



YAML

Serialization format

18, person, pune,

→ raw data → meaningless

↓
Serialization format

↓
information → meaningful

→ XML
→ JSON
→ YAML

XML

<person>

<age> 18 </age>

<name> person </name>

<city> pune </city>

</person>

JSON

{

"age": 18,

"name": "person",

"city": "pune"

}

YAML

age: 18

name: person

city: pune

Overview



- YAML is the abbreviated form of “YAML Ain’t Markup Language”
- It is a data serialization language which is designed to be human -friendly and works well with other programming languages for everyday tasks
- It's often used as a format for configuration files
- It's also easy to for humans to read, which is why it's a good choice for configuration
- It is useful to manage data and includes Unicode printable characters

Features



- Matches native data structures of agile methodology and its languages such as Perl, Python, PHP, Ruby and JavaScript
- YAML data is portable between programming languages
- Includes data consistent data model
- Easily readable by humans ✖ ✖ ✖
- Supports one-direction processing
- Ease of implementation and usage

Basics



- YAML is case sensitive
- The files should have .yaml or .yml as the extension
- YAML does not allow the use of tabs while creating YAML files; spaces are allowed instead
- Comment starts with #
- Comments must be separated from other tokens by whitespaces

comment
 ↑
 space is mandatory

Scalars



- Scalars in YAML are written in block format using a literal type

- E.g.

- Integer

- 20
- 40

- String

- Steve → No quotes
- "Jobs" → double quotes
- 'USA' → single quotes

- Float

- 4.5
- 1.23015e+3

Mapping



- Represents key-value pair
- The value can be identified by using unique key
- Key and value are separated by using colon (:)
- E.g.

→ string

name: person1

address: "India"

phone: +9145434345

age: 40

[hobbies:
- reading
- playing] map
sequence

Sequence



- Represents list of values
- Must be written on separate lines using dash and space
- Please note that space after dash is mandatory
- E.g.

pet animals

- cat

- dog

programming languages

- C

- C++

- Java

Sequence



- Sequence may contains complex objects
- E.g.
 - products:
 - **title**: product 1
price: 100
description: good product
 - **title**: product 2
price: 300
description: useful product