YAML Cheat Sheet

YAML is a simple but robust data serialization language. It achieves high flexibility with just two data structures: sequences (a list) and mappings (key and value pairs).

Sequence (lists)	Mapping (key value pairs)
``---

- Linux

- BSD

- Illumos``` | ```---

Kernel: Linux

CPU: AMD

RAM: 16 GB``` |

These structures can be combined and embedded.

Sequence of mappings (list of pairs)	Mapping sequences (key with many values)
``---
-

  CPU: AMD

  RAM: '16 GB'

-

  CPU: Intel

  RAM: '16 GB'``` | ```---

Linux:

   - Fedora

   - Slackware

FreeBSD:

   - FreeBSD

   - NetBSD``` |
| **Sequence of sequences (a list of lists)** | **Mapping of mapping (many keys and values)** |
| ```---

- [Linux, FreeBSD, Illumos]

- [YAML, XML, JSON]``` | ```---

Desktop:

   CPU: AMD

   RAM: '32 GB'``` |

YAML Cheat Sheet

Multi-line entries

```
---
- Linux: |
    A UNIX-like, open source
    operating system.
```

Use a | or > to allow for data blocks. This example represents this JSON structure:

```
[{"Linux": "A UNIX-like, open source operating system.\n"}]
```

Complex mappings

```
---
?
   - Linux
   - GNU
:
   - Open source
   - GPL
   - UNIX-like
?
   - BSD
   - Illumos
:
   - Open source
   - UNIX
```

A complex mapping uses a **?** (on its own line) to map one sequence to another list indicated by a **:** character on its own line.

This example complex map is read as:

Linux and GNU are open source, GPL, and UNIX-like. BSD and Illumos are open source and UNIX-like.

Here is the same structure in JSON:

```
{"Linux,GNU": ["Open source", "GPL", "UNIX-like"],
"BSD,Illumos": ["Open source","UNIX"}
```