**Introduction**

**What is ?**

* is a human-readable data serialization standard that can be used in conjunction with all programming languages.
* It is often used for configuration files and data exchange between languages with different data structures.

**Basic Concepts**

* **Syntax**: Uses indentation to represent nested structures, similar to Python.
* **Data Types**: Scalars (strings, numbers, booleans), Sequences (arrays/lists), and Mappings (hashes/dictionaries).

**Basic Syntax**

**Scalars**

* **Strings**: Can be unquoted, single-quoted, or double-quoted.

unquoted: unquoted string

single\_quoted: 'single quoted string'

double\_quoted: "double quoted string with \n escape"

**Numbers**: Integers and floats.

integer: 42

float: 3.14159

* **Booleans**: True and False.

boolean\_true: true

boolean\_false: false

* **Null**: Represented as **null**, **~**, or empty.

null\_value: null

tilde\_null: ~

empty\_null:

**Sequences**

* Represented as a list with a dash (**-**) followed by a space.

fruits:

- Apple

- Orange

- Banana

**Mappings**

* Represented as key-value pairs.

person:

name: John Doe

age: 30

married: true

**Advanced Features**

**Nested Structures**

* Combining sequences and mappings.

- name: John Doe

age: 30

children:

- name: Jane Doe

age: 10

- name: Jim Doe

age: 5

**Multi-line Strings**

* **Literal Block** (**|**): Preserves newlines.

description: | This is a description that spans multiple lines.

* **Folded Block** (**>**): Folds newlines to spaces.

description: > This is a description that spans multiple lines.

**Anchors and Aliases**

* **Anchors** (**&**) and **Aliases** (**\***) for reusing nodes.

defaults: &defaults adapter: mysql host: localhost development: database: dev\_db <<: \*defaults

**Merging Data**

* **Merge Key** (**<<**) for merging mappings.

default: &default name: Default Name age: 30 person: <<: \*default name: Jane Doe

**Kubernetes YAML Structure**

**Common Elements in Kubernetes YAML**

* **apiVersion**: Specifies the version of the Kubernetes API.
* **kind**: Specifies the type of Kubernetes object.
* **metadata**: Provides metadata about the object, such as name and labels.
* **spec**: Describes the desired state of the object.

**Basic Kubernetes Objects**

**Pods**

* A Pod is the smallest deployable unit in Kubernetes.

apiVersion: v1

kind: Pod

metadata:

name: my-pod

spec:

containers:

- name: my-container

image: nginx

ports:

- containerPort: 80

**Deployments**

* A Deployment manages a set of identical Pods and ensures they are running.

apiVersion: apps/v1

kind: Deployment

metadata:

name: my-deployment

spec:

replicas: 3

selector:

matchLabels:

app: my-app

template:

metadata:

labels:

app: my-app

spec:

containers:

- name: my-container

image: nginx

ports:

- containerPort: 80

**Services**

* A Service defines a logical set of Pods and a policy to access them.

apiVersion: v1

kind: Service

metadata:

name: my-service

spec:

selector:

app: my-app

ports:

- protocol: TCP

port: 80

targetPort: 80

type: ClusterIP

**Advanced Kubernetes Objects**

**ConfigMaps**

* ConfigMaps store configuration data in key-value pairs.

apiVersion: v1

kind: ConfigMap

metadata:

name: my-config

data:

configKey: configValue

**Secrets**

* Secrets store sensitive data such as passwords.

apiVersion: v1

kind: Secret

metadata:

name: my-secret

type: Opaque

data:

username: bXl1c2VybmFtZQ== # Base64 encoded

password: bXlwYXNzd29yZA== # Base64 encoded

**PersistentVolumeClaims (PVC)**

* PVCs request storage resources.

apiVersion: v1

kind: PersistentVolumeClaim

metadata:

name: my-pvc

spec:

accessModes:

- ReadWriteOnce

resources:

requests:

storage: 1Gi

**Advanced Features and Best Practices**

**Annotations**

* Use annotations to attach metadata to objects.

metadata:

annotations:

key: value

**Templating**

* Use Helm or Kustomize for templating YAML files for better reusability and manageability.

**Organizing YAML Files**

* Keep related resources in separate files or directories for better organization.

├── deployments

│ └── my-deployment.yaml

├── services

│ └── my-service.yaml

├── configmaps

│ └── my-config.yaml

└── secrets

└── my-secret.yaml