INSTITUTO POLITÉCNICO NACIONAL
UNIDAD PROFESIONAL
INTERDISCIPLINARIA EN
INGENIERÍA Y TECNOLOGÍAS
AVANZADAS

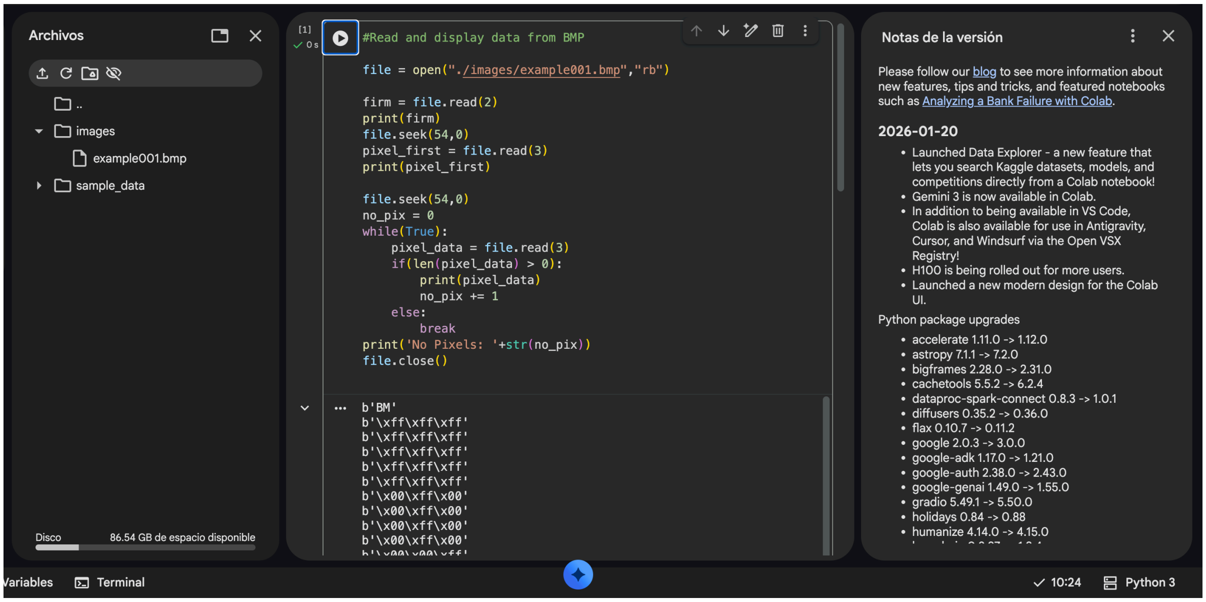
MULTIMEDIA

PRÁCTICA 3
“Filtro BMP”

Alumno: De la Rosa Valdes Itztli Fernanda

Boleta: 2024640062

Abriendo el archivo “example001.bmp” dentro de Google Colab.



```
#Read and display data from BMP

file = open("./images/example001.bmp","rb")

firm = file.read(2)
print(firm)
file.seek(54,0)
pixel_first = file.read(3)
print(pixel_first)

file.seek(54,0)
no_pix = 0
while(True):
    pixel_data = file.read(3)
    if(len(pixel_data) > 0):
        print(pixel_data)
        no_pix += 1
    else:
        break
print('No Pixels: '+str(no_pix))
file.close()
```

```
b'BM'
b'\xff\xff\xff'
b'\xff\xff\xff'
b'\xff\xff\xff'
b'\xff\xff\xff'
b'\xff\xff\xff'
b'\x00\xff\x00'
b'\x00\xff\x00'
b'\x00\xff\x00'
b'\x00\xff\x00'
b'\x00\x00\xff'
```

Notas de la versión

Please follow our blog to see more information about new features, tips and tricks, and featured notebooks such as Analyzing a Bank Failure with Colab.

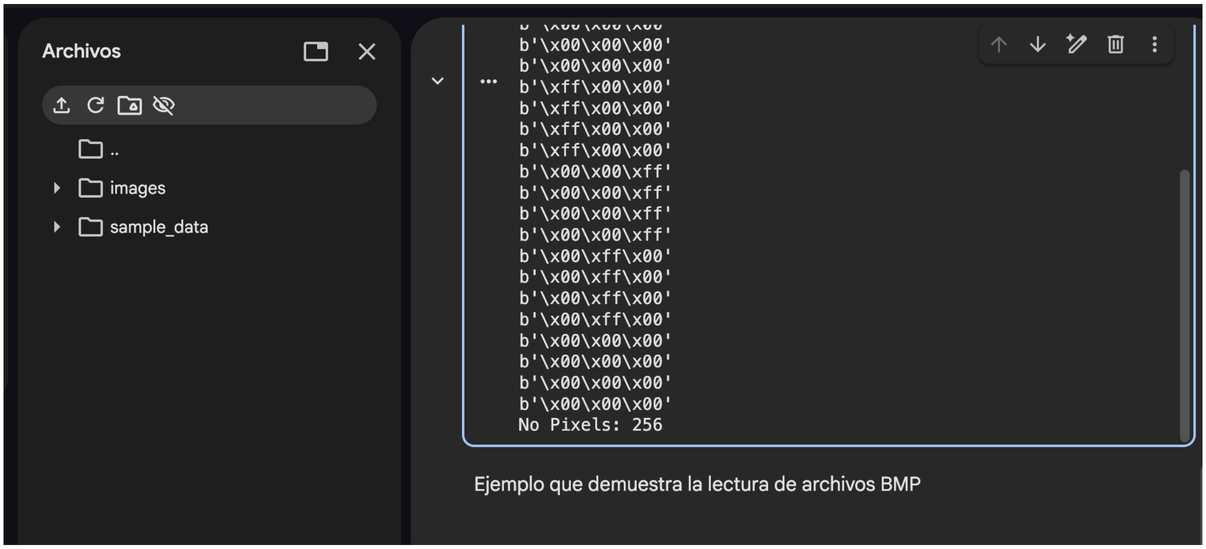
2026-01-20

- Launched Data Explorer - a new feature that lets you search Kaggle datasets, models, and competitions directly from a Colab notebook!
- Gemini 3 is now available in Colab.
- In addition to being available in VS Code, Colab is also available for use in Antigravity, Cursor, and Windsurf via the Open VSX Registry!
- H100 is being rolled out for more users.
- Launched a new modern design for the Colab UI.

Python package upgrades

- accelerate 1.11.0 -> 1.12.0
- astropy 7.1.1 -> 7.2.0
- bigframes 2.28.0 -> 2.31.0
- cachetools 5.5.2 -> 6.2.4
- dataproc-spark-connect 0.8.3 -> 1.0.1
- diffusers 0.35.2 -> 0.36.0
- flax 0.10.7 -> 0.11.2
- google 2.0.3 -> 3.0.0
- google-adk 1.17.0 -> 1.21.0
- google-auth 2.38.0 -> 2.43.0
- google-genai 1.49.0 -> 1.55.0
- gradio 5.49.1 -> 5.50.0
- holidays 0.84 -> 0.88
- humanize 4.14.0 -> 4.15.0

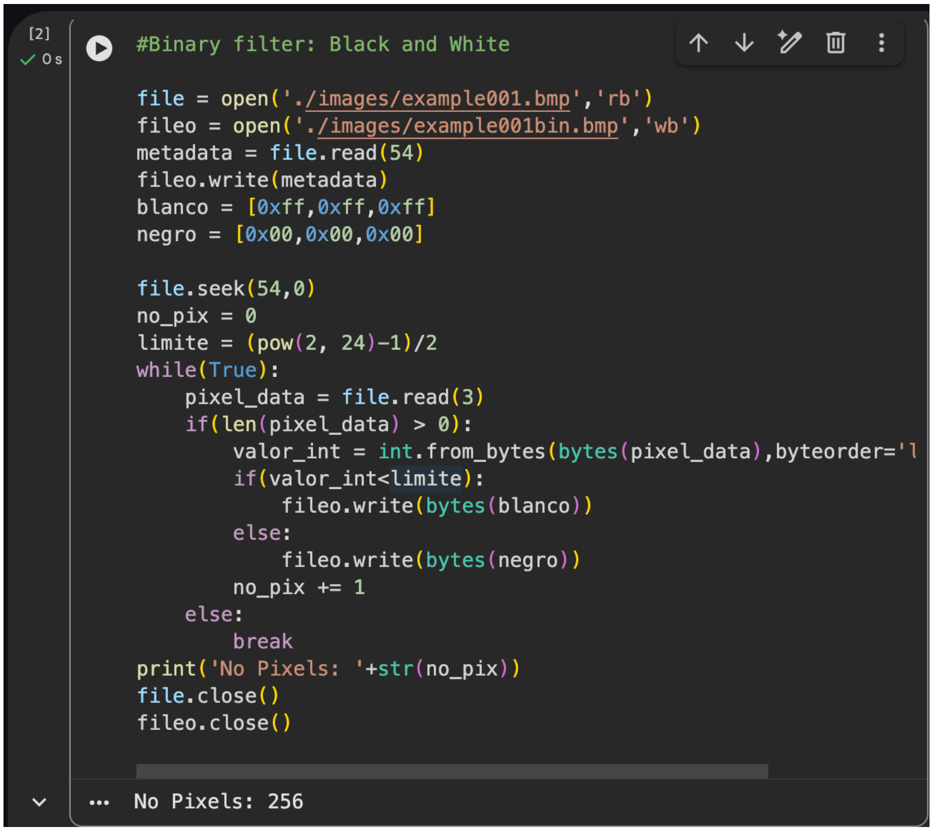
Número de píxeles.



```
b'\x00\x00\x00'
b'\x00\x00\x00'
b'\x00\x00\x00'
b'\xff\x00\x00'
b'\xff\x00\x00'
b'\xff\x00\x00'
b'\xff\x00\x00'
b'\x00\x00\xff'
b'\x00\x00\xff'
b'\x00\x00\xff'
b'\x00\x00\xff'
b'\x00\xff\x00'
b'\x00\xff\x00'
b'\x00\xff\x00'
b'\x00\xff\x00'
b'\x00\x00\x00'
b'\x00\x00\x00'
b'\x00\x00\x00'
b'\x00\x00\x00'
No Pixels: 256
```

Ejemplo que demuestra la lectura de archivos BMP

Código para el filtro blanco y negro.

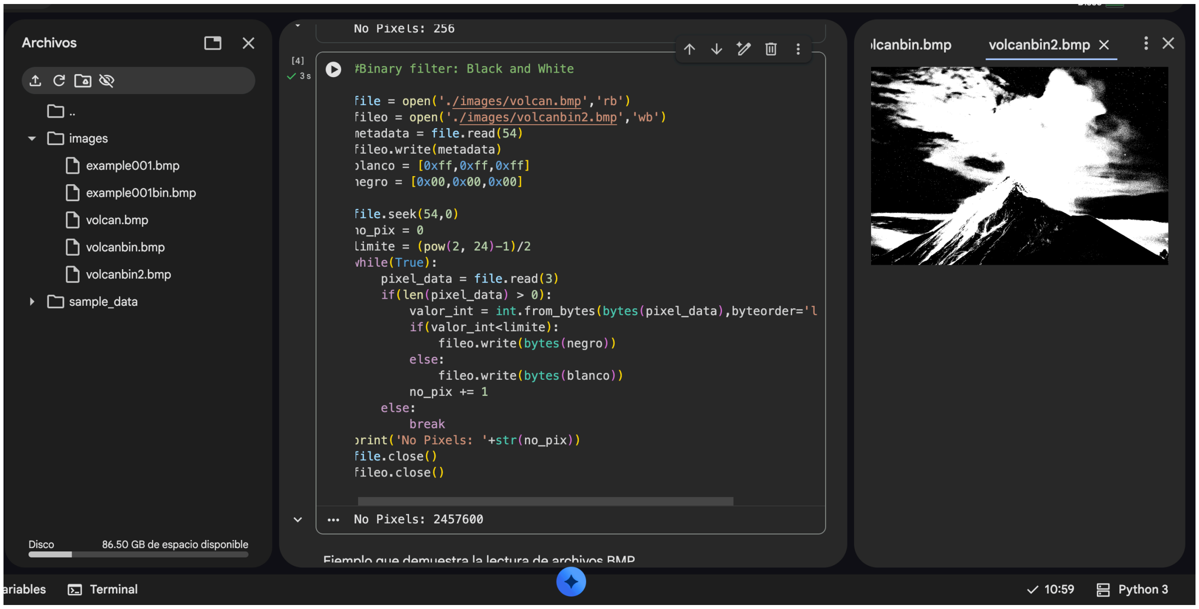


[2]
✓ 0 s

```
#Binary filter: Black and White

file = open('./images/example001.bmp','rb')
fileo = open('./images/example001bin.bmp','wb')
metadata = file.read(54)
fileo.write(metadata)
blanco = [0xff,0xff,0xff]
negro = [0x00,0x00,0x00]

file.seek(54,0)
no_pix = 0
limite = (pow(2, 24)-1)/2
while(True):
    pixel_data = file.read(3)
    if(len(pixel_data) > 0):
        valor_int = int.from_bytes(bytes(pixel_data),byteorder='l
        if(valor_int<limite):
            fileo.write(bytes(blanco))
        else:
            fileo.write(bytes(negro))
        no_pix += 1
    else:
        break
print('No Pixels: '+str(no_pix))
file.close()
fileo.close()
```

```
No Pixels: 256
```

Prueba de código con imagen “volcan”.



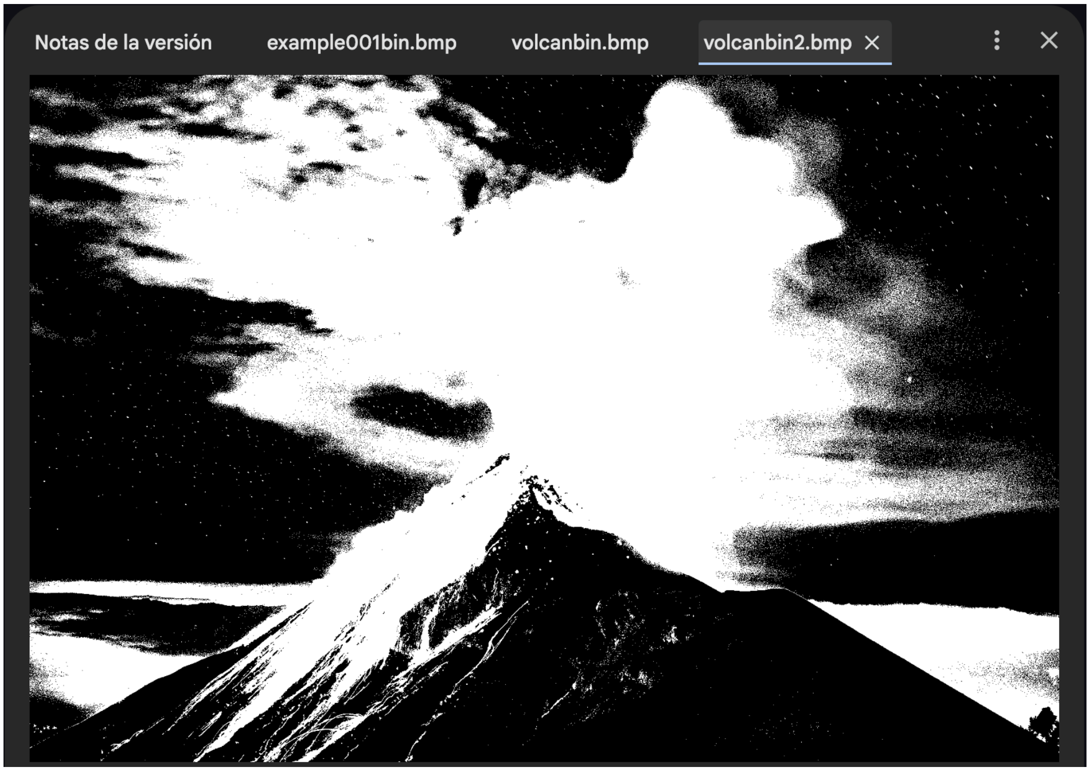
```
No Pixels: 256
```

[4]
✓ 3 s

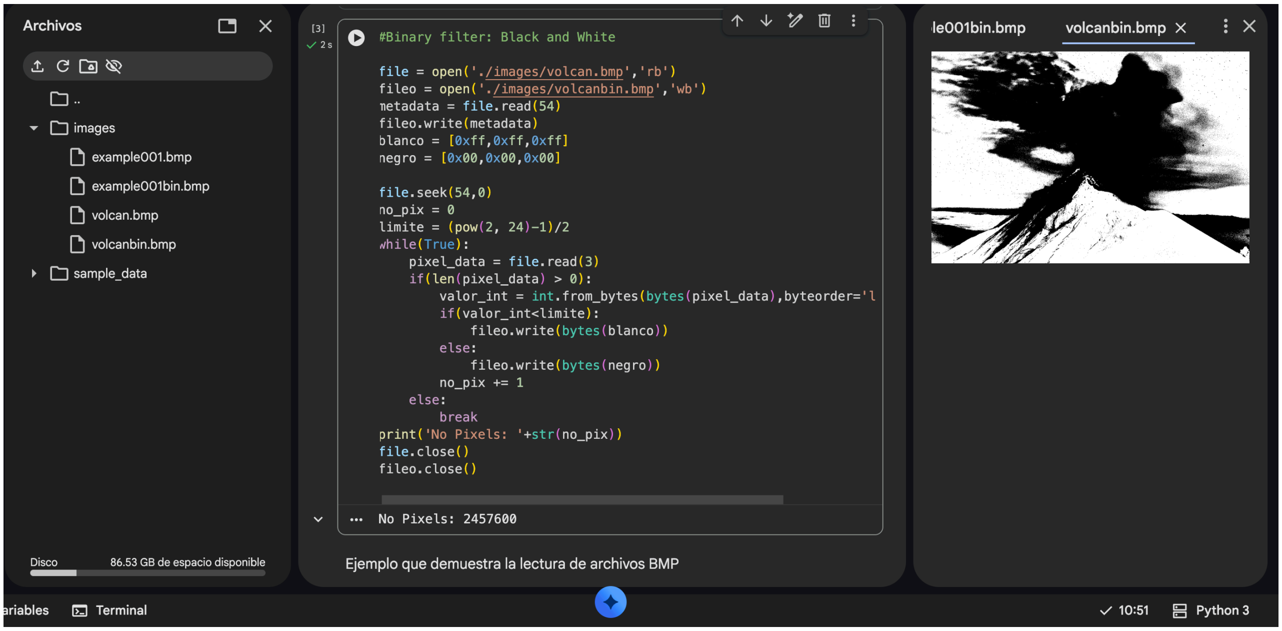
```
#Binary filter: Black and White

file = open('./images/volcan.bmp','rb')
fileo = open('./images/volcanbin2.bmp','wb')
ietadata = file.read(54)
fileo.write(metadata)
lanco = [0xff,0xff,0xff]
iegro = [0x00,0x00,0x00]

file.seek(54,0)
io_pix = 0
limite = (pow(2, 24)-1)/2
vhile(True):
    pixel_data = file.read(3)
    if(len(pixel_data) > 0):
        valor_int = int.from_bytes(bytes(pixel_data),byteorder='l
        if(valor_int<limite):
            fileo.write(bytes(negro))
        else:
            fileo.write(bytes(blanco))
        no_pix += 1
    else:
        break
rint('No Pixels: '+str(no_pix))
file.close()
fileo.close()
```

```
No Pixels: 2457600
```

Ejemplo que demuestra la lectura de archivos BMP



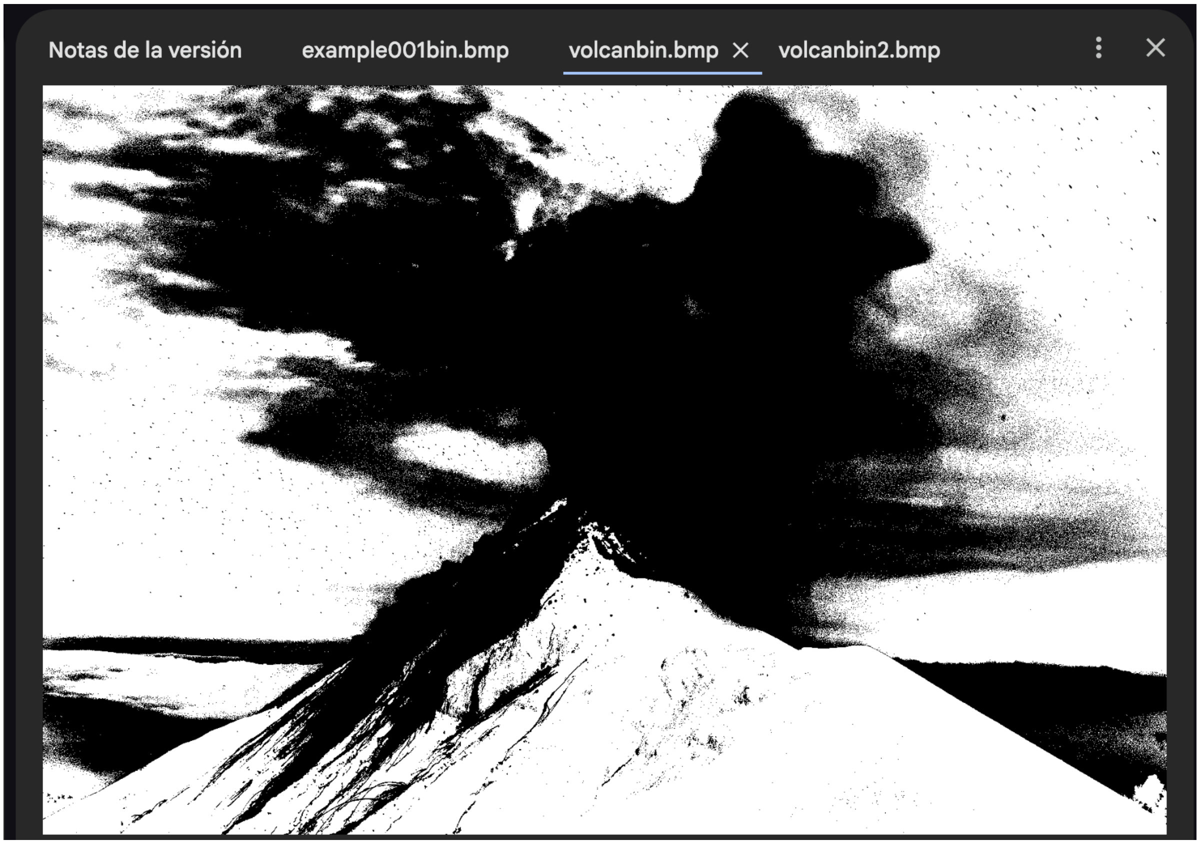
Inversión de filtro.



```
#Binary filter: Black and White

file = open('./images/volcan.bmp','rb')
fileo = open('./images/volcanbin.bmp','wb')
metadata = file.read(54)
fileo.write(metadata)
blanco = [0xff,0xff,0xff]
negro = [0x00,0x00,0x00]

file.seek(54,0)
no_pix = 0
limite = (pow(2, 24)-1)/2
while(True):
    pixel_data = file.read(3)
    if(len(pixel_data) > 0):
        valor_int = int.from_bytes(bytes(pixel_data),byteorder='l
        if(valor_int<limite):
            fileo.write(bytes(blanco))
        else:
            fileo.write(bytes(negro))
        no_pix += 1
    else:
        break
print('No Pixels: '+str(no_pix))
file.close()
fileo.close()
```

```
No Pixels: 2457600
```

Ejemplo que demuestra la lectura de archivos BMP


Notas de la versión
example001bin.bmp
volcanbin.bmp
volcanbin2.bmp