) AS OrderMonth,
        YEAR(soh.OrderDate) AS OrderYear,
        sod.OrderQty
    FROM
        SalesLT.SalesOrderDetail sod
    JOIN
        SalesLT.SalesOrderHeader soh ON sod.SalesOrderID = soh.SalesOrderID
    JOIN
        SalesLT.Product p ON sod.ProductID = p.ProductID
)
-- Pivot para obtener ventas mensuales totales por producto
SELECT *
FROM (
    SELECT
        ProductID,
        ProductName,
        OrderMonth,
        OrderQty
    FROM
        SalesData
) AS MonthlySales
PIVOT (
    SUM(OrderQty)
    FOR OrderMonth IN ([January], [February], [March], [April], [May], [June], [July], [August], [September], [October], [November], [December])
) AS PivotTable
-- UNPIVOT para desagregar las ventas mensuales por producto
UNPIVOT (
    MonthlySales
    FOR OrderMonth IN ([January], [February], [March], [April], [May], [June], [July], [August], [September], [October], [November], [December])
) AS UnpivotTable
SELECT *
FROM SalesLT.SalesOrderDetail;

```

La primera parte de la consulta define una Expresión de Tabla Común (CTE) llamada SalesData. Aquí, se seleccionan los datos relevantes de las tablas SalesOrderDetail y Product, incluyendo ProductID, ProductName, el mes y el año de la fecha de la orden, que son extraídos con las funciones DATENAME y YEAR, y la cantidad de productos vendidos (OrderQty).

Luego, mediante una serie de uniones (JOIN), la consulta conecta SalesOrderDetail con SalesOrderHeader y Product, asignando alias para simplificar su legibilidad. Posteriormente, antes de aplicar PIVOT, se realiza un SELECT sobre los campos ProductID, ProductName, OrderMonth, y OrderQty de SalesData. La función PIVOT se emplea para sumar las cantidades de productos vendidos (OrderQty) por mes (OrderMonth), proporcionando un resumen mensual de ventas por producto.

En la Figura 5 se muestra el resultado del operador PIVOT, donde se visualizan los productos y las cantidades de ventas para el mes de junio, organizado en una tabla de 142 filas. Cabe destacar, que, en este caso, la base de datos solo contiene datos de junio, por lo que únicamente se muestran las ventas de ese mes.

Sin embargo, si se contara con datos de enero a diciembre, la consulta reflejaría un desglose completo de las ventas mensuales a lo largo de todo el año, permitiendo una comparación y análisis de estacionalidad más detallado.

Figura 5

Resultado del operador PIVOT

	ProductID	ProductName	January	February	March	April	May	June	July	August	September	October	November	December
1	712	AWC Logo Cap	NULL	NULL	NULL	NULL	NULL	52	NULL	NULL	NULL	NULL	NULL	NULL
2	877	Bike Wash - Dissolver	NULL	NULL	NULL	NULL	NULL	55	NULL	NULL	NULL	NULL	NULL	NULL
3	952	Chain	NULL	NULL	NULL	NULL	NULL	8	NULL	NULL	NULL	NULL	NULL	NULL
4	865	Classic Vest, M	NULL	NULL	NULL	NULL	NULL	34	NULL	NULL	NULL	NULL	NULL	NULL
5	864	Classic Vest, S	NULL	NULL	NULL	NULL	NULL	87	NULL	NULL	NULL	NULL	NULL	NULL
6	948	Front Brakes	NULL	NULL	NULL	NULL	NULL	12	NULL	NULL	NULL	NULL	NULL	NULL
7	945	Front Derailleur	NULL	NULL	NULL	NULL	NULL	13	NULL	NULL	NULL	NULL	NULL	NULL
8	860	Half-Finger Gloves, L	NULL	NULL	NULL	NULL	NULL	19	NULL	NULL	NULL	NULL	NULL	NULL
9	859	Half-Finger Gloves, M	NULL	NULL	NULL	NULL	NULL	26	NULL	NULL	NULL	NULL	NULL	NULL
10	858	Half-Finger Gloves, S	NULL	NULL	NULL	NULL	NULL	12	NULL	NULL	NULL	NULL	NULL	NULL

A continuación, el operador UNPIVOT revierte la transformación de PIVOT, consolidando los datos de ventas mensuales en una sola columna llamada MonthlySale, que contiene las cantidades de ventas de cada producto para cada mes específico. Este proceso permite observar la información detallada de las ventas mensuales por producto y mes, como se muestra en la Figura 6.

Figura 6

Resultado del operador UNPIVOT

	ProductID	ProductName	MonthlySales	OrderMonth
1	712	AWC Logo Cap	52	June
2	877	Bike Wash - Dissolver	55	June
3	952	Chain	8	June
4	865	Classic Vest, M	34	June
5	864	Classic Vest, S	87	June
6	948	Front Brakes	12	June
7	945	Front Derailleur	13	June
8	860	Half-Finger Gloves, L	19	June
9	859	Half-Finger Gloves, M	26	June
10	858	Half-Finger Gloves, S	12	June

El resultado final es una tabla de 142 filas en la que cada fila representa un producto en un mes específico, detallando la cantidad de ventas para ese mes y mostrando la información organizada por ProductID, ProductName, MonthlySale y OrderMonth, lo que facilita una visualización detalla de las ventas mensuales individuales por producto.

El escenario 2 permite realizar un análisis de las tendencias de ventas mensuales de cada producto. Al organizar las ventas por ProductID y mes, es posible identificar patrones de demanda, estacionalidades y picos de ventas, lo que facilita la toma de decisiones informadas en áreas como gestión de inventarios, promoción de productos y planificación de la producción. Este tipo de visualización mensual ayuda a anticiparse a la demanda y ajustar estrategias de marketing de manera precisa, maximizando la efectividad en cada ciclo de ventas.

Escenario 3: búsqueda que enumera los productos dentro de cada orden de compra generada

En el tercer escenario de consulta, se busca listar los productos dentro de cada orden de compra generada, mostrando información clave como el precio unitario por producto, el precio total por cada producto adquirido, y el total de la orden de compra. Además, se incluye la funcionalidad de filtrar las órdenes de compra por país.

En la parte superior de la Figura 7, se puede observar el detalle de la consulta. La estructura se explica de la siguiente manera:

- En la parte inicial, se seleccionan los campos de interés relacionados con los productos y las órdenes de compra.
- Se establecen relaciones entre las tablas de la base de datos utilizando campos clave. Las tablas involucradas incluyen: SalesOrderHeader, SalesOrderDetail, Product, ProductCategory, y Address.
- En la parte final, se aplica una cláusula de filtrado que permitir introducir criterios de búsqueda, como la filtración de órdenes de compra por país.

Figura 7

Escenario 3 – Lista de productos por orden de compra generada

```

SELECT
ROW_NUMBER()OVER(PARTITION BY SOH.SalesOrderID ORDER BY SOD.SalesOrderID)AS Num,
SOD.SalesOrderID,
P.Name AS ProductName,
pc.Name as ProductCategoryName,
SOD.OrderQty,
SOD.UnitPrice,
(SOD.OrderQty*SOD.UnitPrice)AS Price_QTY,
SOH.TaxAmt,
SOH.Freight,
SOH.SubTotal as SubTotalOrder,
SOH.TotalDue as TotalDueOrder,
a.CountryRegion
FROM [SalesLT].[SalesOrderHeader] SOH
LEFT JOIN [SalesLT].[SalesOrderDetail] SOD ON SOH.SalesOrderID =SOD.SalesOrderID
LEFT JOIN [SalesLT].[Product] P ON SOD.ProductID=P.ProductID
LEFT JOIN [SalesLT].[ProductCategory] pc on p.ProductCategoryID=pc.ProductCategoryID
LEFT JOIN [SalesLT].[Address] a on SOH.ShipToAddressID=a.AddressID
WHERE
a.CountryRegion='United Kingdom'

```

	Num	SalesOrderID	ProductName	ProductCategoryName	OrderQty	UnitPrice	Price_QTY	TaxAmt	Freight	SubTotalOrder	TotalDueOrder	CountryRegion
2	1	71780	ML Mountain Frame-W/ Silver, 42	Mountain Frames	4	218.454	873.816	3073.4952	960.4672	38418.6895	42452.6519	United Kingdom
3	2	71780	Mountain 400-W/ Silver, 46	Mountain Bikes	2	461.694	923.388	3073.4952	960.4672	38418.6895	42452.6519	United Kingdom
4	3	71780	Mountain 500 Silver, 52	Mountain Bikes	6	112.998	677.988	3073.4952	960.4672	38418.6895	42452.6519	United Kingdom
5	4	71780	HL Mountain Frame - Silver, 38	Mountain Frames	2	818.70	1637.40	3073.4952	960.4672	38418.6895	42452.6519	United Kingdom
6	5	71780	Mountain 500 Black, 42	Mountain Bikes	1	323.994	323.994	3073.4952	960.4672	38418.6895	42452.6519	United Kingdom
7	6	71780	LL Mountain Frame - Black, 48	Mountain Frames	1	149.874	149.874	3073.4952	960.4672	38418.6895	42452.6519	United Kingdom
8	7	71780	HL Mountain Frame - Black, 42	Mountain Frames	1	809.76	809.76	3073.4952	960.4672	38418.6895	42452.6519	United Kingdom
9	8	71780	Mountain 200 Black, 38	Mountain Bikes	4	1376.994	5507.976	3073.4952	960.4672	38418.6895	42452.6519	United Kingdom
10	9	71780	LL Mountain Frame - Silver, 44	Mountain Frames	2	158.43	316.86	3073.4952	960.4672	38418.6895	42452.6519	United Kingdom
11	10	71780	Mountain 200 Silver, 42	Mountain Bikes	4	1391.994	5567.976	3073.4952	960.4672	38418.6895	42452.6519	United Kingdom
12	11	71780	HL Mountain Pedal	Pedals	1	48.594	48.594	3073.4952	960.4672	38418.6895	42452.6519	United Kingdom
13	12	71780	Women's Mountain Shorts, S	Shorts	6	41.994	251.964	3073.4952	960.4672	38418.6895	42452.6519	United Kingdom
14	13	71780	Mountain 500 Silver, 42	Mountain Bikes	1	112.998	112.998	3073.4952	960.4672	38418.6895	42452.6519	United Kingdom
15	14	71780	Mountain 500 Black, 40	Mountain Bikes	2	323.994	647.988	3073.4952	960.4672	38418.6895	42452.6519	United Kingdom
16	15	71780	Mountain 500 Black, 44	Mountain Bikes	3	323.994	971.982	3073.4952	960.4672	38418.6895	42452.6519	United Kingdom
17	16	71780	Mountain 500 Black, 48	Mountain Bikes	1	323.994	323.994	3073.4952	960.4672	38418.6895	42452.6519	United Kingdom
18	17	71780	Mountain 500 Black, 52	Mountain Bikes	2	323.994	647.988	3073.4952	960.4672	38418.6895	42452.6519	United Kingdom
19	18	71780	Mountain 500 Silver, 40	Mountain Bikes	2	112.998	225.996	3073.4952	960.4672	38418.6895	42452.6519	United Kingdom
20	19	71780	Mountain 500 Silver, 44	Mountain Bikes	3	112.998	338.994	3073.4952	960.4672	38418.6895	42452.6519	United Kingdom
21	20	71780	Mountain 500 Silver, 48	Mountain Bikes	3	112.998	338.994	3073.4952	960.4672	38418.6895	42452.6519	United Kingdom
22	21	71780	Mountain 400-W/ Silver, 40	Mountain Bikes	2	461.694	923.388	3073.4952	960.4672	38418.6895	42452.6519	United Kingdom
23	22	71780	Mountain 400-W/ Silver, 42	Mountain Bikes	3	461.694	1385.082	3073.4952	960.4672	38418.6895	42452.6519	United Kingdom
24	23	71780	Mountain 200 Black, 42	Mountain Bikes	5	1376.994	6884.97	3073.4952	960.4672	38418.6895	42452.6519	United Kingdom
25	24	71780	ML Mountain Handlebars	Handlebars	3	37.152	111.456	3073.4952	960.4672	38418.6895	42452.6519	United Kingdom
26	25	71780	HL Mountain Handlebars	Handlebars	1	72.162	72.162	3073.4952	960.4672	38418.6895	42452.6519	United Kingdom
27	26	71780	LL Mountain Pedal	Pedals	2	24.294	48.588	3073.4952	960.4672	38418.6895	42452.6519	United Kingdom
28	27	71780	LL Mountain Frame - Black, 44	Mountain Frames	1	149.874	149.874	3073.4952	960.4672	38418.6895	42452.6519	United Kingdom
29	28	71780	Women's Mountain Shorts, L	Shorts	7	41.994	293.958	3073.4952	960.4672	38418.6895	42452.6519	United Kingdom
30	29	71780	Hydration Pack - 70 oz.	Hydration Packs	1	32.994	32.994	3073.4952	960.4672	38418.6895	42452.6519	United Kingdom

El resultado de la consulta muestra un desglose detallado de los productos incluidos en cada orden de compra. Para cada producto, se presenta información

como el nombre del producto (ProductName), el precio unitario (UnitPrice), el total correspondiente a la cantidad de productos comprados (Price_QTY), los impuestos (TaxAmt), y el total de la orden (TotalDueOrder). Además, se incluye la información del país de origen de la compra, lo que permite aplicar filtros específicos por región, tal como se muestra en la tabla de resultados en la parte inferior de la Figura 7.

Este tipo de consulta es particularmente útil para visualizar y analizar las ventas de manera detallada por cliente y región, lo que facilita un mayor control sobre las transacciones comerciales y permite la toma de decisiones estratégicas más informadas.

Discusión

Para la exploración de consultas presentadas en este artículo, se ha utilizado el Lenguaje de Consulta Estructurada (SQL) para extraer información de una base de datos relacional. SQL es ampliamente utilizado en diversas aplicaciones, tanto comerciales como académicas, debido a su robustez y versatilidad para gestionar y manipular datos.

A través de los tres escenarios mostrados, se presentan varias funciones analíticas y operadores de SQL que ayudan a resolver problemas de análisis de datos. Estas herramientas permiten extraer información detallada y transformarla en conocimientos útiles para la toma de decisiones. Entre las funciones destacadas, se ha utilizado OVER () y ROW_NUMBER (), que operan sobre subconjuntos de datos, facilitando el cálculo de métricas y la organización de los resultados. Además, se utilizan componentes como PARTITION BY y ORDER BY, que proporcionan un mayor control sobre la agrupación y ordenación de los datos.

También se aplican operadores relacionales como PIVOT y UNPIVOT, que permiten transformar la estructura de los datos, convirtiendo filas en columnas y viceversas. Estas funcionalidades son clave para reorganizar datos y visualizar patrones que de otra manera serían difíciles de identificar. Aunque herramientas actuales de análisis visual como Power BI y Tableau ofrecen soluciones más

intuitivas para estas transformaciones, es valioso comprender cómo aplicar estas operaciones directamente en SQL, en particular cuando se trabaja en entornos que no disponen de estas herramientas externas.

En el primer escenario, se utiliza la función `PERCENTIL_CONT ()` para determinar qué productos conformaban el 80% de las ventas totales. Esta función permite identificar los productos más significativos en términos de ingresos, así como aquellos que representan el 20% restante de las ventas, ofreciendo una visión clara de la concentración de ventas y ayudando a tomar decisiones estratégicas.

En el segundo escenario, se demuestra la utilidad de `PIVOT` y `UNPIVOT` para reorganizar los datos. `PIVOT` transforma valores únicos de una columna en múltiples columnas en la salida, ejecutando agregaciones, mientras que `UNPIVOT` invierte este proceso. Estas operaciones permiten analizar los datos desde perspectivas diferentes, algo importante en la toma de decisiones basada en información multidimensional.

El tercer escenario mostró el uso de la función `ROW_NUMBER ()`, que asigna un número incremental único a las filas en el resultado de la consulta. Esto es útil cuando se necesita identificar el orden en que aparecen los resultados o realizar operaciones como seleccionar el primer o último elemento de un grupo de filas.

En conjunto, las funciones y operadores explorados en este estudio son de uso frecuente al trabajar con bases de datos relaciones y forman parte del núcleo de SQL en su capacidad para transformar y analizar datos de manera eficiente. Estos escenarios prácticos con SQL Server demuestran cómo las consultas avanzadas pueden proporcionar una visión profunda del comportamiento de los datos, ayudando a los analistas y profesionales de datos a tomar decisiones más informadas y precisas.

Además, todos los escenarios fueron ejecutados con éxito en una base de datos alojada en la nube, lo cual facilitó el acceso eficiente y remoto a los datos, optimizando la ejecución de consultas desde diversos puntos de acceso y

garantizando un entorno de trabajo escalable y seguro. Báez y Clunie (2020) destacan que este tipo de entorno ofrece ventajas significativas al proporcionar seguridad, escalabilidad, confiabilidad y acceso global, aspectos que facilitan la ejecución de proyectos de gran alcance sin requerir infraestructura propia ni personal especializado para su gestión. En este contexto, la integración de SQL Server en Azure proporciona un rendimiento óptimo al manejar grandes volúmenes de datos y procesar las consultas de manera eficiente, lo que se traduce en tiempos de respuesta rápidos y consistentes.

Por otro lado, García et al. (2023) señalan que:

Es importante tener en cuenta que los servicios en la nube también conllevan ciertos riesgos, tales como la pérdida de control sobre los datos, problemas de privacidad y cumplimiento normativo, dependencia del proveedor y posibles amenazas de seguridad. Por ello, es esencial identificar y evaluar estos riesgos junto con los beneficios para gestionar adecuadamente el uso de la nube (p. 10).

Se espera que los ejemplos de consulta presentados en este artículo ofrezcan una comprensión más profunda de las capacidades de SQL desde una plataforma en la nube y que sean útiles para quienes desean optimizar sus consultas y análisis en bases de datos relacionales.

Conclusión

En conclusión, la implementación de una base de datos SQL en la plataforma en la nube de Microsoft Azure demuestra ser una decisión estratégica que ofrece una gama de ventajas significativas para la gestión y análisis eficiente de grandes volúmenes de datos. En este artículo, se explora cómo el modelo relacional de bases de datos y el lenguaje SQL brindan una estructura sólida y robusta para organizar, consultar y manipular datos de manera eficaz.

Los escenarios de consulta desarrollado en este estudio ponen de relieve la capacidad de SQL Server Management Studio (SSMS) para manejar consultas complejas con funciones y operadores avanzados como PERCENTIL_CONT, PIVOT, UNPIVOT, y ROW_NUMBER (). En el primer escenario, el análisis de ventas mediante la distribución de Pareto proporcionó una visión estratégica de los productos que generan el mayor ingreso. El segundo escenario demostró la utilidad de los operadores PIVOT y UNPIVOT para reorganizar los datos de manera flexible, mientras que, en el tercer escenario, se utilizó la función ROW_NUMBER () para ordenar y numerar filas de datos, mejorando la organización y el análisis de la información.

La combinación de Azure y SSMS no solo ha facilitado la ejecución de estas consultas con eficiencia, sino que también ha proporcionado un entorno seguro y confiable para alojar la base de datos. Azure garantiza la integridad, seguridad y disponibilidad de los datos en todo momento, lo que es fundamental para aplicaciones empresariales y entornos de análisis de datos.

Los resultados de este estudio resaltan la efectividad de Azure como una solución integral para bases de datos en la nube. Al combinarlo con herramientas como SQL Server Management Studio, se puede simplificar tareas complejas de análisis y toma de decisiones informadas. Estos hallazgos refuerzan la importancia de elegir la infraestructura en la nube adecuada junto con herramientas potentes de gestión de bases de datos, lo que permite a las organizaciones maximizar su eficiencia operativa y obtener un control detallado sobre la información clave para sus operaciones.

Referencias bibliográficas

Arana López, L. M., Ruiz Rivera, M. E., & La Serna Palomino, N. (2015). Análisis de aplicaciones empleando la computación en la nube de tipo PaaS y la metodología ágil Scrum. *Industrial Data*, 18(1), 149-160.

- Báez-Pérez, C. I., & Clunie-Beaufond, C. E. (2020). El modelo tecnológico para la implementación de un proceso de educación ubicua en un ambiente de computación en la nube móvil. *Revista UIS Ingenierías*, 19(4), 77-88. <https://doi.org/10.18273/revuin.v19n4-2020007>
- Bernal, M. C., & Molina, Y. (2022). A test model for database architectures: an assessment. *Journal of Applied Research and Technology*, 20(3), 306-319. <https://doi.org/10.22201/icat.24486736e.2022.20.3.1169>
- Chingo Esquivel, W., & López Sevilla, G. (2021). Paralelismos entre bases de datos relacionales y no relacionales (un enfoque en seguridad). *ReCIBE, Revista electrónica de Computación, Informática, Biomédica y Electrónica*, 10(2), C1-16. <https://doi.org/10.32870/recibe.v10i2.189>
- Codd, E. F. (1970). A relational model of data for large shared data banks. *Communications of the ACM*, 13(6), 377-387. <https://doi.org/10.1145/362384.362685>
- Derfoufi Ouahbi, M. (2024). *FitNotion: Aplicación web de nutrición* [Trabajo de grado, Universidad de Alicante]. RUA - Repositorio Institucional. <http://hdl.handle.net/10045/145598>
- García, A. B. (2015). *UF2405 - Modelo de programación web y bases de datos* (5.0 ed.). Editorial Elearning S.L.
- García Charcape, A. P., Samamé Uceda, M. A., & Mendoza De los Santos, A. (2023). Análisis de la gestión de servicios de TI en la nube: beneficios y riesgos de su implementación. *INGENIERÍA INVESTIGA*, 5, 1-10. <https://doi.org/10.47796/ing.v5i0.795>
- Lions, P. (2020, 6 de abril). Visualize the 80/20 rule using Oracle Analytics. *Oracle Analytics*. <https://blogs.oracle.com/analytics/post/visualize-the-8020-rule-using-oracle-analytics>

- Microsoft. (2024). *Azure*. <https://azure.microsoft.com/es-es/resources/cloud-computing-dictionary/what-is-azure>
- Microsoft. (2024). *Azure documentation*. <https://learn.microsoft.com/en-us/azure/?product=popular>
- Microsoft. (2024, 9 de mayo). Bases de datos de ejemplo AdventureWorks. <https://learn.microsoft.com/es-es/sql/samples/adventureworks-install-configure?view=sql-server-ver16&tabs=ssms>
- Microsoft. (2024, 9 de abril). Download SQL Server Management Studio (SSMS). <https://learn.microsoft.com/en-us/sql/ssms/download-sql-server-management-studio-ssms?view=sql-server-ver16>
- Microsoft. (2024, 27 de febrero). Prueba Azure SQL Database gratis. <https://learn.microsoft.com/es-es/azure/azure-sql/database/free-offer?view=azuresql>
- Microsoft Learn. (2024, 17 de octubre). FROM: uso de PIVOT y UNPIVOT. <https://learn.microsoft.com/es-es/sql/t-sql/queries/from-using-pivot-and-unpivot?view=sql-server-ver16>
- Parra, R. (2022, 22 de agosto). Microsoft Azure: colaboración y soluciones integrales para industrias y gobiernos. *DPL News Cloud Digital Series*, pp. 1-8. <https://dplnews.com/dpl-cloud-microsoft-azure-colaboracion-y-soluciones-integrales-para-industrias-y-gobiernos-2/>
- Ramakrishnan, R., & Gehrke, J. (2003). *Database management systems*. McGraw-Hill Education.
- Treviño-Reyes, R., Rivera-Rodríguez, F. S., & Garza-Alonso, J. A. (2020). La analítica de datos como ventaja competitiva en las organizaciones. *Vinculatégica EFAN*, 6(2), 1063-1074. <https://doi.org/10.29105/vtga6.2-520>

 REVISTA Más TIC	Vol. 1, No. 2 	diciembre 2024 – mayo 2025 pp. 8-33 ISSN L 3072-9696
--	--	---

Valverde, V., Portalanza, N., & Mora, P. (2019). Análisis descriptivo de base de datos relacional y no relacional. *Atlante: Cuadernos de Educación y Desarrollo*, 2-15.

West, R., Zacharias, M., Assaf, W., Aelterman, S., Davidson, L., & D'Antoni, J. (2019). *SQL Server 2019 administration inside out*. Microsoft Press.