



ESCUELA POLITÉCNICA NACIONAL

ESCUELA DE FORMACIÓN DE TECNÓLOGOS



ANÁLISIS DE DATOS

ASIGNATURA:

ANÁLISIS DE DATOS

PROFESOR:

Ing. Lorena Chulde

PERÍODO ACADÉMICO:

2023-B

PROYECTO FINAL – BIMESTRE 2

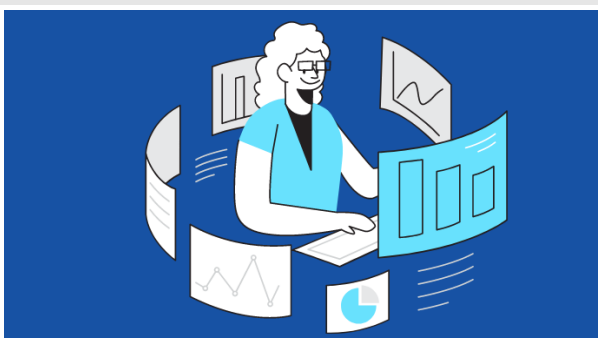
NOMBRES DE LOS INTEGRANTES:

Marcelo Pinzón

Silvia Chaluisa

Jhonatan Bautista

Erick Caiza



2023-B

Definición de Caso de estudio

En el marco teórico-práctico, se pretende la comprensión y análisis de los datos que se almacenan en una arquitectura de Data Lake para la toma de una decisión o examinar fenómenos globales y el impacto que generan en el panorama social.

El caso de estudio abarcará temas como: política en el Ecuador, eventos y conciertos en el mundo, ciencia y salud en el Ecuador, con ello, se propone la creación de un repositorio que albergará los archivos y data sets para su limpieza y exportación a un concentrador de datos mediante scripts, y para la visualización y análisis de los datos importados se utiliza la herramienta Power BI Desktop.

Objetivo General

Analizar los datos de distintas fuentes de información tales como Wikipedia, Datos Abiertos, Kaggle, entre otras fuentes, mediante el uso de Web Scraping, visualización de la información mediante la herramienta Power BI Desktop, con la finalidad de obtener información limpia y entendible para el proyecto.

Objetivos Específicos.

1. Crear y diseñar un Data Lake que permita el almacenamiento de distintos data sets para su previa

limpieza, exportación al concentrador de datos MySQL y visualización mediante la herramienta Power BI Desktop.

2. Desarrollar un análisis de datos de cada caso de estudios mediante la redacción de un informe técnico y detallado
3. Resumir de forma detallada las conclusiones de cada caso de estudio para una mejor comprensión de los dashboards.
4. Contrastar la información de los data sets mediante la creación de dashboards con el uso de herramienta Power BI Desktop

Descripción del equipo de trabajo y actividades realizadas por cada uno.

El equipo de trabajo está conformado por cuatro personas y las actividades realizadas por cada uno se describen en las

Integrantes	Actividades realizadas					
	Creación del repositorio para el trabajo en conjunto	Búsqueda de al menos tres fuentes de información/persona	Limpieza de los data sets con el uso de Jupyter Notebook	Exportación a la base de datos MySQL WorkBrench	Realización de los dashboards en Power BI	Realización del informe
Marcelo Pinzón	X	X	X	X	X	X
Silvia Chaluisa		X	X	X	X	X
Jhonatan Bautista		X	X	X	X	X
Erick Caiza		X	X	X	X	X

Cronograma de actividades

En **Figura 1** se detalla el cronograma de las actividades realizadas para la realización del proyecto:

Figura 1.

Cronograma de actividades mediante el uso de la herramienta Project

		Modo de tarea	Nombre de tarea	Duración	Comienzo	Fin
1			INTEGRACIÓN DEL GRUPO DE TRABAJO	1 día	jue 08/02/24	jue 08/02/24
2			ELECCIÓN DE CASOS DE ESTUDIOS	1 día	vie 09/02/24	vie 09/02/24
3			BÚSQUEDA DE LAS FUENTES DE INFORMACIÓN - DATASETS	3 días	sáb 10/02/24	mar 13/02/24
4			IMPORTACIÓN DE LOS DATA SETS A MYSQL WORKBRENCH	3 días	mié 14/02/24	vie 16/02/24
5			GENERACIÓN DE LOS DASHBOARDS - POWER BI	4 días	sáb 17/02/24	mié 21/02/24
6			REDACCIÓN DEL INFORME	5 días	dom 25/02/24	jue 29/02/24
7			REALIZACIÓN DEL VIDEO EXPLICATIVO	1 día	lun 04/03/24	lun 04/03/24
8			PRESENTACIÓN DEL PROYECTO	1 día	mar 05/03/24	mar 05/03/24

En la **Figura 2**, se visualiza el Diagrama de Gantt generado por el cronograma de actividades

Figura 2.

Diagrama de Gantt

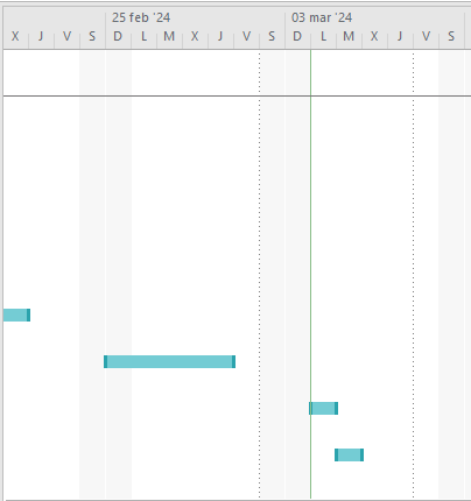


Diagrama de Gantt

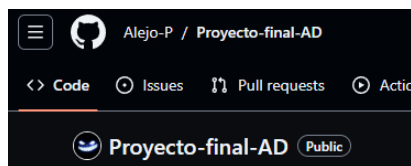
Recursos y herramientas utilizadas

- Power BI Desktop
- MySQL Workbench
- Clever Cloud
- GitHub
- Concentrador de datos
- Project
- Jupyter Notebook

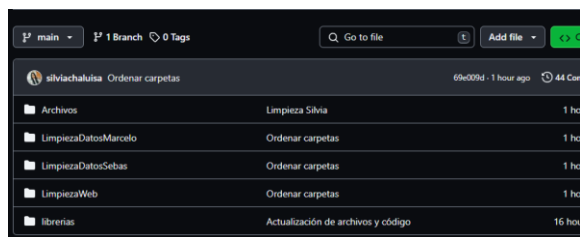
Arquitectura de la solución

La arquitectura del Data Lake permite almacenar todas las fuentes de información por carpetas con el número de caso correspondiente.

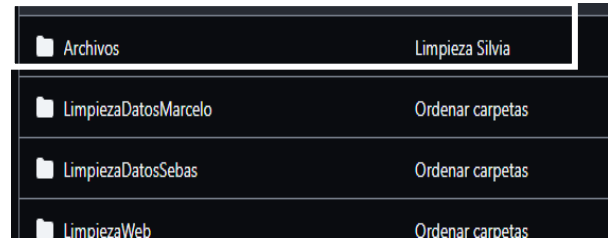
1. Se crea un repositorio en GitHub



2. Se crea las carpetas que se organizan de la siguiente manera:



3. Cada integrante del grupo tiene que importar sus fuentes de información por cada caso de estudio entrando en la carpeta “Archivos”



Extracción de datos.

4. Cuando se cumpla con al menos doce fuentes de información, se procede a la limpieza de nulos/vacíos que puede contener los data sets mediante la librería Pandas y el uso de la herramienta Jupyter Notebook

Instalación de librerías

```
pip install --upgrade pip
Requirement already satisfied: pip in c:\users\user\appdata\local\programs\python\python38\python.exe (20.0.2)
Requirement already satisfied: setuptools in c:\users\user\appdata\local\programs\python\python38\python.exe (57.0.0)
Requirement already satisfied: wheel in c:\users\user\appdata\local\programs\python\python38\python.exe (0.36.0)
Requirement already satisfied: pip in c:\users\user\appdata\local\programs\python\python38\python.exe (20.0.2)
Requirement already satisfied: setuptools in c:\users\user\appdata\local\programs\python\python38\python.exe (57.0.0)
Requirement already satisfied: wheel in c:\users\user\appdata\local\programs\python\python38\python.exe (0.36.0)
Requirement already satisfied: pip in c:\users\user\appdata\local\programs\python\python38\python.exe (20.0.2)
Requirement already satisfied: setuptools in c:\users\user\appdata\local\programs\python\python38\python.exe (57.0.0)
Requirement already satisfied: wheel in c:\users\user\appdata\local\programs\python\python38\python.exe (0.36.0)
```

La importación de las librerías y el uso correcto para la exportación de los datos

```
import pandas as pd

# Cargar Los datos del archivo CSV
registro = pd.read_csv("inabio_inaturalist_bdd_2024enero.csv", encoding="latin-1")
registro
```

	Fecha Inicio	AÑO	Fecha Fin	Observaciones	Especies	Identificadores	Observadores
0	01/01/2019	2023	31/12/2023	1.155.656	28.130	12.865	23.804
1	01/01/2019	2023	30/11/2023	1.139.475	27.909	12.762	23.577
2	01/01/2019	2023	31/10/2023	1.106.016	27.535	12.573	22.966
3	01/01/2019	2023	30/09/2023	1.088.803	27.332	12.468	22.675
4	01/01/2019	2023	31/08/2023	1.076.104	27.144	12.390	22.492
5	01/01/2019	2023	30/07/2023	1.055.841	26.865	12.274	22.038

Eliminación de nulos y datos vacíos

```

: #Revisión de nulos
registro.isnull().sum()

: Fecha Inicio      0
  A%0              0
  Fecha Fin        0
  Observaciones     0
  Especies          0
  Identificadores   0
  Observadores      0
dtype: int64

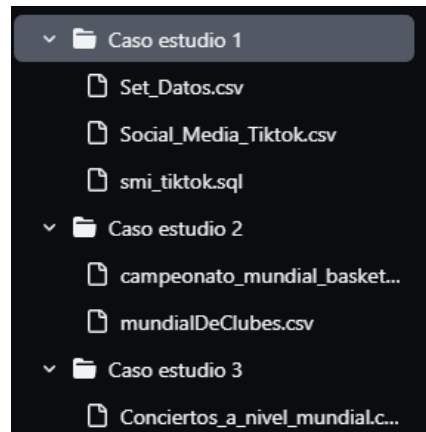
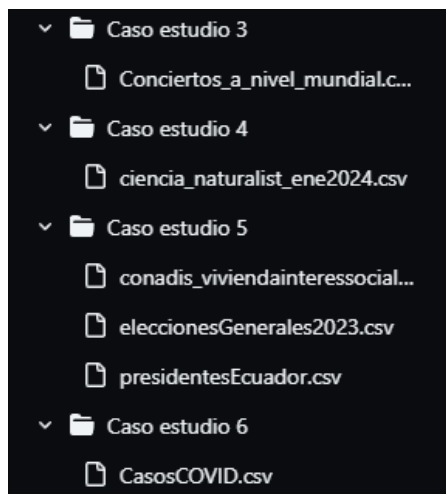
```

Exportación de los datos limpios a formato .csv

```
registro.to_csv('ciencia_naturalist_ene2024.csv')
```

ciencia_naturalist_ene2024 04/03/2024 05:14 p. m. Archivo de

5. Cuando las fuentes de datos sean completamente limpias, se procede a importar a las carpetas de cada caso de estudio.



Importación a la base de datos creada en Clever Cloud utilizando WorkBench

a) Se pretende generar la conexión entre la base de datos creada en Clever Cloud y los archivos limpios mediante el uso de la librería sqlalchemy

```

#Importación de Librerías
import pandas as pd
from sqlalchemy import create_engine

```

b) Para elaborar la conexión con la base de datos se genera el siguiente script:

```

#Importación de Librerías
import pandas as pd
from sqlalchemy import create_engine

# Cargar los archivos .CSV mediante la librería pandas
aves = pd.read_csv("conciertos.csv")
sonidos = pd.read_csv("numero_de_vociferos.csv")

# Generar la conexión con la base de datos
engine = create_engine("mysql+mysqlconnector://usuario:password@localhost:3306/sonidos")

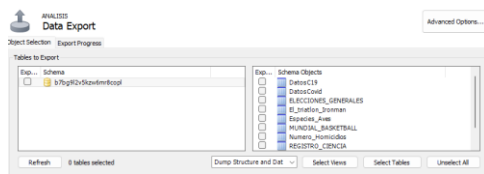
# Insertar datos
def insertar_datos():
    # Si existe se utiliza para crear la tabla, en caso de existir, se elimina y
    # se crea una nueva tabla con los mismos nombres y esquema por el valor replace
    aves.to_sql('conciertos', con=engine, if_exists='replace', index=False)
    sonidos.to_sql('numero_vociferos', con=engine, if_exists='replace', index=False)
    # El parámetro 'con' se utiliza para especificar la conexión a la base de datos

    print("Datos insertados correctamente.")

# Verificar la conexión
try:
    insertar_datos()
except Exception as e:
    print(f"Error al insertar los datos: {e}")

```

c) Una vez los datos se hayan cargado de forma correcta, se presentarán en las respectivas tablas



Resultados obtenidos:

Los resultados de la importación y limpieza por la herramienta de Jupyter Notebook mediante la librería Pandas se procede a

Caso 1: Redes Sociales

Fuente:

<https://www.kaggle.com/datasets/ramjasmaurya/top-1000-social-media-channels>

- Se importan las librerías necesarias para trabajar con el set de datos

```

Click here to ask Blackbox to help you code faster
import pandas as pd      La importación "pandas" n
import numpy as np       No se ha podido resolver l
from librerias.ConexionBDD import ConexionMySQL
import sqlite3
import os
✓ 10.9s

```

- Se carga el DataFrame con pandas para su tratamiento

```

Modificar los archivos
Caso 1
Click here to ask Blackbox to help you code faster
# Cargar los datos del archivo para el caso 1
Datos = pd.read_csv(ruta_casos1+"social media influencers - Tiktok sep 2022.csv", delimiter=";", if
Datos.isnull().sum()
[3]
...
Tiktoker name    0
Tiktok name      1
Subscribers      0
Views avg.       0
Likes avg.       0
Comments avg.    0
Shares avg.      0
dtype: int64

```

- Se limpian los datos remplazando valores nulos

```

Click here to ask Blackbox to help you code faster
# Rellenar valores nulos
Nombres = Datos["Tiktok name"]
Datos_no_nulos = Datos["Tiktok name"].dropna().to_list()
for clave, valor in Nombres.items():
    if pd.isna(valor):
        indice = np.random.randint(0, len(Datos_no_nulos))
        Datos.at[clave, "Tiktok name"] = Datos_no_nulos[indice]
Datos.isnull().sum()
[3]
...
Tiktoker name    0
Tiktok name      0
Subscribers      0
Views avg.       0
Likes avg.       0
Comments avg.    0
Shares avg.      0
dtype: int64

```

- Se extrae el DataFrame limpio como archivo CSV y también se lo lleva a una base de datos SQLite

```

Click here to ask Blackbox to help you code faster
# Guardar la información en un archivo CSV
Datos.to_csv(ruta_guardado_casos1+"Set_Datos.csv", index=False, encoding="utf-8")

Click here to ask Blackbox to help you code faster
# Guardar la información a una base de datos SQLite
conn = sqlite3.connect(ruta_guardado_casos1+"Social_Media_Tiktok.db")
cursor = conn.cursor()
Datos.to_sql(name="smi_tiktok", con=conn, if_exists="replace")
conn.commit()
conn.close()

```

Caso 2: Deportes en el mundo

Fuente:

https://es.wikipedia.org/wiki/Anexo:Clubes_de_fútbol_campeones_del_mundo

- En este grupo de datos, se conoce sobre todos los campeones del mundial de clubes, los resultados, los años que fueron celebrados y las sedes.

```
[42]: campeones = pd.read_csv("CampeonesClubes.csv")
```

```
[43]: campeones
```

	Año	Campeón	Subcampeón	Resultado	Sede
0	2023	Manchester City	Fluminense	4-0	Arabia Saudí
1	2022	Real Madrid	Al-Hilal	5-3	Marruecos
2	2021	Chelsea	Palmeiras	2-1	Emiratos Árabes
3	2020	Bayern	Tigres	1-0	Catar
4	2019	Liverpool	Flamengo	1-0	Catar
...
60	1964	Inter	Independiente	0-1 / 2-0 / 1-0	Avellaneda / Milán / Madrid
61	1963	Santos	Milán	2-4 / 4-2 / 1-0	Milán / Río de Janeiro
62	1962	Santos	Benfica	3-2 / 5-2	Río de Janeiro / Lisboa
63	1961	Peñarol	Benfica	0-1 / 5-0 / 2-1	Lisboa / Montevideo
64	1960	Real Madrid	Peñarol	0-0 / 5-1	Montevideo / Madrid

65 rows x 5 columns

- Se procede a la verificación de la existencia de valores nulos

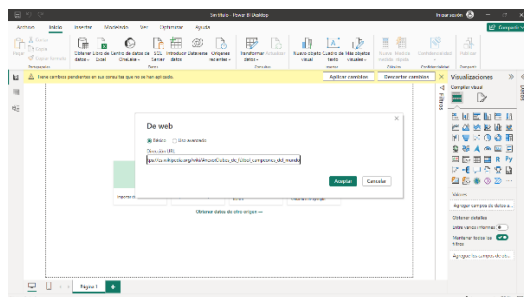
```
[44]: campeones.isnull().sum()
```

```
[44]: Año 0
Campeón 0
Subcampeón 0
Resultado 0
Sede 0
dtype: int64
```

- Se importan los datos limpios a .csv

```
campeones.to_csv('campeonesClubes.csv')
```

- Se extrae los datos



- En Power BI Desktop, se puede realizar limpieza dentro del data set

- Al finalizar la limpieza, se puede realizar consultas con la herramienta de Power BI Desktop

Header	Temporada	Campeón	Resultado	Subcampeón	Sede Final
Copa Intercontinental	1960	Real Madrid C. F.	0-0, 5-1	C. A. Peñarol	Estadio Centenario, Montevideo / Estadio Santiago Bernabéu
Copa Intercontinental	1961	C. A. Peñarol	0-1, 5-0, 2-1	S. L. Benfica	Estadio da Luz, Lisboa / Estadio Centenario, Montevideo / Est
Copa Intercontinental	1962	Santos F. C.	3-2, 5-2	S. L. Benfica	Estadio Maracanã, Río de Janeiro / Estadio da Luz, Lisboa
Copa Intercontinental	1963	Santos F. C.	2-4, 4-2, 1-0	A. C. Milan	Stadio San Siro, Milán / Estadio Maracanã, Río de Janeiro / Est
Copa Intercontinental	1964	F. C. Internazionale	0-1, 3-0, 1-0	C. A. Independiente	Estadio Libertadores de América, Avellaneda / Stadio San Siro
Copa Intercontinental	1965	F. C. Internazionale	3-0, 0-0	C. A. Independiente	Stadio San Siro, Milán / Estadio Libertadores de América, Avell
Copa Intercontinental	1966	C. A. Peñarol	2-0, 2-0	Real Madrid C. F.	Estadio Centenario, Montevideo / Estadio Santiago Bernabéu
Copa Intercontinental	1967	Racing Club	0-1, 2-1, 1-0	Celtic F. C.	Hampden Park, Glasgow / Juan Domingo Perón, Avellaneda /
Copa Intercontinental	1968	C. Estudiantes de La Plata	1-0, 1-1	Manchester United F. C.	Estadio Alberto J. Armando, Buenos Aires / Old Trafford, Man
Copa Intercontinental	1969	A. C. Milan	3-0, 1-2	C. Estudiantes de La Plata	Stadio San Siro, Milán / Estadio Alberto J. Armando, Buenos A
Copa Intercontinental	1970	Feyenoord Rotterdam	2-2, 1-0	C. Estudiantes de La Plata	Estadio Alberto J. Armando, Buenos Aires / De Kuip, Rotterdam
Copa Intercontinental	1971	C. Nacional de F.	1-1, 2-1	Panathinaikos A. O.	Estadio Karaiskaki, El Pireo / Estadio Centenario, Montevideo
Copa Intercontinental	1972	A. F. C. Ajax	1-1, 3-0	C. A. Independiente	Estadio Libertadores de América, Avellaneda / Estadio Olímpic
Copa Intercontinental	1973	C. A. Independiente	1-0	Juventus F. C.	Estadio Olímpico, Roma
Copa Intercontinental	1974	C. Atlético de Madrid(2)	0-1, 2-0	C. A. Independiente	Estadio Libertadores de América, Avellaneda / Estadio Vicente
Copa Intercontinental	1975	No disputada	No disputada	No disputada	
Copa Intercontinental	1976	F. C. Bayern Múnich	2-0, 0-0	Cruzeiro E. C.	Olympiastadion, Múnich / Estádio Mineirão, Belo Horizonte
Copa Intercontinental	1977	C. A. Boca Juniors	2-2, 3-0	Borussia Mönchengladbach	Estadio Alberto J. Armando, Buenos Aires / Wildparkstadion, F
Copa Intercontinental	1978	No disputada	No disputada	No disputada	
Copa Intercontinental	1979	C. Olimpia	1-0, 2-1	Malmö F.F.	Malmö Stadion, Malmö / Estadio Defensores del Chaco, Asun
Copa Intercontinental	1980	C. Nacional de F.	1-0	Nottingham Forest F. C.	Estadio Olímpico de Tokio, Tokio
Copa Intercontinental	1981	C. R. Flamengo	3-0	Liverpool F. C.	Estadio Olímpico de Tokio, Tokio
Copa Intercontinental	1982	C. A. Peñarol	2-0	Aston Villa F. C.	Estadio Olímpico de Tokio, Tokio
Copa Intercontinental	1983	Gilense F. B. R. A.	2-1 (pñ)	Hamburg S.V.	Estadio Olímpico de Tokio, Tokio

Caso 2: Deportes en el mundo

Fuente:

<https://www.kaggle.com/datasets/emmanueluelleai/iron-man-world-championship-allyears>

Sinopsis:

El triatlón Ironman es una serie de carreras organizadas por World Triathlon Corporation. Los participantes tienen que cubrir 3 distancias: 3,86 km de natación, 180 km de ciclismo y 42,2 km de carrera a pie. La carrera tiene un tiempo límite de 17 horas.

- Se procede a la importación de las librerías y el archivo .csv

```

s = pd.read_csv("C:\Users\USER\Documents\GitHub\Proyecto-final-AD\Archivos\Conciertos_a_nivel_mundial.csv")

```

Year	Place	Athlete	Country	Time	Gender	
0	2019	1	Jan Frodeno	GER	7:51:13	Male
1	2018	1	Patrick Lange	GER	7:52:39	Male
2	2017	1	Patrick Lange	GER	8:01:40	Male
3	2016	1	Jan Frodeno	GER	8:06:30	Male
4	2015	1	Jan Frodeno	GER	8:14:40	Male
...
247	1984	3	Julie Olson	USA	10:38:10	Female
248	1983	3	Eva Uelzen	USA	11:01:49	Female
249	1982	3	Sally Edwards	USA	11:03:00	Female
250	1982	3	Lyn Brooks	USA	11:51:00	Female
251	1981	3	Lyn Brooks	USA	12:42:15	Female

- b) Una vez el archivo este importado, se realiza la limpieza y verificación de valores nulos

```

s.isnull().sum()

```

```

Year      0
place     0
Athlete   0
Country   0
Time      0
Gender    0
dtype: int64

```

```

print("Archivo generado con éxito.")

```

Caso 3: Eventos en el mundo

Conciertos en el mundo

Fuentes:

https://es.wikipedia.org/wiki/Anexo:Giras_musicales_m%C3%A1s_recaudadoras

- a) Se procede a la importación de las librerías

```

import pandas as pd
import numpy as np
from librerias.ConexionBDD import ConexionMySQL
import sqlite3
import os
from sqlalchemy import create_engine

```

- b) Exportar archivo .CSV y limpiar

```

url2 = "https://es.wikipedia.org/wiki/Anexo:Giras_musicales_m%C3%A1s_recaudadoras"
concierto = pd.read_html(url2)
df_concierto = concierto[0]
df_concierto.isnull().sum()

```

- c) Se verifica si existen valores nulos

```

Puesto      0
Año(s)      0
Recaudación (en USD)  0
Inflación (para 2024)  0
Artista     0
Gira        0
Número de conciertos  0
Asistencia  0
Recaudación promedio  0
Ref.        0
dtype: int64
df_concierto.to_csv('Conciertos_a_nivel_mundial.csv')
print("Archivo generado con éxito.")

```

- d) Una vez, los datos estén limpios, se procede a la exportación base de datos


```

# Códig here to add libraries to help you code better
# Importación de librerías
import pandas as pd
from sqlalchemy import create_engine

# Cargar los archivos .csv mediante la librería pandas
Concierto = pd.read_csv("Conciertos_a_nivel_mundial.csv", encoding="latin1")

# Corregir los nombres de las columnas
Concierto.columns = Concierto.columns.str.strip()

# Crear una nueva tabla con los mismos nombres y reemplazar por el valor nulo
Concierto.to_sql("ELECTIONS_GENERALES", con=engine, if_exists="replace", index=False)

print("Datos insertados correctamente.")

except Exception as e:
    print("Error al insertar los datos:", e)

```

Caso 4: Ciencia en el Ecuador

Fuente: <https://datosabiertos.gob.ec/dataset>

/articulos-cientificos-en-los-ambitos-geologicos-y-energeticos/resource/9bf2bb4d-c07b-4d73-8690-e039e447b3bd

- Después se realiza la limpieza de los datos para su previa exportación a la herramienta Power BI Desktop, para realizar las respectivas consultas
- Primero, se debe transformar los datos, eliminar filas innecesarias, nulos datos vacíos.

ciencia_naturalist_ene2024.csv

Origen de archivo

65001 Unificado (UTF-8)

Delimitador

Coma

Detección del tipo de datos

Basado en las primeras 200 filas

Caso 4: Ciencia en el Ecuador

Especies de Aves

Fuente:

<https://www.ecoregistros.org/site/pais.php?id=18&idgrupoclase=1&page=1>

Mediante la importación de la librería Pandas se puede revisar el estado de los datos, y se visualiza la existencia de datos vacíos y nulos.

- Se utiliza la herramienta Jupyter Notebook

```

[28]: import pandas as pd
[29]: aves = pd.read_csv("especiesAvesEcuador.csv", encoding="latin1")
[30]: print(aves)

```

Column	Nombre Común	Nombre Inglés	Nombre Científico	Registros
0	1	Tinamú Tao	Grey Tinamou	1
1	2	Tinamú Oliváceo	Great Tinamou	1
2	3	Tinamú Sombrio	Cinereous Tinamou	1
3	4	Tinamú Chico	Little Tinamou	1
4	5	Tataupá Listado	Undulated Tinamou	1
...
943	944	Carpintero Cara Negra	Black-cheeked Woodpecker	1
944	945	Carpintero Oliva Oscuro	Smoky-brown Woodpecker	1
945	946	Carpintero Culirrujo	Red-rumped Woodpecker	1
946	947	Carpintero Dorisacarlata	Scarlet-backed Woodpecker	1
947	948	Carpintero Ventriamarillo	Yellow-vented Woodpecker	1
...
943	No Existe Registro	Melanerpes pucherani	Melanerpes pucherani	8
944	No Existe Registro	Leuconotopus fumigatus	Leuconotopus fumigatus	3
945	Pica-pau-de-sobre-vermelho	Veniliornis kirkii	Veniliornis kirkii	1
946	No Existe Registro	Veniliornis callonotus	Veniliornis callonotus	10
...

```
aves.isnull().sum()

Column1      0
Nombre Común  0
Nombre Inglés 0
Nombre Portugués 0
Nombre Científico 0
Registros    0
dtype: int64
```

- Luego, se exporta los datos limpios a un formato .CSV

```
aves.to_csv('EspeciesAves.csv')
```

- Una vez se tenga los datos limpios en los data sets, se procede a la importación a Power BI para realizar consultas de interés.

Column1	Nombre Común	Nombre Inglés	Nombre Portugués	Nombre Científico	Registros
0	1 Tinamú Tío	Grey Tinamou	Azulena	Tinamus tao	4
1	2 Tinamú Oliváceo	Great Tinamou	Inhambu-de-cabeça-vermelha	Tinamus major	2
2	3 Tinamú Sombrio	Cinereous Tinamou	Inhambu-grinto	Crypturellus cinereus	2
3	4 Tinamú Chico	Little Tinamou	Turum	Crypturellus soul	6
4	5 Tataupá Listado	Undulatus Tinamou	Jabá	Crypturellus undulatus	3
5	6 Tinamú Capulí	Pale-browed Tinamou	No Existe Registro	Crypturellus transfasciatus	3
6	7 Pava Fencalar	Sixte-winged Guan	No Existe Registro	Chamaepetes goudotti	3
7	8 Pava Andina	Andean Guan	No Existe Registro	Penelope montagni	4
8	9 Pava Amadónica	Splix's Guan	Jacar-de-igile	Penelope jacquacu	4
9	10 Pava Crestada	Crested Guan	No Existe Registro	Penelope purpurascens	2
10	11 Cuyave	Blue-throated Piping guan	No Existe Registro	Pipile cumanensis	2
11	12 Pava Aborita	Wattled Guan	No Existe Registro	Aburria aburri	2
12	13 Chachalaca Cabecinegra	Rufous-headed Chachalaca	No Existe Registro	Orealis erythrogaster	6
13	14 Chachalaca Manchada	Speckled Chachalaca	Aracali-pintado	Orealis guttata	8
14	15 Pajaro Nocturno	Nocturnal Curassow	Ururumut	Nothocrax urumutum	2
15	16 Pavón Curunculado	Wattled Curassow	Mutum-de-fava	Crao globulosa	2
16	17 Corcovado Dorsooscuro	Dark-backed Wood-quail	No Existe Registro	Odonophanes melanotos	2
17	18 Arca	Horned Soreanser	Arhuana	Ardeioa cornuta	2
18	19 Sini Vientre Negro	Black-bellied Whistling-duck	Marruca-cabolla	Dendrocygna autumnalis	22
19	20 Sini Colorado	Fulvous Whistling-duck	Marruca-caneleira	Dendrocygna bicolor	7

Caso 5: Política en Ecuador

Fuente:

<https://www.datosabiertos.gob.ec/dataset/vivienda-de-interes-social>

Fuente:

https://es.wikipedia.org/wiki/Elecciones_en_Ecuador

- a) El procedimiento sigue siendo el de los anteriores casos, considerando que los archivos sean de tipo .csv.

	codigo_iccs	subtipo_objeto_robado	nombre_objeto	cantidad	calibre	clase_armas	calidad_de_fabricacion	fecha_evento	hora_evento	nombre_lugar
0	05.02.013	ARMA DE FUEGO	PICULAS	1	9 mm	AUTOMÁTICO	APREHENDIDO	ARTESANAL	1/1/2024	4:30 -- 5 DE OCTUB
1	05.02.013	ARMA DE FUEGO	PICULAS	1	CERO	SEMIAUTOMÁTICO	APREHENDIDO	IMPORTADA	1/1/2024	2:50 -- FLORE
2	05.02.013	ARMA DE FUEGO	PICULAS	1	38	AUTOMÁTICO	DECOMISO	IMPORTADA	1/1/2024	2:30 -- PAS
3	05.02.013	ARMA DE FUEGO	PICULAS	1	9 mm	SEMIAUTOMÁTICO	APREHENDIDO	IMPORTADA	1/1/2024	3:30 -- PIR
4	05.02.013	ARMA DE FUEGO	PICULAS	1	9 mm	AUTOMÁTICO	APREHENDIDO	IMPORTADA	1/1/2024	8:00 -- LIBRE

- b) Se verifica si los datos dentro del data set poseen datos nulos

```
script3.Silvia.ipynb U
script3.Silvia.ipynb > armasilicitas_pm_2024_enero = pd.read_csv('armasilicitas_pm_2024_enero.csv')
+ Código + Markdown | Ejecutar todo | Reiniciar | Borrar

Click here to ask Blackbox to help you code faster |
armasilicitas_pm_2024_enero.isnull().sum()

[34]
...
codigo_iccs      0
subtipo_objeto_robado  0
nombre_objeto    0
cantidad         0
calibre          0
clase_armas      0
calidad_de_fabricacion  0
fecha_evento     0
hora_evento      0
tipo_lugar       0
lugar            0
codigo_zona      0
codigo_distrito  0
codigo_circuito  0
codigo_subcircuito  0
nombre_zona      0
nombre_distrito  0
nombre_circuito  0
nombre_subcircuito  0
codigo_provincia  0
codigo_canton    0
codigo_parroquia  0
nombre_provincia  0
nombre_canton    0
nombre_parroquia  0
tipo_delito      0
dtype: int64
```

- c) Se procede a la importación de la data frame con datos limpios a un formato de archivo .csv

```

Click here to ask Blackbox to help you code faster
armasilicita pm_2024_enero.to_csv("armasilicita pm_2024_enero.csv")
print("Archivo generado con éxito.")

[36]
... Archivo generado con éxito.

```

d) Se procede a la exportación de los data sets a la base de datos mediante el siguiente script:

```

script3libia.py:36
armasilicita pm_2024_enero = pd.read_csv(C:\Users\USER\Documents\Github\Proyecto-final-AD\Archivos\Caso de estudio 5\armasilicita pm_2024_enero.csv)
# Guardar el DataFrame en una base de datos MySQL
conn = ConexionMySQL() # Instanciar la clase ConexionMySQL para obtener la conexión a BD
try:
    conectado = conn.conectar(
        host="b7hg912vskzawer8cpl-mysql.services.clever-cloud.com",
        user="ush2qzqwraktoed",
        password="4h0ic0f7f0iaad807L",
        db="b7hg912vskzawer8cpl"
    ) # Conectar a la base de datos
    if conectado:
        print("Conexión exitosa")
        # Crear la tabla
        creacion = conn.crear_tabla(dataframe=armasilicita pm_2024_enero, nombre_tabla="conadis_viviendainteresocial_2023")
        if creacion==0:
            print("Tabla creada en la base de datos MySQL")
            # Insertar los registros
            insercion = conn.insertar_desde_dataframe(nombre_tabla="conadis_viviendainteresocial_2023", dataframe=armasilicita pm_2024_enero)
            if insercion==0:
                print("Datos insertados en la tabla", "Filas afectadas:", insercion)
            else:
                print("No se insertaron los registros")
        else:
            print("No se creo la tabla")
    else:
        print("Error en la conexión")
except Exception as e:
    print("Error al insertar los registros:", e)
finally:
    conn.cerrar_conexion() # Cerrar la conexión a la base de datos

[37]
... Conexión exitosa
Tabla creada en la base de datos MySQL
Datos insertados en la tabla Filas afectadas: 1200

```

e) Se realiza consulta con ayuda de Power BI Desktop

Caso 6: Casos de Covid 19

Fuente:

<https://data.ct.gov/browse?tags=covid+epi+data>

- Una vez obtenido el Data Set, se procede a cargarlo con Pandas para su tratamiento

```

Click here to ask Blackbox to help you code faster
DatosC19_2=pd.read_csv(ruta_caso6+"COVID-19_County_Level_Data_-_Archive_20240301.csv", sep=";", dtype="int64")
DatosC19_2.isnull().sum()

report_date      0
county            0
county_population 348
cumulative_cases  0
cumulative_tests_reportable 0
cumulative_deaths 0
cases_7days      0
tests_reportable_7days 0
positive_naet_7days 0
test_naet_7days  0
naet_positivity_7days 153
cumulative_positive_naet 0
cumulative_tests_naet 0
positive_ag_7days 0
census_today     348
fullyvax_today   348
partialvax_today 348
nomvax_today     348
adddose_today    348
census_7days_ago 348
census_change    348
census_not_fully_vax 348
census_pct_not_fully_vax 539
case_rate_weekly 348
flips            348
data_updated     0
dtype: int64

```

- Se procede a limpiar el Data Set rellenando los valores nulos que tenga

```

Click here to ask Blackbox to help you code faster
Poblacion = DatosC19_2["county_population"]
Valores_no_nulos = Poblacion.dropna().to_list()
# Rellenar valores nulos
for clave, valor in Poblacion.items():
    if pd.isna(valor):
        indice = np.random.randint(0, len(Valores_no_nulos))
        DatosC19_2.at[clave, "county_population"] = Valores_no_nulos[indice]

# Verificar valores nulos
DatosC19_2.isnull().sum()

```

- Una vez limpiados los valores nulos se procede a subir el Data Set a MySQL y exportarlo como archivo CSV, para la exportacion a la base de datos se utilizo una clase creada en python que permitiera hacer eso

```

Guardar el DataFrame en una base de datos MySQL

Click here to ask Blackbox to help you code faster
# Guardar el DataFrame en una base de datos MySQL
conn = ConexionMySQL() # Instanciar la clase ConexionMySQL para obtener la conexión a BD
try:
    conectado = conn.conectar(
        host="b7hg912vskzawer8cpl-mysql.services.clever-cloud.com",
        user="ush2qzqwraktoed",
        password="4h0ic0f7f0iaad807L",
        db="b7hg912vskzawer8cpl"
    ) # Conectar a la base de datos
    if conectado:
        print("Conexión exitosa")
        # Crear la tabla
        creacion = conn.crear_tabla(dataframe=DatosC19_2, nombre_tabla="DatosC19") # Crear la tabla en la base de datos
        if creacion==0:
            print("Tabla creada en la base de datos MySQL")
            # Insertar los registros
            insercion = conn.insertar_desde_dataframe(nombre_tabla="DatosC19", dataframe=DatosC19_2) # Insertar los registros en la tabla
            if insercion==0:
                print("Datos insertados en la tabla", "Filas afectadas:", insercion)
            else:
                print("No se insertaron los registros")
        else:
            print("No se creo la tabla")
    else:
        print("Error en la conexión")
except Exception as e:
    print("Error al insertar los registros:", e)
finally:
    conn.cerrar_conexion() # Cerrar la conexión a la base de datos

Conexión exitosa
Tabla creada en la base de datos MySQL
Datos insertados en la tabla Filas afectadas: 3132

```

- A continuación , se exporta el archivo con formato CSV

```

Click here to ask Blackbox to help you code faster
# Guardar el DataFrame como archivo CSV
DatosC19_2.to_csv(ruta_guardado_caso6+"CasosCOVID.csv", index=False, encoding="utf-8")

```

Geolocalización del Ecuador utilizando latitudes y longitudes

Fuente:

https://www.geodatos.net/coordenadas/ecuador#google_vignette

- Se procede a la importación del archivo para su limpieza por pandas

```
[38]: coordenadas = pd.read_csv("CoordenadasEcuador.csv")
```

```
[39]: coordenadas
```

```
[39]:
```

		Ciudad	Latitud	Longitud
0		Guayaquil	-2.19616	-79.88621
1		Quito	-0.22985	-78.52495
2		Cuenca	-2.90055	-79.00453
3		Machala	-3.25861	-79.96053
4		Manta	-0.96212	-80.71271
5		Portoviejo	-1.05458	-80.45445
6		Esmeraldas	0.95920	-79.65397
7		Ambato	-1.24908	-78.61675
8		Milagro	-2.13404	-79.59415
9		Ibarra	0.35171	-78.12233
10		Tulcán	0.81187	-77.71727
11		Riobamba	-1.67098	-78.64712
12		Quevedo	-1.02863	-79.46352
13		Babahoyo	-1.80217	-79.53443
14		Santo Domingo de los Colorados	-0.25305	-79.17536

- Se verifica la existencia de valores nulos

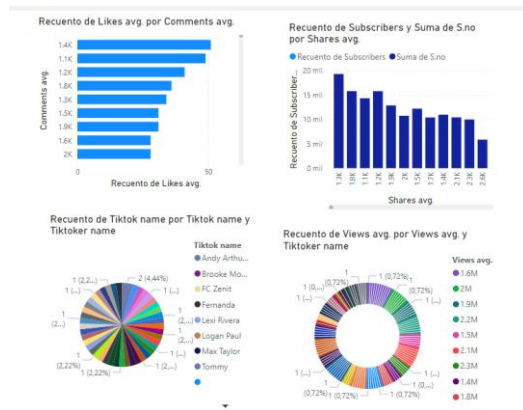
```
[40]: coordenadas.isnull().sum()
```

```
[40]: Ciudad      0
      Latitud    0
      Longitud   0
      dtype: int64
```

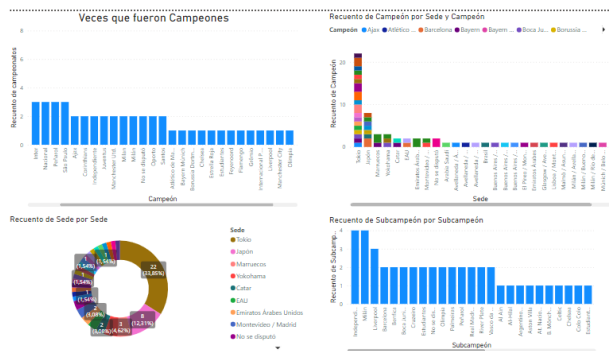
```
[41]: coordenadas.to_csv('coordenadasEcuador.csv')
```

Visualizaciones por cada caso de estudio

Caso 1: Redes Sociales



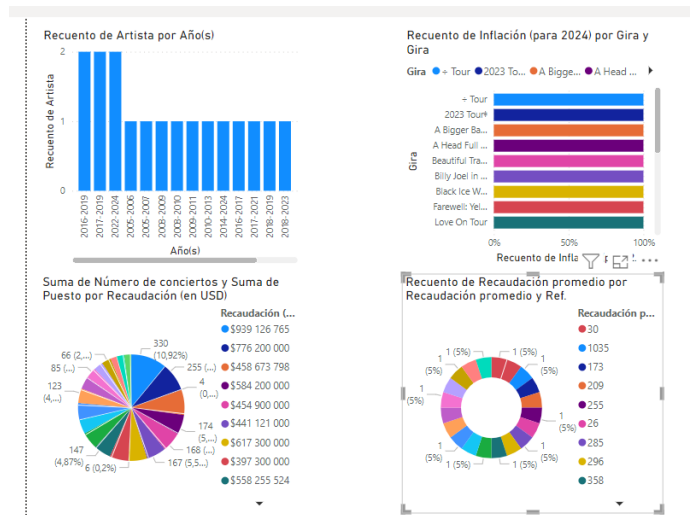
Caso 2: Deporte en el mundo



Geolocalización



Caso 3: Eventos en el mundo

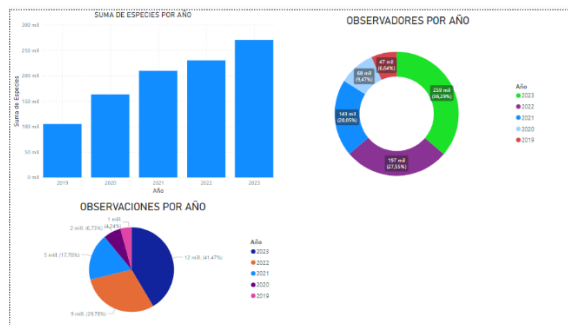


SUMA DE ESPERANZA POR AÑO

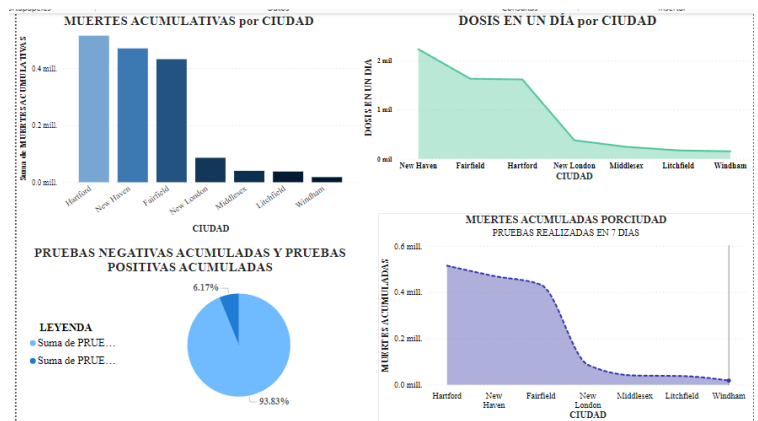
Año	Suma de Esperanza
2019	100
2020	160
2021	200
2022	210
2023	230

OBSERVADORES POR AÑO

Año	Porcentaje
2023	33.33%
2022	27.27%
2021	17.17%
2020	13.64%
2019	8.18%



A map of Ecuador with latitude and longitude coordinates marked for several cities. The map shows the country's borders with Peru to the east and Colombia to the north and east. The Pacific Ocean is to the west. Major cities like QUITO, Guayaquil, and Cuenca are labeled. The map also shows the Amazon region to the east and the Andes mountains. The coordinates are marked with blue dots and lines, indicating the location of each city.



Análisis de la información

Resumen de suma de suscriptores por comparticiones

- La suma de suscriptores por comparticiones, la mayor suma es de 20 mil y 15 mil que

corresponden a 3.2k de participaciones.

Caso 2:

Resumen de Campeones del Mundial de Clubes

- El equipo español Real Madrid es el máximo ganador de esta competencia, debido a que tiene muchas participaciones por ser también el máximo ganador de la liga de campeones.
- Tokio es la sede que más participaciones ha albergado, por lo que también es la sede donde ha visto más equipos coronarse campeones.
- El equipo Argentino Independiente, es el equipo que más veces ha sido vicecampeón del certamen, debido al poderío futbolístico que está demostrando el viejo continente.

Caso 3:

Resumen se suma de número de conciertos y suma de puesto por recaudación

- El Mayor número de recaudación es de 939 126 75

Resumen de artista por año

- El año en el que se denota un mayor recuento de numero de artistas es 2018-203

Resumen de recaudación promedio.

- El mayor número de recaudación promedio es de 1035

Caso 4:

Resumen de Observaciones de especies:

- Desde que existe el registro de Observaciones a especies, desde 2019, cada año aumenta su número, porque hay mucho más presupuesto y más ayudantes.
- Al igual que las observaciones han aumentado, también aumento el personal, es decir los observadores.

Resumen De Registro de Aves en el Ecuador:

- Aunque exista una gran variedad de especies de aves, no existen muchos registros de ellos.
- En el ecuador, la mayoría de las especies de aves son endémicas del país, y pocas especies no catalogan o son exóticas.

Resumen de latitud y longitud de las provincias del Ecuador:

- El registro de las latitudes y longitudes de cada provincia del Ecuador es beneficioso para poder localizarnos en el mundo.

Caso 5:

Resumen de Presidentes del Ecuador:

- En la participación de expresidente Rafael Correa se visualiza una

mayor participación (30%) por parte de la población ante las urnas.

- Guayaquil es la ciudad con mayor número de presidentes electos que tengan su lugar de procedencia.
- En conclusión, Guayaquil posee el mayor número de presidentes que estuvieron en el mando entre 2000-2024, teniendo en cuenta, que el expresidente Rafael Correa tiene como lugar de procedencia a esta ciudad.

Resumen de CONADIS-VIVIENDA SOCIAL:

- Pichincha es la provincia de la región Interandina con mayores personas que posee una capacidad especial
- Manabí es la provincia de la región Costa que tiene una mayor población de personas con capacidades especiales
- En una tabulación de datos, se evidenció que el género masculino tiene mayor población en personas con capacidades especiales

Caso 6:

Resumen del COVID-19 en el 2021:

- La ciudad con mayores casos de mortalidad en 2021 por COVID-19 es Hartford
- Durante un estudio, se evidenció

que las pruebas negativas poseen un mayor porcentaje ante las pruebas positivas, de esta manera, se comprueba que se redujo el contagio de COVID-19

- En conclusión, en 2021 se generó una reducción de muertes por coronavirus, comparación con años anteriores.

Recomendaciones

1. Se recomienda el uso de la página Clever Cloud que permite la creación de bases de datos en la nube, de esta manera, se puede trabajar en conjunto.
2. Se recomienda buscar datos desde fuentes confiables, para no caer en falsa información y perjudicar al informe.
3. Se recomienda la utilización de la librería Pandas para la limpieza de datos, ya que resulta ser muy fácil y sencilla de usar.

Desafíos y problemas encontrados.

- I. La conexión a la base de datos MySQL WorkBrench resultó ser un desafío para los integrantes del equipo, pero se logró resolver de forma eficaz con autoaprendizaje.

- II. Un desafío encontrado para los integrantes fue la obtención de fuentes confiables y verídicas para la limpieza y análisis de los datos.
- III. Un desafío como equipo fue la exportación de los data sets a la base de datos, existieron complicaciones, que eficazmente se supo resolver con autoaprendizaje y revisión de temas impartidos en la clase.

Link de GitHub del proyecto

<https://github.com/Alejo-P/Proyecto-final-AD>

Enlace a Youtube:

<https://youtu.be/wvHVd7ThEOc>