

2023

HITO GRUPAL PROGRAMACIÓN

GRUPO VERMAX

FASE 1

Esta fase es una explicación “teórica” de cuatro puntos fundamentales:

1. **Hablamos de fuentes de datos. De grandes volúmenes de datos. Por ejemplo, data Lake o similar. También es importante tratar la diferencia entre datos estructurados y no estructurados en relación al big Data.**

Una fuente de datos es un lugar en el cual se recopila la información.

Puede ser una base de datos, un archivo plano, o documentos XML o de cualquier otro formato.

Las fuentes de datos se pueden diferenciar por tipos (fuente de datos de máquinas y fuente de datos de archivos.)

Tipos de fuente de datos:

1.Fuentes de datos automáticas:

Las fuentes de datos automáticas, al igual que otras fuentes de datos, ofrecen toda la información necesaria para conectarse a los datos, incluidas las fuentes de software necesarias y los administradores de controladores

2.Fuente de datos de archivos:

Al igual que con cualquier base de datos informática, las fuentes de datos de archivos se pueden editar y copiar. Los usuarios y los sistemas pueden compartir una conexión común (transmitir fuentes de datos entre ordenadores o servidores individuales), y el proceso de conexión de datos puede simplificarse.

3.Funciones de una fuente de datos:

Las fuentes de datos están diseñadas para ayudar a los usuarios y las aplicaciones a conectarse y mover datos a la ubicación adecuada. Recopilan y almacenan datos técnicos importantes en un solo lugar. De esta manera, los usuarios pueden procesar y determinar el mejor uso de los datos.

Si hablamos de grandes volúmenes de fuentes de datos, estamos hablando por ejemplo de Big Data.

Big Data (grandes datos o grandes volúmenes de datos), consiste en un proceso de análisis e interpretación de grandes volúmenes de datos estructurados y no estructurados. Sirve para que los datos almacenados de forma remota puedan ser utilizados por las empresas como base para su toma de decisiones

Diferencias entre datos estructurados y no estructurados:

Los datos estructurados son datos predefinidos, generalmente solo texto y fáciles de buscar, mientras que los datos no estructurados no son modelos de datos predefinidos, pueden ser en texto, imágenes, sonido, vídeos u otros formatos, y su búsqueda y análisis es más difícil.

2. Entre las herramientas más interesantes a la hora de gestionar grandes volúmenes de datos nos encontramos con Hadoop y Spark. Habría que tratar sus características y finalidad.

Hadpool: Es una estructura de software de código abierto útil para almacenar datos y ejecutar aplicaciones en clústeres de hardware. Proporciona almacenamiento para cualquier tipo de datos.

Tienen un enorme poder de procesamiento y capacidad de procesar tareas o trabajos complejos.

Hadpool tiene un procesamiento distribuido, eficiente, económico y fácil de utilizar.

Spark: Apache Spark es un motor de procesamiento distribuido que administra, distribuye y monitorea aplicaciones de software que se ejecutan en un grupo de estaciones de trabajo, (en clúster).

Spark tiene un alto rendimiento del uso de datos y transmisión por lotes.

3. Existen lenguajes de programación “recomendables” para gestionar datos. Entre ellos, están Python y Scala. Sería explicar brevemente por qué.

Python es un lenguaje de programación interpretado ampliamente utilizado en las aplicaciones web, el desarrollo de software, la ciencia de datos y el machine learning (ML).

Es un lenguaje fácil de aprender, eficiente y se integra bien a todos los tipos de sistemas.

Tiene una sintaxis básica y cuenta con una amplia biblioteca estándar que permite reutilizar código y ahorrar tiempo en el desarrollo. Se puede utilizar en dispositivos con diferentes sistemas operativos como Windows, macOS, Linux y Unix.

Python es un lenguaje de programación orientado a objetos de código abierto, que agrupa datos y funciones para lograr flexibilidad. En la ciencia de datos, Python suele utilizarse para el procesamiento de datos, la implementación de algoritmos de análisis de datos y el entrenamiento de algoritmos de aprendizaje automático y aprendizaje profundo.

Scala es un lenguaje de programación de propósito general, es de código abierto, integra principios de orientación a objetos y la programación funcional, permitiendo a los programadores ser más productivos y aprovechar los conocimientos y estructuras de otros lenguajes como Java.

Scala es una extensión de Java, un lenguaje fuertemente asociado a la ingeniería de datos, con interoperabilidad gracias a la compilación del bytecode de Java y su ejecución en la máquina virtual de Java. Construido como respuesta a los problemas percibidos en Java, es un lenguaje más nuevo y elegante. Scala permite crear marcos de trabajo de alto rendimiento para el manejo de datos en silos, perfectos para la ciencia de datos a nivel empresarial.

4. En la parte de visualización de datos, de mostrar dashboards nos encontramos con PowerBI y Tableau entre otros. Debemos explicar qué son.

Power BI

Es una plataforma unificada y escalable de inteligencia empresarial (BI) con funciones de autoservicio apta para grandes empresas.

Nos permite acceder a nuestros datos de forma segura y rápida, generando grandes beneficios para nosotros y para nuestra empresa. Es un sistema predictivo, inteligente y de gran apoyo, capaz de traducir los datos (simples o complejos) en gráficas, paneles o informes por sus cualidades como la capacidad gráfica de presentación de la información, o la integración de Power Query: el motor de extracción, transformación y carga (ETL) incluido en Excel.

Tableau

Es una herramienta de análisis empresarial creada con el objetivo de mejorar el flujo del análisis y poner los datos al alcance de las personas a través de la visualización. Puede conectarse a varias fuentes de datos sin necesidad de ninguna programación, como por ejemplo Redshift, Cloudera Hadoop, SQL Server, Salesforce, Google Analytics y Google Sheets, MongoDB, archivos PDF, Dropbox, Amazon Athena, entre otros.

Fase 2.

En esta segunda fase se realiza la implementación de la investigación. En concreto sería acceder a un volumen de datos y mostrarlo. Podríamos utilizar Scala o Python y mostrar el resultado en PowerBI o Tableau. La idea es que sea algo muy impactante por la calidad de contenido tratado, velocidad de acceso, volumen de datos....

Para este apartado lo primero que tendremos que hacer será instalar las librerías *pandas* y *matplotlib*, que las necesitaremos para poder visualizar los datos desde PowerBI.

```
C:\Users\David>pip install pandas
Collecting pandas
  Downloading pandas-1.5.3-cp310-cp310-win_amd64.whl (10.4 MB)
    ----- 10.4/10.4 MB 86.2 kB/s eta 0:00:00
Collecting pytz>=2020.1
  Downloading pytz-2022.7.1-py2.py3-none-any.whl (499 kB)
    ----- 499.4/499.4 kB 89.2 kB/s eta 0:00:00
Requirement already satisfied: python-dateutil>=2.8.1 in c:\users\david\AppData\Local\Programs\Python\Python310\lib\site-packages (from pandas) (2.8.2)
Collecting numpy>=1.21.0
  Downloading numpy-1.24.2-cp310-cp310-win_amd64.whl (14.8 MB)
    ----- 14.8/14.8 MB 120.6 kB/s eta 0:00:00
Requirement already satisfied: six>=1.5 in c:\users\david\AppData\Local\Programs\Python\Python310\lib\site-packages (from python-dateutil>=2.8.1->pandas) (1.16.0)
Installing collected packages: pytz, numpy, pandas
Successfully installed numpy-1.24.2 pandas-1.5.3 pytz-2022.7.1

C:\Users\David>pip install matplotlib
Collecting matplotlib
  Downloading matplotlib-3.7.1-cp310-cp310-win_amd64.whl (7.6 MB)
    ----- 7.6/7.6 MB 1.6 MB/s eta 0:00:00
Collecting contourpy>=1.0.1
  Downloading contourpy-1.0.7-cp310-cp310-win_amd64.whl (162 kB)
    ----- 163.0/163.0 kB 1.6 MB/s eta 0:00:00
Collecting pillow>=6.2.0
  Downloading Pillow-9.4.0-cp310-cp310-win_amd64.whl (2.5 MB)
    ----- 2.5/2.5 MB 665.6 kB/s eta 0:00:00
Collecting cycler>=0.10
  Downloading cycler-0.11.0-py3-none-any.whl (6.4 kB)
Requirement already satisfied: six>=1.5 in c:\users\david\AppData\Local\Programs\Python\Python310\lib\site-packages (from contourpy>=1.0.1->matplotlib) (1.16.0)
Requirement already satisfied: python-dateutil>=2.7->matplotlib) (1.16.0)
Installing collected packages: pyparsing, pillow, packaging, kiwisolver, fonttools, cycler, contourpy, matplotlib
Successfully installed contourpy-1.0.7 cycler-0.11.0 fonttools-4.39.2 kiwisolver-1.4.4 matplotlib-3.7.1 packaging-23.0 pyparsing-3.0.9 pillow-9.4.0 pyparsing-3.0.9
```

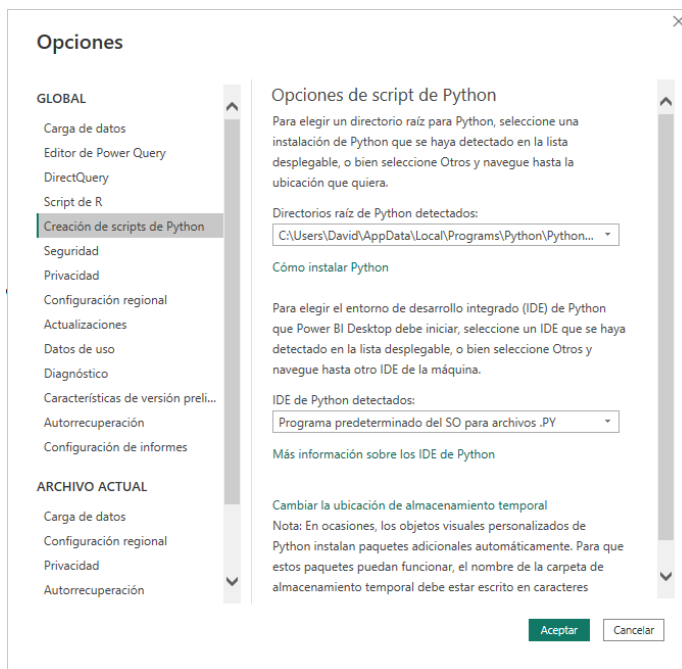
Ahora, instalaremos PowerBI Desktop, donde más tarde visualizaremos los datos.



Para crear el volumen de datos, aquí tengo un código que nos genera un csv con 100 registros.

```
crearCSV.py x Release Notes: 1.76.2
C: > Users > David > Desktop > crearCSV.py
1 import csv
2 import random
3
4 # Abrir archivo CSV en modo escritura
5 with open('datos_prueba.csv', mode='w', newline='') as file:
6
7     # Crear objeto writer de CSV
8     writer = csv.writer(file)
9
10    # Escribir encabezados
11    writer.writerow(['id', 'nombre', 'edad', 'ciudad'])
12
13    # Escribir 100 registros aleatorios
14    for i in range(1, 101):
15        writer.writerow([i, f'Usuario {i}', random.randint(18, 65), random.choice(['Ciudad A', 'Ciudad B', 'Ciudad C'])])
```

Ahora que ya tenemos el volumen de datos, abriremos PowerBI Desktop e iremos a la configuración, donde activaremos la creación de scripts de Python, confirmaremos que la ruta es correcta y le daremos a aceptar.

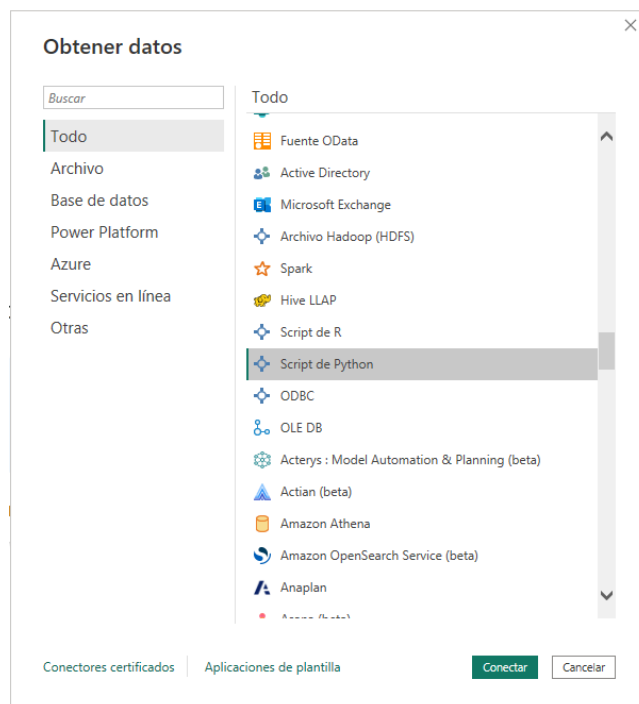


Ahora, crearemos y ejecutaremos un script de Python que nos permita mostrar los 50 primeros datos de nuestro archivo.

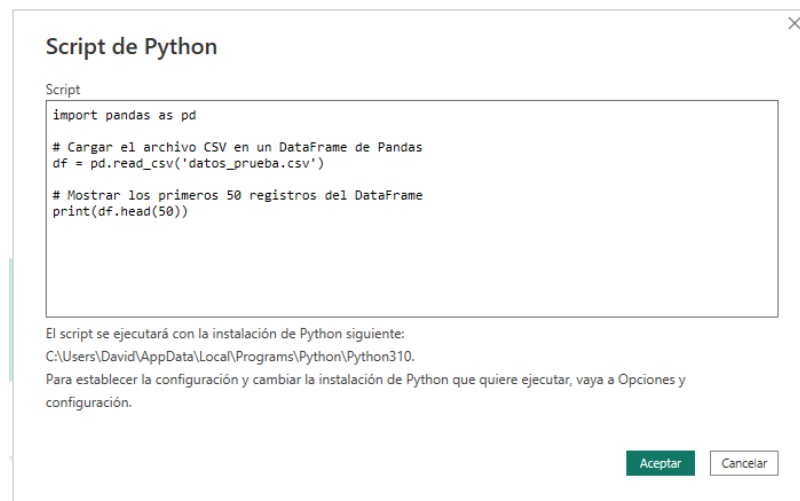
- En **PowerBI**, iremos al apartado de arriba y le daremos a **Obtener Datos**.




- Bajaremos hasta encontrar la opción **Script de Python**.





- Escribimos nuestro script de Python.




- # Navegador



Opciones de presentación ▾


 Python [1]

☒  df

df

id	nombre	edad	ciudad
1	Usuario 1	45	Ciudad A
2	Usuario 2	24	Ciudad B
3	Usuario 3	18	Ciudad A
4	Usuario 4	35	Ciudad B
5	Usuario 5	48	Ciudad C
6	Usuario 6	65	Ciudad C
7	Usuario 7	47	Ciudad C
8	Usuario 8	49	Ciudad C
9	Usuario 9	52	Ciudad A
10	Usuario 10	25	Ciudad B
11	Usuario 11	44	Ciudad B
12	Usuario 12	58	Ciudad C
13	Usuario 13	61	Ciudad C
14	Usuario 14	30	Ciudad C
15	Usuario 15	34	Ciudad B
16	Usuario 16	50	Ciudad B
17	Usuario 17	45	Ciudad A
18	Usuario 18	51	Ciudad C
19	Usuario 19	39	Ciudad B
20	Usuario 20	39	Ciudad A
21	Usuario 21	27	Ciudad A
22	Usuario 22	32	Ciudad A
23	Usuario 23	53	Ciudad A
24	Usuario 24	60	Ciudad B

Cargar

Transformar datos

-

Fase 3.

Para finalizar, realizamos una evaluación o consideraciones de cómo han evolucionado el acceso a datos en los últimos años. Desde acceso a ficheros, pasando por base de datos y consumiendo APIs.

El **acceso a los datos** ha aumentado significativamente en los últimos años. En el pasado, el acceso a los datos estaba principalmente restringido a los archivos almacenados localmente en su ordenador. Sin embargo, con la llegada de Internet y las nuevas tecnologías, se han desarrollado nuevas formas de acceder a los datos.

Una de las formas más comunes de acceder a los datos hoy en día es a través de **bases de datos**. Una base de datos es un sistema que permite el almacenamiento, la organización y la recuperación eficientes de la información.

Las **bases de datos** se utilizan en una amplia variedad de aplicaciones, desde la gestión de inventario hasta la creación de aplicaciones empresariales. Además, a medida que las aplicaciones web y móviles crecen en popularidad, también lo hace la demanda de datos accesibles a través de API (interfaces de programación de aplicaciones). Una API es una interfaz que permite la comunicación entre aplicaciones y sistemas externos a través de un conjunto de comandos.

Hoy, el acceso a los datos también se extiende a nuevas tecnologías como la inteligencia artificial, el aprendizaje automático y el análisis de datos en tiempo real. Esto ha impulsado la demanda de datos en tiempo real de alta calidad y ha estimulado la creación de nuevas tecnologías de acceso a datos.

En resumen, el acceso a los datos ha evolucionado significativamente en los últimos años, pasando de archivos locales a bases de datos y API y expandiéndose a nuevas tecnologías como inteligencia artificial y análisis. Esto ha aumentado la demanda de datos de alta calidad y ha estimulado el desarrollo de nuevas tecnologías de acceso a datos.