

[Actividad extracurricular 2b] Módulo logging

- Mateo Molina Moises Estrella

Link Repositorio: https://github.com/R3LM15/MetodosNumericos_MateoMolina

Objetivo

Conocer y utilizar el módulo de logging en Python.

Indicaciones

- Investigue sobre el módulo de logging.
- Cree un ejemplo que: realice logging en consola, en un archivo, en un mensaje de Telegram.
- Modifique el logging para mostrar el nombre del archivo, incluir la fecha y la hora, cambiar de color, etc.

¿Qué es?

El módulo logging en Python es una herramienta estándar que permite registrar mensajes (logs) desde los programas. Es muy útil para:

- Depuración (debugging) de errores.
- Monitoreo del comportamiento del programa.
- Guardar eventos importantes en archivos o mostrarlos por consola.

Te permite emitir mensajes de diferentes niveles de gravedad, como por ejemplo:

Nivel	Uso típico
DEBUG	Detalles internos para depuración.
INFO	Información general de ejecución.
WARNING	Advertencias que no detienen el programa.
ERROR	Errores que afectan una parte del programa.
CRITICAL	Errores graves que detienen el programa.

Ejemplos

1. Logging en consola

```
In [10]: import logging

# Crear Logger
logger = logging.getLogger('consola_logger')
logger.setLevel(logging.DEBUG)

# Crear handler para consola
console_handler = logging.StreamHandler()
console_handler.setLevel(logging.INFO)

# Formato del mensaje
formatter = logging.Formatter('%(asctime)s - %(levelname)s - %(message)s')
console_handler.setFormatter(formatter)

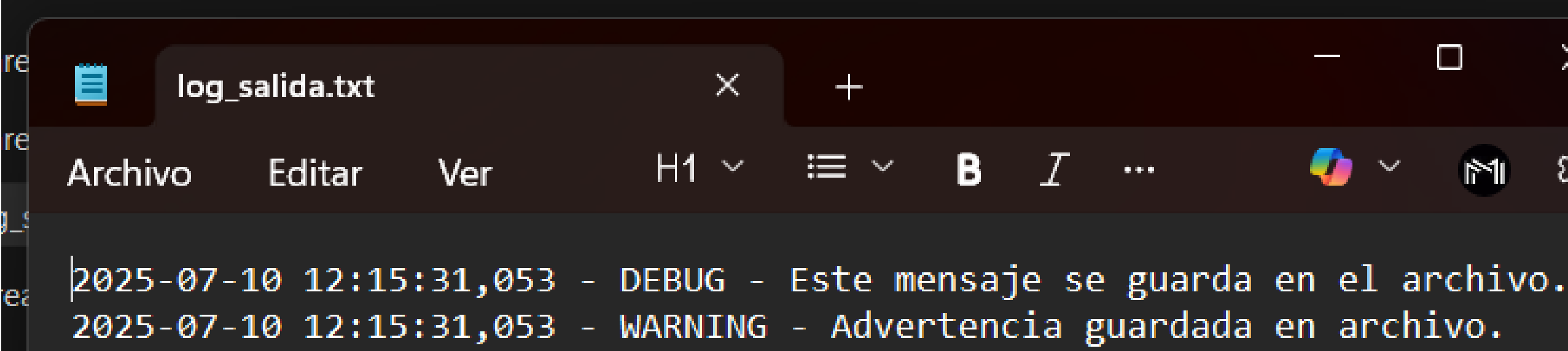
# Agregar handler al Logger
logger.addHandler(console_handler)

# Mensajes de prueba
logger.info("Este mensaje aparece en la consola.")
logger.error("Este error también se muestra en consola.")
```

```
2025-07-10 13:03:24,012 - INFO - Este mensaje aparece en la consola.
2025-07-10 13:03:24,012 - INFO - Este mensaje aparece en la consola.
2025-07-10 13:03:24,017 - ERROR - Este error también se muestra en consola.
2025-07-10 13:03:24,017 - ERROR - Este error también se muestra en consola.
2025-07-10 13:03:24,012 - INFO - Este mensaje aparece en la consola.
2025-07-10 13:03:24,017 - ERROR - Este error también se muestra en consola.
2025-07-10 13:03:24,017 - ERROR - Este error también se muestra en consola.
```

2. En un archivo

```
In [ ]: 2. **En un archivo**
```



3. Mensaje de Telegram

```
In [15]: import logging
import requests

# Handler personalizado para Telegram
class TelegramHandler(logging.Handler):
    def __init__(self, bot_token, chat_id):
        super().__init__(level=logging.CRITICAL) # Solo envia mensajes criticos
        self.bot_token = bot_token
        self.chat_id = chat_id

    def emit(self, record):
        log_entry = self.format(record)
        url = f"https://api.telegram.org/bot{self.bot_token}/sendMessage"
        data = {"chat_id": self.chat_id, "text": f"[LOG CRÍTICO] {log_entry}"}
        try:
            requests.post(url, data=data)
        except Exception as e:
            print(f"Error al enviar mensaje a Telegram: {e}")

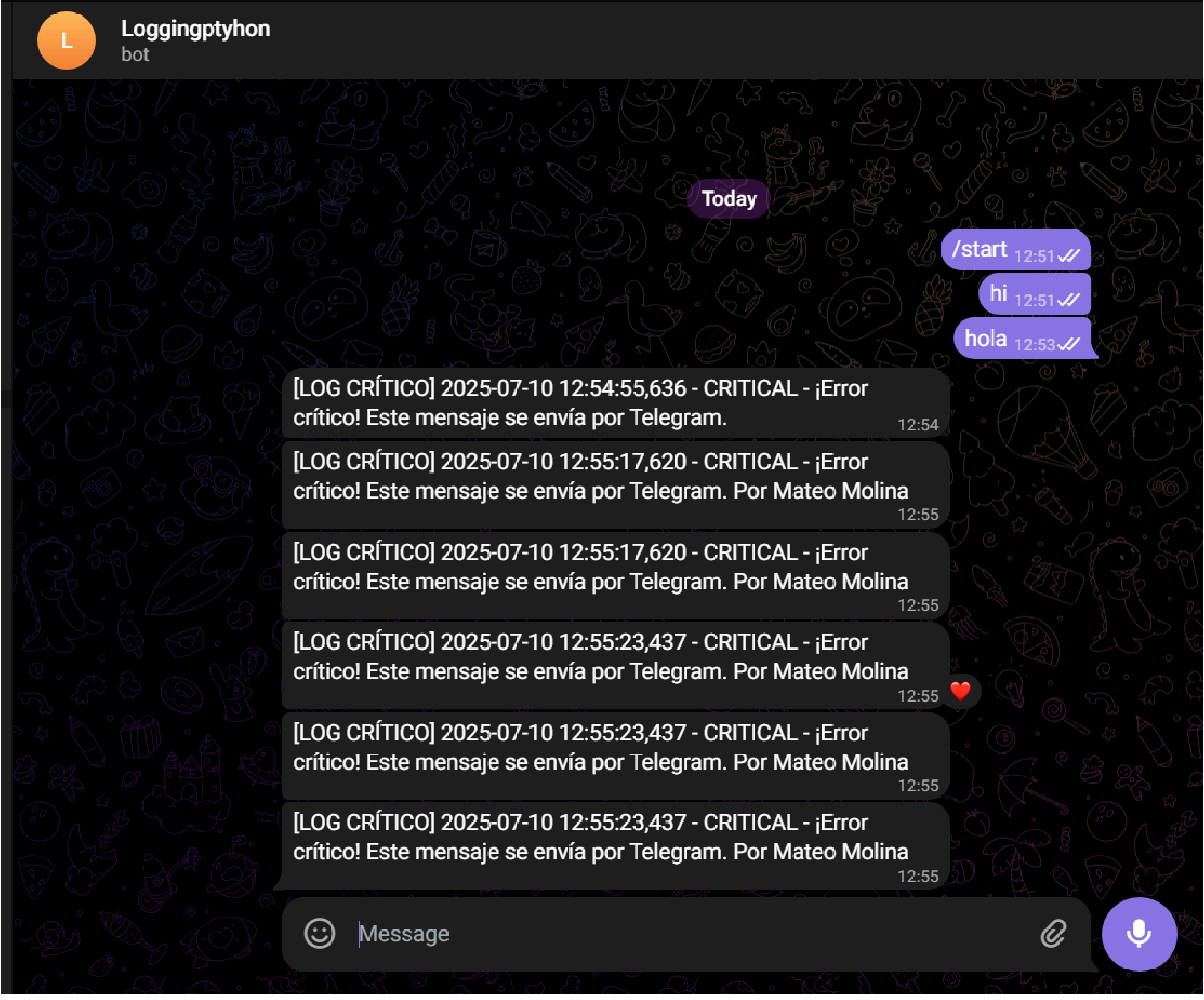
# Configura tu bot
BOT_TOKEN = "8131292212:AAEiMUsaetA3b4Ji695KmEKXUk5-fSehEw0"
CHAT_ID = "6789942138"

# Crear Logger
logger = logging.getLogger('telegram_logger')
logger.setLevel(logging.DEBUG)

# Crear y configurar handler
telegram_handler = TelegramHandler(BOT_TOKEN, CHAT_ID)
formatter = logging.Formatter('%(asctime)s - %(levelname)s - %(message)s')
telegram_handler.setFormatter(formatter)

# Agregar handler
logger.addHandler(telegram_handler)

# Enviar mensaje critico
logger.critical("¡Error critico! Este mensaje se envía por Telegram. Por Mateo Molina ")
```



Personalización

```
In [20]: import logging
import sys
from colorama import Fore, Style, init

# Inicializa colorama para colores en consola Windows/Linux/Mac
init(autoreset=True)

# Crea Logger
logger = logging.getLogger("logger_coloreado")
logger.setLevel(logging.DEBUG)

# Crea handler para consola (stdout)
console_handler = logging.StreamHandler(sys.stdout)
console_handler.setLevel(logging.DEBUG)

# Formato base con fecha, nivel, archivo y línea
log_format = "%(asctime)s - %(levelname)s - %(filename)s:%(lineno)d - %(message)s"

# Clase para formatear con colores según nivel
class ColorFormatter(logging.Formatter):
    COLORS = {
        logging.DEBUG: Fore.CYAN,
        logging.INFO: Fore.GREEN,
        logging.WARNING: Fore.YELLOW,
        logging.ERROR: Fore.RED,
        logging.CRITICAL: Fore.MAGENTA + Style.BRIGHT
    }

    def format(self, record):
        msg = super().format(record)
        color = self.COLORS.get(record.levelno, Fore.WHITE)
        return f"{color}{msg}{Style.RESET_ALL}"

# Asignar formatter al handler
formatter = ColorFormatter(log_format, datefmt="%Y-%m-%d %H:%M:%S")
console_handler.setFormatter(formatter)

# Agregar handler al Logger
logger.addHandler(console_handler)

# Ejemplos de Logs
logger.debug("Mensaje DEBUG")
logger.info("Mensaje INFO")
logger.warning("Mensaje WARNING")
logger.error("Mensaje ERROR")
logger.critical("Mensaje CRITICAL")
```

```
2025-07-10 13:11:52 - DEBUG - 2246431092.py:42 - Mensaje DEBUG
2025-07-10 13:11:52 - DEBUG - 2246431092.py:42 - Mensaje DEBUG
2025-07-10 13:11:52 - INFO - 2246431092.py:42 - Mensaje INFO
2025-07-10 13:11:52 - INFO - 2246431092.py:43 - Mensaje INFO
2025-07-10 13:11:52 - WARNING - 2246431092.py:44 - Mensaje WARNING
2025-07-10 13:11:52 - WARNING - 2246431092.py:44 - Mensaje WARNING
2025-07-10 13:11:52 - ERROR - 2246431092.py:45 - Mensaje ERROR
2025-07-10 13:11:52 - ERROR - 2246431092.py:45 - Mensaje ERROR
2025-07-10 13:11:52 - CRITICAL - 2246431092.py:46 - Mensaje CRITICAL
2025-07-10 13:11:52 - CRITICAL - 2246431092.py:46 - Mensaje CRITICAL
```

```
PS C:\Users\Usuario\Documents\MetodosNumericos_MateoMolina> & C:/Users/Usuario/AppData/Local/Microsoft/WindowsApps/python3.11.exe "c:/Users/Usuario/
Documents/MetodosNumericos_MateoMolina/Actividades Extra/Logging colores.py"
2025-07-10 13:10:22 - DEBUG - Logging colores.py:42 - Mensaje DEBUG
2025-07-10 13:10:22 - INFO - Logging colores.py:43 - Mensaje INFO
2025-07-10 13:10:22 - WARNING - Logging colores.py:44 - Mensaje WARNING
2025-07-10 13:10:22 - ERROR - Logging colores.py:45 - Mensaje ERROR
2025-07-10 13:10:22 - CRITICAL - Logging colores.py:46 - Mensaje CRITICAL
PS C:\Users\Usuario\Documents\MetodosNumericos_MateoMolina>
```