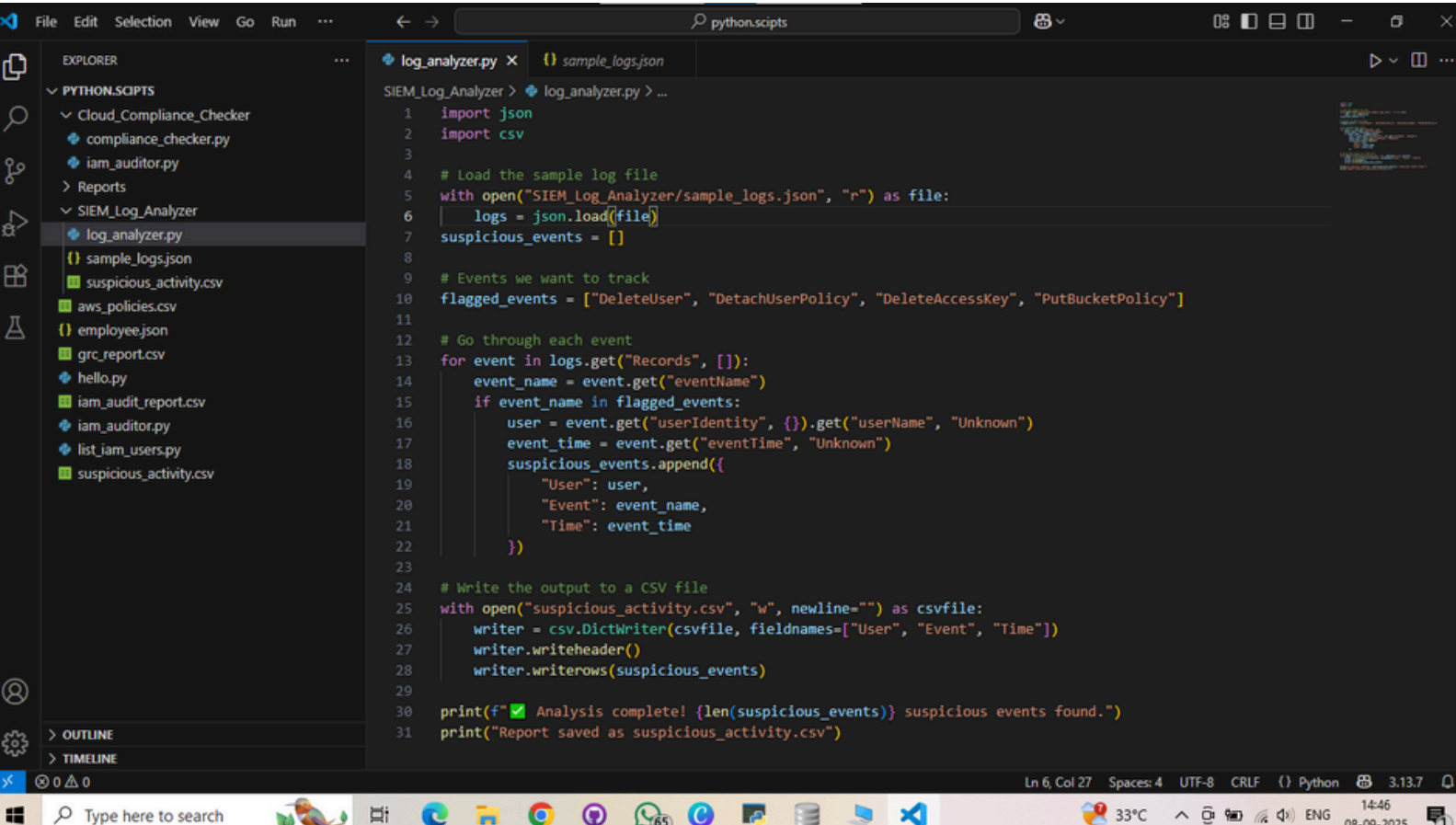


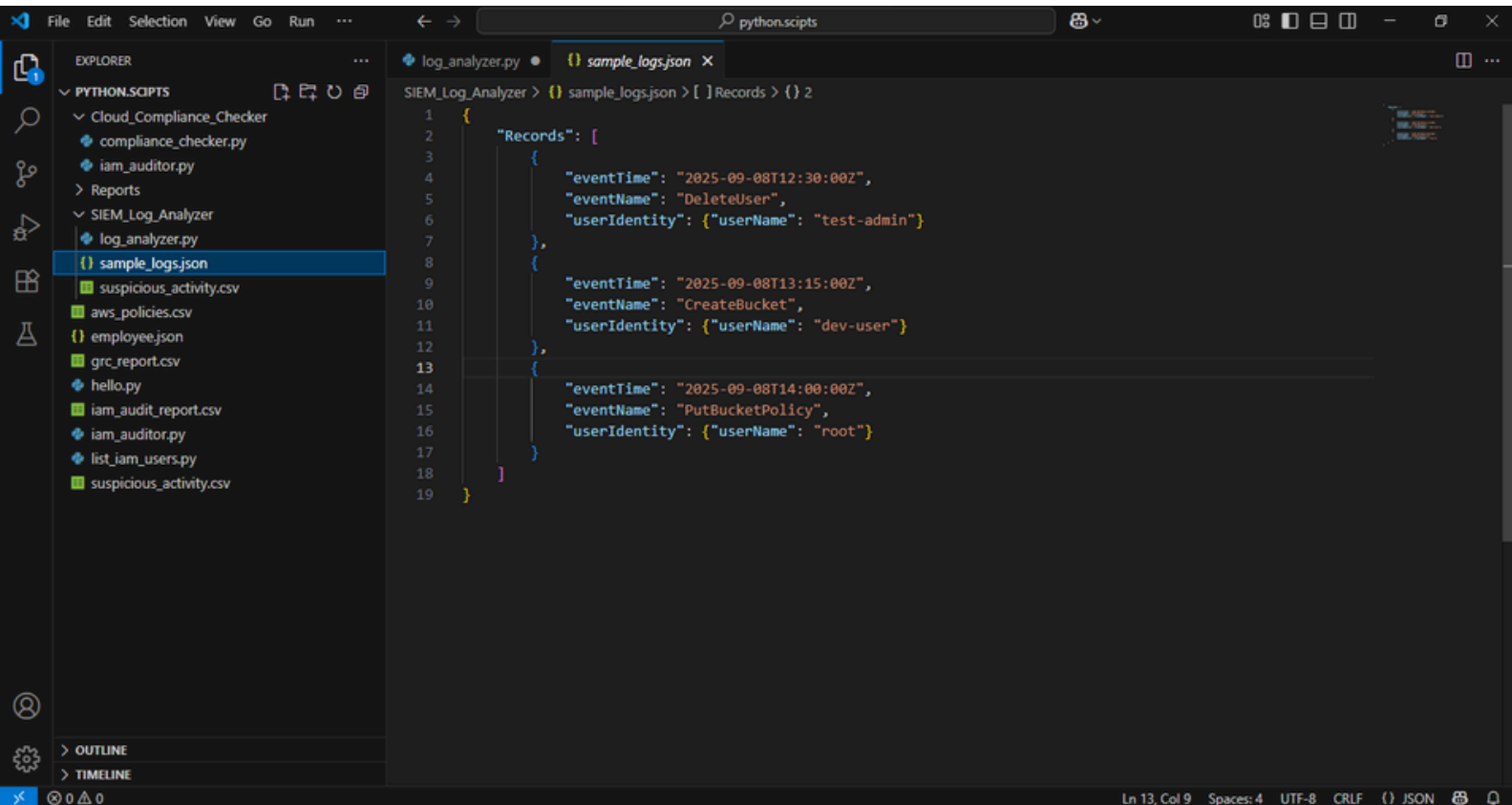
→ Use a Python script to analyze security logs .



The screenshot shows the Visual Studio Code editor with a Python script named `log_analyzer.py` open. The script is designed to analyze security logs from a JSON file (`sample_logs.json`) and extract suspicious events into a CSV file (`suspicious_activity.csv`). The script uses the `json` and `csv` modules. It loads the log file, iterates through the records, and checks for events that are flagged as suspicious (e.g., `DeleteUser`, `DetachUserPolicy`, `DeleteAccessKey`, `PutBucketPolicy`). The script then writes the extracted data to the CSV file and prints a completion message.

```
1 import json
2 import csv
3
4 # Load the sample log file
5 with open("SIEM_Log_Analyzer/sample_logs.json", "r") as file:
6     logs = json.load(file)
7     suspicious_events = []
8
9 # Events we want to track
10 flagged_events = ["DeleteUser", "DetachUserPolicy", "DeleteAccessKey", "PutBucketPolicy"]
11
12 # Go through each event
13 for event in logs.get("Records", []):
14     event_name = event.get("eventName")
15     if event_name in flagged_events:
16         user = event.get("userIdentity", {}).get("userName", "Unknown")
17         event_time = event.get("eventTime", "Unknown")
18         suspicious_events.append({
19             "User": user,
20             "Event": event_name,
21             "Time": event_time
22         })
23
24 # Write the output to a CSV file
25 with open("suspicious_activity.csv", "w", newline="") as csvfile:
26     writer = csv.DictWriter(csvfile, fieldnames=["User", "Event", "Time"])
27     writer.writeheader()
28     writer.writerows(suspicious_events)
29
30 print(f"Analysis complete! {len(suspicious_events)} suspicious events found.")
31 print("Report saved as suspicious_activity.csv")
```

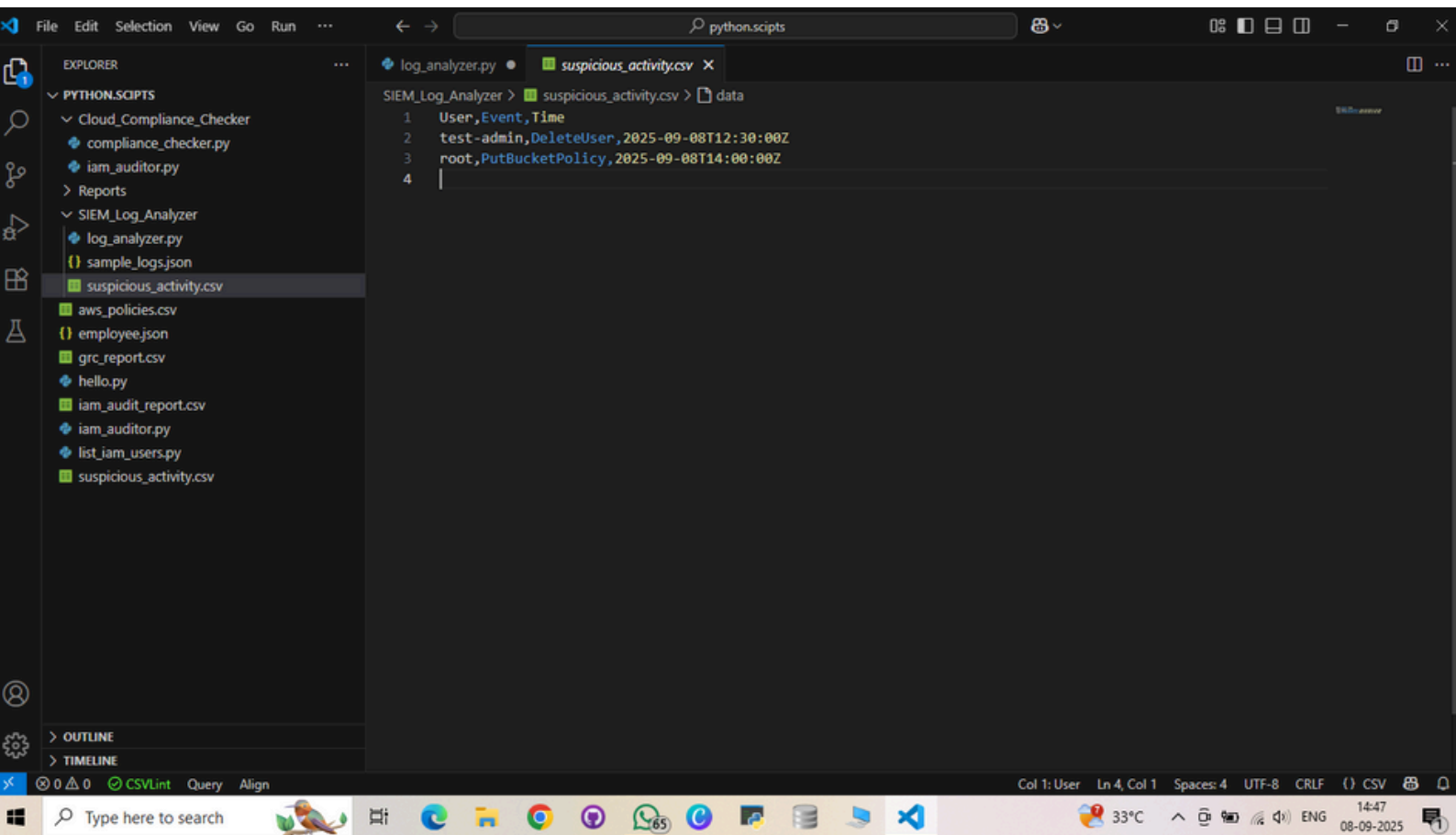
→ Created sample JSON test data in the folder, then extracted data user, event and timestamp.



The screenshot shows the Visual Studio Code editor with the `sample_logs.json` file open. The JSON file contains a list of security log records. Each record includes an `eventTime`, an `eventName`, and a `userIdentity` object containing a `userName`. The records are as follows:

```
1 {
2   "Records": [
3     {
4       "eventTime": "2025-09-08T12:30:00Z",
5       "eventName": "DeleteUser",
6       "userIdentity": {"userName": "test-admin"}
7     },
8     {
9       "eventTime": "2025-09-08T13:15:00Z",
10      "eventName": "CreateBucket",
11      "userIdentity": {"userName": "dev-user"}
12    },
13    {
14      "eventTime": "2025-09-08T14:00:00Z",
15      "eventName": "PutBucketPolicy",
16      "userIdentity": {"userName": "root"}
17    }
18  ]
19 }
```

→ Output showing User, event and timestamp as CSV format for review.



The screenshot shows a Visual Studio Code editor window with the file explorer on the left and a code editor on the right. The file explorer shows a project named 'PYTHON\_SCRIPTS' with a subdirectory 'SIEM\_Log\_Analyzer'. Inside this subdirectory, the file 'suspicious\_activity.csv' is selected. The code editor displays the contents of this CSV file, which is formatted as a table with three columns: 'User', 'Event', and 'Time'. The data is as follows:

User	Event	Time
test-admin	DeleteUser	2025-09-08T12:30:00Z
root	PutBucketPolicy	2025-09-08T14:00:00Z

The status bar at the bottom indicates the current position is 'Col 1: User', 'Ln 4, Col 1', with 'Spaces: 4', 'UTF-8', 'CRLF', and 'CSV' encoding.