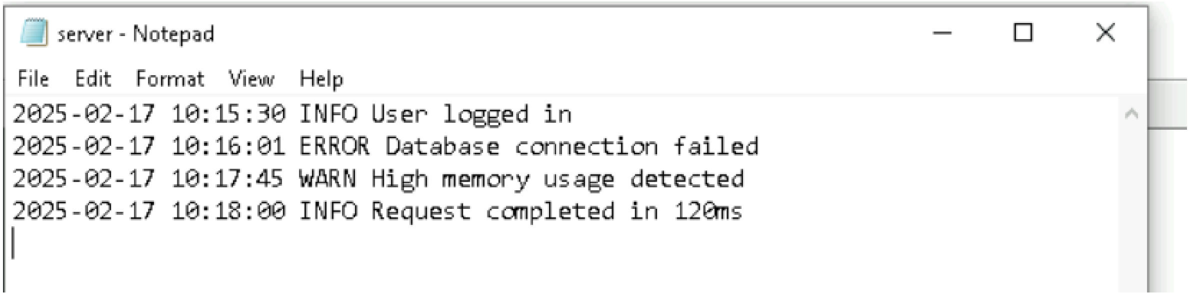
Groovy Projects

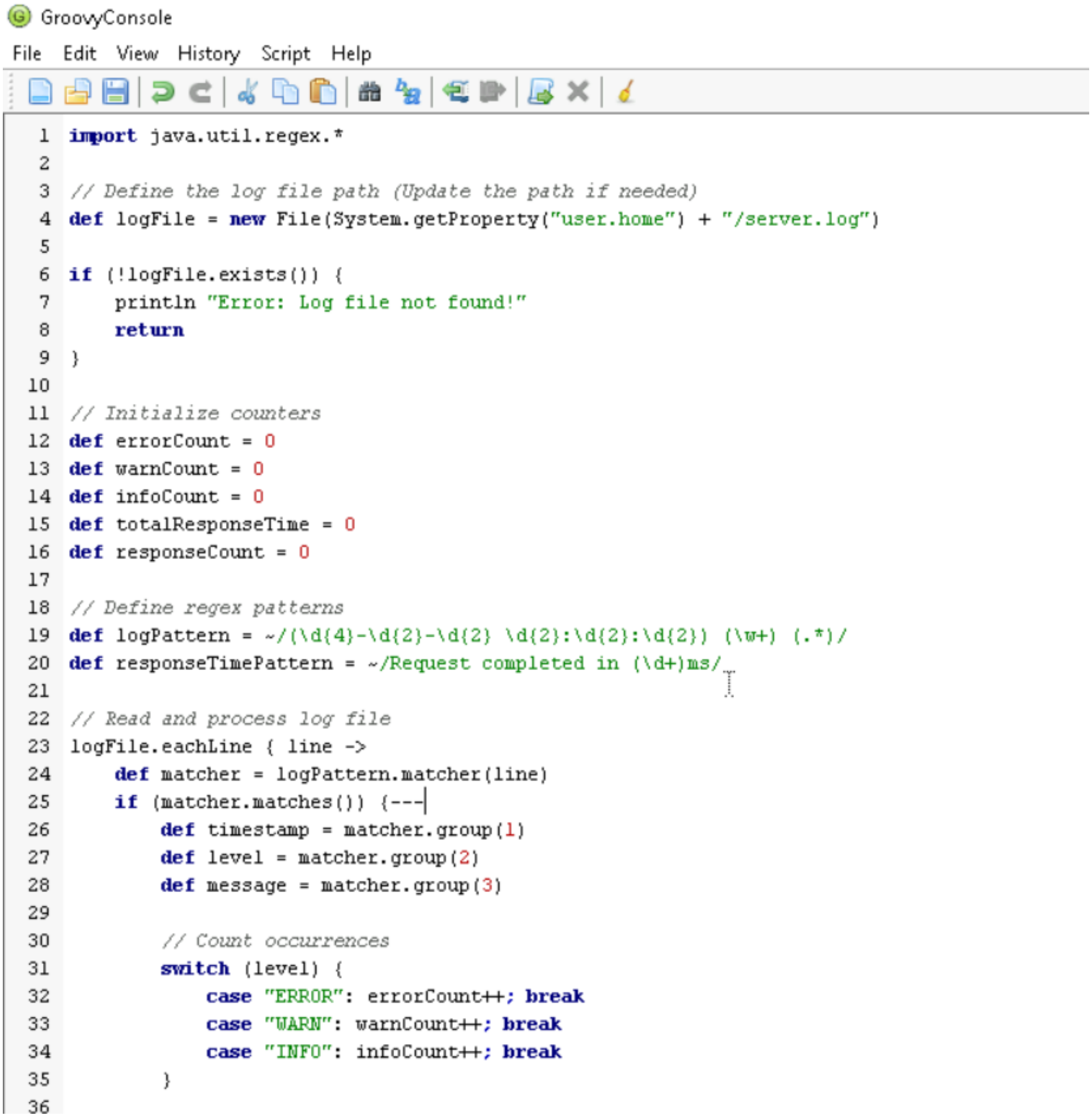
1. Groovy Script for Log File Analysis

1. **Open Nodepad and paste this info and save in the c://user/Administator as server.log text file.**



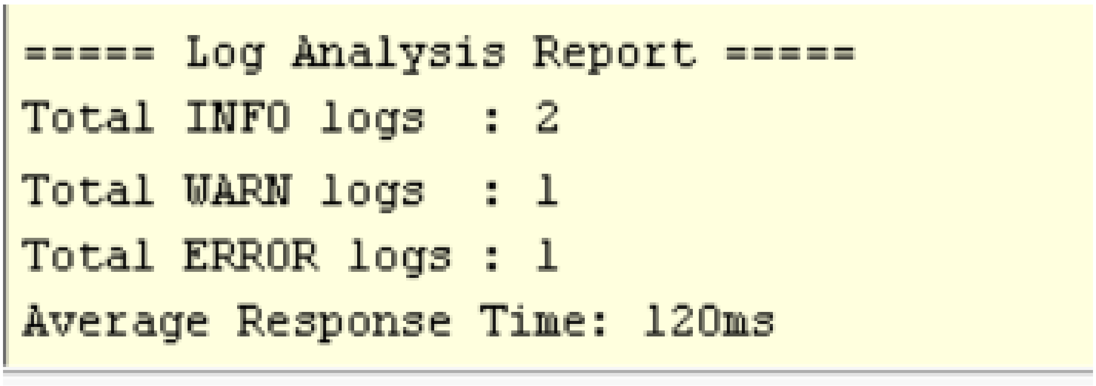
```
2025-02-17 10:15:30 INFO User logged in
2025-02-17 10:16:01 ERROR Database connection failed
2025-02-17 10:17:45 WARN High memory usage detected
2025-02-17 10:18:00 INFO Request completed in 120ms
```

2. **Right the script in the Groovy Console**



```
import java.util.regex.*

// Define the log file path (Update the path if needed)
def logFile = new File(System.getProperty("user.home") + "/server.log")

if (!logFile.exists()) {
    println "Error: Log file not found!"
    return
}

// Initialize counters
def errorCount = 0
def warnCount = 0
def infoCount = 0
def totalResponseTime = 0
def responseCount = 0

// Define regex patterns
def logPattern = ~/(\d{4}-\d{2}-\d{2} \d{2}:\d{2}:\d{2}) (\w+) (.*)/
def responseTimePattern = ~/Request completed in (\d+)ms/

// Read and process log file
logFile.eachLine { line ->
    def matcher = logPattern.matcher(line)
    if (matcher.matches()) {---
        def timestamp = matcher.group(1)
        def level = matcher.group(2)
        def message = matcher.group(3)

        // Count occurrences
        switch (level) {
            case "ERROR": errorCount++; break
            case "WARN": warnCount++; break
            case "INFO": infoCount++; break
        }

```

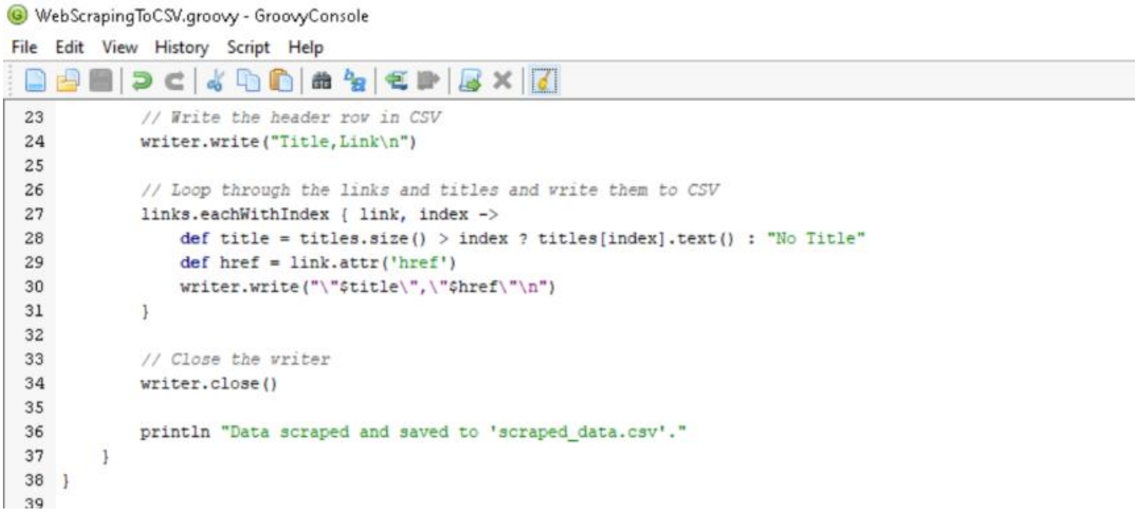
3. Output



```
===== Log Analysis Report =====
Total INFO logs  : 2
Total WARN logs  : 1
Total ERROR logs : 1
Average Response Time: 120ms
```

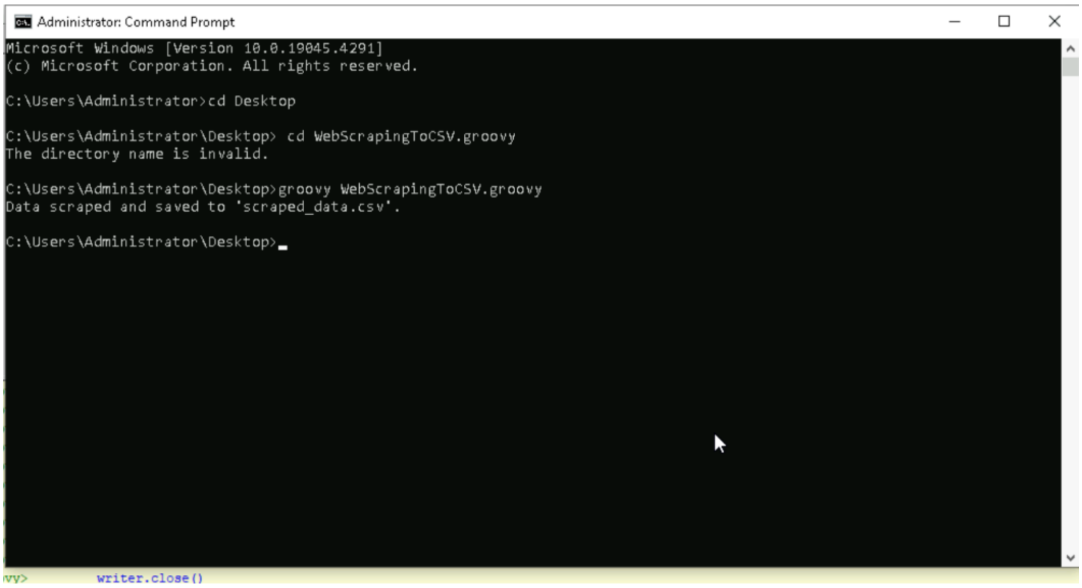
2. Web Scraping with Groovy & Jsoup

Step 1: Run the script in groovy


WebScrapingToCSV.groovy - GroovyConsole
File Edit View History Script Help

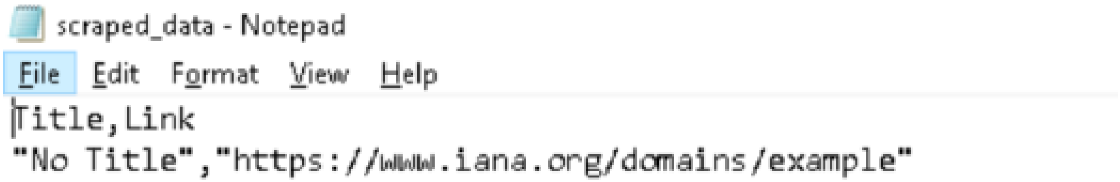
```
        // Write the header row in CSV
        writer.write("Title,Link\n")

        // Loop through the links and titles and write them to CSV
        links.eachWithIndex { link, index ->
            def title = titles.size() > index ? titles[index].text() : "No Title"
            def href = link.attr('href')
            writer.write("\"$title\",\"$href\"\n")
        }

        // Close the writer
        writer.close()

        println "Data scraped and saved to 'scraped_data.csv'."
    }
}
```

Step 2: Run the script in terminal



```
Microsoft Windows [Version 10.0.19045.4291]
(c) Microsoft Corporation. All rights reserved.

C:\Users\Administrator>cd Desktop

C:\Users\Administrator\Desktop> cd WebScrapingToCSV.groovy
The directory name is invalid.

C:\Users\Administrator\Desktop>groovy WebScrapingToCSV.groovy
Data scraped and saved to 'scraped_data.csv'.

C:\Users\Administrator\Desktop>_
```

Step 3: Output



```
Title,Link
"No Title","https://www.iana.org/domains/example"
```