

YAML Fundamentals

YAML Basic Syntax



Maaike van Putten
Software Developer & Trainer

www.brightboost.nl



Version Check



This version was created by using:

- YAML 1.2.2 (October 2021)



Overview



What YAML is

Basic syntax

- Indentation
- Sequences / lists
- Mapping / dictionary
- Scalars
- Comments
- Documents





YAML

Yet Another Markup Language.

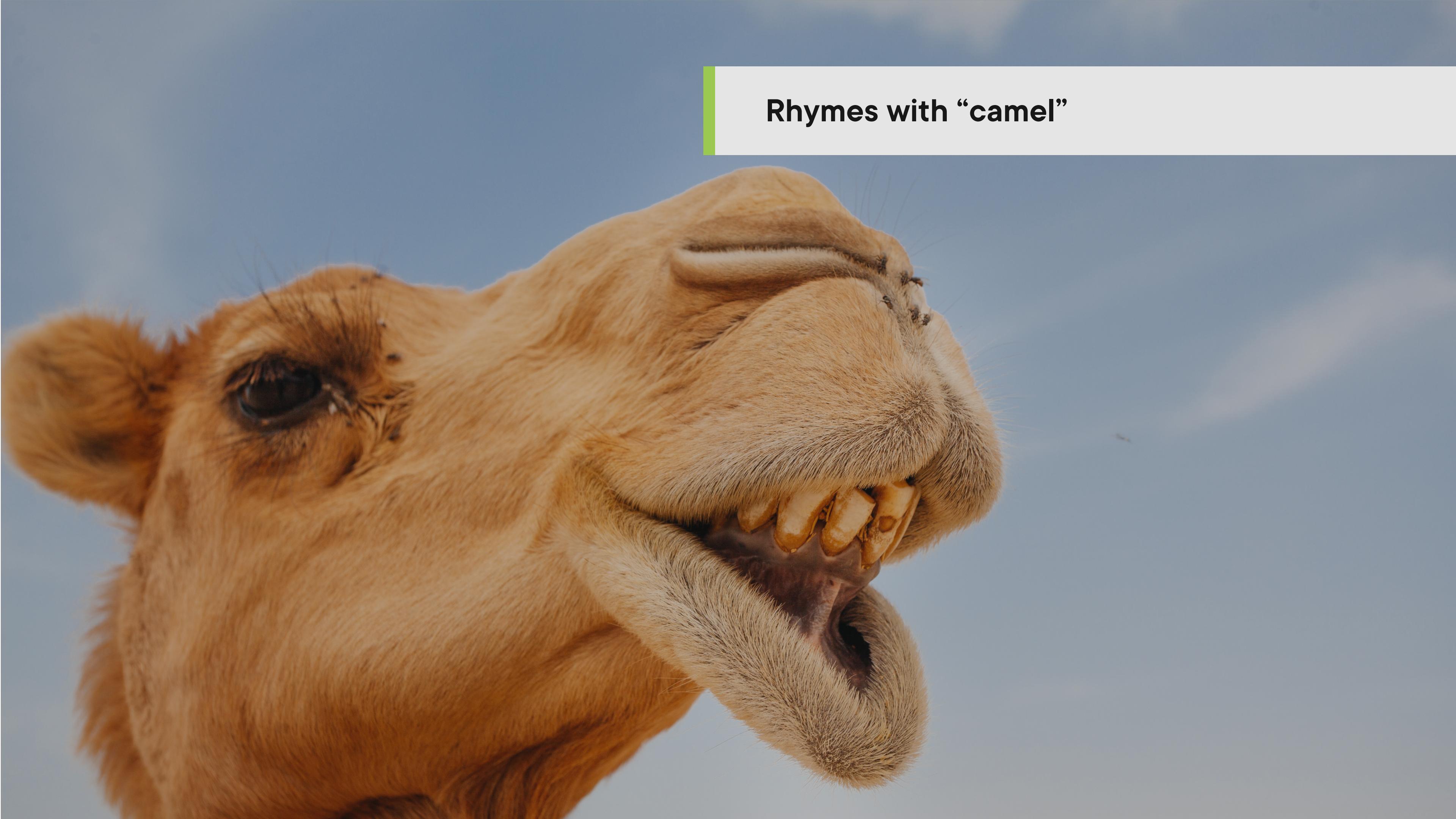




YAML

Yet Another Markup Language.
YAML Ain't Markup Language.





Rhymes with “camel”



What Is YAML?

Data serialization language

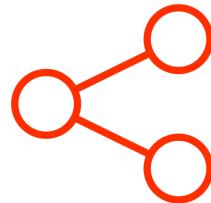
Unicode printable characters

Increasingly popular over the last few years

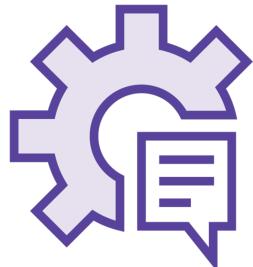
Often used for data about hosts and infrastructure



YAML Use Cases



Cross-language data sharing



Configuration files



Log files



Object persistence



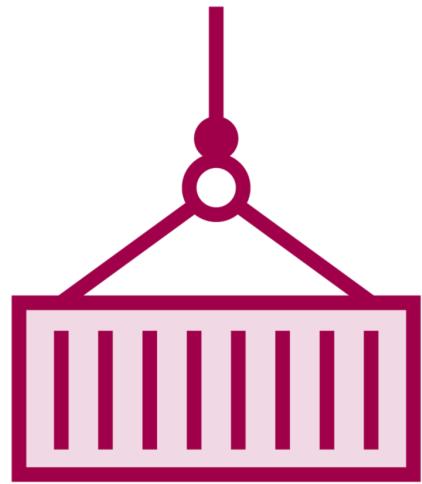
Working with languages such as PHP, JS, Python, Perl and Ruby



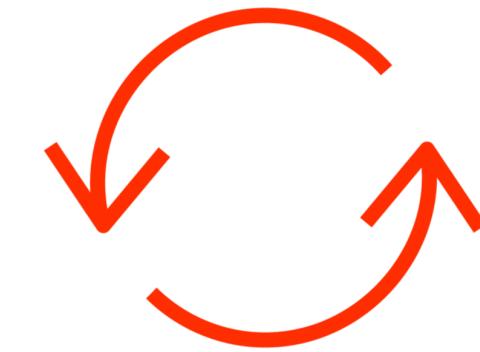
YAML Features



Human readable



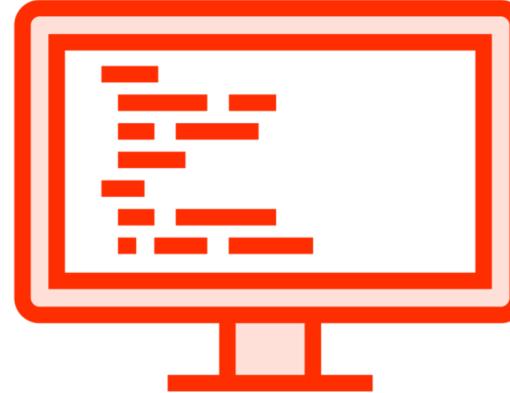
Portable



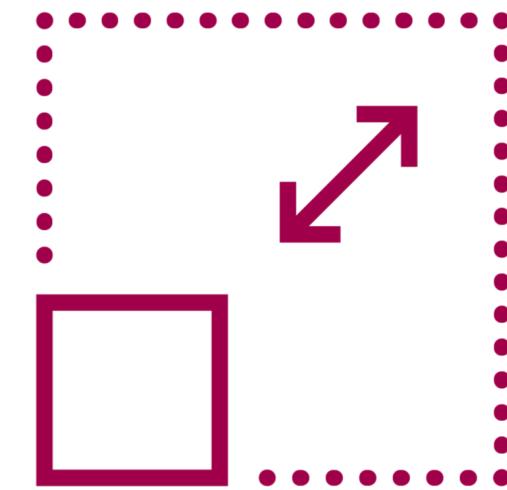
Consistent



**Match native data
structure**



Expressive



Extensible



YAML and JSON

{JSON}

JSON

**Data interchange format focused
on being interchangeable and easy
to process for programming
languages.**

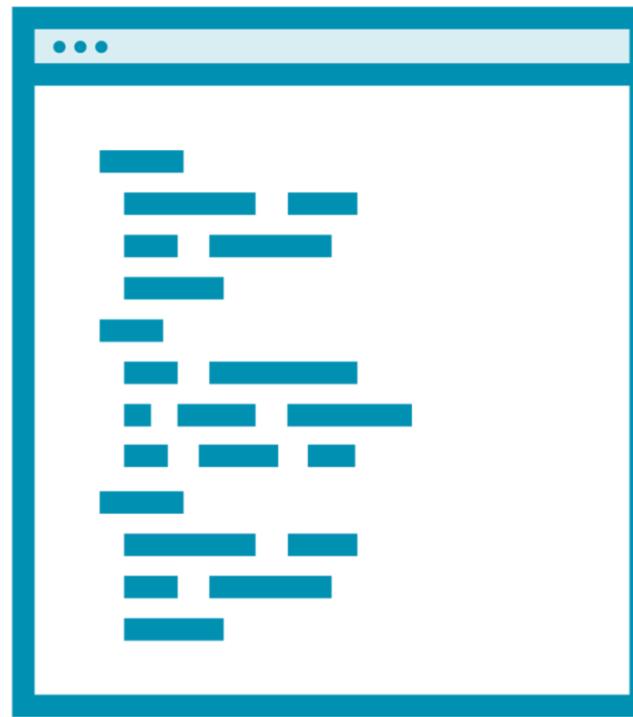


YAML

**Superset of JSON.
Data interchange format. Focus on
being human readable and
supporting native structures.**

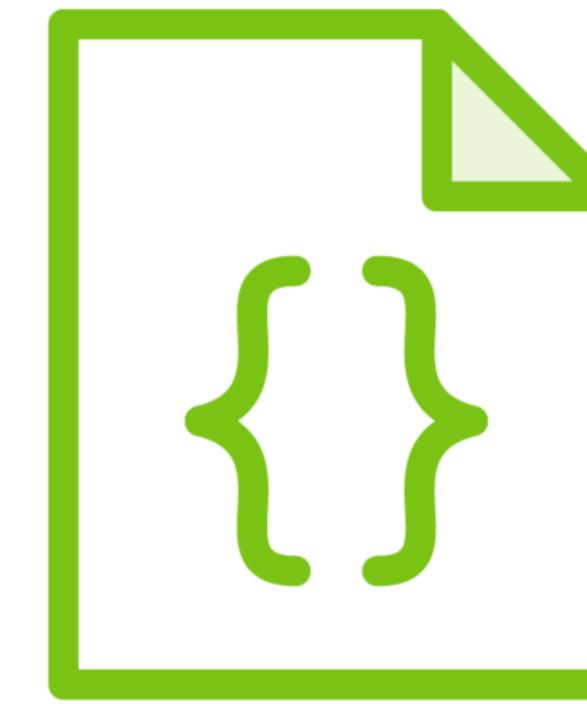


Block vs Flow Style



Block style

YAML's format, indentation to give the data structure.

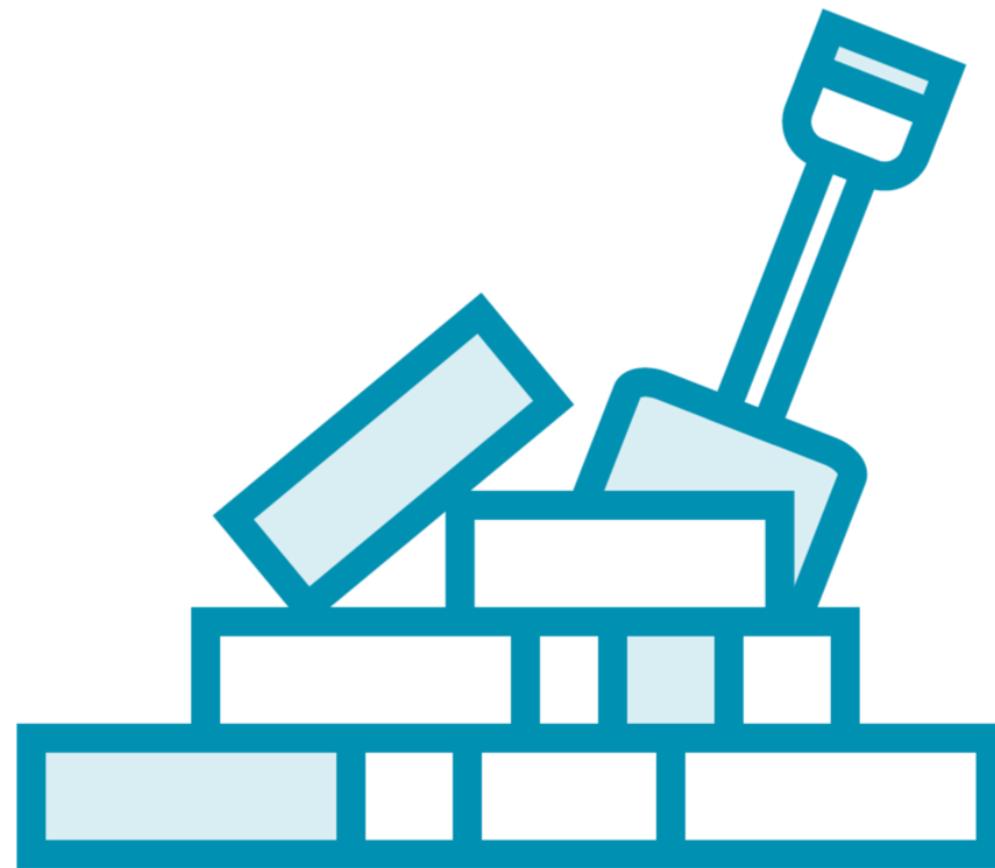


Flow style

JSON style format, using characters as explicit indicators to indicate structure.



YAML Building Blocks



Sequence

Mapping

Scalar





Indentation

Indentation is done with one or more spaces, but
never with tabs.



Sequence

```
[1, 2, 3]
```

Arrays / lists

Different styles:

- Block style
- Flow style



YAML Sequence Block Style

- first item in the list
- second item in the list
- third item in the list



YAML Sequence Flow Style

[just, a list, in flow style]



Demo



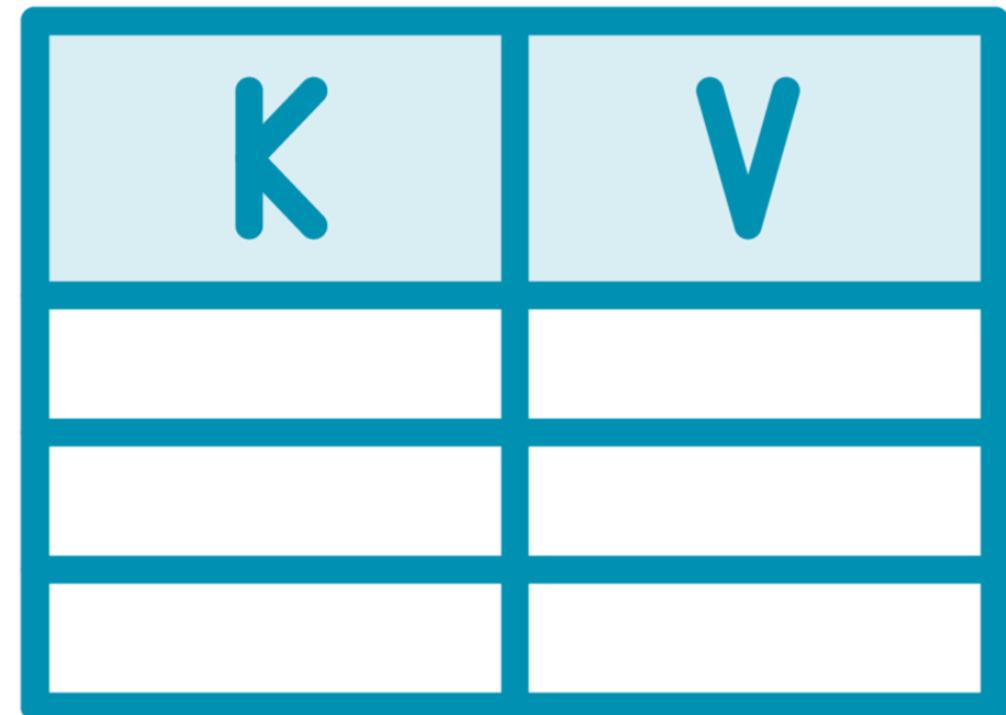
Sequence

Block style: using indentation and -

Flow style: using [] and ,



Mapping



Key-value pair

Indicated by : **(colon and space)**

Alternative terms: dictionary, hash, key-value pair



YAML Mapping Block Style

```
key: value
```

```
list:
```

- first item
- second item
- third item

```
flowlist: [just, a list, in flow style]
```



YAML Mapping Flow Style

```
{ key: value, another: value, flowlist: [just, a list, in flow style] }
```



Demo



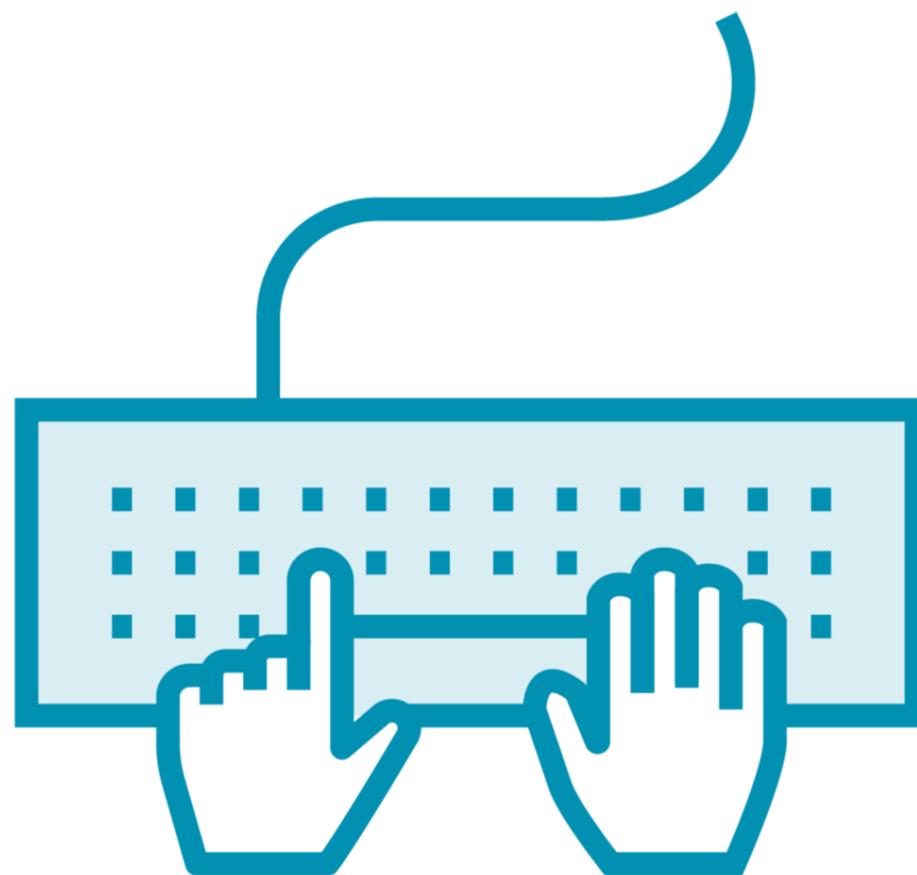
Mapping

Block style: using colon and space

Flow style: using { }, : and ,



Scalar



Basic values: strings, numbers, booleans, dates

Different styles:

- Flow scalar
- Block scalar

**Escaping with **



YAML Scalar

```
tool: yaml
version: 1.2
awesome: true
nothing: null
duplicateKeys: not allowed
```



Demo



Basic scalars

For different data types

- String
- Numeric
- Date
- Boolean



Comment



Presentation detail

No effect on serialization tree

Meant for humans

Start with a “#”



YAML Comment

```
# this is a comment  
# it will be lost during parsing
```



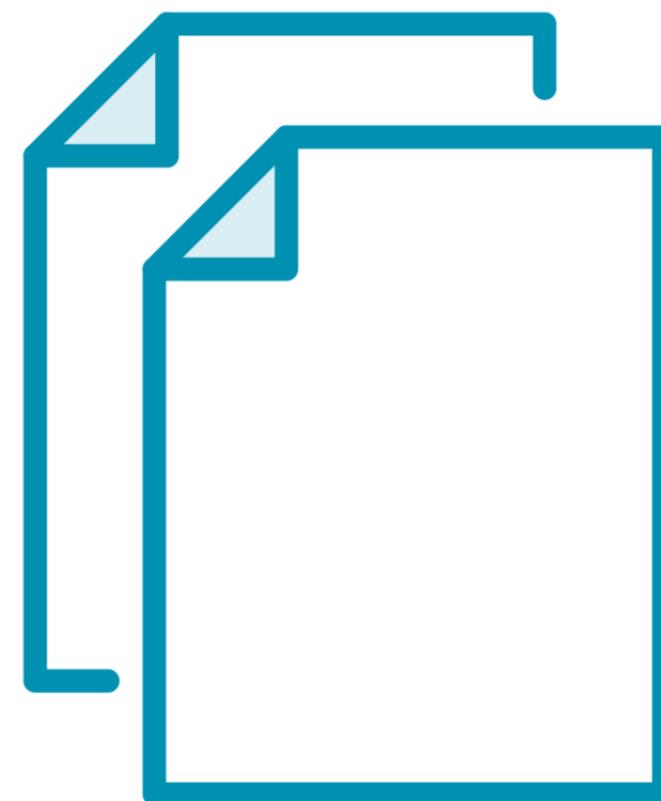
Demo



Comments



Document



One file can contain multiple documents

Documents are separated by 3 hyphens (---)

Can be ended with suffix of 3 dots (...)



YAML Document

```
tool: yaml
version: 1.2
awesome: true
nothing: null
duplicateKeys: not allowed
```

This is a new document

One file can contain multiple documents



Demo



Multi documents



Summary



Sequences

Mappings

Scalars

Comments

Documents



Up Next:
YAML Advanced Syntax

