



**INSTITUTO POLITÉCNICO NACIONAL**



**CENTRO DE INVESTIGACIÓN EN COMPUTACIÓN**

**INTRODUCCIÓN A CHATGPT CON PHYTON**

**PROFESOR: ALAN BADILLO SALAS**

**PRACTICA 102**

**ALUMNA: CARMEN M. REZA SAN GERMÁN**

**[creza@ipn.mx](mailto:creza@ipn.mx)**

**11 AGOSTO 2023**

## P102.1 - Reporte estadístico

.1 Sigue a CHATGPT que genere el código para abrir un archivo weight-height.csv desde la URL <https://raw.githubusercontent.com/dragonnomada/chat-gpt-2023/main/datasets/weight-height.csv> usando Pandas

CR

Podrías apoyarme con el código para abrir un archivo weight-height.csv desde la URL <https://raw.githubusercontent.com/dragonnomada/chat-gpt-2023/main/datasets/weight-height.csv> usando Pandas

Claro, para abrir el archivo "weight-height.csv" desde la URL proporcionada utilizando Pandas, puedes seguir estos pasos:

1. Instala Pandas si aún no lo tienes instalado. Puedes hacerlo utilizando el siguiente comando:

```
bash
Copy code
pip install pandas
```

1. Luego, puedes usar el siguiente código para abrir el archivo desde la URL:

```
python
Copy code
import pandas as pd

# URL del archivo CSV
url = "https://raw.githubusercontent.com/dragonnomada/chat-gpt-2023/main/datasets/weight-height.csv"

# Leer el archivo CSV desde la URL
df = pd.read_csv(url)

# Mostrar las primeras filas del DataFrame (opcional)
print(df.head())
```

El código utilizará la función `pd.read\_csv()` de Pandas para leer el archivo CSV desde la URL y almacenará los datos en un DataFrame llamado `df`. Luego, puedes manipular y analizar los datos utilizando las capacidades de Pandas para trabajar con DataFrames.

El método `head()` se utiliza aquí para mostrar las primeras filas del DataFrame (por defecto, muestra las primeras 5 filas). Esto es opcional y puedes omitirlo si no deseas visualizar los datos en la consola.

Con estos pasos, tendrás el archivo "weight-height.csv" cargado en un DataFrame de Pandas y podrás realizar cualquier análisis o manipulación de datos que necesites.

Esto lo que entrega Google Colab

+ Código + Texto

✓ RAM Disco

```
import pandas as pd

# URL del archivo CSV
url = "https://raw.githubusercontent.com/dragonnomada/chat-gpt-2023/main/datasets/weight-height.csv"

# Leer el archivo CSV desde la URL
df = pd.read_csv(url)

# Mostrar las primeras filas del DataFrame (opcional)
print(df.head())
```

	Gender	Height	Weight
0	Male	73.847017	241.893563
1	Male	68.781904	162.310473
2	Male	74.110105	212.740856
3	Male	71.730978	220.042470
4	Male	69.881796	206.349801

.2 Solicita a CHATGPT que genere el código que imprima 10 valores aleatorios del dataset anterior usando la función sample de Pandas

me proporcionas el código que imprima 10 valores aleatorios del dataset anterior usando la función sample de Pandas

CR Por supuesto, puedes utilizar la función `sample` de Pandas para obtener 10 valores aleatorios del dataset. Aquí tienes el código para hacerlo:

```
python

import pandas as pd

# URL del archivo CSV
url = "https://raw.githubusercontent.com/dragonnomada/chat-gpt-2023/main/datasets/weight-height.csv"

# Leer el archivo CSV desde la URL
df = pd.read_csv(url)

# Obtener 10 valores aleatorios del dataset
valores_aleatorios = df.sample(n=10)

# Imprimir los 10 valores aleatorios
print(valores_aleatorios)
```

La función `sample` de Pandas se utiliza para obtener una muestra aleatoria de filas del DataFrame. En este caso, estamos utilizando el argumento `n=10` para indicar que queremos obtener 10 filas aleatorias del DataFrame `df`. Los 10 valores aleatorios se almacenarán en el DataFrame `valores\_aleatorios`, y luego los imprimiremos en la consola.

## Esto lo que entrega Google Colab

```
+ Código + Texto
```

```
✓ Import pandas as pd
# URL del archivo CSV
url = "https://raw.githubusercontent.com/dragonnomada/chat-gpt-2023/main/datasets/weight-height.csv"
# Leer el archivo CSV desde la URL
df = pd.read_csv(url)
# Obtener 10 valores aleatorios del dataset
valores_aleatorios = df.sample(n=10)
# Imprimir los 10 valores aleatorios
print(valores_aleatorios)
```

	Gender	Height	Weight
907	Male	68.893418	172.654238
3061	Male	69.219568	218.314451
1373	Male	67.988585	191.860774
2215	Male	68.856797	172.444399
19	Male	67.236085	182.426648
251	Male	68.231132	199.827758
6546	Female	63.714930	127.876873
178	Male	64.888621	162.379541
4425	Male	73.846225	222.620664
5472	Female	68.552126	113.232336

.3 Solicita a CHATGPT que genere el código que describa la columna Gender que es categórica, pide que extraiga los valores únicos y sus conteos

CR como sería un código que describa la columna Gender que es categórica, sólo debes extraer los valores únicos y sus conteos

Para describir la columna "Gender" que es categórica y extraer los valores únicos junto con sus conteos, puedes utilizar la función `value_counts()` de Pandas. Aquí tienes el código para hacerlo:

```
python
```

```
import pandas as pd

# URL del archivo CSV
url = "https://raw.githubusercontent.com/dragonnomada/chat-gpt-2023/main/datasets/weight-height.csv"

# Leer el archivo CSV desde la URL
df = pd.read_csv(url)

# Describir la columna "Gender" y extraer valores únicos y sus conteos
descripcion_gender = df["Gender"].value_counts()

# Imprimir la descripción de la columna "Gender"
print(descripcion_gender)
```

La función `value_counts()` de Pandas se aplica directamente sobre la columna "Gender" del DataFrame `df`, lo que nos permite contar la frecuencia de cada valor único en esa columna. El resultado de `descripcion_gender` será una serie de Pandas que muestra los valores únicos de "Gender" junto con sus respectivos conteos.

Al imprimir `descripcion_gender`, obtendrás algo similar a esto:

```
yaml
```

```
Male      5000
Female    5000
Name: Gender, dtype: int64
```

Este resultado indica que hay 5000 filas con "Male" y 5000 filas con "Female" en la columna "Gender". Esto proporciona una descripción clara de la distribución de los valores en la columna categórica "Gender" del dataset.

## Esto lo que entrega Google Colab

```
+ Código + Texto  
✓  
import pandas as pd  
  
# URL del archivo CSV  
url = "https://raw.githubusercontent.com/dragonnomada/chat-gpt-2023/main/datasets/weight-height.csv"  
  
# Leer el archivo CSV desde la URL  
df = pd.read_csv(url)  
  
# Describir la columna "Gender" y extraer valores únicos y sus conteos  
descripcion_gender = df["Gender"].value_counts()  
  
# Imprimir la descripción de la columna "Gender"  
print(descripcion_gender)  
  
Male      5000  
Female    5000  
Name: Gender, dtype: int64
```

.4 Solicita a CHATGPT que genere el código que describa la columna Height que es numérica y obtenga los estadísticos: total, mínimo, máximo, promedio, desviación estándar, intervalo de confianza al 95%, y los cuartiles

 puedes generar el código que describa la columna Height que es numérica y obtenga los estadísticos: total, mínimo, máximo, promedio, desviación estándar, intervalo de confianza al 95%, y los cuartiles

 Claro, para describir la columna "Height" que es numérica y obtener sus estadísticos, puedes utilizar la función `describe()` de Pandas para obtener un resumen estadístico de la columna. También puedes calcular manualmente algunos estadísticos adicionales, como el promedio, desviación estándar, intervalo de confianza al 95% y los cuartiles. Aquí tienes el código para hacerlo:

```
python  
Copy code  
import pandas as pd  
import numpy as np  
from scipy import stats  
  
# URL del archivo CSV  
url = "https://raw.githubusercontent.com/dragonnomada/chat-gpt-2023/main/datasets/weight-height.csv"  
  
# Leer el archivo CSV desde la URL  
df = pd.read_csv(url)  
  
# Describir la columna "Height" usando la función describe()  
descripcion_height = df["Height"].describe()
```

```
# Calcular manualmente el promedio, desviación estándar y cuartiles  
promedio = df["Height"].mean()  
desviacion_estandar = df["Height"].std()  
cuartiles = df["Height"].quantile([0.25, 0.5, 0.75])  
  
# Calcular el intervalo de confianza al 95%  
confianza_95 = stats.t.interval(0.95, df["Height"].count() - 1, loc=df["Height"].mean(), scale=df["Height"].std())  
  
# Imprimir los resultados  
print("Descripción de la columna Height:")  
print(descripcion_height)  
  
print("\nPromedio:", promedio)  
print("Desviación estándar:", desviacion_estandar)  
print("Cuartiles:")  
print(cuartiles)  
print("Intervalo de confianza al 95%:", confianza_95)
```

La función `describe()` proporciona un resumen estadístico que incluye el conteo, promedio, desviación estándar, valor mínimo, cuartiles y valor máximo de la columna "Height". Luego, se calcula manualmente el promedio, desviación estándar y cuartiles utilizando las funciones de Pandas. También se calcula el intervalo de confianza al 95% utilizando la biblioteca `scipy`.

Con este código, obtendrás una descripción completa de la columna numérica "Height" y sus estadísticos correspondientes.

```

+ Código + Texto
✓ RAM Disco
+ URL del archivo CSV
url = "https://raw.githubusercontent.com/dragonnomada/chat-gpt-2023/main/datasets/weight-height.csv"

# Leer el archivo CSV desde la URL
df = pd.read_csv(url)

# Describir la columna "Height" usando la función describe()
descripcion_height = df["Height"].describe()

# Calcular manualmente el promedio, desviación estándar y cuartiles
promedio = df["Height"].mean()
desviacion_estandar = df["Height"].std()
cuartiles = df["Height"].quantile([0.25, 0.5, 0.75])

# Calcular el intervalo de confianza al 95%
confianza_95 = stats.t.interval(0.95, df["Height"].count() - 1, loc=df["Height"].mean(), scale=df["Height"].std())

# Imprimir los resultados
print("Descripción de la columna Height:")
print(descripcion_height)

print("\nPromedio:", promedio)
print("Desviación estandar:", desviacion_estandar)
print("Cuartiles:")
print(cuartiles)
print("Intervalo de confianza al 95%:", confianza_95)

```

D: Descripción de la columna Height:

```

count    10000.000000
mean     66.367560
std      3.847528
min     64.261113
25%     63.505620
50%     66.318870
75%     69.174262
max     78.998742
Name: Height, dtype: float64

Promedio: 66.3675975482124
Desviación estandar: 3.8475281287732293
Cuartiles:
0.25    63.505620
0.50    66.318870
0.75    69.174262
Name: Height, dtype: float64
Intervalo de confianza al 95%: (66.29214045997966, 66.44297904966282)

```

.5 Solicita a CHATGPT que genere el código que describa la columna Weight que es numérica y obtenga los estadísticos: total, mínimo, máximo, promedio, desviación estándar, intervalo de confianza al 95%, y los cuartiles

cómo será el código que describa la columna Weight que es numérica y obtenga los estadísticos: total, mínimo, máximo, promedio, desviación estándar, intervalo de confianza al 95%, y los cuartiles



Para describir la columna "Weight" que es numérica y obtener sus estadísticos, podemos seguir un proceso similar al código anterior. Utilizaremos la función `describe()` de Pandas para obtener un resumen estadístico de la columna y calcularemos manualmente algunos estadísticos adicionales, como el promedio, desviación estándar, intervalo de confianza al 95% y los cuartiles. Aquí tienes el código:

```

python
Copy code

import pandas as pd
import numpy as np
from scipy import stats

# URL del archivo CSV
url = "https://raw.githubusercontent.com/dragonnomada/chat-gpt-2023/main/datasets/weight-height.csv"

# Leer el archivo CSV desde la URL
df = pd.read_csv(url)

# Describir la columna "Weight" usando la función describe()
descripcion_weight = df["Weight"].describe()

```

```

# Calcular manualmente el promedio, desviación estándar y cuartiles
promedio_weight = df["Weight"].mean()
desviacion_estandar_weight = df["Weight"].std()
cuartiles_weight = df["Weight"].quantile([0.25, 0.5, 0.75])

# Calcular el intervalo de confianza al 95%
confianza_95_weight = stats.t.interval(0.95, df["Weight"].count() - 1, loc=df["Weight"].mean(), scale=df["Weight"].std())

# Imprimir los resultados
print("Descripción de la columna Weight:")
print(descripcion_weight)

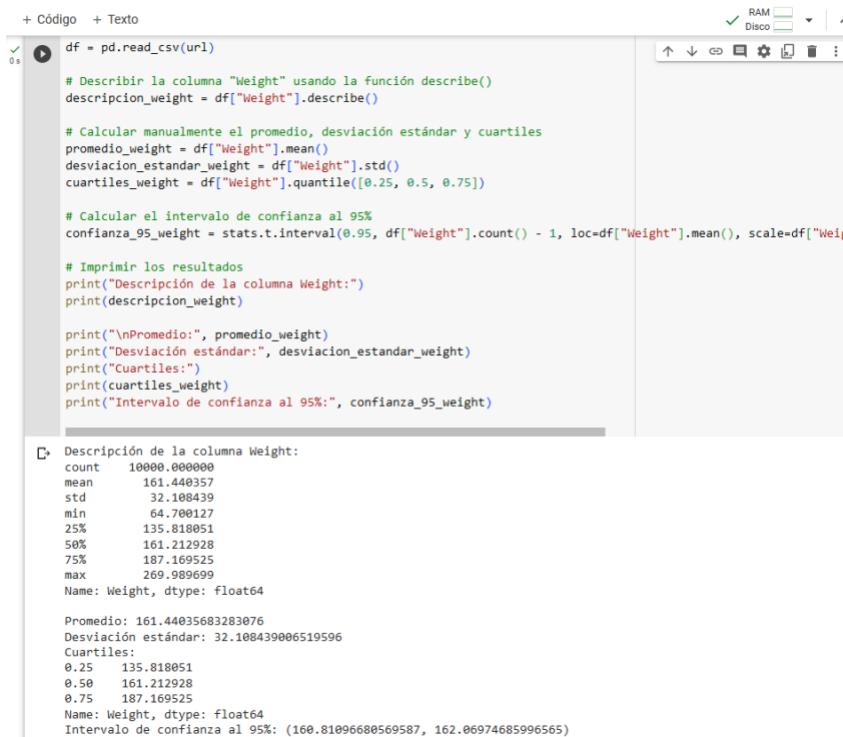
print("\nPromedio:", promedio_weight)
print("Desviación estándar:", desviacion_estandar_weight)
print("Cuartiles:")
print(cuartiles_weight)
print("Intervalo de confianza al 95%:", confianza_95_weight)

```

El código es similar al anterior, pero ahora se aplica a la columna "Weight".

Utilizamos la función `describe()` para obtener el resumen estadístico básico de la columna. Luego, calculamos manualmente el promedio, desviación estándar y cuartiles utilizando las funciones de Pandas. También calculamos el intervalo de confianza al 95% utilizando la biblioteca `scipy`.

Con este código, obtendrás una descripción completa de la columna numérica "Weight" y sus estadísticos correspondientes.



The screenshot shows a Jupyter Notebook cell with the following code:

```

+ Código + Texto
0s df = pd.read_csv(url)

# Describir la columna "Weight" usando la función describe()
descripcion_weight = df["Weight"].describe()

# Calcular manualmente el promedio, desviación estándar y cuartiles
promedio_weight = df["Weight"].mean()
desviacion_estandar_weight = df["Weight"].std()
cuartiles_weight = df["Weight"].quantile([0.25, 0.5, 0.75])

# Calcular el intervalo de confianza al 95%
confianza_95_weight = stats.t.interval(0.95, df["Weight"].count() - 1, loc=df["Weight"].mean(), scale=df["Weight"].std())

# Imprimir los resultados
print("Descripción de la columna Weight:")
print(descripcion_weight)

print("\nPromedio:", promedio_weight)
print("Desviación estándar:", desviacion_estandar_weight)
print("Cuartiles:")
print(cuartiles_weight)
print("Intervalo de confianza al 95%:", confianza_95_weight)

```

The output pane displays the results of the executed code:

```

Descripción de la columna Weight:
count    10000.000000
mean     161.440357
std      32.108439
min      64.700127
25%     135.818051
50%     161.212928
75%     187.169525
max      269.989699
Name: Weight, dtype: float64

Promedio: 161.44035683283076
Desviación estándar: 32.108439006519596
Cuartiles:
0.25    135.818051
0.50    161.212928
0.75    187.169525
Name: Weight, dtype: float64
Intervalo de confianza al 95%: (160.81096680569587, 162.06974685996565)

```

## .6 Sigue a CHATGPT que genere el código que grafique el histograma de la columna Height usando Seaborn

CR

cómo será el código que grafique el histograma de la columna Height usando Seaborn



Para graficar el histograma de la columna "Height" utilizando Seaborn, necesitamos seguir estos pasos:

1. Instalar Seaborn si aún no lo tienes instalado. Puedes hacerlo utilizando el siguiente comando:

bash

Copy code

```
pip install seaborn
```

1. Importar las bibliotecas necesarias y leer los datos desde la URL del archivo CSV.
2. Utilizar Seaborn para crear el histograma.

Aquí tienes el código para graficar el histograma de la columna "Height":

python

Copy code

```
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
```

```
# URL del archivo CSV
url = "https://raw.githubusercontent.com/dragonnomada/chat-gpt-2023/main/height_weight.csv"

# Leer el archivo CSV desde la URL
df = pd.read_csv(url)

# Configurar el estilo de Seaborn (opcional)
sns.set(style="whitegrid")

# Crear el histograma de la columna "Height"
plt.figure(figsize=(8, 6)) # Ajustar el tamaño de la figura
sns.histplot(df["Height"], bins=20, kde=True, color="skyblue")

# Configurar el título y etiquetas del gráfico
plt.title("Histograma de la columna Height")
plt.xlabel("Height")
plt.ylabel("Frecuencia")

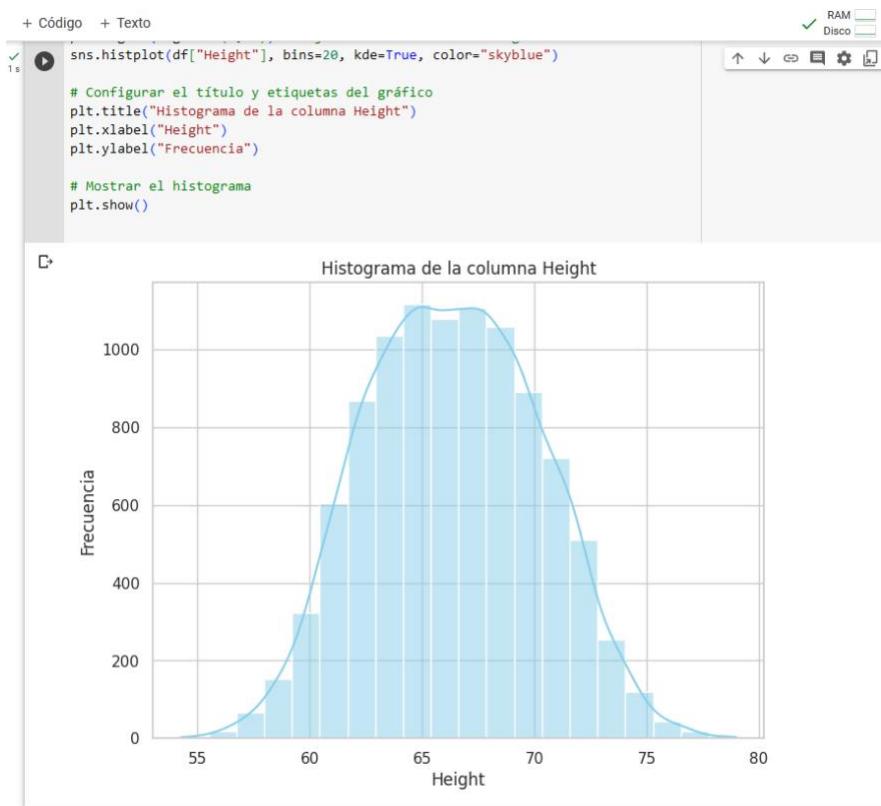
# Mostrar el histograma
plt.show()
```

En este código, utilizamos Seaborn para crear el histograma de la columna "Height" con `sns.histplot()`. Especificamos el número de "bins" (intervalos) para el histograma con el argumento `bins`, y activamos la representación de la función de densidad del kernel (KDE) con `kde=True`. También configuramos el color de las barras del histograma con `color="skyblue"`.

El código muestra el histograma en una figura de tamaño 8x6. También hemos agregado títulos y etiquetas al gráfico para una mejor comprensión.

Ejecutando este código, deberás obtener un histograma de la columna "Height" del dataset que muestra la distribución de alturas.

## En Google Colab



## .7 Sigue a CHATGPT que genere el código que grafique el histograma de la columna Weight usando Seaborn



genera el código que grafique el histograma de la columna Weight usando Seaborn



Por supuesto, aquí tienes el código para graficar el histograma de la columna "Weight" utilizando Seaborn:

```
python
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt

# URL del archivo CSV
url = "https://raw.githubusercontent.com/dragonnomada/chat-gpt-2023/"

# Leer el archivo CSV desde la URL
df = pd.read_csv(url)

# Configurar el estilo de Seaborn (opcional)
sns.set(style="whitegrid")

# Crear el histograma de la columna "Weight"
plt.figure(figsize=(8, 6)) # Ajustar el tamaño de la figura
sns.histplot(df["Weight"], bins=20, kde=True, color="salmon")

# Crear el histograma de la columna "Weight"
plt.figure(figsize=(8, 6)) # Ajustar el tamaño de la figura
sns.histplot(df["Weight"], bins=20, kde=True, color="salmon")

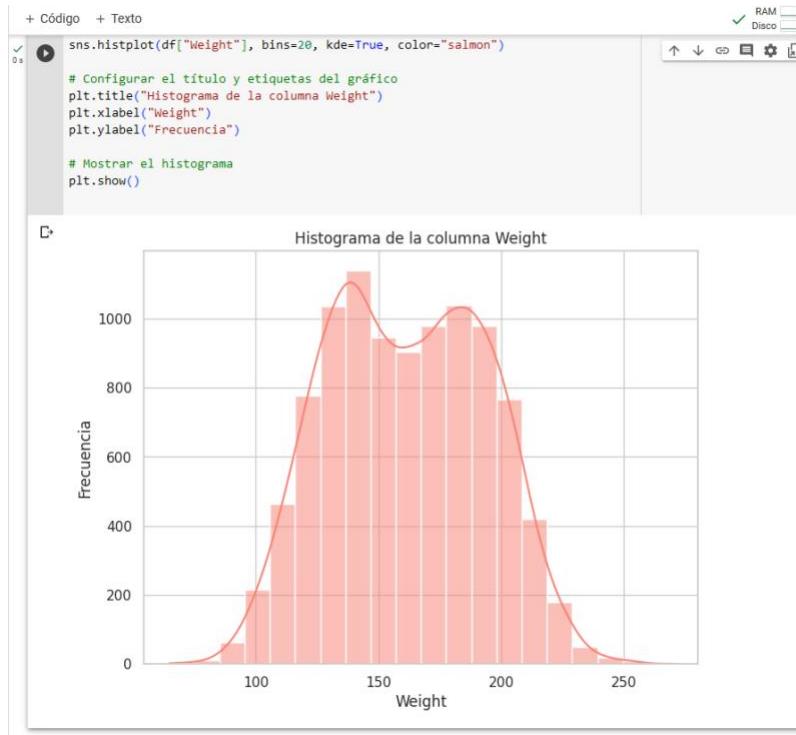
# Configurar el título y etiquetas del gráfico
plt.title("Histograma de la columna Weight")
plt.xlabel("Weight")
plt.ylabel("Frecuencia")

# Mostrar el histograma
plt.show()
```

En este código, utilizamos Seaborn para crear el histograma de la columna "Weight" con `sns.histplot()`. Especificamos el número de "bins" (intervalos) para el histograma con el argumento `bins`, y activamos la representación de la función de densidad del kernel (KDE) con `kde=True`. También configuramos el color de las barras del histograma con `color="salmon"`.

El código muestra el histograma en una figura de tamaño 8x6. También hemos agregado títulos y etiquetas al gráfico para una mejor comprensión.

Ejecutando este código, deberías obtener un histograma de la columna "Weight" del dataset que muestra la distribución de pesos.



.8 Solicita a CHATGPT que genere el código que grafique la correlación entre las columnas Height y Weight usando **Seaborn**

CR genera el código que grafique la correlación entre las columnas Height y Weight usando Seaborn



Claro, para graficar la correlación entre las columnas "Height" y "Weight" utilizando Seaborn, puedes seguir estos pasos:

1. Instalar Seaborn si aún no lo tienes instalado. Puedes hacerlo utilizando el siguiente comando:

bash

Copy code

```
pip install seaborn
```

1. Importar las bibliotecas necesarias y leer los datos desde la URL del archivo CSV.
2. Utilizar Seaborn para crear la gráfica de correlación.

Aquí tienes el código para graficar la correlación entre las columnas "Height" y "Weight":

```
python
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt

# URL del archivo CSV
url = "https://raw.githubusercontent.com/dragonnomada/chat-gpt-2023/i

# Leer el archivo CSV desde la URL
df = pd.read_csv(url)

# Configurar el estilo de Seaborn (opcional)
sns.set(style="whitegrid")

# Crear la gráfica de correlación entre "Height" y "Weight"
plt.figure(figsize=(8, 6)) # Ajustar el tamaño de la figura
sns.scatterplot(x="Height", y="Weight", data=df, color="purple")

# Configurar el título y etiquetas del gráfico
plt.title("Correlación entre Height y Weight")
plt.xlabel("Height")
plt.ylabel("Weight")
```

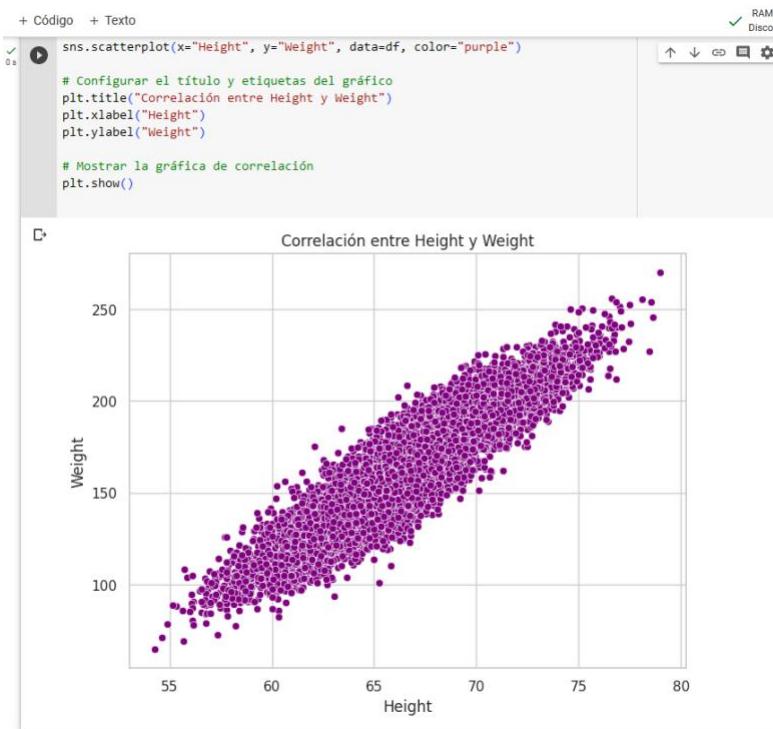
```
# Mostrar la gráfica de correlación
plt.show()
```

En este código, utilizamos Seaborn para crear la gráfica de correlación entre las columnas "Height" y "Weight" con `sns.scatterplot()`. Esta gráfica de dispersión muestra cómo están relacionadas ambas variables.

El código muestra la gráfica en una figura de tamaño 8x6. También hemos agregado títulos y etiquetas al gráfico para una mejor comprensión.

Ejecutando este código, deberías obtener una gráfica de dispersión que muestra la correlación entre las columnas "Height" y "Weight" del dataset. Si los puntos se encuentran más o menos alineados de manera ascendente, indicará una correlación positiva entre ambas variables.

## En Google Colab



.9 Solicita a CHATGPT que genere el código que obtenga la correlación entre las columnas Height y Weight usando corr de **Numpy**

genera el código que obtenga la correlación entre las columnas Height y Weight usando corr de Numpy

Para obtener la correlación entre las columnas "Height" y "Weight" utilizando la función `corr()` de Numpy, necesitamos seguir estos pasos:

1. Importar las bibliotecas necesarias y leer los datos desde la URL del archivo CSV.
2. Utilizar la función `corr()` de Numpy para calcular la correlación entre las columnas.

Aquí tienes el código para obtener la correlación entre las columnas "Height" y "Weight" usando Numpy:

```
python
import pandas as pd
import numpy as np

# URL del archivo CSV
url = "https://raw.githubusercontent.com/dragonnomada/chat-gpt-2023/r

# Leer el archivo CSV desde la URL
df = pd.read_csv(url)
```

```
# Obtener la correlación entre las columnas "Height" y "Weight" usando Numpy
correlacion_height_weight = df["Height"].corr(df["Weight"])

# Imprimir el resultado
print("Correlación entre Height y Weight:", correlacion_height_weight)
```

En este código, utilizamos la función `corr()` de Numpy para calcular la correlación entre las columnas "Height" y "Weight". La función `corr()` toma dos columnas como argumentos y devuelve el coeficiente de correlación entre ellas.

El resultado impreso será un valor que oscila entre -1 y 1. Un valor cercano a 1 indica una correlación positiva, mientras que un valor cercano a -1 indica una correlación negativa. Un valor cercano a 0 indica una correlación débil o inexistente entre las columnas.

Ejecutando este código, obtendrás la correlación entre las columnas "Height" y "Weight" del dataset.

## En Google Colab

The screenshot shows the Google Colab interface. At the top, there are tabs for '+ Código' and '+ Texto'. Below the tabs, a progress bar indicates the cell is at step [22]. The main area contains the following Python code:

```
import pandas as pd
import numpy as np

# URL del archivo CSV
url = "https://raw.githubusercontent.com/dragonnomada/chat-gpt-2023/main/datasets/weight-height.csv"

# Leer el archivo CSV desde la URL
df = pd.read_csv(url)

# Obtener la correlación entre las columnas "Height" y "Weight" usando corr() de Numpy
correlacion_height_weight = df["Height"].corr(df["Weight"])

# Imprimir el resultado
print("Correlación entre Height y Weight:", correlacion_height_weight)
```

The output cell shows the result of the print statement: `Correlación entre Height y Weight: 0.9247562987409146`.

.10 Sigue la instrucción de CHATGPT que genere el código que reporte el valor promedio de la columna Height donde Gender es Male y donde Gender es Female y repita lo mismo para la columna Weight

CR generame el código que reporte el valor promedio de la columna Height donde Gender es Male y donde Gender es Female y repita lo mismo para la columna Weight



Para reportar el valor promedio de las columnas "Height" y "Weight" separado por género ("Male" y "Female"), puedes utilizar la función `groupby()` de Pandas junto con la función `mean()`. Aquí tienes el código:

```
python
import pandas as pd

# URL del archivo CSV
url = "https://raw.githubusercontent.com/dragonnomada/chat-gpt-2023/main/height_weight.csv"

# Leer el archivo CSV desde la URL
df = pd.read_csv(url)

# Calcular el valor promedio de la columna Height y Weight para cada género
promedio_height_gender = df.groupby("Gender")["Height"].mean()
promedio_weight_gender = df.groupby("Gender")["Weight"].mean()

# Imprimir los resultados
print("Valor promedio de Height por género:")
print(promedio_height_gender)
```

```
print("\nValor promedio de Weight por género:")
print(promedio_weight_gender)
```

En este código, primero utilizamos la función `groupby()` para agrupar los datos por el valor de la columna "Gender". Luego, aplicamos la función `mean()` a las columnas "Height" y "Weight" para calcular el valor promedio separado por género.

El resultado impreso mostrará el valor promedio de "Height" y "Weight" para cada género ("Male" y "Female").

Ejecutando este código, obtendrás los valores promedio de "Height" y "Weight" para ambos géneros en el dataset.

## En Google Colab

```
+ Código + Texto RAM Disco ✓
```

```
import pandas as pd

# URL del archivo CSV
url = "https://raw.githubusercontent.com/dragonnomada/chat-gpt-2023/main/datasets/weight-height.csv"

# Leer el archivo CSV desde la URL
df = pd.read_csv(url)

# Calcular el valor promedio de la columna Height y Weight para cada género
promedio_height_gender = df.groupby("Gender")["Height"].mean()
promedio_weight_gender = df.groupby("Gender")["Weight"].mean()

# Imprimir los resultados
print("Valor promedio de Height por género:")
print(promedio_height_gender)

print("Valor promedio de Weight por género:")
print(promedio_weight_gender)
```

```
Valor promedio de Height por género:
Gender
Female    63.708774
Male      69.026346
Name: Height, dtype: float64

Valor promedio de Weight por género:
Gender
Female   135.860093
Male     187.028621
Name: Weight, dtype: float64
```

## P102.2 - Extracción de Información

.1 Sigue la guía que te proporciono en el siguiente link para extraer el texto de una imagen PNG usando OCR en Python

apoyame en la generación del código que extraiga el texto de una imagen PNG desde la URL [https://raw.githubusercontent.com/dragonnomada/chat-gpt-2023/main/datasets/UITD\\_PAPER.png](https://raw.githubusercontent.com/dragonnomada/chat-gpt-2023/main/datasets/UITD_PAPER.png) usando OCR en Python

Para extraer el texto de una imagen PNG desde una URL utilizando OCR (Reconocimiento Óptico de Caracteres) en Python, vamos a utilizar la biblioteca `pytesseract` junto con la biblioteca `Pillow` para procesar la imagen. Asegúrate de tener ambas bibliotecas instaladas previamente. Puedes hacerlo utilizando los siguientes comandos:

```
bash Copy code
```

```
pip install pytesseract
pip install Pillow
```

Una vez que tengas las bibliotecas instaladas, aquí tienes el código para extraer el texto de la imagen:

```
python Copy code
```

```
import requests
from PIL import Image
import pytesseract
```

```
# URL de la imagen PNG
url = "https://raw.githubusercontent.com/dragonnomada/chat-gpt-2023/"

# Obtener el contenido de la imagen desde la URL
response = requests.get(url)
image = Image.open(BytesIO(response.content))

# Extraer el texto de la imagen usando pytesseract
texto_extraido = pytesseract.image_to_string(image)

# Imprimir el texto extraido
print(texto_extraido)
```

En este código, utilizamos `requests` para obtener el contenido de la imagen desde la URL y `Pillow` para abrir y procesar la imagen. Luego, utilizamos `pytesseract.image\_to\_string()` para extraer el texto de la imagen.

Es importante mencionar que para utilizar `pytesseract`, debes tener instalado Tesseract OCR en tu sistema. Puedes descargar e instalar Tesseract desde su sitio web oficial: <https://github.com/tesseract-ocr/tesseract>

Recuerda que la calidad de los resultados de OCR puede variar dependiendo de la calidad de la imagen y la legibilidad del texto en la imagen. Asegúrate de tener una imagen clara y bien definida para obtener mejores resultados.



resuelve el error NameError: name 'BytesIO' is not defined



Mis disculpas por el error. El error `NameError: name 'BytesIO' is not defined` ocurre porque falta importar la clase `BytesIO` desde el módulo `io`. Para solucionarlo, agrega la importación necesaria. Aquí tienes el código corregido:

```
python
Copy code

import requests
from PIL import Image
from io import BytesIO
import pytesseract

# URL de la imagen PNG
url = "https://raw.githubusercontent.com/dragonnomada/chat-gpt-2023/"

# Obtener el contenido de la imagen desde la URL
response = requests.get(url)
image = Image.open(BytesIO(response.content))

# Extraer el texto de la imagen usando pytesseract
texto_extraido = pytesseract.image_to_string(image)
```

```
# Extraer el texto de la imagen usando pytesseract
texto_extraido = pytesseract.image_to_string(image)

# Imprimir el texto extraido
print(texto_extraido)
```

Con esta corrección, el código debería funcionar correctamente y podrás extraer el texto de la imagen PNG utilizando OCR con `pytesseract`. Recuerda tener instalado Tesseract OCR en tu sistema para que `pytesseract` funcione correctamente.



resuelve error TesseractNotFoundError: tesseract is not installed or it's not in your PATH. See README file for more information



El error `TesseractNotFoundError` indica que Tesseract OCR no está instalado o no está en tu PATH. Tesseract es una herramienta de OCR necesaria para que `pytesseract` funcione correctamente.

Para solucionar este problema, asegúrate de que Tesseract OCR esté instalado en tu sistema y sea accesible desde la línea de comandos.

Sigue estos pasos para instalar Tesseract OCR:

1. Para Windows:

a. Descarga el instalador para Windows desde el sitio oficial de Tesseract:

<https://github.com/UB-Mannheim/tesseract/wiki>

b. Ejecuta el instalador y sigue las instrucciones para completar la instalación.

c. Asegúrate de seleccionar la opción "Add to PATH" durante la instalación para que Tesseract esté disponible desde la línea de comandos.

2. Para macOS:

a. Instala Tesseract OCR usando Homebrew ejecutando el siguiente comando en la terminal:

Copy code

```
brew install tesseract
```

3. Para Linux (Ubuntu/Debian):

a. Instala Tesseract OCR usando el administrador de paquetes apt ejecutando el siguiente comando en la terminal:

Copy code

```
sudo apt-get update
sudo apt-get install tesseract-ocr
```

Una vez que hayas instalado Tesseract OCR, asegúrate de reiniciar el entorno de Python y vuelve a ejecutar el código. Con Tesseract instalado y configurado correctamente, el error `TesseractNotFoundError` debería resolverse y podrás extraer el texto de la imagen con `pytesseract`.

En Google Colab después de resolver los errores marcados



```
import requests
from PIL import Image
from io import BytesIO
import pytesseract

# URL de la imagen PNG
url = "https://raw.githubusercontent.com/dragonnomada/chat-gpt-2023/main/datasets/UI_TD_PAPER.png"

# Obtener el contenido de la imagen desde la URL
response = requests.get(url)
image = Image.open(BytesIO(response.content))

# Extraer el texto de la imagen usando pytesseract
texto_extraido = pytesseract.image_to_string(image)

# Imprimir el texto extraído
print(texto_extraido)
```

1. "Saving Time with the User Interface Transition Diagrams (UITD) Editor"  
2. "The UITD Editor: A Tool to Simplify User-System Interaction Modeling"  
3. "Streamlining User Interface Design with the UITD Editor: A Comparative  
Study"  
4. "UITD Editor: Simplifying User-System Interaction Modeling for  
Developers"  
Can the User Interface Transition Diagrams editor (UITD editor) save users' time?  
Cervantes-Ojeda J., Badillo-Salas A., Gomez-Fuentes M.C.  
Department of Applied Mathematics and Systems  
Universidad Autonoma Metropolitana  
México City, México  
[jcervantes@cua.uam.mx](mailto:jcervantes@cua.uam.mx), [dragonnomal23@gmail.com](mailto:dragonnomal23@gmail.com), [mgomez@cua.uam.mx](mailto:mgomez@cua.uam.mx)

.2 Sigue la instrucción de CHATGPT que traduzca el texto a español (pega el texto extraído en <https://chat.openai.com/>)

CR podrías apoyarme en traducir al español el siguiente texto?

iPor supuesto! Estoy aquí para ayudarte. Por favor, proporciona el texto que deseas traducir

1. "Saving Time with the User Interface Transition Diagrams (UITD) Editor"  
2. "The UITD Editor: A Tool to Simplify User-System Interaction Modeling"  
3. "Streamlining User Interface Design with the UITD Editor: A Comparative  
Study"  
4. "UITD Editor: Simplifying User-System Interaction Modeling for  
Developers"  
Can the User Interface Transition Diagrams editor (UITD editor) save users' time?  
Cervantes-Ojeda J., Badillo-Salas A., Gomez-Fuentes M.C.  
Department of Applied Mathematics and Systems  
Universidad Autonoma Metropolitana  
México City, México  
[jcervantes@cua.uam.mx](mailto:jcervantes@cua.uam.mx), [dragonnomal23@gmail.com](mailto:dragonnomal23@gmail.com), [mgomez@cua.uam.mx](mailto:mgomez@cua.uam.mx)  
Abstract— The User Interface Transition Diagram  
(UITD) is a formal modeling notation that simplifies  
the specification and design of user-system



iPor supuesto! Aquí tienes la traducción al español del texto que proporcionaste:



1. "Ahorrando tiempo con el Editor de Diagramas de Transición de Interfaz de Usuario (UITD)"
2. "El Editor UITD: Una Herramienta para Simplificar la Modelización de la Interacción Usuario-Sistema"
3. "Optimizando el Diseño de la Interfaz de Usuario con el Editor UITD: Un Estudio Comparativo"
4. "Editor UITD: Simplificando la Modelización de la Interacción Usuario-Sistema para Desarrolladores"

¿Puede el editor de Diagramas de Transición de Interfaz de Usuario (UITD) ahorrar tiempo a los usuarios?

Cervantes-Ojeda J., Badillo-Salas A., Gomez-Fuentes M.C.

Departamento de Matemáticas Aplicadas y Sistemas

Universidad Autónoma Metropolitana

Ciudad de México, México

[jcervantes@cua.uam.mx](mailto:jcervantes@cua.uam.mx), [dragonnomal23@gmail.com](mailto:dragonnomal23@gmail.com), [mgomez@cua.uam.mx](mailto:mgomez@cua.uam.mx)

Resumen: El Diagrama de Transición de Interfaz de Usuario (UITD) es una notación de modelización formal que simplifica la especificación y el diseño de interacciones usuario-sistema. Es una valiosa herramienta de comunicación para partes interesadas técnicas y no técnicas durante la fase de obtención de requisitos, ya que proporciona una notación simple y completa que es fácil de entender. En este artículo, investigamos la eficiencia de la creación de UITDs utilizando draw.io, un software de diagramación ampliamente utilizado, en comparación con un editor de UITD dedicado. Realizamos un estudio para comparar el tiempo necesario para completar la tarea, así como la facilidad subjetiva de uso y la

Regenera

### .3 Solicita a CHATGPT que te diga quiénes son los autores del artículo



sabes quienes son los autores del texto anterior?



Sí, los autores mencionados en el texto son:



1. Cervantes-Ojeda J.
2. Badillo-Salas A.
3. Gomez-Fuentes M.C.

Pertenecen al Departamento de Matemáticas Aplicadas y Sistemas de la Universidad Autónoma Metropolitana en Ciudad de México, México. Los correos electrónicos proporcionados para contactar a los autores son:

- [jcervantes@cua.uam.mx](mailto:jcervantes@cua.uam.mx)
- [dragonnomal23@gmail.com](mailto:dragonnomal23@gmail.com)
- [mgomez@cua.uam.mx](mailto:mgomez@cua.uam.mx)

Estos autores son los responsables del estudio y el artículo sobre la eficiencia del Editor de Diagramas de Transición de Interfaz de Usuario (UITD) en la simplificación de la especificación y el diseño de interacciones usuario-sistema.

.4 Solicita a CHATGPT que te diga cuáles son los correos relacionados al artículo

Esto ya lo comentó en la pregunta anterior

.5 Solicita a CHATGPT que te diga quién es la universidad, el departamento, la ciudad y el país del artículo

Esto ya lo comentó en la pregunta numero 3

.5 Solicita a CHATGPT que te explique el ABSTRACT

 puedes explicarme lo que quiere decir en el abstracto, toma en cuenta que no se mucho del tema



Por supuesto, estaré encantado de explicarte el resumen (abstract) del texto, teniendo en cuenta que no estás muy familiarizado con el tema.

El resumen del texto se refiere a un estudio que analiza una herramienta llamada "Editor de Diagramas de Transición de Interfaz de Usuario" (UITD, por sus siglas en inglés) y cómo puede ser útil en el proceso de diseño de interacciones entre usuarios y sistemas informáticos.

1. **Importancia del UITD:** El "User Interface Transition Diagram" (Diagrama de Transición de Interfaz de Usuario, UITD) es una especie de dibujo o diagrama que ayuda a las personas a entender cómo los usuarios interactúan con un sistema, como una aplicación o un sitio web. Es como un mapa que muestra las diferentes partes de una interfaz y cómo los usuarios pueden moverse entre ellas.
2. **Simplificación y Comunicación:** El UITD es una herramienta muy valiosa porque simplifica la forma en que describimos y diseñamos estas interacciones entre los usuarios y los sistemas. Es especialmente útil durante la fase inicial en la que se definen los requisitos, ya que facilita la comunicación entre las personas que diseñan el sistema y las que lo usarán, incluso si estas últimas no tienen conocimientos técnicos.
3. **Comparación de Herramientas:** En este estudio, compararon dos formas de crear estos diagramas UITD. Una forma es usando "draw.io", que es un software muy utilizado para crear diferentes tipos de diagramas. La otra forma es utilizando un editor específico diseñado para crear estos diagramas UITD.
4. **Resultados del Estudio:** Los resultados del estudio indicaron que el editor dedicado para crear diagramas UITD fue más eficiente (es decir, más rápido y más fácil de usar) en comparación con el uso de "draw.io". Los participantes prefirieron el editor especializado, lo que sugiere que es importante utilizar herramientas específicas cuando se trata de crear este tipo de modelos formales.
5. **Importancia para Desarrolladores y Diseñadores:** Los resultados tienen implicaciones para las personas que desarrollan software (programadores, desarrolladores) y los diseñadores de sistemas, ya que usar una herramienta diseñada para este propósito puede ser más eficiente y producir mejores resultados en la definición de cómo los usuarios interactuarán con un sistema.

Espero que esta explicación te ayude a entender mejor el contenido del resumen. Si tienes más preguntas o algo no está claro, no dudes en preguntar.

## .6 Sigue a CHATGPT que te explique la introducción

CR

ahora podrías explicarme la introducción, sigue tomando en cuenta que no conozco mucho del tema



iPor supuesto! Vamos a desglosar la introducción teniendo en cuenta que no estás muy familiarizado con el tema.



### I. INTRODUCCIÓN

La introducción es la parte inicial del artículo donde se establece el contexto y el propósito de la investigación. Aquí se presenta de manera general de qué trata el estudio y por qué es relevante.

1. **Importancia del UITD:** El texto comienza mencionando una herramienta llamada "User Interface Transition Diagram" (Diagrama de Transición de Interfaz de Usuario, UITD). Esta herramienta es usada para describir cómo los usuarios interactúan con sistemas informáticos, como aplicaciones de software o sitios web. Es importante entender estas interacciones para diseñar sistemas efectivos y satisfactorios para los usuarios.
2. **Comunicación entre Diferentes Personas:** Se dice que el UITD es una herramienta de comunicación valiosa. Ayuda a que las personas que están diseñando un sistema (como programadores o diseñadores) puedan entenderse mejor con las personas que lo van a usar, incluso si estas últimas no tienen conocimientos técnicos. Esto es especialmente útil en la etapa temprana del diseño, cuando se están definiendo los requisitos.
3. **Eficacia de Herramientas:** En esta introducción, se plantea una pregunta importante: ¿cuál es la mejor manera de crear estos diagramas UITD? Se mencionan dos enfoques: uno es usar "draw.io", un software general para crear diagramas, y el otro es utilizar un editor especializado diseñado específicamente para hacer estos diagramas UITD.
4. **Objetivo del Estudio:** El objetivo de la investigación es comparar la eficiencia de estos dos enfoques. Quieren saber si es más rápido y más fácil usar el editor especializado en comparación con el uso de "draw.io". Además, también quieren saber cuál de las dos opciones es más preferida por las personas que participan en el estudio.
5. **Importancia de los Resultados:** Los resultados de este estudio tienen implicaciones significativas para los desarrolladores de software, diseñadores y otras personas involucradas en la creación de sistemas. Si se encuentra que el editor especializado es más eficiente y preferido, esto podría influir en cómo se diseñan y comunican las interacciones entre los usuarios y los sistemas.

En resumen, la introducción establece la importancia del UITD como herramienta de comunicación en el diseño de sistemas, plantea una pregunta sobre la eficiencia de diferentes enfoques para crear estos diagramas UITD y destaca la relevancia de los resultados para profesionales en el campo de desarrollo y diseño de sistemas.

## .7 Sigue a CHATGPT que extraiga las palabras principales

CR

cuales son las palabras clave o principales en el texto anterior



Las palabras clave o principales en el texto anterior son:



1. **UITD (User Interface Transition Diagram):** El acrónimo que representa el Diagrama de Transición de Interfaz de Usuario, que es la herramienta central de este estudio.
2. **Editor:** Se refiere al software específico que se utiliza para crear los Diagramas de Transición de Interfaz de Usuario (UITD).
3. **Eficacia:** Se menciona la eficacia como un aspecto que se evalúa en el estudio, especialmente en términos de cuán rápido y fácil es usar el editor especializado en comparación con otros enfoques.
4. **Draw.io:** Un software ampliamente utilizado para crear diferentes tipos de diagramas, que se compara con el editor de UITD en el estudio.
5. **Sistemas de Usuario:** La interacción entre los usuarios y sistemas informáticos es el foco central del UITD y del estudio.
6. **Diseñadores:** Las personas involucradas en la creación de sistemas, especialmente desde una perspectiva de diseño, son un público clave para las implicaciones de este estudio.
7. **Desarrolladores:** Aquellas personas que crean y programan sistemas informáticos también son parte importante de la audiencia relevante para los resultados de este estudio.
8. **Comunicación:** La eficacia del UITD como herramienta de comunicación entre diferentes personas con diversos niveles de conocimiento técnico.
9. **Requisitos:** Se menciona que el UITD es relevante durante la fase de definición de requisitos de un sistema, lo que resalta su papel en el proceso de desarrollo.
10. **Simplicidad:** Se enfatiza que el UITD es una herramienta que debe ser comprensible, especialmente para personas que no son expertas técnicas.

Recuerda que estas palabras clave son esenciales para entender la temática del estudio y sus implicaciones.

.8 Sigue solicitando a CHATGPT que genere el código para que reescriba el artículo en formato de Markdown

CR

puedes generar el código para que reescriba el artículo en formato de Markdown



Por supuesto, aquí tienes el texto reescrito en formato Markdown:

Copy code

```
markdown

# Ahorrando Tiempo con el Editor de Diagramas de Transición de Interfaz de Usuari

**Autores:** Cervantes-Ojeda J., Badillo-Salas A., Gomez-Fuentes M.C.

**Departamento:** Matemáticas Aplicadas y Sistemas

**Universidad:** Autónoma Metropolitana

**Ciudad:** Ciudad de México, México

**Correos Electrónicos:** jcervantes@cua.uam.mx, dragonnomadal23@gmail.com, m

## Resumen

El Diagrama de Transición de Interfaz de Usuario (UITD) es una notación de modelización que permite describir el comportamiento de un sistema en términos de estados y transiciones.

**Palabras Clave:** Editor de UITD, Notación de modelización, Obtención de requerimientos, Diseño de interfaces de usuario.

## I. Introducción

La importancia del User Interface Transition Diagram (UITD) radica en su capacidad para simplificar la descripción del comportamiento de los usuarios en un sistema.

La simplicidad es fundamental, especialmente para quienes no son expertos en modelización.

Hasta la fecha, han surgido varias herramientas para apoyar la creación de diagramas de UITD.

Para abordar esta pregunta, llevamos a cabo un estudio en el que los participantes
```

Este fragmento contiene las secciones principales del artículo reescritas en formato Markdown, incluyendo encabezados, resumen y la introducción. Si deseas continuar con el resto del artículo en este formato o si necesitas algún ajuste específico, no dudes en decírmelo.

En Google Colab marcó error, volví a ChatGPT, me corrige de esa forma:

CR

del código anterior puedes apoyarme para corregirlo, será en google colab



Por supuesto, aquí tienes el código corregido para Google Colab en formato Markdown:

```
markdown Copy code

# Ahorrando Tiempo con el Editor de Diagramas de Transición de Interfaz de Usuario

**Autores:** Cervantes-Ojeda J., Badillo-Salas A., Gomez-Fuentes M.C.
**Departamento:** Matemáticas Aplicadas y Sistemas
**Universidad:** Autónoma Metropolitana
**Ciudad:** Ciudad de México, México
**Correos Electrónicos:** jcervantes@cua.uam.mx, dragonnomadal23@gmail.com

## Resumen

El *Diagrama de Transición de Interfaz de Usuario (UITD)* es una notación de modelización que permite describir el comportamiento de un sistema interactivo de forma clara y concisa.

**Palabras Clave:** Editor de UITD, Notación de modelización, Obtención de requerimientos, Diseño de interfaces de usuario.

## I. Introducción

La importancia del *User Interface Transition Diagram (UITD)* radica en su capacidad para simplificar la descripción del comportamiento de los sistemas interactivos, permitiendo una mejor comprensión y comunicación entre los diferentes stakeholders.

La simplicidad es fundamental, especialmente para quienes no son expertos en modelización de sistemas.

Hasta la fecha, han surgido varias herramientas para apoyar la creación de diagramas de UITD.

Para abordar esta pregunta, llevamos a cabo un estudio en el que los participantes...
```

Este fragmento ahora debería ser adecuado para ser utilizado en Google Colab en formato Markdown. Recuerda que en Google Colab, puedes agregar encabezados, texto en negrita, en cursiva y otros elementos de formato Markdown. Si necesitas más ajustes o ayuda adicional, no dudes en decírmelo!

Reg

En google colab, sigue marcando error

 sigue marcando error, este es el siguiente: File "<ipython-input-3-f8d08966a2f7>", line 3  
  \*\*Autores:\*\* Cervantes-Ojeda J., Badillo-Salas A., Gomez-Fuentes M.C.  
  ^  
SyntaxError: invalid syntax

 Entendido, parece que estás intentando ejecutar el texto Markdown directamente en una celda de código en Google Colab, lo cual generará un error de sintaxis porque el Markdown no es un lenguaje válido en una celda de código. En Google Colab, para agregar contenido en formato Markdown, debes usar celdas de tipo "Texto".

Aquí tienes un nuevo código para Google Colab, que utiliza celdas de tipo "Texto" para el contenido en formato Markdown:

1. **Crear celda de texto para el título y autores:**

```
ruby
# Ahorrando Tiempo con el Editor de Diagramas de Transición de Interfaz de Usu
**Autores:** Cervantes-Ojeda J., Badillo-Salas A., Gomez-Fuentes M.C.
**Departamento:** Matemáticas Aplicadas y Sistemas
**Universidad:** Autónoma Metropolitana
**Ciudad:** Ciudad de México, México
**Correos Electrónicos:** jcervantes@cua.uam.mx, dragonnomadal23@gmail.com,
```

1. **Crear celda de texto para el resumen:**

```
swift
## Resumen

El *Diagrama de Transición de Interfaz de Usuario (UITD)* es una notación de
**Palabras Clave:** Editor de UITD, Notación de modelización, Obtención de
```

1. **Crear celda de texto para la introducción:**

```
css
## I. Introducción

La importancia del *User Interface Transition Diagram (UITD)* radica en su
La simplicidad es fundamental, especialmente para quienes no son expertos en
Hasta la fecha, han surgido varias herramientas para apoyar la creación de
Para abordar esta pregunta, llevamos a cabo un estudio en el que los partici
```

Crea celdas de tipo "Texto" en Google Colab y copia el contenido correspondiente en cada celda para que el Markdown sea renderizado correctamente.

The screenshot shows a LaTeX editor interface with the following content:

```

# Ahorrando Tiempo con el Editor de Diagramas de Transición de Interfaz de Usuario (UITD)

**Autores:** Cervantes-Ojeda J., Badillo-Salas A., Gomez-Fuentes M.C.
**Departamento:** Matemáticas Aplicadas y Sistemas
**Universidad:** Autónoma Metropolitana
**Ciudad:** Ciudad de México, México
**Correos Electrónicos:** jcervantes@cua.uam.mx, dragonnomadal23@gmail.com, mgomez@cua.uam.mx

## Resumen

El «Diagrama de Transición de Interfaz de Usuario (UITD)» es una notación de modelización formal que simplifica la especificación y el diseño de interacciones usuario-sistema. Es una valiosa herramienta de comunicación para partes interesadas técnicas y no técnicas durante la fase de obtención de requisitos, ya que proporciona una notación simple y completa que es fácil de entender. En este artículo, investigamos la eficiencia de la creación de UITDs utilizando draw.io, un software de diagramación ampliamente utilizado, en comparación con un editor de UITD dedicado. Realizamos un estudio para comparar el tiempo necesario para completar la tarea, así como la facilidad subjetiva de uso y la satisfacción de los participantes. Nuestros resultados muestran que el editor de UITD es más eficiente y es preferido por los participantes, destacando la importancia de utilizar herramientas especializadas para crear modelos formales como los UITD. Los hallazgos de este estudio tienen implicaciones para desarrolladores de software, diseñadores y otras partes interesadas involucradas en la especificación y el diseño de interacciones usuario-sistema.

## Palabras Clave

Editor de UITD, Notación de modelización, Obtención de requisitos

## I. Introducción

La importancia del «User Interface Transition Diagram (UITD)» radica en su capacidad para simplificar la especificación y el diseño de las interacciones entre usuarios y sistemas. Este diagrama es esencial para la comunicación entre diferentes partes interesadas, incluso ...

```

.9 Solicita a CHATGPT que genere el código para que reescriba el artículo en forma Latex

puedes reescribir el articulo en látex?

Por supuesto, aquí tienes el artículo reescrito en formato LaTeX. Te proporcionaré el contenido para un documento LaTeX básico que puedes utilizar como base:

```

\documentclass{article}

\title{Ahormando Tiempo con el Editor de Diagramas de Transición de Interfaz de Usuario (UITD)}
\author{Cervantes-Ojeda J., Badillo-Salas A., Gomez-Fuentes M.C.}
\date{}
\usepackage[utf8]{inputenc}
\usepackage[spanish]{babel}

\begin{document}

\maketitle

\section*{Resumen}

El \textit{Diagrama de Transición de Interfaz de Usuario (UITD)} es una nota

\textbf{Palabras Clave:} Editor de UITD, Notación de modelización, Obtención de requisitos

\section*{I. Introducción}


```

```
La importancia del \textit{User Interface Transition Diagram (UITD)} radica  
en la simplicidad es fundamental, especialmente para quienes no son expertos en  
el diseño de interfaces de usuario. Hasta la fecha, han surgido varias herramientas para apoyar la creación de  
diagramas de transición de interfaz de usuario, pero su uso puede ser complejo y requiere conocimientos avanzados.  
Para abordar esta pregunta, llevamos a cabo un estudio en el que los participantes  
utilizaron una herramienta intuitiva y fácil de aprender para crear diagramas de transición de interfaz de usuario.  
\end{document}
```

Este código LaTeX contiene el título, autores, resumen e introducción, todos en el formato LaTeX adecuado. Puedes utilizar este código como base para tu documento LaTeX. Recuerda que necesitarás un compilador LaTeX para generar un PDF a partir de este código. Si tienes alguna pregunta adicional o necesitas más ayuda, no dudes en decírmelo. ¡Buena suerte con tu trabajo!



el compilador LaTex lo puedo descargar de alguna pagina?



Sí, puedes descargar un compilador LaTeX, también conocido como sistema TeX, de diversas páginas web. Uno de los sistemas TeX más populares y ampliamente utilizado es "TeX Live". Aquí tienes los pasos para descargar e instalar "TeX Live" en sistemas Windows, macOS y Linux:



#### Windows:

1. Visita la página oficial de "TeX Live" para Windows: <https://www.tug.org/texlive/acquire-netinstall.html>
2. Descarga el instalador "install-tl-windows.exe".
3. Ejecuta el instalador y sigue las instrucciones.

#### macOS:

1. Visita la página oficial de "TeX Live" para macOS: <https://www.tug.org/mactex/mactex-download.html>
2. Descarga el archivo de instalación para macOS.
3. Ejecuta el archivo de instalación y sigue las instrucciones.

#### Linux (Debian/Ubuntu):

1. Abre una terminal.
2. Ejecuta el siguiente comando para instalar "TeX Live":

```
bash                                     Copy code  
sudo apt-get install texlive-full
```

Este comando instalará una versión completa de "TeX Live" en tu sistema.

Recuerda que la instalación puede llevar un tiempo ya que es una suite completa de herramientas para procesar documentos LaTeX. Una vez instalado, podrás compilar documentos LaTeX en archivos PDF utilizando un editor LaTeX como TeXmaker, TeXstudio o cualquier otro editor compatible con LaTeX.

Si tienes alguna pregunta adicional o necesitas ayuda con la instalación, no dudes en preguntar.

.10 Sigue la instrucción de CHATGPT

CR puedes apoyarme a generar un código para escribir el artículo generado de Markdown en un archivo PDF?

Sí, puedo proporcionarte un ejemplo de cómo puedes generar un archivo PDF a partir del contenido Markdown utilizando la biblioteca "Pandoc". Pandoc es una herramienta de línea de comandos que puede convertir documentos entre diferentes formatos, incluyendo Markdown a PDF.

Dado que no puedes instalar Pandoc directamente en Google Colab, necesitarás ejecutar el comando de conversión de Pandoc en tu entorno local. Aquí está el proceso general que puedes seguir:

1. Escribe tu contenido en formato Markdown en un archivo de texto, por ejemplo, "mi\_articulo.md".
2. Usa Pandoc para convertir el archivo Markdown a PDF.

Aquí tienes un ejemplo de cómo podrías hacerlo en tu entorno local (no en Google Colab) si tienes Pandoc instalado:

1. Crea un archivo de Markdown llamado "mi\_articulo.md" con el contenido del artículo.
2. Abre una terminal en el mismo directorio donde tienes el archivo "mi\_articulo.md".
3. Ejecuta el siguiente comando para convertir el archivo Markdown a PDF:

```
bash
pandoc mi_articulo.md -o mi_articulo.pdf
Copy code
```

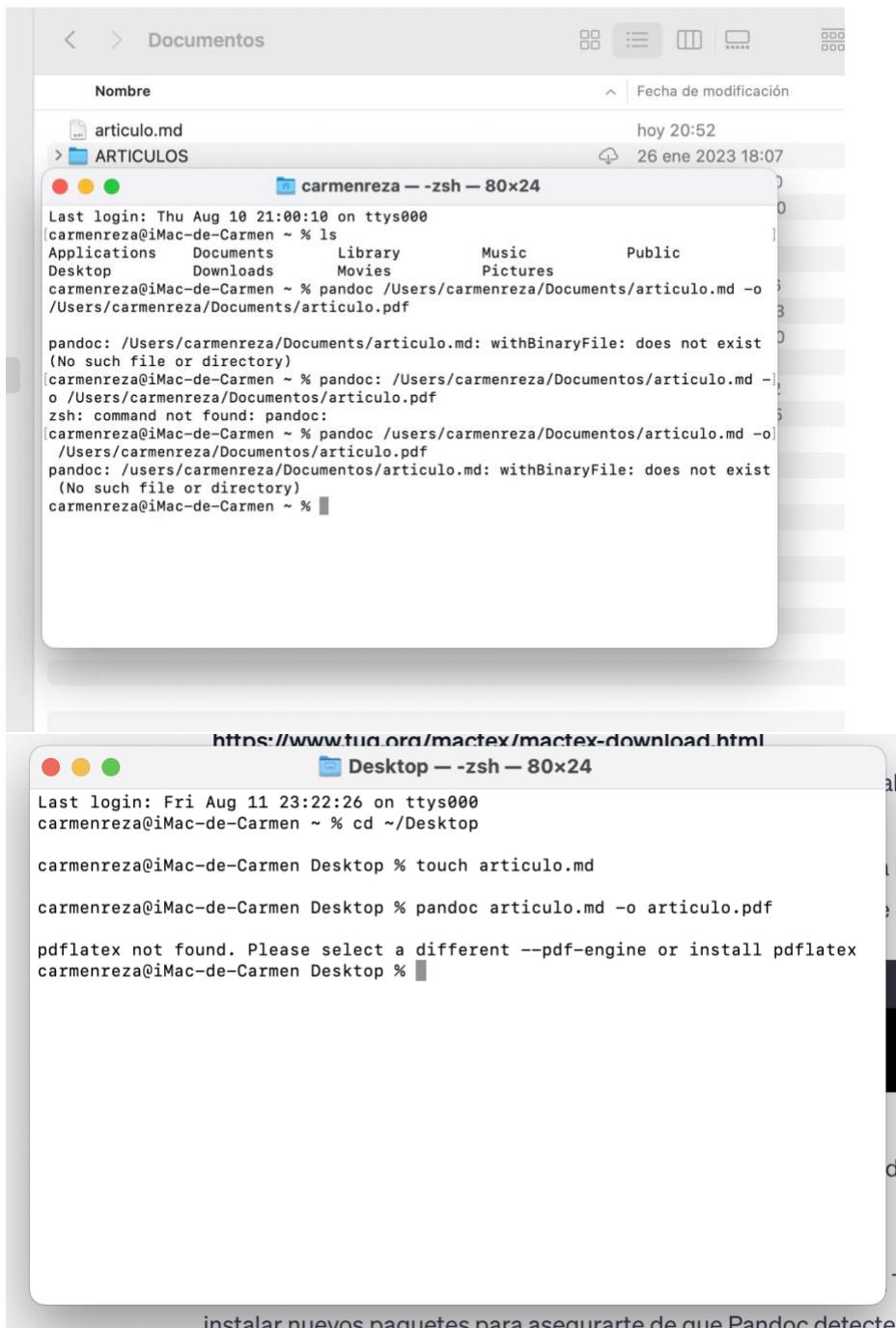
Este comando toma el archivo Markdown "mi\_articulo.md" y lo convierte en un archivo PDF llamado "mi\_articulo.pdf" en el mismo directorio.

Recuerda que necesitarás tener Pandoc instalado en tu sistema local para ejecutar este comando. Consulta la documentación de Pandoc para obtener instrucciones de instalación según tu sistema operativo.

Si deseas hacer esto directamente en Google Colab, no es posible generar un archivo PDF directamente, pero podrías generar el PDF en tu sistema local como se describe arriba y luego subirlo a Google Colab para trabajar con él en el entorno de Colab.

NOTA: Prueba que los códigos sean funcionales y genera archivo PDF mostrando el resultado

Instalé varias veces Pandoc en mi computadora, pero nunca me salio en el finder y tampoco en Launchpad, lo busqué varias veces pero no lo encontré, le pedí a ChatGPT que me ayudara, me hizo varias recomendaciones, pero aun asi no pude realizar este punto. No pude instalar LaTex en mi computadora



The screenshot shows a Mac desktop environment. At the top, there's a menu bar with 'Documentos' selected. Below it is a file browser window titled 'Documentos' showing files 'articulo.md' and a folder 'ARTICULOS'. A terminal window is open in the foreground, titled 'carmenreza -- zsh -- 80x24'. It displays a command-line session where the user attempts to run Pandoc on a Markdown file. The session starts with a login message, then the user runs 'ls' to show directory contents, followed by a command to convert 'articulo.md' to a PDF. The terminal shows several errors from Pandoc indicating it cannot find the file or the pdflatex engine. The user then tries to touch the file and run Pandoc again, but receives a similar error about pdflatex not being found.

```
Last login: Thu Aug 10 21:00:10 on ttys000
carmenreza@iMac-de-Carmen ~ % ls
Applications    Documents    Library    Music      Public
Desktop        Downloads   Movies     Pictures
carmenreza@iMac-de-Carmen ~ % pandoc /Users/carmenreza/Documents/articulo.md -o
/Users/carmenreza/Documents/articulo.pdf
pandoc: /Users/carmenreza/Documents/articulo.md: withBinaryFile: does not exist
(No such file or directory)
carmenreza@iMac-de-Carmen ~ % pandoc: /Users/carmenreza/Documentos/articulo.md -o
/Users/carmenreza/Documentos/articulo.pdf
zsh: command not found: pandoc:
carmenreza@iMac-de-Carmen ~ % pandoc /users/carmenreza/Documentos/articulo.md -o
/Users/carmenreza/Documentos/articulo.pdf
pandoc: /users/carmenreza/Documentos/articulo.md: withBinaryFile: does not exist
(No such file or directory)
carmenreza@iMac-de-Carmen ~ %

https://www.tug.org/mactex/mactex-download.html
Desktop -- zsh -- 80x24
Last login: Fri Aug 11 23:22:26 on ttys000
carmenreza@iMac-de-Carmen ~ % cd ~/Desktop
carmenreza@iMac-de-Carmen Desktop % touch articulo.md
carmenreza@iMac-de-Carmen Desktop % pandoc articulo.md -o articulo.pdf
pdflatex not found. Please select a different --pdf-engine or install pdflatex
carmenreza@iMac-de-Carmen Desktop %
```

instalar nuevos paquetes para asegurarte de que Pandoc detecte

## P102.3 - Caso de Estudio

.1 Diseña un caso de estudio dónde tomarías varias imágenes que contengan información relevante, por ejemplo, un acta constitutiva para extraer la empresa y la notaría que generaron el acta constitutiva y demás valores, una acta judicial dónde extraerías a personajes involucrados y causas, un estado de cuenta dónde extraerías montos y beneficiarios, etc.

El caso de estudio es analizar varias imágenes del diario oficial de: **ACUERDO SO/IV-22/08,R, mediante el cual se aprueba el nuevo Estatuto Orgánico de la Comisión de Operación y Fomento de Actividades Académicas del Instituto Politécnico Nacional, del 3 de enero del 2023, en donde se le pedirá a CHAGPT realizar algunas consultas como un resumen, aplicar algunas ideas que se han dado en la practica 2.**

.2 Genera el código en Python (a través de CHATGPT) para extraer el texto con OCR y luego extraer la información con CHATGPT.

 apoyarme en la generación de un código que extraiga el texto de varias imágenes desde la url [https://correoipn-my.sharepoint.com/:f/g/personal/creza\\_ipn\\_mx/EjzOGX3-gslAm2ga8VYMMc4Bz8k6rIPW5MfXVtX\\_YJHjBA?e=g3nQ2a](https://correoipn-my.sharepoint.com/:f/g/personal/creza_ipn_mx/EjzOGX3-gslAm2ga8VYMMc4Bz8k6rIPW5MfXVtX_YJHjBA?e=g3nQ2a), usando OCR en Phyton



Por supuesto, puedo proporcionarte un código en Python para extraer texto de imágenes utilizando OCR (Optical Character Recognition). Sin embargo, es importante mencionar que no puedo acceder a enlaces externos, por lo que necesitaré que me proporciones una URL directa a una imagen específica que deseas procesar con OCR. Una vez que tengas una URL directa a una imagen, puedo mostrarte cómo usar la biblioteca `pytesseract` junto con `requests` para lograr esto.

Primero, asegúrate de instalar las bibliotecas necesarias ejecutando estos comandos en tu entorno de Python:

```
bash
Copy code
pip install pytesseract
pip install requests
```

Luego, puedes usar el siguiente código como ejemplo para extraer texto de una imagen:

```
python
import requests
from PIL import Image
import pytesseract

# URL directa a la imagen que deseas procesar
image_url = "URL_DIRECTA_A_LA_IMAGEN"

# Obtener la imagen desde la URL
response = requests.get(image_url)
img = Image.open(io.BytesIO(response.content))

# Procesar la imagen con OCR
text = pytesseract.image_to_string(img)

# Imprimir el texto extraído
print(text)
```

Asegúrate de reemplazar `"**URL\_DIRECTA\_A\_LA\_IMAGEN**"` con la URL directa a la imagen que deseas procesar. Este código descargará la imagen desde la URL, la procesará utilizando OCR y luego imprimirá el texto extraído.

Recuerda que **pytesseract** utiliza la biblioteca Tesseract OCR en segundo plano, por lo que es necesario tener Tesseract OCR instalado en tu sistema. Puedes descargarlo e instalarlo desde: <https://github.com/tesseract-ocr/tesseract>

Si tienes varias imágenes a procesar, puedes colocar el código en un bucle para iterar sobre las URLs de las imágenes. ¡Espero que esto te ayude! Si tienes alguna pregunta adicional, por favor házmelo saber.

CR

estoy trabajando en Google colab



Genial, Google Colab es una excelente plataforma para trabajar en Python, especialmente para proyectos de aprendizaje automático y procesamiento de datos. A continuación, te mostraré cómo utilizar OCR con **pytesseract** en Google Colab para extraer texto de imágenes. Asegúrate de tener acceso a las imágenes que deseas procesar.

1. Importa las bibliotecas y configura Tesseract OCR:

```
python
!pip install pytesseract
!apt-get install -y tesseract-ocr

import pytesseract
from PIL import Image
```

1. Monta Google Drive para acceder a las imágenes. Esto es necesario si las imágenes están en tu Google Drive:

```
python
from google.colab import drive
drive.mount('/content/drive')
```

1. Define una función para extraer texto de una imagen:

```
python
Copy code

def extract_text_from_image(image_path):
    img = Image.open(image_path)
    text = pytesseract.image_to_string(img)
    return text
```

1. Llama a la función `extract\_text\_from\_image` proporcionando la ruta a la imagen que deseas procesar:

```
python
Copy code

image_path = "/content/drive/My Drive/path/to/your/image.jpg" # Ruta a tu imagen
text_extracted = extract_text_from_image(image_path)
print(text_extracted)
```

Recuerda reemplazar `"/content/drive/My Drive/path/to/your/image.jpg"` con la ruta real a la imagen que deseas procesar.

Si las imágenes están disponibles en una URL, puedes descargarlas primero y luego proporcionar la ruta local en Colab. Asegúrate de que la imagen se pueda acceder desde Colab. Con esto deberías poder extraer texto de imágenes utilizando OCR en Google Colab. ¡Espero que esto te sea útil!



por favor apoyarme a resolver este error: FileNotFoundError  
recent call last)

```
<ipython-input-6-e10f7138a25e> in <cell line: 5>()
      3
      4 # Ahora puedes usar esta ruta para extraer el texto de la imagen
----> 5 text_extracted = extract_text_from_image(image_path)
      6 print(text_extracted)
```

```
1 frames
/usr/local/lib/python3.10/dist-packages/PIL/Image.py in open(fp, mode, formats)
3225
3226     if filename:
-> 3227         fp = builtins.open(filename, "rb")
3228         exclusive_fp = True
3229
```

FileNotFoundError: [Errno 2] No such file or directory: 'https://drive.google.com/uc? id=1g6AecoO8CpsbN5fbOxqfm\_Jf0z23-2MR'

Traceback (most



El error que estás enfrentando, "FileNotFoundError", indica que el archivo de imagen no pudo ser encontrado en la ruta que proporcionaste. En este caso, parece que estás intentando cargar una imagen desde una URL de Google Drive utilizando la URL directa, pero la biblioteca `PIL` (Python Imaging Library) no puede acceder a archivos desde URLs de esta manera.

Para resolver este problema, primero necesitas descargar la imagen desde la URL de Google Drive y guardarla localmente en tu entorno de Colab antes de poder procesarla con OCR. Aquí hay una manera de hacerlo:

```
python Copy code

import requests
from PIL import Image
import io

# URL directa a la imagen en Google Drive
image_url = "https://drive.google.com/uc?id=1g6Aeco08CpsbN5fb0xqfm_Jf0z23-2MR"

# Descargar la imagen y guardarla localmente
response = requests.get(image_url)
image = Image.open(io.BytesIO(response.content))

# Procesar la imagen con OCR
text_extracted = pytesseract.image_to_string(image)
print(text_extracted)
```

En este código, primero descargamos la imagen desde la URL utilizando la biblioteca `'requests'`, luego abrimos la imagen usando `'PIL'`, y finalmente, procesamos la imagen con OCR utilizando `'pytesseract'`.

Asegúrate de que la imagen en la URL proporcionada tenga permisos de acceso compartido o sea pública para que puedas descargarla. Esto debería resolver el error y permitirte extraer el texto de la imagen en Google Colab.

+ Código + Texto

```
✓ 17s import requests
from PIL import Image
import io

# URL directa a la imagen en Google Drive
image_url = "https://drive.google.com/uc?id=1g6Aeco08CpsbN5fb0xqfm_Jf0z23-2MR"

# Descargar la imagen y guardarla localmente
response = requests.get(image_url)
image = Image.open(io.BytesIO(response.content))

# Procesar la imagen con OCR
text_extracted = pytesseract.image_to_string(image)
print(text_extracted)
```

DOF: 03/01/2023

ACUERDO SO/IV-22/08,R, mediante el cual se aprueba el nuevo Estatuto Orgánico de la Comisión de Operación y Fomento de Actividades Académicas del Instituto Politécnico Nacional.

Comisión de Operación y Fomento de Actividades Académicas del I.P.N.- Secretaría Ejecutiva.

ESTATUTO ORGÁNICO DE LA COMISIÓN DE OPERACIÓN Y FOMENTO DE ACTIVIDADES ACADEMÍCAS DEL INSTITUTO  
POLITECNICO NACIONAL

La Junta Directiva de la Comisión de Operación y Fomento de Actividades Académicas del Instituto Politécnico Nacional, durante el desarrollo de su Cuarta Sesión Ordinaria correspondiente al ejercicio fiscal 2022, aprobó la modificación del presente Estatuto Orgánico del Organismo Descentralizado de la Administración Pública Federal, con fundamento en lo dispuesto en los artículos 58, fracción VIII de la Ley Federal de las Entidades Paraestatales y QUINTO, fracción III de su DECRETO constitutivo, por lo que se tiene a bien expedir el siguiente:

ACUERDO SO/IV-22/08,R, Mediante el cual se aprueba el nuevo Estatuto Orgánico de la Comisión de Operación y Fomento de Actividades Académicas del Instituto Politécnico Nacional.

ARTICULO UNICO Se abroga el Estatuto Orgánico de la Comisión de Operación y Fomento de Actividades Académicas del Instituto Politécnico Nacional, publicado en el Diario Oficial de la Federación el 13 de agosto de 2021 y aprobado en la sesión del 26 de junio de 2020 y, en su lugar se aprueba el nuevo Estatuto Orgánico de la Comisión de Operación y Fomento de Actividades Académicas del Instituto Politécnico Nacional, para quedar como sigue:

ESTATUTO ORGÁNICO DE LA COMISIÓN DE OPERACIÓN Y FOMENTO DE ACTIVIDADES  
ACADEMÍCAS DEL INSTITUTO POLITECNICO NACIONAL

 puedes resumir el siguiente texto?

 Lamento la confusión, pero parece que no proporcionaste el texto que deseas que resuma.    
Si puedes proporcionar el texto que quieres que resuma, estaré encantado de ayudarte. Por favor, copia y pega el texto aquí, y te proporcionaré un resumen.

 DOF: 03/01/2023 

ACUERDO SO/IV-22/08.R, mediante el cual se aprueba el nuevo Estatuto Orgánico de la Comisión de Operación y Fomento de Actividades Académicas del Instituto Politécnico Nacional.

Comisión de Operación y Fomento de Actividades Académicas del I.P.N.- Secretaría Ejecutiva.

ESTATUTO ORGÁNICO DE LA COMISIÓN DE OPERACIÓN Y FOMENTO DE ACTIVIDADES ACADEMÍCAS DEL INSTITUTO POLITÉCNICA NACIONAL

La Junta Directiva de la Comisión de Operación y Fomento de Actividades Académicas del Instituto Politécnico Nacional, durante el desarrollo de su Cuarta Sesión Ordinaria correspondiente al ejercicio fiscal 2022, aprobó la modificación del presente Estatuto Orgánico del Organismo Descentralizado de la Administración Pública Federal, con fundamento en lo dispuesto en los artículos 58, fracción VIII de la Ley Federal de las Entidades Paraestatales y QUINTO, fracción III de su DECRETO constitutivo, por lo que se tiene a bien expedir el siguiente:

ACUERDO SO/IV-22/08.R, Mediante el cual se aprueba el nuevo Estatuto Orgánico de la Comisión de Operación y Fomento de Actividades Académicas del Instituto Politécnico Nacional.

ARTICULO UNICO Se abroga el Estatuto Orgánico de la Comisión de Operación y Fomento de Actividades Académicas del Instituto Politécnico Nacional, publicado en el Diario Oficial de la Federación el 13 de agosto de 2021 y aprobado en la sesión del 26 de junio de 2020 y, en su lugar se aprueba el nuevo Estatuto Orgánico de la Comisión de Operación y Fomento de Actividades Académicas del Instituto Politécnico Nacional, para quedar como sigue:

ESTATUTO ORGÁNICO DE LA COMISIÓN DE OPERACIÓN Y FOMENTO DE ACTIVIDADES ACADEMÍCAS DEL INSTITUTO POLITÉCNICO NACIONAL

CONSIDERANDOS

UNICO.- El propósito fundamental de este documento es delimitar las funciones de las Unidades que conforman la entidad, orientando a los responsables a través de la definición de los tramos de control, las líneas de mando, la asignación de autoridad, responsabilidad de cada una de ellas, evitando la evasión o duplicidad de funciones y propiciando el uso racional de los recursos de que dispone la Comisión para la realización de sus fines.

MISIÓN.

hay mucho más texto extraido en google colab

 El texto es un acuerdo que aprueba el nuevo Estatuto Orgánico de la Comisión de Operación y Fomento de Actividades Académicas del Instituto Politécnico Nacional (IPN). El estatuto establece la misión, visión y disposiciones generales de la comisión, así como su objetivo de apoyar técnica y económicamente al IPN para lograr sus finalidades educativas y contribuir al desarrollo tecnológico, económico, social y cultural del país. El estatuto también define términos importantes, como "Adquisiciones, Arrendamientos y Servicios", "Instituto", "Ley", "Comisión", "COCODI", "Unidades Administrativas" y "Dependencia Politécnica", y establece la autonomía de gestión de la Comisión para cumplir con sus objetivos y metas. Además, se menciona la existencia de la Junta Directiva como el órgano máximo de gobierno del organismo público descentralizado.

CR

cuales son las ideas principales del texto?



Las ideas principales del texto son:



1. **Aprobación del Nuevo Estatuto Orgánico:** El texto se refiere al acuerdo mediante el cual se aprueba el nuevo Estatuto Orgánico de la Comisión de Operación y Fomento de Actividades Académicas del Instituto Politécnico Nacional (IPN).
2. **Modificación del Estatuto:** La Junta Directiva del IPN aprobó la modificación del Estatuto Orgánico durante una sesión ordinaria del ejercicio fiscal 2022.
3. **Objetivo del Estatuto:** El Estatuto tiene como objetivo definir las funciones y organización de las unidades que componen la entidad, estableciendo tramos de control, líneas de mando, asignación de autoridad y responsabilidades para evitar duplicidad de funciones y utilizar racionalmente los recursos de la Comisión.
4. **Misión de la Comisión:** La misión de la Comisión es fomentar las actividades académicas del IPN mediante la captación de donativos del sector público, privado y social, así como otorgar becas y apoyos económicos a docentes, investigadores y alumnos para contribuir al desarrollo tecnológico, económico, social y cultural del país.
5. **Visión de la Comisión:** La Comisión aspira a ser un modelo de apoyo a la labor educativa, de investigación científica y desarrollo tecnológico y social del IPN, contribuyendo a soluciones innovadoras para problemas nacionales y promoviendo un entorno de inclusión, igualdad y transparencia.
6. **Estructura y Atribuciones:** El Estatuto establece las bases de organización de la Comisión, su estructura básica y sus atribuciones, otorgándole autonomía de gestión para cumplir con sus objetivos y metas.
7. **Términos Definidos:** Se definen términos importantes, como "Adquisiciones, Arrendamientos y Servicios", "Instituto", "Ley", "Comisión", "COCODI", "Unidades Administrativas" y "Dependencia Politécnica", que tienen relevancia en el contexto del Estatuto.
8. **Junta Directiva:** Se menciona la existencia de la Junta Directiva como el órgano máximo de gobierno del organismo público descentralizado.

CR

a quien beneficia el nuevo estatuto



El nuevo Estatuto Orgánico de la Comisión de Operación y Fomento de Actividades Académicas del Instituto Politécnico Nacional (IPN) beneficia principalmente a las actividades académicas y a la comunidad involucrada en el IPN. A través de la Comisión, se busca impulsar el desarrollo tecnológico, económico, social y cultural del país, así como mejorar las condiciones de docentes, investigadores y alumnos. Algunos de los principales beneficiarios incluyen:



1. **Alumnos:** La Comisión podría otorgar becas y apoyos económicos a los estudiantes del IPN, lo que les proporcionaría recursos para continuar sus estudios y promover un ambiente de aprendizaje más enriquecedor.
2. **Docentes e Investigadores:** Los profesores e investigadores del IPN pueden recibir apoyo económico y recursos para desarrollar proyectos académicos y de investigación, lo que podría mejorar la calidad de la enseñanza y la producción de conocimiento.
3. **Desarrollo Tecnológico:** La Comisión puede captar donativos y recursos para adquirir equipo especializado de vanguardia, lo que beneficia el desarrollo tecnológico y científico en el IPN, y potencialmente contribuir a la innovación en el país.
4. **Mantenimiento de Instalaciones:** El apoyo económico también puede destinarse al mantenimiento de las instalaciones del IPN, lo que mejora el entorno de estudio y trabajo para todos.
5. **Contribución Nacional:** Al impulsar actividades académicas y tecnológicas, la Comisión puede contribuir al desarrollo económico y social del país al fomentar la generación de conocimiento y soluciones innovadoras.

6. **Inclusión y Transparencia:** La visión de la Comisión incluye un entorno de inclusión, igualdad y transparencia, lo que puede beneficiar a la comunidad al promover un ambiente justo y transparente en el que se gestionan los recursos.

En resumen, el nuevo estatuto busca beneficiar a la comunidad académica del IPN, promover el desarrollo tecnológico, mejorar las condiciones de estudio y trabajo, y contribuir al avance del país en términos de conocimiento e innovación.

CR

por qué es importante este estatuto?



El Estatuto Orgánico de la Comisión de Operación y Fomento de Actividades Académicas del Instituto Politécnico Nacional (IPN) es importante por varias razones fundamentales:

1. **Organización y Estructura:** El estatuto establece la organización y estructura de la Comisión, definiendo las funciones, atribuciones y relaciones de las unidades que componen la entidad. Esto es esencial para garantizar una gestión efectiva de las actividades académicas y administrativas del IPN.
2. **Misión y Visión Claras:** El estatuto establece la misión y visión de la Comisión, proporcionando una guía clara sobre los objetivos y propósitos de la entidad. Esto asegura que todas las actividades y decisiones de la Comisión estén alineadas con su propósito fundamental.
3. **Apoyo Económico:** Uno de los roles esenciales de la Comisión es captar donativos y recursos económicos para apoyar las actividades académicas, becas, investigación y desarrollo tecnológico en el IPN. Esto es vital para asegurar que el IPN cuente con los recursos necesarios para mantener su excelencia en la educación y la investigación.
4. **Contribución al Desarrollo Nacional:** La Comisión contribuye al desarrollo tecnológico, económico, social y cultural del país al fomentar la generación de conocimiento, la innovación y el apoyo a proyectos académicos. Esto tiene un impacto directo en el progreso del país.
5. **Transparencia y Equidad:** El estatuto promueve la transparencia en la gestión de recursos y establece principios de igualdad, inclusión y transparencia en el trabajo de la Comisión. Esto es importante para garantizar una administración justa y eficiente.
6. **Autonomía y Eficiencia:** El estatuto otorga autonomía de gestión a la Comisión, permitiendo que tome decisiones de manera eficiente y directa para cumplir con sus objetivos. Esto es crucial para agilizar procesos y lograr resultados efectivos.

En resumen, el Estatuto Orgánico de la Comisión es fundamental para establecer las bases de operación de la entidad, asegurar el apoyo económico necesario, promover la transparencia y eficiencia, y contribuir al desarrollo académico y tecnológico del IPN y del país en general.

CR

qué pasaría si no se cumple el estatuto?



Si no se cumple el Estatuto Orgánico de la Comisión de Operación y Fomento de Actividades Académicas del Instituto Politécnico Nacional (IPN), podrían ocurrir una serie de consecuencias negativas que afectarían a la institución, sus actividades académicas, la comunidad involucrada y su contribución al desarrollo del país. Algunas de las posibles repercusiones incluyen:

1. **Desorganización:** El incumplimiento del estatuto podría llevar a la desorganización en la gestión de las actividades académicas y administrativas de la Comisión. Esto podría generar ineficiencias, falta de coordinación y confusión en la toma de decisiones.
  2. **Falta de Recursos:** El estatuto establece la captación de donativos y recursos económicos como uno de los roles principales de la Comisión. Si esto no se cumple, podría haber una escasez de fondos para apoyar becas, investigación, desarrollo tecnológico y mantenimiento de instalaciones, lo que afectaría negativamente la calidad de la educación y la investigación en el IPN.
  3. **Impacto en la Comunidad:** La falta de recursos y apoyo económico podría impactar a la comunidad académica del IPN, incluidos estudiantes, docentes e investigadores, al no poder acceder a becas, apoyos económicos o recursos para proyectos.
  4. **Pérdida de Contribución Nacional:** La Comisión juega un papel importante en el desarrollo tecnológico, económico, social y cultural del país. Si no se cumplen sus objetivos, se perdería una fuente valiosa de contribución a la innovación y soluciones a problemas nacionales.
  5. **Falta de Transparencia y Equidad:** El incumplimiento del estatuto podría llevar a la falta de transparencia en la gestión de recursos y la falta de equidad en la distribución de apoyos y recursos, lo que generaría desconfianza y conflictos.
6. **Perdida de Autonomía:** El estatuto otorga autonomía de gestión a la Comisión para cumplir con sus objetivos. Si no se cumple, la Comisión podría perder su capacidad de tomar decisiones efectivas y gestionar sus recursos de manera independiente.

En resumen, el incumplimiento del Estatuto Orgánico podría tener un impacto negativo en la organización, la comunidad académica, los recursos disponibles y la contribución al desarrollo del IPN y del país. Es importante que se cumpla el estatuto para garantizar el funcionamiento eficiente y el logro de los objetivos de la Comisión.

 qué organismo elaboró el estatuto?

 El estatuto fue elaborado por la Junta Directiva de la Comisión de Operación y Fomento de Actividades Académicas del Instituto Politécnico Nacional (IPN). El texto menciona que durante la Cuarta Sesión Ordinaria correspondiente al ejercicio fiscal 2022, la Junta Directiva aprobó la modificación del Estatuto Orgánico. La Junta Directiva es el órgano máximo de gobierno de la Comisión y tiene la autoridad para aprobar cambios y decisiones importantes relacionadas con su funcionamiento y estructura. Por lo tanto, la Junta Directiva del IPN es la entidad que elaboró y aprobó el nuevo Estatuto Orgánico de la Comisión de Operación y Fomento de Actividades Académicas.

 cual fue la fecha de elaboración?

 El texto proporcionado no especifica la fecha exacta de elaboración del nuevo Estatuto Orgánico de la Comisión de Operación y Fomento de Actividades Académicas del Instituto Politécnico Nacional (IPN). El documento hace referencia a la "Cuarto Sesión Ordinaria correspondiente al ejercicio fiscal 2022" en la que se aprobó la modificación del estatuto.

Para obtener la fecha exacta de elaboración o aprobación, se requeriría consultar fuentes adicionales o documentos oficiales proporcionados por el Instituto Politécnico Nacional o la propia Comisión de Operación y Fomento de Actividades Académicas, donde es probable que se encuentre la fecha precisa en la que se realizó esta modificación. Sin embargo, con la información proporcionada en el texto original, no podemos determinar con certeza la fecha exacta de elaboración del estatuto.

 el 03/01/2023 será el dato de la fecha?

 Sí, en el encabezado del texto proporcionado se menciona "DOF: 03/01/2023", lo que sugiere que la publicación o el registro del texto en cuestión se realizó en el Diario Oficial de la Federación (DOF) el 3 de enero de 2023. Sin embargo, es importante aclarar que esta fecha podría referirse a la fecha de publicación oficial del estatuto en el DOF y no necesariamente a la fecha exacta en que se elaboró o se aprobó el estatuto. Para obtener la fecha precisa de elaboración o aprobación, es necesario buscar información adicional en documentos oficiales o comunicados proporcionados por el Instituto Politécnico Nacional (IPN) o la Comisión de Operación y Fomento de Actividades Académicas.

 quien es el IPN?

 El Instituto Politécnico Nacional (IPN) es una institución de educación superior y de investigación ubicada en México. Fundado en 1936, es una de las universidades técnicas más destacadas en América Latina y tiene un enfoque significativo en ciencia, tecnología e ingeniería. El IPN se destaca por su contribución a la formación de profesionales altamente capacitados en áreas técnicas y científicas.



Algunas características del IPN incluyen:

1. **Enfoque en Ciencia y Tecnología:** El IPN es conocido por sus programas de estudio en áreas como ingeniería, ciencias exactas, tecnología, ciencias de la salud y ciencias sociales.
2. **Educación Técnica:** Ofrece programas de educación técnica y profesional en diversos niveles, desde carreras de técnico hasta programas de posgrado.
3. **Investigación:** Tiene una fuerte tradición de investigación científica y tecnológica en una variedad de campos, con la creación de centros de investigación y la promoción de proyectos innovadores.
4. **Innovación:** El IPN ha sido un motor de innovación y desarrollo en México, contribuyendo a la industria y al avance tecnológico del país.
5. **Formación de Profesionales:** Ha contribuido a la formación de ingenieros, científicos, tecnólogos y profesionales en muchas áreas, y ha desempeñado un papel importante en el desarrollo económico y social de México.

El IPN tiene múltiples campus y unidades académicas distribuidas en distintos estados de México, y es una institución reconocida tanto a nivel nacional como internacional por su excelencia en educación e investigación técnica.

 quien es Comisión de Operación y Fomento de Actividades Académicas?

 La Comisión de Operación y Fomento de Actividades Académicas es un órgano o entidad relacionada con el Instituto Politécnico Nacional (IPN) en México. El nombre completo de esta comisión es "Comisión de Operación y Fomento de Actividades Académicas del Instituto Politécnico Nacional".



La función de esta comisión es apoyar técnicamente y económicaamente al IPN en la realización de sus actividades académicas y en el fomento del desarrollo tecnológico, económico, social y cultural del país. En particular, se dedica a la captación de donativos del sector público, privado y social, así como a otorgar becas y apoyos económicos a docentes, investigadores y alumnos.

La comisión tiene un papel relevante en el funcionamiento y desarrollo del IPN, contribuyendo a mejorar las condiciones de estudio, investigación y formación de profesionales en el ámbito académico y científico en México. Su labor incluye la gestión de recursos para impulsar la educación y la investigación en el IPN, y también busca promover la innovación y el avance tecnológico en el país.



que pasaría si no existiera la Comisión de Operación y Fomento de Actividades Académicas



Si la Comisión de Operación y Fomento de Actividades Académicas dejara de existir, tendría varios impactos significativos en el Instituto Politécnico Nacional (IPN) y en las actividades académicas que realiza. Algunas de las consecuencias posibles podrían ser:

1. **Escasez de Recursos:** La comisión tiene la responsabilidad de captar donativos y recursos económicos para apoyar actividades académicas, becas, investigación y desarrollo tecnológico en el IPN. Sin la comisión, el IPN podría enfrentar dificultades para obtener fondos adicionales, lo que podría afectar el funcionamiento de programas, proyectos e infraestructura.
2. **Reducción en Apoyos y Becas:** La comisión otorga becas y apoyos económicos a docentes, investigadores y alumnos del IPN. Su ausencia podría llevar a una reducción en la disponibilidad de estas becas, lo que afectaría a la comunidad académica y a la posibilidad de acceso a la educación superior para estudiantes con necesidades económicas.
3. **Impacto en la Investigación y Desarrollo:** La comisión juega un papel importante en el apoyo a la investigación científica y el desarrollo tecnológico en el IPN. Sin su contribución, podrían verse limitados los proyectos de investigación, la adquisición de equipo especializado y la financiación de iniciativas innovadoras.
4. **Disminución de la Contribución al Desarrollo Nacional:** La comisión contribuye al desarrollo tecnológico, económico, social y cultural del país a través de su apoyo a las actividades académicas. Su ausencia podría reducir la capacidad del IPN para contribuir a soluciones innovadoras a problemas nacionales y limitar su impacto positivo en el desarrollo de México.
5. **Mayor Dependencia en Recursos Gubernamentales:** Sin la comisión, el IPN podría depender más de recursos gubernamentales, lo que podría influir en su autonomía y capacidad para gestionar sus actividades de manera eficiente.

5. **Mayor Dependencia en Recursos Gubernamentales:** Sin la comisión, el IPN podría depender más de recursos gubernamentales, lo que podría influir en su autonomía y capacidad para gestionar sus actividades de manera eficiente.

En resumen, la falta de la Comisión de Operación y Fomento de Actividades Académicas podría tener un impacto negativo en la financiación, apoyo a la investigación, desarrollo tecnológico y contribución al desarrollo nacional del IPN. Es importante para la comunidad académica y el país contar con entidades que fomenten el avance de la educación superior y la investigación.

### .3 Genera el código para reportar la información extraída y resumida en un documento PDF.



podrías apoyarme en la generación de un código para reportar una información extraída en pdf, estoy trabajando en google colaborando



Por supuesto, puedo ayudarte a generar un código en Python para reportar información extraída de un PDF en Google Colab. Aquí tienes un ejemplo básico que utiliza la biblioteca PyPDF2 para extraer texto de un archivo PDF y muestra el contenido:

```
python
import PyPDF2

# Ruta del archivo PDF
pdf_file_path = "/content/drive/MyDrive/path_to_your_pdf.pdf"

# Función para extraer texto de un PDF
def extract_text_from_pdf(pdf_path):
    text = ""
    with open(pdf_path, "rb") as pdf_file:
        pdf_reader = PyPDF2.PdfReader(pdf_file)
        for page_number in range(len(pdf_reader.pages)):
            page = pdf_reader.pages[page_number]
            text += page.extract_text()
    return text

# Llamada a la función y reporte del contenido extraido
extracted_text = extract_text_from_pdf(pdf_file_path)
print(extracted_text)
```



Asegúrate de reemplazar `"/content/drive/MyDrive/path\_to\_your\_pdf.pdf"` con la ruta real de tu archivo PDF en Google Drive.

Este código utilizará la biblioteca PyPDF2 para extraer el texto de todas las páginas del PDF y lo imprimirá en la salida estándar. Puedes personalizar este código para adaptarlo a tus necesidades específicas de procesamiento y presentación de la información extraída.

Supongo que no entendió la pregunta, al parecer me esta dando un codigo para extraer un archivo en pdf, es importante redactar bien la pregunta para que no haya confusiones, aun asi seguí intentando realizar la extraccion del archivo en pdf, pero me marca error, al parecer cuando le doy una URL en drive no me deja extraerlos, aún cuando se le otorgan los permisos.

## CONCLUSIONES

- Es necesario verificar muy bien la entrada del usuario para que no haya confusion en la IA de acuerdo a lo que se le pide realizar.
- Necesito tener varias aplicaciones instaladas en mi computadora para poder generar archivos pdf, no pude instalar el latex o IA-writer (lo que me proponía la IA del CHATGPT), pero aun asi estoy analizando qué aplicaciones serian las mas básicas y/o importantes para poder realizar el trabajo que necesito para entregar las practicas.
- La IA, aún cuando le especifico que estoy trabajando en Google Colab hay ocasiones que no me proporciona todos los datos necesarios para poder correr bien el codigo en google colab
- Aún con todos los inconvenientes que tuve en esta practica, sigo pensando que CHATGPT me apoya en varias circunstancias que necesito, aunque hasta el momento no he necesitado utilizar OCR, es interesante saber que se pueden obtener textos desde Google Colab.
- El ChatGPT, volviendo a lo que comenté en la practica 101, no sé mucho de parogramación en Phyton, lo cual me hace disminuida en verificar si los codigos que me proporciona la IA estan bien realizados, pero al parecer me ha ayudado mucho, aunque necesito aprender un poco más de ello.
- Para el punto numero 2, es importante mencionar que soy ago renuente a instalar aplicaciones en mi computadora sin antes verificar si me va a ser muy útil más adelante.