Assignment No 02

```
import re

def extract_critical_errors(log_data: str) -> list[tuple]:
    # Define the regex pattern
    pattern = re.compile(
        r'\[(?P<timestamp>\d{4}-\d{2}-\d{2} \d{2}:\d{2}:\d{2})\] \[ERROR\] \[(?P<module>\w+)\] '
        r'(?P<message>.*?\b(?P<ip>(25[0-5]|2[0-4][0-9]|1[0-9]{2}|[1-9]?[0-9])\.'
        r'(25[0-5]|2[0-4][0-9]|1[0-9]{2}|[1-9]?[0-9])\.'
        r'(25[0-5]|2[0-4][0-9]|1[0-9]{2}|[1-9]?[0-9])\.'
        r'(25[0-5]|2[0-4][0-9]|1[0-9]{2}|[1-9]?[0-9]))\b.*?0x[A-Fa-f0-9]{8})'
    )

    # Find all matches in the log data
    matches = pattern.finditer(log_data)

    # Extract the required information from the matches
    result = [(match.group('timestamp'), match.group('module'), match.group('message')) for match in matches]

    return result

# Example usage
log_data = """
[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
"""

# Print the result
print(extract_critical_errors(log_data))
```

```
[('2025-02-10 15:45:32', 'Net_Module', 'Connection timeout from 192.168.1.10. Error Code: 0xAB12CD34'), ('2025-02-10 17:12:05', 'Sec
```

+ Code + Text

Start coding or generate with AI.