YAML Syntax

**YAML** = serialization language, such as JSON and XML.  
Application written in different technologies can communicate in these serialization languages.  
  
*.yaml* & *.yml* are the same extension  
  
**Advantages:**

- Human readable and intuitive

Comparing YAML with other serialization languages  
  
YAML:

# YAML comment

microservices: ====> YAML object

- app: user-authentication ====> YAML item of a list (adding a dash, converts the parent object in a list). In this case, this is a list of objects.

port: 900 ====> Simple key-value pairs (use quotes when using special characters such as /n)

version: ====> This is a list of simple items. There can be list of objects and simple items.

- 1.0

- 2.0

- 2.1

primitiveList: [1.0, 2.0, 2.1] ====> Another way to express a list of primitive items (no objects)

XML:

<microservices>

<microservice>

<app>user-authentication</app>

<port>9000</port>

<version>1.0</version>

</microservice>

</microservices>

JSON:

{

microservices: [

{

app: "user-authentication",

port: 9000,

version: "1.0"

}

]

}

**Note:** YAML is superset of JSON, any valid JSON file is also a valid YAML file.

**Note:** YAML syntax is validated through line separation and indentation. That is why we cannot indent or separate lines as desired ... those have a format.

**Note:** Widely used for Docker config, Kubernetes config, Ansible, etc...

**Syntax of YAML:**

**-** **Simple key-value pairs**

**- comments:** #

**-** **objects:** indent key value pairs and adding an object name above

**-** **lists:** using a dash (-) and the attributes must be indented in the same level.

The dash converts the immediate parent object in a list.

**-** **boolean expr:** a boolean can be assigned with true/false or yes/no or on/off

**- env variable:** we can access env variables with a $ sign

**- placeholders:** {{ .Values.service.name }} the value inside the brackets gets replaced using template generator.

**- multiline strings:** use pipe "|" if you want the lines to be preserved, or ">" is you want a single line (even if you wrote it in multiple lines).  
  
 Example #1:

data:

mosquitto.conf: | ======> Notice the pipe symbol

log\_dest stdout

log\_type all

log\_timestamp true

listener 9001

Example #2:

command:

- sh

- -c

- |

#!/usr/bin/env bash -e

http() {

local path="${1}"

set -- -XGET -s --fail

# some more stuff here

curl -k "$@" "http://localhost:5601${path}"

}

http "/app/kibana"

**- multiple components:**

if we want to add multiple components in s single YAML file (useful in kubernetes),

we separate the components using three dashes (---)

apiVersion: v1

kind: ConfigMap

...

---

apiVersion: v1

kind: Secret

...