**YAML** **SCRITING**

Introduction:

* YAML stands for “YAML ain’t another markup language”.
* YAML is a data format used to exchange data.
* In YAML, we can store only data but not the commands.
* It is similar to XML & JSON.
* YAML is used for data serialization.
* It is used to write configuration files in docker, kubernetes etc...

Benefits of YAML file:

* Simple and easy to read.
* It has a strict syntax. [Indentation is important].
* Easily convertible to JSON and XML.
* Most languages use YAML.
* More powerful while representing complex data.
* Parsing is easy. [Reading the data].

“YAML files should have .yaml or .yml extension”.

Data types and representation:

* Strings:

Ex:-

Name: kumar

Fruit: “apple”

job: ‘swe’

# multiple lines string.

Bio: |

Hey my name is kumar.

I work in Bangalore.

# writing a single line in multiple lines.

message: >

this will

all be

in one single line.

* Integers:

Number: 5442

* Float:

Marks: 58.91

* Boolean:

booleanValue: No

#Specifying the data types explicitly.

zero: !!int 0

positiveNum: !!int 45

negativeNum: !!int -45

binaryNum: !!int 0b1101

octalNum: !!int 06574

hexa: !!int 0\*45

commaValue: !!int +540\_000 # 540,000

marks: !!float 54.24

infinites: !!float .inf

not a num: .nan

surname: !!null Null

~: this is a null key

# dates and time

Date: 2012-12-12

exponentialNumbers: 6.023E56

* Comments:

Comments begin with a pound sign. They can appear after a document value or take up an entire line.

Ex:-

\_\_\_

# This is a full line comment

foo: bar # this is a comment, too

* Key-Value Pairs and Dictionaries:

The key-value is YAML's basic building block. Every item in a YAML document is a member of at least one dictionary. The key is always a string. The value is a scalar so that it can be any datatype. So, as we've already seen, the value can be a string, a number, or another dictionary.

* Arrays:

You can specify arrays or lists on a single line.

---

items: [ 1, 2, 3, 4, 5 ]

names: [ "one", "two", "three", "four" ]

Or, you can put them on multiple lines.

---

items:

 - 1

 - 2

 - 3

 - 4

 - 5

names:

 - "one"

 - "two"

 - "three"

 - "four"

### Dictionaries

### We covered dictionaries above, but there's more to them. Like arrays, you can put dictionaries inline. We saw this format above. It's how python prints dictionaries.

---

foo: { thing1: huey, thing2: louie, thing3: dewey }

### 