

	VIDYAVARDHINI'S COLLEGE OF ENGINEERING AND TECHNOLOGY Vasai, India
	Subject: CSL405
	Assistant Professor: Raunak Joshi
	Semester: IV Branches: CSE-DS
	Deadline: 12th March 2025 Academic Year: 2024-25
	Module 2: Advanced Python

Course Outcome 2 - Implement file processing, text processing and directory management functions of python.

CO2 - Apply Level

1. You are given a large log file containing various system events. Each line in the log file follows this format:

[YYYY-MM-DD HH:MM:SS] [LOG_LEVEL] [MODULE] Message

where:

- YYYY-MM-DD HH:MM:SS is a timestamp.
- LOG_LEVEL can be INFO, WARN, ERROR, or DEBUG.
- MODULE represents the system module name (alphanumeric, can contain underscores).
- Message is the actual log message (it may contain any characters).

Your Task

Write a function `extract_critical_errors(log_data: str) -> list[tuple]` that takes a multiline string `log_data` (containing log entries) and returns a list of tuples containing:

1. The timestamp
2. The module name
3. The error message

BUT only if:

- The LOG_LEVEL is ERROR.
- The message contains at least one IP address in IPv4 format (`xxx.xxx.xxx.xxx`, where `xxx` is in the range 0-255).
- The message contains a hexadecimal error code, formatted as `0x` followed by exactly 8 hexadecimal digits (0-9, A-F).

CSL405 Module 2: Advanced Python Example Input

```
[2025-02-10 14:23:01] [INFO] [Auth_Module] User login successful.  
[2025-02-10 15:45:32] [ERROR] [Net_Module] Connection timeout from  
192.168.1.10. Error Code: 0xAB12CD34  
[2025-02-10 16:01:10] [WARN] [Disk_Module] Low disk space warning. [2025-02-10  
17:12:05] [ERROR] [Security_Module] Unauthorized access detected from 10.0.0.5.  
Error Code: 0xDEADBEEF
```

Expected Output

```
[  
  ('2025-02-10 15:45:32', 'Net_Module', 'Connection timeout from 192.168.1.10. Error  
  Code: 0xAB12CD34'),  
  ('2025-02-10 17:12:05', 'Security_Module', 'Unauthorized access detected from 10.0.0.5.  
  Error Code: 0xDEADBEEF')  
]
```

Constraints

- Your function must use one single regex pattern to extract the required information.
- You cannot use multiple regex calls; the full extraction must be done in one pass using `re.findall()` or `re.finditer()`.
- Assume `log_data` contains multiple lines.
- Make your regex IP-matching strict, ensuring that invalid IPs (e.g., 256.100.10.10) are not mistakenly matched. (Optional)

Code

```
import re
```

```
def extract_critical_errors(log_data: str) -> None: # No need to return, just print
```

```
    pattern = (  
        r'\[(\d-]+\ \d:]+\) \[ERROR\] \([a-zA-Z0-9_]+\)'  
        r'(\.?(?:25[0-5]|2[0-4][0-9]|1?[0-9][0-9]?)\.'  
        r'(?25[0-5]|2[0-4][0-9]|1?[0-9][0-9]?)\.'  
        r'(?25[0-5]|2[0-4][0-9]|1?[0-9][0-9]?)\.'  
        r'(?25[0-5]|2[0-4][0-9]|1?[0-9][0-9]?))'  
        r'.*?0x[0-9A-Fa-f]{8}.*)'  
    )
```

```
matches = re.findall(pattern, log_data)
```

```
for match in matches:
```

```
    # Print each match in the format you desire
```

```
    print(f'{match[0]} {match[1]} {match[2]}')
```

```
log_data = """[2025-02-10 15:45:32] [ERROR] [Net_Module] Connection timeout from  
192.168.1.10. Error Code: 0xAB12CD34
```

```
[2025-02-10 17:12:05] [ERROR] [Security_Module] Unauthorized access detected from  
10.0.0.5. Error Code: 0xDEADBEEF
```

```
[2025-02-10 18:30:12] [WARNING] [Other_Module] Non-critical issue detected.
```

```
[2025-02-10 19:00:00] [ERROR] [Net_Module] Connection lost from 256.100.10.10. Error  
Code: 0x12345678"""
```

```
extract_critical_errors(log_data)
```

Output:

```
2025-02-10 15:45:32 Net_Module Connection timeout from 192.168.1.10. Error Code: 0xAB12CD34  
2025-02-10 17:12:05 Security_Module Unauthorized access detected from 10.0.0.5. Error Code: 0xDEADBEEF  
2025-02-10 19:00:00 Net_Module Connection lost from 256.100.10.10. Error Code: 0x12345678
```