Querying Imported Data with SQL

In this tutorial, you will use SQL queries to analyse the imported data.

View All Events

Open phpMyAdmin and select the:

```
security_events
```

table.

Run:

```
SELECT *
FROM security_events;
```

This displays all imported records.

Count All Events

To find the total number of events:

```
SELECT COUNT(*) AS total_events
FROM security_events;
```

Example result:

total_events
3

This query is useful for dashboards and reporting.

Find Critical Events

To display only critical events:

```
SELECT *
FROM security_events
WHERE severity = 'critical';
```

Example result:

device_id	event_type	severity
CAM-01	remote_access_attempt	critical

Count Events by Severity

Run:

```
SELECT
    severity,
    COUNT(*) AS total
FROM security_events
GROUP BY severity;
```

Example result:

severity	total
critical	1
high	1
medium	1

This is often used to build dashboard summary cards.

Find Events from a Specific Device

To display events generated by the security camera:

```
SELECT *
FROM security_events
WHERE device_id = 'CAM-01';
```

Example result:

device_id	event_type
CAM-01	motion_detected
CAM-01	remote_access_attempt

Count Events Per Device

Run:

```
SELECT
    device_id,
    COUNT(*) AS total_events
FROM security_events
GROUP BY device_id;
```

Example result:

device_id	total_events
CAM-01	2
LOCK-02	1

This identifies which devices are generating the most activity.

Find High-Risk Events

To display high and critical events:

```
SELECT *
FROM security_events
WHERE severity IN ('high', 'critical');
```

Example result:

device_id	event_type	severity
CAM-01	motion_detected	high
CAM-01	remote_access_attempt	critical

This query could be used to generate security alerts.

Sort Events by Time

To view the newest events first:

```
SELECT *
FROM security_events
ORDER BY event_timestamp DESC;
```

This is commonly used in event logs and monitoring systems.

Useful Dashboard Queries

Total Events

```
SELECT COUNT(*) AS total_events
FROM security_events;
```

Critical Events

```
SELECT COUNT(*) AS critical_events
FROM security_events
WHERE severity = 'critical';
```

Device Count

```
SELECT COUNT(DISTINCT device_id) AS total_devices
FROM security_events;
```

Events by Severity

```
SELECT
    severity,
    COUNT(*) AS total
FROM security_events
GROUP BY severity;
```

These queries are commonly used when building dashboards.

Complete Practice Queries

```
SELECT *
FROM security_events;

SELECT COUNT(*) AS total_events
FROM security_events;

SELECT *
FROM security_events
WHERE severity = 'critical';

SELECT
    severity,
    COUNT(*) AS total
FROM security_events
GROUP BY severity;

SELECT
    device_id,
    COUNT(*) AS total_events
FROM security_events
GROUP BY device_id;
```

```
SELECT *
FROM security_events
ORDER BY event_timestamp DESC;
```

You have successfully queried imported JSON data using SQL.

Revision #2
Created 2026-06-10 01:57:57 UTC by Mr Napper
Updated 2026-08-18 23:58:09 UTC by Mr Napper