Dive into the world of Kusto Query Language

Sarah Lean

Senior Technical Specialist

https://www.techielass.com

Who am I?

- Senior Technical Specialist @ Microsoft
- Blog at www.techielass.com
- Founder of the Glasgow Azure User Group
- STEM Ambassador
- 18+ years in the IT industry
- IT Pro

Agenda

- What is Kusto Query Language
- Why learn Kusto Query Language?
- Basics
- Kusto Query Language Syntax
- Demo

What is Kusto Query Language?

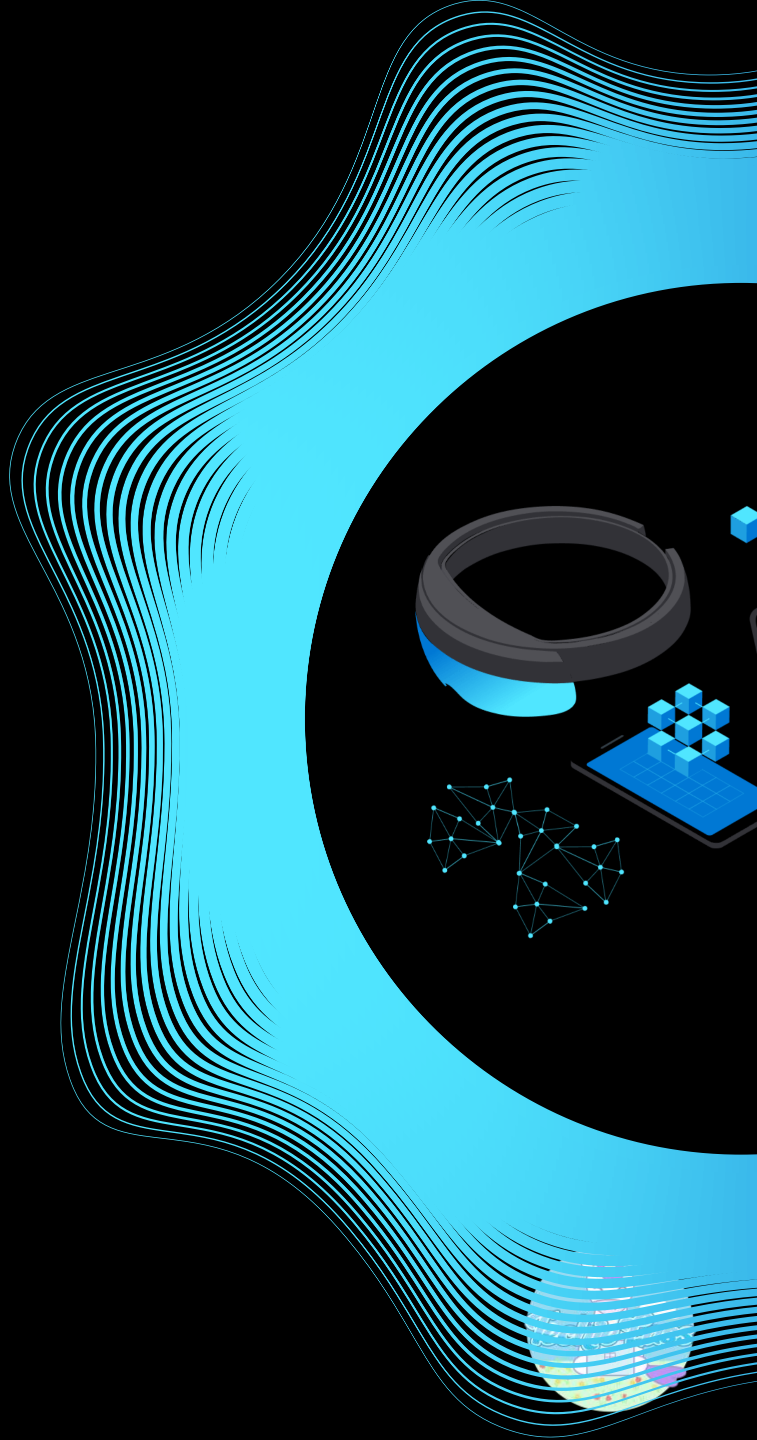
History of Kusto Query Language

- Launched in 2017
- Expanded from its initial remit
- Popular within the community

KQL is shorthand for Kusto Query Language

What products use KQL?

- Azure Data Explorer (ADX)
- Azure Monitor
- Log Analytics
- Application Insights
- Microsoft Sentinel
- Microsoft Defender for Cloud
- Resource Graph Explorer



Why learn KQL?

What are the basics of KQL?

Terminology

- **Statement/Query:** a complete command that perform a specific operation.

- **Operator**: A symbol or keyword that performs an operation on one or more expressions.

- **Step**: An individual operation within a query.

KQL example

SQL Statement:

SELECT * FROM Sales WHERE Manager = ‘William Wallace’

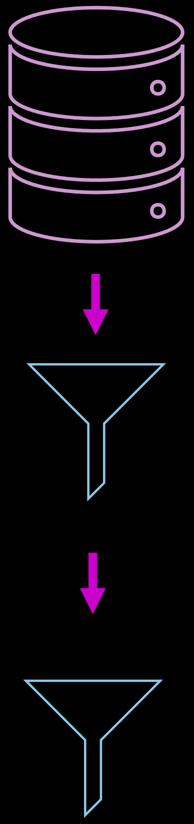
KQL Statement:

Sales

| where Manager == ‘William Wallace’

Query Order

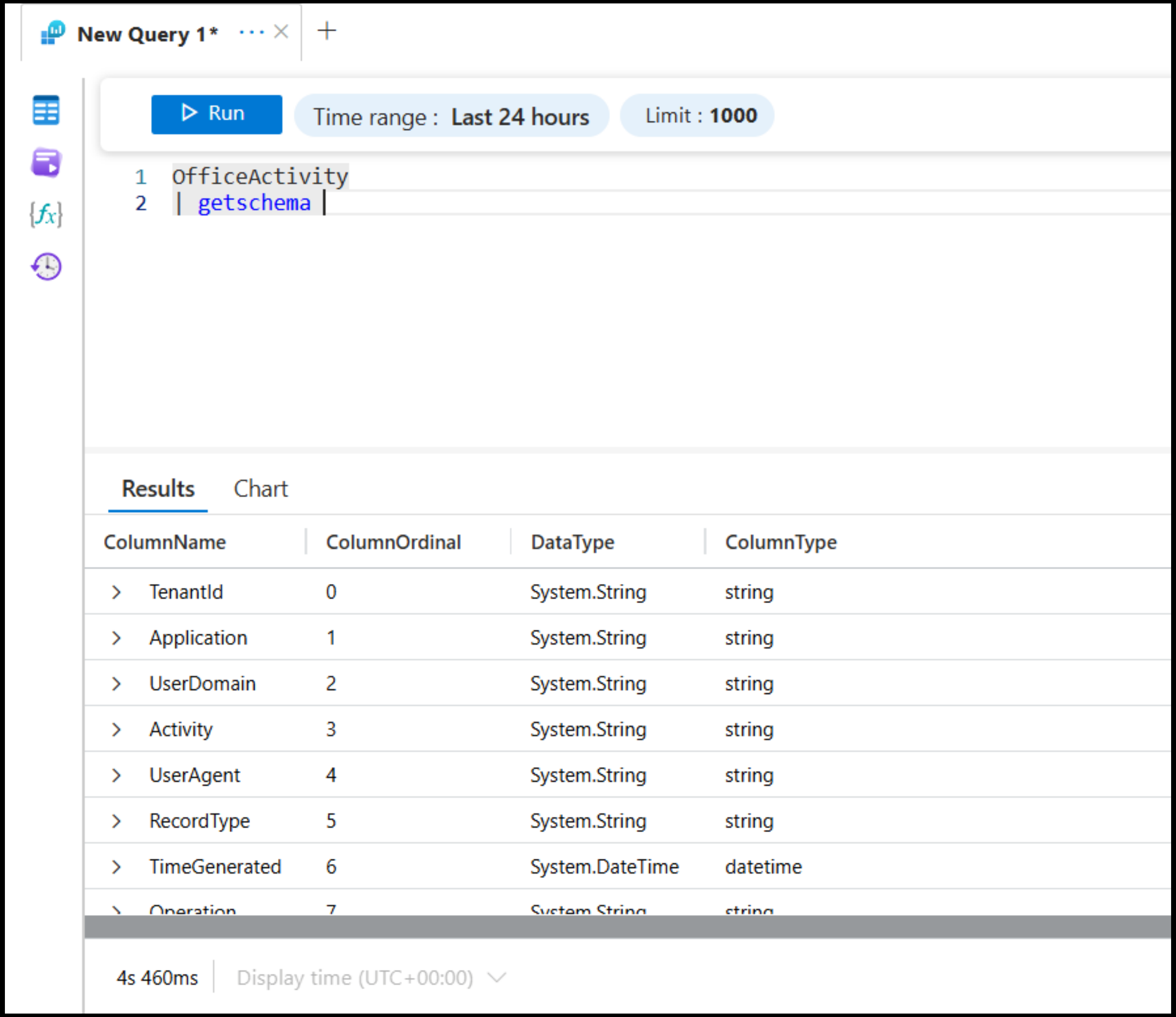
Query order matters.



Schema

```
OfficeActivity

| getschema
```

Schema



Time range : Last 24 hours

Limit : 1000

```
OfficeActivity
| getschema
```

ColumnName	ColumnOrdinal	DataType	ColumnType
TenantId	0	System.String	string
Application	1	System.String	string
UserDomain	2	System.String	string
Activity	3	System.String	string
UserAgent	4	System.String	string
RecordType	5	System.String	string
TimeGenerated	6	System.DateTime	datetime
Operation	7	System.String	string

4s 460ms | Display time (UTC+00:00)

Comparison Operators

== Exact Match

!= Does not Include

Display relevant fields

The project operator selects the columns to include, rename or drop and insert new computed columns.

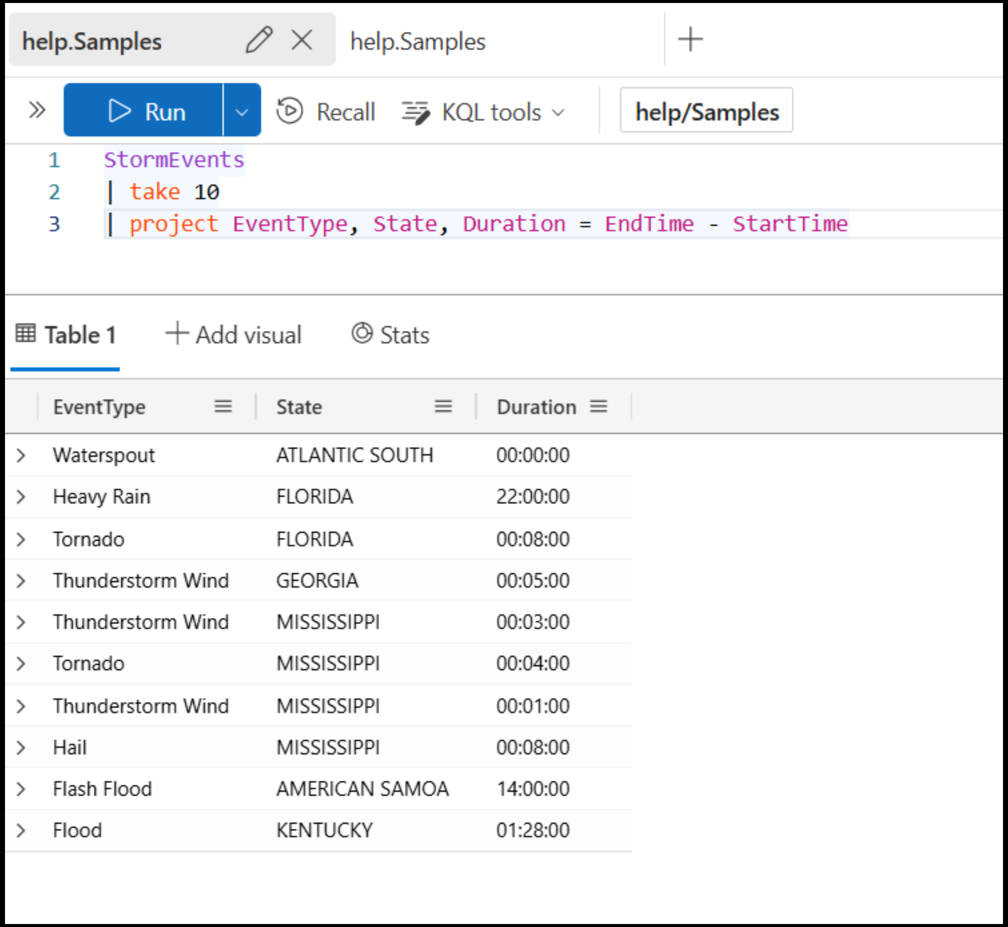
Project

```
VMComputer

| where OperatingSystemFamily == "windows"

| project HostName, Cpus, AzureLocation
```

Project – create column

```
StormEvents
| take 10
| project EventType, State, Duration = EndTime - StartTime
```

Project – create column



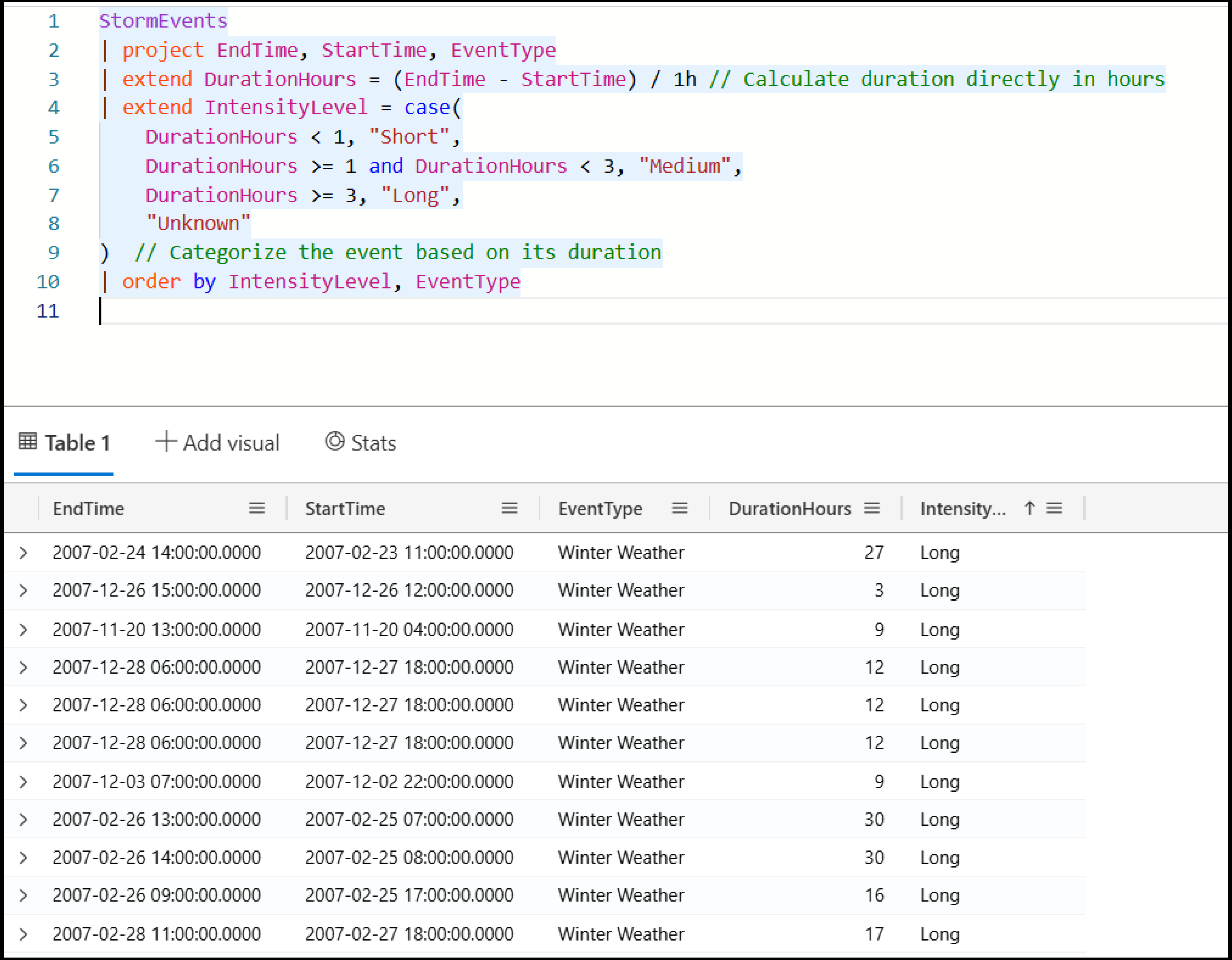
```
StormEvents
| take 10
| project EventType, State, Duration = EndTime - StartTime
```

EventType	State	Duration
Waterspout	ATLANTIC SOUTH	00:00:00
Heavy Rain	FLORIDA	22:00:00
Tornado	FLORIDA	00:08:00
Thunderstorm Wind	GEORGIA	00:05:00
Thunderstorm Wind	MISSISSIPPI	00:03:00
Tornado	MISSISSIPPI	00:04:00
Thunderstorm Wind	MISSISSIPPI	00:01:00
Hail	MISSISSIPPI	00:08:00
Flash Flood	AMERICAN SAMOA	14:00:00
Flood	KENTUCKY	01:28:00

Perform calculations on fields

The extend operator allows you to carry out calculations against fields and add in a column to the query output.

Extend



```
StormEvents
| project EndTime, StartTime, EventType
| extend DurationHours = (EndTime - StartTime) / 1h // Calculate duration directly in hours
| extend IntensityLevel = case(
    DurationHours < 1, "Short",
    DurationHours >= 1 and DurationHours < 3, "Medium",
    DurationHours >= 3, "Long",
    "Unknown"
)  // Categorize the event based on its duration
| order by IntensityLevel, EventType
```

EndTime	StartTime	EventType	DurationHours	Intensity...
2007-02-24 14:00:00.0000	2007-02-23 11:00:00.0000	Winter Weather	27	Long
2007-12-26 15:00:00.0000	2007-12-26 12:00:00.0000	Winter Weather	3	Long
2007-11-20 13:00:00.0000	2007-11-20 04:00:00.0000	Winter Weather	9	Long
2007-12-28 06:00:00.0000	2007-12-27 18:00:00.0000	Winter Weather	12	Long
2007-12-28 06:00:00.0000	2007-12-27 18:00:00.0000	Winter Weather	12	Long
2007-12-28 06:00:00.0000	2007-12-27 18:00:00.0000	Winter Weather	12	Long
2007-12-03 07:00:00.0000	2007-12-02 22:00:00.0000	Winter Weather	9	Long
2007-02-26 13:00:00.0000	2007-02-25 07:00:00.0000	Winter Weather	30	Long
2007-02-26 14:00:00.0000	2007-02-25 08:00:00.0000	Winter Weather	30	Long
2007-02-26 09:00:00.0000	2007-02-25 17:00:00.0000	Winter Weather	16	Long
2007-02-28 11:00:00.0000	2007-02-27 18:00:00.0000	Winter Weather	17	Long

Summarize

The summarize operator is used to aggregate or group data in your dataset and perform calculations such as sums, averages, counts, and more.

Summarize

```
SigninLogs

| project TimeGenerated, Location, AppDisplayName, RiskDetail, UserType

| summarize count() by Location
```

Tools

- Kusto.Explorer
- Kusto CLI
- Visual Studio Code with Kusto extension pack
- Real-Time KQL
- Azure Resource Graph Explorer
- Azure Data Explorer

Demo

Resources

Learn more:

https://aka.ms/kqlwithsarah

Thank you!

Any questions?

Learn more:

https://aka.ms/kqlwithsarah