



INSTITUTO POLITECNICO NACIONAL



UNIDAD PROFESIONAL INTERDISCIPLINARIA EN INGENIERIA
Y TECNOLOGIAS AVANZADAS - IPN

MATERIA

Multimedia

PROFESOR

Noé Sierra Romero

ALUMNO

Pelcastre Gaitan Emiliano

TEMA

Formato BMP Práctica 2

Grupo: 3TM2

Parte 1

The screenshot shows a Jupyter Notebook interface with the following details:

- Header:** LecturaBMP.ipynb
- Menu Bar:** Archivo, Editar, Ver, Insertar, Entorno de ejecución, Herramientas, Ayuda.
- Toolbar:** Compartir, RAM, Disco.
- File Explorer:** Shows a local directory structure with files like 'example001.bmp' and 'sample_data'.
- Code Cell:** Displays the Python code used to read the BMP file, which outputs binary data representing the image pixels.
- Output Cell:** Shows the output of the code as a large block of binary data (hex values) followed by the text "No Pixels: 256".
- Bottom Status Bar:** Shows disk usage (86.54 GB disponibles), a progress bar for a download, and the current time (10:22 a.m.).

Parte 2

LecturaBMP.ipynb

Archivo Editar Ver Insertar Entorno de ejecución Herramientas Ayuda

Comandos Código Texto Ejecutar todo

RAM Disco

Archivos

- ..
- images
 - example001.bmp
 - example001bin.bmp
- sample_data

#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='little')
        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

Crea un filtro binario para el archivo BMP



Parte 3

The screenshot shows a Jupyter Notebook interface with the following components:

- File Explorer (Archivos):** Shows a directory structure with files: example001.bmp, example001bin.bmp, volcán.bmp, volcánbin.bmp, and volcánbin2.bmp.
- Code Cell:** Displays Python code for a "Binary filter: Black and White". The code reads a BMP file, processes it to create a binary version, and then writes it back. It includes comments explaining the process and prints the number of pixels.

```
#Binary filter: Black and White
file = open('./images/volcan.bmp','rb')
fileo = open('./images/volcánbin2.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)
    valor_int = int.from_bytes(bytes(pixel_data),byteorder='little')
    if(valor_int>limite):
        fileo.write(bytes(blanco))
    else:
        fileo.write(bytes(negro))
    no_pix += 1
    if(no_pix == limite):
        break
print('No Pixels: '+str(no_pix))
file.close()
fileo.close()
```

Crea un filtro binario para el archivo BMP

- Output:** Two images are shown: "volcánbin.bmp" and "volcánbin2.bmp". Both images are black and white binary versions of the original volcano image, showing high contrast between the dark lava field and the bright sky.
- Bottom Panel:** Shows the notebook variables and terminal, and the status bar indicates the date and time (10:50 a.m.) and Python 3 environment.

Parte 4

The screenshot shows a Jupyter Notebook interface with the following components:

- Archivos:** A sidebar showing a directory structure with files like `example001.bmp`, `example01bin.bmp`, `volcan.bmp`, `volcanbin.bmp`, and `volcanbin2.bmp`.
- Código:** A code editor containing Python code for a binary filter. The code reads a BMP file (`volcan.bmp`), writes its metadata, and then iterates through the pixels. For each pixel, it checks if the value is greater than zero. If so, it writes a byte of `0xff` (black). Otherwise, it writes a byte of `0x00` (white). It also prints the total number of pixels.
- Output:** Two image preview windows titled "icanbin.bmp" and "volcanbin2.bmp". Both images show a high-contrast, black-and-white version of a volcano scene, where the sky and clouds are mostly black and the land and volcano are white.
- Variables:** A tab labeled "Variables" at the bottom left.
- Terminal:** A tab labeled "Terminal" at the bottom left.
- System Status:** At the bottom right, it shows "10:50 a.m.", "Python 3", and disk usage information ("Disco" with 86.50 GB available).

