



**curso gratuito**

# [Trabajo integrador: *Buscando nuestro público objetivo* 🧑]

*Análisis de datos*

# Público objetivo

## Premisa

Hoy día, los datos se han convertido en el **mayor activo** que poseen las organizaciones para **fundamentar sus decisiones y dirigir estrategias empresariales efectivas**; esto va desde poder analizar patrones de comportamiento hasta prever tendencias futuras.

En este contexto, entra **InnovateTech**, una empresa líder en tecnología que, con nuestra ayuda, busca **construir y mantener su plataforma de datos**. Esta plataforma no solo respalda la toma de decisiones estratégicas a través del análisis e interpretación de datos, sino que también permite que la empresa pueda examinar y organizar su información con diferentes propósitos: ajustar sus productos y servicios, detectar nichos de mercado, expandirse, mejorar su atención al cliente, entre otros.

En este caso, el desafío al que se enfrentan es **comprender mejor a su público objetivo**. Están buscando, con nuestro apoyo, entender a fondo a sus consumidores: a qué se dedican, qué hacen, qué edad tienen, qué les gusta, qué les duele, y qué necesitan. Esto les permitirá optimizar, contextualizar y personalizar los mensajes hacia sus clientes lo que a su vez les ayudará a definir mejores campañas y obtener leads de mayor calidad, es decir, potenciales clientes más alineados con sus productos y servicios.

Como miembro del equipo de Ingeniería de Datos de la empresa y aplicando los conocimientos adquiridos en la unidad de Excel, Python y análisis de datos, te enfocarás en **extraer, analizar y gestionar datos clave**.

## Propósito general

### Manipulación de Datos con Python:

- Utiliza la biblioteca Pandas en Python para cargar un archivo CSV que contiene datos de ventas de diferentes sucursales de la empresa.
- Limpia y prepara los datos utilizando funciones de manipulación de datos en Python, como la eliminación de filas duplicadas y el manejo de valores nulos.

## Análisis de Datos con Python:

- Utiliza NumPy para realizar cálculos de álgebra lineal, como la multiplicación de matrices y el cálculo de determinantes.
- Utiliza Pandas para generar un reporte descriptivo que incluya estadísticas descriptivas básicas y tablas dinámicas sobre las ventas de las diferentes sucursales.

## Presentación de Informes con Excel:

- Exporta los datos limpios y el reporte descriptivo generado en Python a un archivo Excel.
- Utiliza fórmulas y funciones de Excel para calcular métricas adicionales, como el total de ventas y el promedio de ventas por sucursal.
- Aplica formatos personalizados y estilos visuales para mejorar la presentación de la información en el archivo Excel.

## Entregables

- Archivo Excel que contiene los datos limpios, el reporte descriptivo y las métricas adicionales calculadas.
- Archivo .py con el código utilizado para el análisis de datos en Python.

## ¿Qué vamos a validar?

Estos son algunos puntos que vamos a validar:

### A. Cumplimiento de lo requerido por archivo:

- ✓ Utiliza Pandas: para cargar y limpiar los datos de ventas.
- ✓ Utiliza NumPy: para realizar cálculos de álgebra lineal según sea necesario.
- ✓ Utiliza Pandas: para generar el reporte descriptivo de ventas.
- ✓ Exporta los datos y el reporte a un archivo Excel.

- ✓ Utiliza Excel: para calcular métricas adicionales y mejorar la presentación del informe.

## **B. Calidad del código: Programación en Python**

- ✓ Modularización: Revisar si el código está organizado en módulos o funciones de manera lógica, facilitando su comprensión, mantenimiento y reutilización.
- ✓ Documentación: Comprobar que el código esté adecuadamente documentado, incluyendo comentarios explicativos, descripciones de funciones y notas sobre la lógica utilizada.
- ✓ Buenas prácticas: Evaluar si se siguen las buenas prácticas de programación, como nombramiento coherente de variables, estructuras de código limpias, y uso eficiente de recursos.

## **C. Eficiencia y optimización:**

- ✓ Rendimiento del Código: Revisar el código en busca de posibles ineficiencias que puedan ralentizar la ejecución o consumir recursos innecesarios.
- ✓ Optimización de Consultas: Asegurar que las consultas SQL y las operaciones de Pandas se realicen de manera eficiente, especialmente en el manejo de grandes volúmenes de datos.

## **D. Seguridad y privacidad:**


- ✓ Manejo de Datos Sensibles: Verificar que se respeten las prácticas de seguridad y privacidad de datos, especialmente si se manejan datos personales o confidenciales.
- ✓ Protección contra Vulnerabilidades: Asegurarse de que el código esté libre de vulnerabilidades comunes, como inyecciones SQL en las consultas.

## **E. Presentación y reporte de resultados:**

- ✓ Claridad en los Informes: Comprobar que los informes generados sean claros, bien estructurados y que presenten los hallazgos de manera comprensible.

- ✓ Fundamentación de Conclusiones: Asegurarse de que las conclusiones y recomendaciones estén bien fundamentadas y basadas en los datos y análisis realizados.

## Requerimientos

→ Analizar los archivos proporcionados para **comprender su estructura y los datos** que contienen  :

- ecommerce\_customers\_dataset.csv
- ecommerce\_orders\_dataset.csv
- ecommerce\_order\_items\_dataset.csv
- ecommerce\_products\_dataset.csv
- ecommerce\_order\_payments\_dataset.csv

→ Realizar las siguientes **actividades con pandas** :

- Cargar los archivos CSV en DataFrames de Pandas
- Establecer la columna índices de las tablas como la clave primaria en el DataFrame, ejemplo para customers la clave será customer\_id.
- Obtener el número total de clientes únicos en el conjunto de datos
- Calcular el promedio de valor de pago por pedido
- Determinar la categoría de producto más vendida
- Calcular el número total de pedidos realizados

## Recursos

- ▶ [ecommerce\\_customers\\_dataset.csv](#)
- ▶ [ecommerce\\_order\\_items\\_dataset.csv](#)
- ▶ [ecommerce\\_order\\_payments\\_dataset.csv](#)
- ▶ [ecommerce\\_orders\\_dataset.csv](#)
- ▶ [ecommerce\\_products\\_dataset.csv](#)

¡Manos a la obra! 