PROCESO DE ETL PRUEBA TÉCNICA

Miguel Enrique Fernández Azucena

Este documento presenta la documentación de la realización de un proceso de ETL realizado en los datos proveídos en la descripción de la prueba técnica

Programas utilizados

- SQL Server
- SQL Server Integration Services
- Visual Studio
- Visual Studio SQL Server Integration Services Projects Extension

Diagrama de archivos en Kaggle

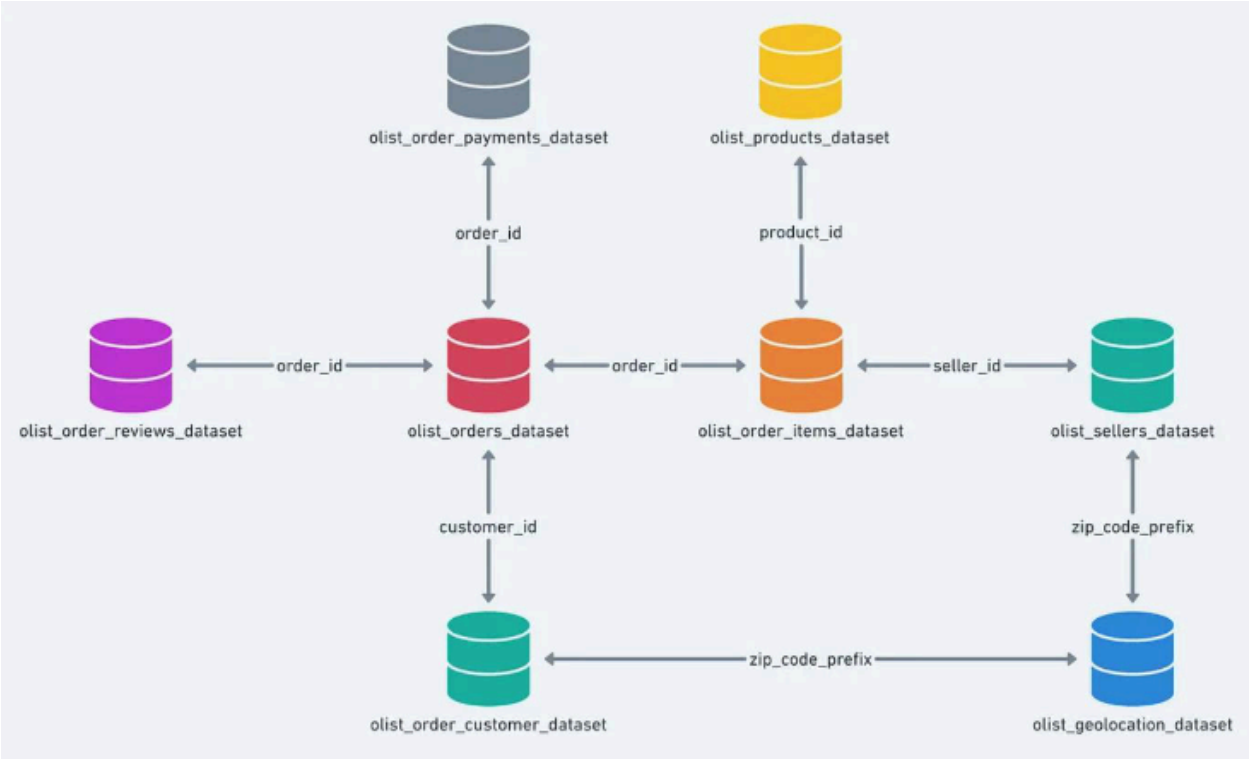

Figura 1. Diagrama de dataset Olist

El diagrama de la figura 1 describe la estructura de un dataset correspondiente a un negocio de e-commerce Brasileño. Cada uno de los iconos representa un archivo csv que contiene data correspondiente a esa tabla, es decir que el archivo olist_order_customer_dataset contiene información acerca de los clientes quienes hacen uso del servicio de e-commerce.

De estas tablas se utilizaron tres archivos para realizar un proceso de extracción, transformación y carga de datos a una base de datos. Los archivos seleccionados son:

- **olist_orders_dataset:** tabla que describe la información crítica de una orden, incluyendo detalles como el cliente, si la orden ya fue entregada y las diferentes fechas y horas relacionadas a la orden.
- **olist_order_items_dataset:** tabla que describe los ítems incluidos en cada orden junto con su precio nominal y su costo de envío.
- **olist_product_dataset:** tabla con información de los productos publicados en el sitio de e-commerce.

Estas tres tablas están relacionadas, siendo la tabla de order_items una tabla normalizada que granuraliza la relación entre las órdenes y los ítems incluidos en cada orden. Se seleccionaron estos archivos ya que proveen información necesaria para formar perspectivas y conocimiento importante con respecto al negocio de e-commerce. De estas tablas se seleccionaron solamente ciertas columnas para su carga a la base de datos. Tanto la relación entre las tablas y las columnas seleccionadas se muestran en el siguiente diagrama entidad-relación:

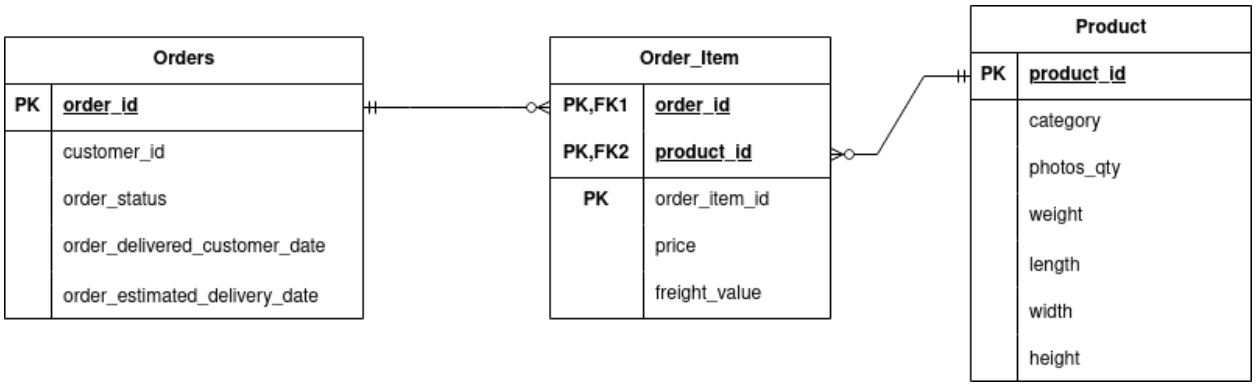

Figura 2. Diagrama entidad-relación

El proceso de ETL se realizó por medio de la herramienta de integración de SQL Server(SSIS por sus siglas en inglés). El proceso se realizó en un servidor de SQL Server local y una solución realizada en Visual Studio. Todos los archivos utilizados en el proceso y sus productos se encuentran en el repositorio de Github adjuntado a la entrega de este trabajo. Las instrucciones de creación de la base de datos se pueden ver en la figura 3.

```sql
CREATE DATABASE PruebaTecnica1
USE PruebaTecnica1
GO

CREATE TABLE orders(
    order_id varchar(50),
    customer_id varchar(50),
    order_status varchar(50),
    delivered_date datetime,
    estimated_delivery_date datetime,
    CONSTRAINT PK_order_id PRIMARY KEY(order_id)
);
GO

CREATE TABLE product(
    product_id varchar(50),
    category varchar(50),
    photos_qty int,
    product_weight float,
    product_length float,
    product_height float,
    product_width float,
    CONSTRAINT PK_product_id PRIMARY KEY(product_id)
    );
GO

CREATE TABLE order_item(
    order_id varchar(50),
    product_id varchar(50),
    order_item_id int,
    price float,
    freight_value float
    CONSTRAINT PK_order_item PRIMARY KEY(order_id,product_id,order_item_id),
    CONSTRAINT FK_item_order_id FOREIGN KEY(order_id) REFERENCES orders(order_id),
    CONSTRAINT FK_item_product_id FOREIGN KEY(product_id) REFERENCES product(product_id)
);
GO
```

Figura 3. Instrucciones de creación de las tablas en la base de datos

Estas columnas se ven reflejadas dentro del proyecto de integración realizado en Visual Studio. Como parte del proyecto se definen los archivos csv como fuentes de datos de archivo plano, como parte de este proceso los datos se mapean con los siguientes tipos.

- **Orders**
    - **Order_id:** string
    - **Customer_id:** string
    - **Order_status:** string
    - **Order_pruchase_timestamp:** database timestamp [DB_DBTIMESTAMP]
    - **Order_approved_at:** database timestamp [DB_DBTIMESTAMP]

    - **Order_delivered_carrier_date:** database timestamp [DB_DBTIMESTAMP]
    - **Order_delivered_customer_date:** database timestamp [DB_DBTIMESTAMP]
    - **Order_estimated_delivery_date:** database timestamp [DB_DBTIMESTAMP]
- **Order_item**
    - **Order_id:** string
    - **Order_item_id:** integer
    - **Product_id:** string
    - **Seller_id:** string
    - **Shipping_limit_date:** database timestamp [DB_DBTIMESTAMP]
    - **Price:** float
    - **Freight_value:** float
- **Product**
    - **Product_id:** string
    - **Product_category_name:** string
    - **Product_name_length:** integer
    - **Product_description_length:** integer
    - **Product_photos_qty:** integer
    - **Product_weigth_g:** float
    - **Product_length_cm:** float
    - **Product_height_cm:** float
    - **Product_width_cm:** float

De esta forma Visual Studio extrae los datos de los archivos para que estos se carguen a la base de datos. En el siguiente paso se seleccionan las columnas de interés y se cargan en la base de datos.
El proceso de carga se debe de hacer en dos pasos, ya que si se busca paralelizar la carga de la tabla order_item junto con las demás tablas se produce un problema de dependencia, por ello se realiza primero la carga de las tablas de orders y product, por último se cargan los valores de la tabla order_item. El primer paso de carga se encuentra dentro del primer data flow mostrado en la figura 4.

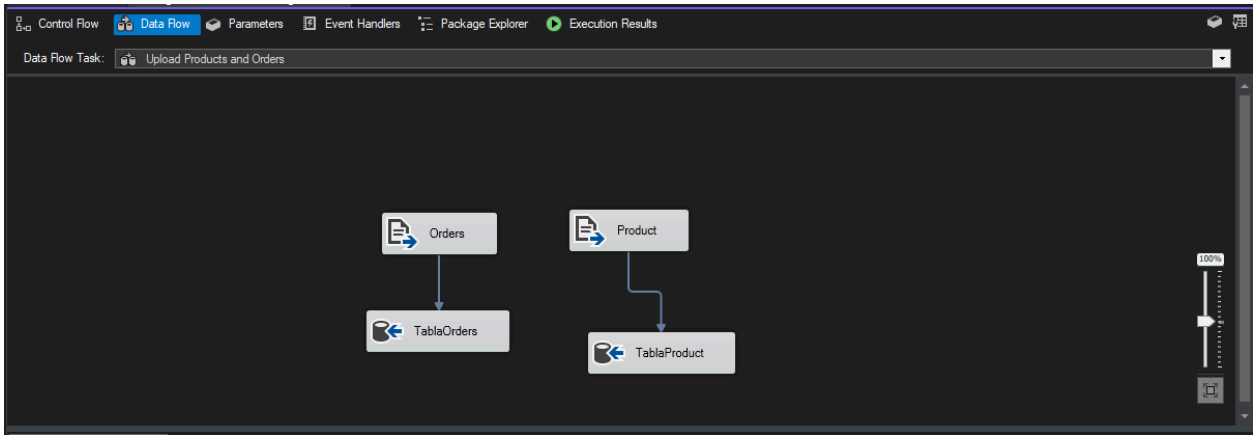

Figura 4. Primer proceso de data flow para subir los datos de productos y orders

Una vez ha terminado ese proceso se procede a cargar los datos de la tabla order_item indicado en su proceso de data flow mostrado en la figura 5.

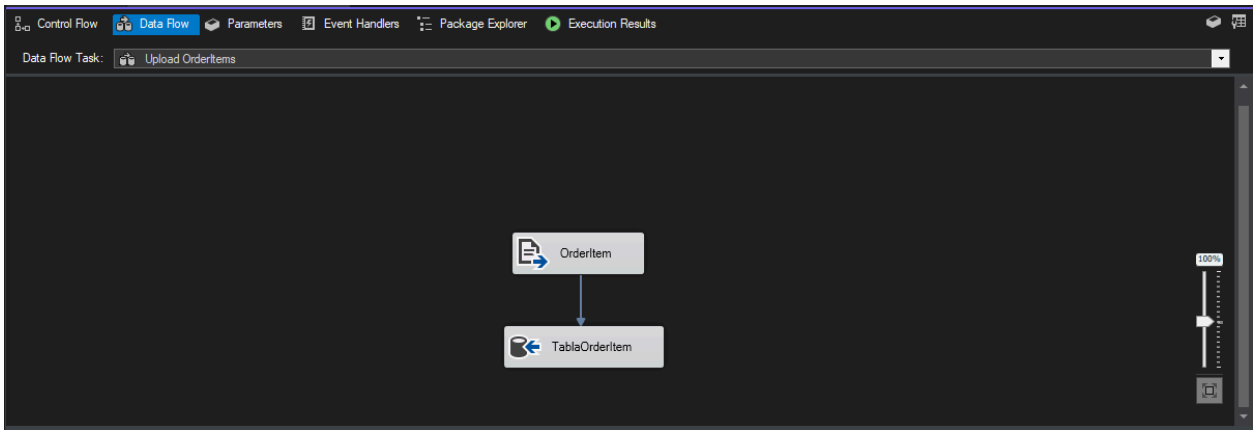

Figura 5. Segundo proceso de dataflow para la carga de datos a tabla order_item.

En la figura 6 se muestra el flujo general del proceso ETL en Visual Studio, donde el primer Dataflow titulado ‘Carga de Product y Orders’ corresponde a la figura 4 y el segundo data flow corresponde a la carga de los datos de la tabla order_item mostrada en la figura 5.

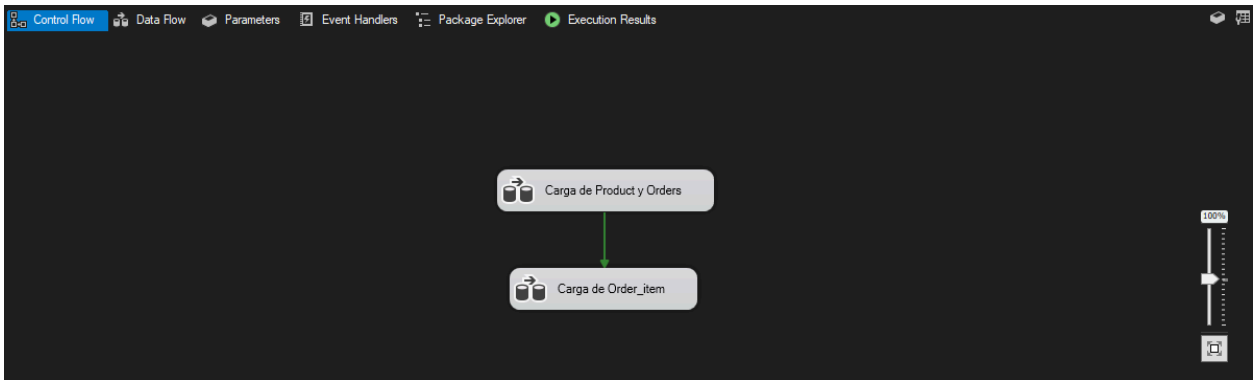

Figura 6. Flujo general de proceso ETL en Visual Studio.

Los resultados se pueden ver cargados en la base de datos por medio de las queries ilustradas en las figuras 7, 8 y 9.

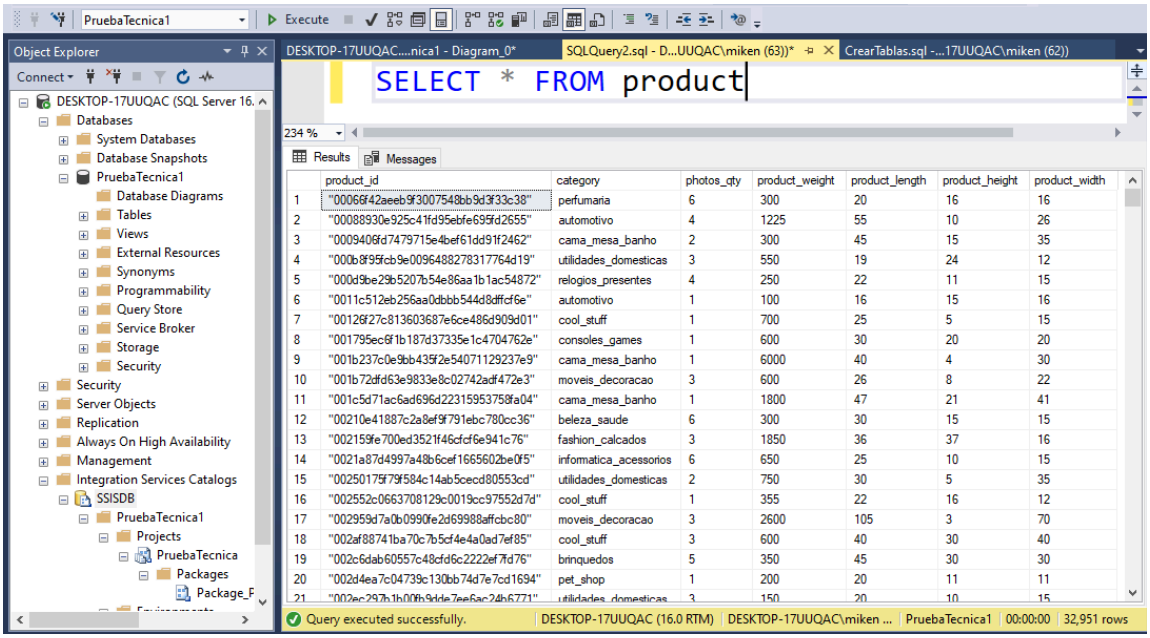

Figura 7. Datos cargados en tabla product

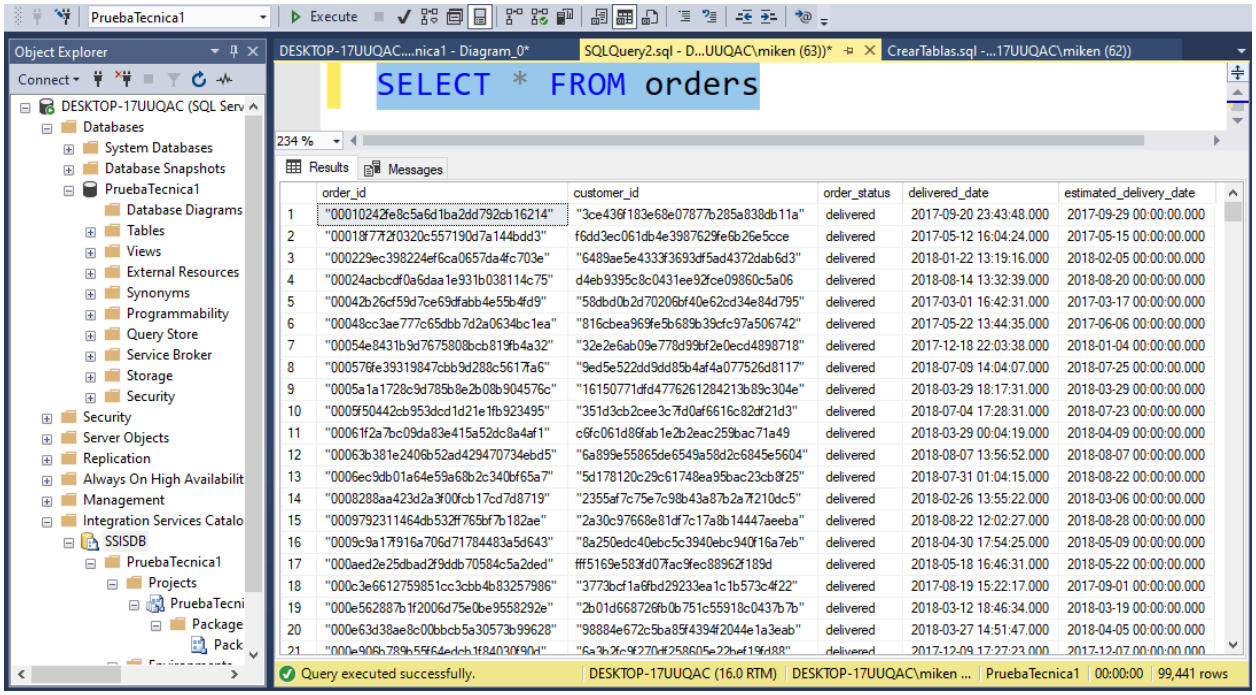

Figura 8. Datos cargados en la tabla orders

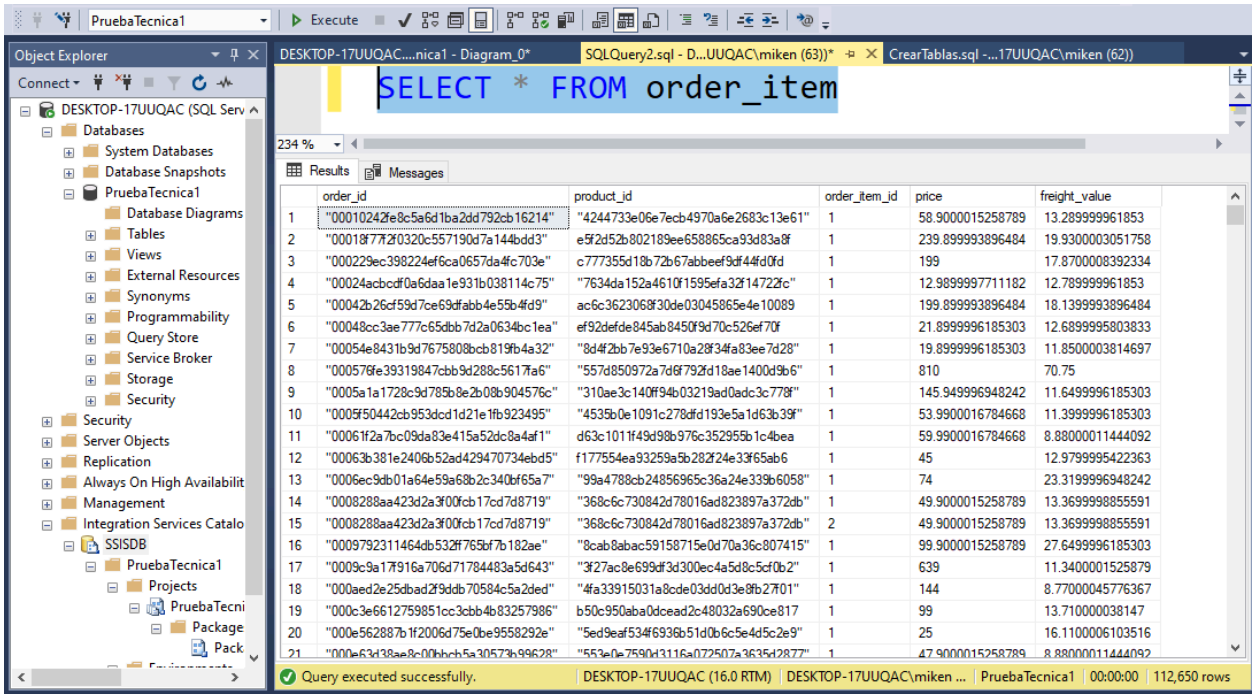

Figura 9. Datos cargados en la tabla order_item

Finalmente se realiza el despliegue del paquete que resulta del proceso de ETL, de esta forma se facilita el acceso al proceso automático de ETL ya que se puede ejecutar desde el mismo servidor de base de datos. Desde Visual Studio se despliega el paquete al catálogo de servicios de integración de SQL Server. En la figura 10 se muestra este proceso realizado de forma satisfactoria.

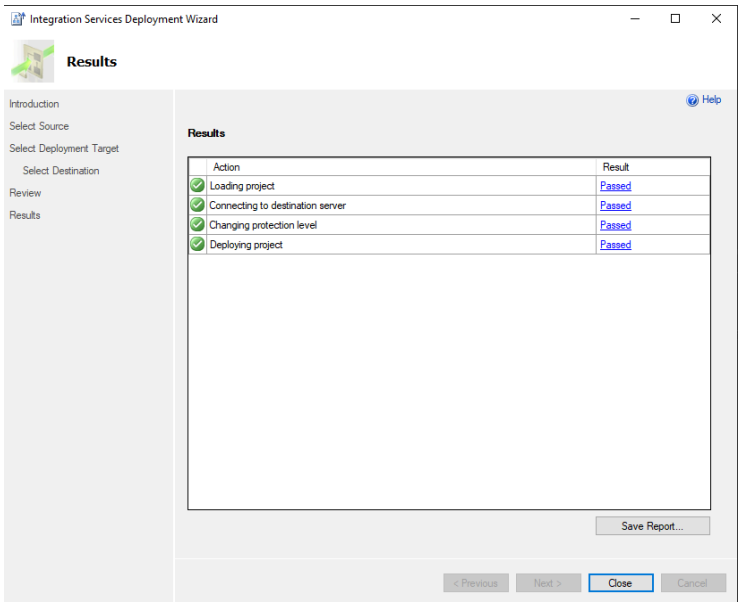

Figura 10. Despliegue de paquete a servicio de integración dentro de servidor SQL

La ejecución de este paquete se puede realizar de la siguiente manera. Se expande el folder de Integration Services Catalog dentro del programa de manejo de SQL Server (SSMS por sus siglas en inglés). Se expanden las carpetas de ‘SSISDB’, ‘PruebaTecnica1’, ‘Projects’, ‘PruebaTecnica’ y ‘Package’ hasta llegar al propio archivo de package.
Luego hacer click derecho en el archivo Package y seleccionar ejecutar tal como lo muestra la figura 11.

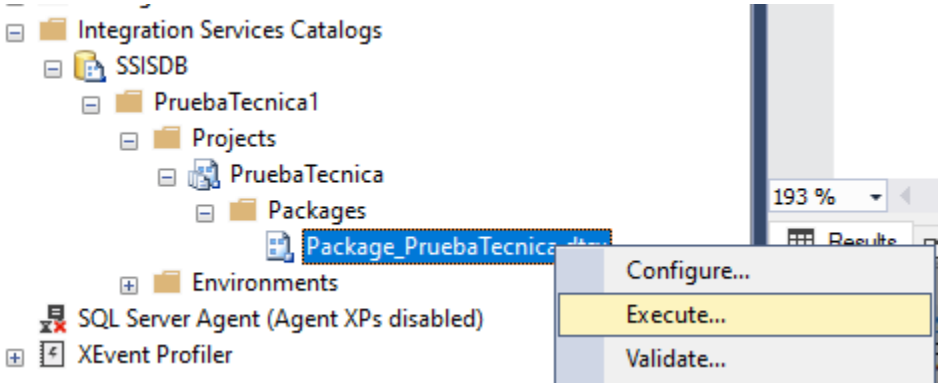

Figura 11. Ejecución de proceso ETL desde SSMS

Fuentes

1. Fuente de datos de servicio brasileño de e-commerce, Kaggle. https://www.kaggle.com/datasets/olistbr/brazilian-ecommerce
2. Proyectos de ETL con SSIS y Visual Studio, Microsoft. https://learn.microsoft.com/en-us/sql/integration-services/ssis-how-to-create-an-etl-package?view=sql-server-ver16
3. Instalación de SSIS, Microsoft. https://learn.microsoft.com/en-us/sql/integration-services/install-windows/install-integration-services?view=sql-server-ver16
4. Tipos de datos de SSIS en Visual Studio, Microsoft. https://learn.microsoft.com/en-us/sql/integration-services/data-flow/integration-services-data-types?view=sql-server-ver16