**Integrantes**

Andrea Estefania Timaran Buchely

Laura Daniela Espinosa Maca

Carlos Enrique Jaramillo Aros

**Titulo**: Proyecto Final Curso Infraestructura y Arquitectura de TI

**Objetivo**: Se busca llevar a cabo el análisis, desarrollo y ejecución de transformaciones (ETL), utilizando la herramienta Pentaho Data Integrator (), usando diferentes fuentes de información como archivo plano (Ames Property.csv) , base de datos estructurada Postgres (creada a partir de scripts compartidos en la plataforma ).

**Contexto:** El conjunto de datos recibido proporciona información detallada sobre diversas propiedades inmobiliarias, representadas por sus atributos y características asociadas. Cada fila representa una propiedad, y las columnas contienen información como el ID de la propiedad (PID), el estilo de la casa, la calidad general, el año de construcción, el área del lote, el área habitable, entre otros detalles, los cuales están divididos en tres grupos de fuentes.

En primer lugar están Ames Properties (Archivo CSV), que contiene información general de la construcción; por otro lado, la base de datos estructurada (postgres) tiene, información relacionada con la construcción de la misma, información como, detalle por cada piso (clase, habitaciones, cocina), al igual que tiene unos clasificadores como tipo de calidad, detalle de ventas de la propiedad. Y por último, en la base de datos NoSQL MongoDB, se cuenta con información del sótano, garage, piscina, entre otros distribuidos en diferentes colecciones.

**ETL (Diseño/Desarrollo):** Para la realización de las Extracciones, transformaciones y cargar se realizó de la sgte manera.

1. En la base de datos POSTGRES, se realizó la creación e inserción de datos en las tablas que fueron entregadas como insumos para el proyecto, se realizaron consultas de base de datos previas para simular las transformaciones a realizar.

Ej.



Nota importante: fue necesario crear dos instancias de base de datos en ElephanSQL por la restricción de conexiones por instancia (dado que es free permite solo 5)

1. Para la base de información de propiedades (CSV); Dada la información compartida como insumo para la columna Neighborhood, se crea la tabla y se inserta para luego hacer búsqueda (lookup) y tomar las abreviaciones correspondientes, para los demás casos (Condition1, Condition2, LotShape) se realizaron mappings directos (Value Mapping) en el proceso de ETL, dado es una cantidad de registros limitada y controlable, seria preferible manejar otra fuente de información para controlar estos parámetros de homolohaciones.

Script:



Nota, para los valores con entregados 'Landmrk','Greens' ,'GrnHill' fue necesario agregar tres registros faltantes en los insumos.

1. Para el caso de la base de datos MongoDB, se realizaron los merges correspondientes para unir la información de bsmt, garage, misc y pool pero se encontraron novedades considerables que podrían generar perdida de información, de igual manera, dada su naturaleza (NoSQL) se recibe información en vacio, por tanto es necesario realizar dummificacion de valores nulos.

Compare:



Para ver mas detalle, se recomienda ingresar al archivo ktr, que contiene la información de los procesos de ETL realizados.



Lecciones aprendidas:

La función de Lookup es herramienta para optimizar las operaciones de fusión y búsqueda de datos, proporcionando una alternativa eficaz.

La correcta definición de los tipos de datos facilita significativamente la tarea de cruce, evitando posibles incidentes relacionados con incompatibilidades de tipos

Referencias:

[1] Hitachi Vantara. "Pentaho Tutorials." [Online]. Available: <https://www.hitachivantara.com/en-us/products/pentaho-plus-platform/data-integration-analytics/pentaho-tutorials.html#getting-started>.

[2] PostgreSQLTutorial.com. "PostgreSQL Cheat Sheet." [Online]. Available: <https://www.postgresqltutorial.com/wp-content/uploads/2018/03/PostgreSQL-Cheat-Sheet.pdf>.

[3] Pentaho. "Pentaho Documentation." [Online]. Available: <https://pentaho-public.atlassian.net/wiki/spaces/EAI/pages/376442127/If+field+value+is+null>.

[4] ElephantSQL. "ElephantSQL API." [Online]. Available: <https://api.elephantsql.com/>.