Dynatrace Query Language (DQL) – Sample Queries

# 1. Basic Log Fetch

fetch logs  
| filter content contains "ERROR"  
| limit 100

Fetches the latest 100 logs containing the word “ERROR”.

# 2. Filter by Time (Last 2 Hours)

fetch logs  
| filter timestamp >= now() - 2h

Filters logs from the past 2 hours.

# 3. Count Errors per Service

fetch logs  
| filter content contains "ERROR"  
| summarize error\_count = count(), by:{dt.entity.service}

Counts the number of errors per service.

# 4. Timeseries of Request Count (5-minute intervals)

fetch dt.service.request  
| timeseries request\_count = sum(count), resolution: 5m

Generates a 5-minute interval time series of service request counts.

# 5. CPU Usage % by Host

fetch dt.host.cpu.usage  
| summarize avg\_cpu = avg(value), by:{dt.entity.host}

Shows average CPU usage per host.

# 6. Problems Created in Last 24 Hours

fetch problems  
| filter startTime >= now() - 24h  
| summarize total = count(), by:{severityLevel, impactLevel}

Summarizes problems by severity and impact levels in the last 24 hours.

# 7. Errors Grouped by Application

fetch logs  
| filter contains(content, "Exception")  
| summarize count = count(), by:{dt.entity.application}

Counts exceptions grouped by application.

# 8. HTTP 5xx Error Count

fetch dt.web.error  
| filter statusCode >= 500  
| summarize total\_errors = count()

Counts all HTTP 5xx errors.

# 9. Remove Nulls and Format Output

fetch logs  
| filter isNotNull(content)  
| fieldsAdd status = coalesce(status, "UNKNOWN")

Removes null values and replaces them with 'UNKNOWN'.

# 10. Create Metric from Log Query

fetch logs  
| filter content contains "login failed"  
| summarize failed\_logins = count()

Counts login failures – can be saved as a custom metric.