

# REFERÊNCIA RÁPIDA DE YAML

Escalares, seqüências, mapeamentos, âncoras, multi-linha

## Sintaxe

### Regras Básicas

<b>Indentação</b>	Apenas espaços (sem tabs), profundidade consistente
<b>#</b>	Comentário (até o final da linha)
<b>--</b>	Marcaador de início de documento
<b>...</b>	Marcaador de fim de documento
<b>Sensível a maiúsculas</b>	Chaves e valores diferenciam maiúsculas e minúsculas

### Exemplo Mínimo

```
name: Alice
age: 30
active: true
```

## Escalares

### Tipos de Escalar

<b>hello</b>	String (sem aspas)
<b>"hello"</b>	String (aspas duplas, permite escapes)
<b>'hello'</b>	String (aspas simples, literal)
<b>42 / 3.14</b>	Número (inteiro / decimal)
<b>true / false</b>	Booleano
<b>null / ~</b>	Valor nulo
<b>2026-03-26</b>	Data (ISO 8601)

### Regras de Aspas

```
plain: sem aspas necessárias
special: "dois-pontos: e cerquilha # precisam de aspas"
escape: "linha\nnova"
literal: 'sem \n escape aqui'
```

## Seqüências

### Seqüência em Bloco

```
fruits:
- apple
- banana
- cherry
```

### Seqüência de Fluxo (Em Linha)

```
colors: [red, green, blue]
matrix: [[1, 2], [3, 4]]
```

### Seqüência de Objetos

```
users:
- name: Alice
  role: admin
- name: Bob
  role: user
```

## Mapeamentos

### Mapeamento em Bloco

```
server:
  host: localhost
  port: 8080
  debug: false
```

### Mapeamento de Fluxo (Em Linha)

```
point: {x: 10, y: 20}
```

### Chaves Complexas

```
? [first, second]
: "valor de chave composta"
```

## Âncoras e Aliases

### Definir e Reutilizar

```
defaults: &defaults
timeout: 30
retries: 3
production:
<< *defaults
timeout: 60
```

### Referência

<b>&amp;name</b>	Definir uma âncora
<b>*name</b>	Referenciar (alias) uma âncora
<b>&lt;&lt;</b>	Chave de mesclagem — mesclar mapeamento no atual

## Strings Multi-linha

### Escalares em Bloco

<b>  (literal)</b>	Preserva quebras de linha exatamente
<b>&gt; (dobrado)</b>	Dobra quebras de linha em espaços
<b>!+ (manter)</b>	Literal, manter quebras de linha finais
<b>!- (remover)</b>	Literal, remover quebras de linha finais
<b>&gt;- (dobrar+remover)</b>	Dobrado, remover quebras de linha finais

### Bloco Literal

```
script: |
echo "line one"
echo "line two"
```

### Bloco Dobrado

```
description: >
Este texto longo será
dobrado em uma única linha
com espaços.
```

## Tags

### Tags de Tipo

<b>!!str 42</b>	Forçar valor para string "42"
<b>!!int "42"</b>	Forçar valor para inteiro 42
<b>!!float 1</b>	Forçar valor para decimal 1.0
<b>!!bool "true"</b>	Forçar valor para booleano
<b>!!null ""</b>	Forçar valor para nulo
<b>!!binary</b>	Dados binários codificados em Base64

### Exemplo de Tag

```
port: !!str 8080
enabled: !!bool "yes"
version: !!float 3
```

## Padrões Comuns

### Docker Compose

```
services:
  web:
    image: nginx:alpine
    ports: ["80:80"]
    environment:
      NODE_ENV: production
```

### GitHub Actions

```
on: [push, pull_request]
jobs:
  test:
    runs-on: ubuntu-latest
    steps:
      - uses: actions/checkout@v4
```

### Dicas

<b>yamllint</b>	Validar YAML quanto a sintaxe e estilo
<b>yq</b>	Processador YAML de linha de comando (como jq)
<b>Evitar Norway</b>	NO é false booleano — colocar aspas em códigos de países
<b>Evitar !!python</b>	Nunca carregar YAML não confiável com carregadores inseguros