**YAML**

**YAML:**

It was first called Yet Another MarkUp Langugae but now it is Yaml Ain't MarkUp Language.

**Definition:**

Applications written with different technologies or languages which have different *Data Structures* can transfer data to each other using a *Common/ Standerd Format*. YAML, JSON and XML are these formats

It is a Data Serialization Language

• Data Serialization: Translation of Object (Code + Data) into stream of bytes that saves the state of this object in a form that is transmittable. Using Serialization, data can be converted into code and code can be converted into data again.

Benifits of YAML:

• Simple and easy to read

• Nice and Strict Syntax

• Most Languages use it

• More powerful when representing complex data

***Note: Any JSON input can be converted to YAML output and vice virse***

**SYNTAX**

**Data Type**

**Yaml Automatically Determines the data type if not specified**

name: Hammad

age: 20

graduate: no

**But you can specify a dataType**

Syntax --> variable: !!datatype value

age: !!int 20

**Some Common are**

zero: !!int 0

positiveNum: !!int 45

negativeNum: !!int -27

commaValue: !!int 245\_000 #245,000

**Other Types**

PI: !!float 3.14

infinite: !!infinite .inf

not a num: !!nan .nan

booleanValue: !!boolean no

nameString: !!str Hammad

nullDataType: !!null null #null or NULL or ~

**Lists**

- Item1

- Item2

- Item3

**Nested Lists**

-

- Item1

- Item2

- Item3

-

- Item1

- Item2

- Item3

**Block Type / Object**

ObjectTypeKeyValue:

KeyValue: Pair

KeyValue2: Pair2

**Or**

ObjectTypeSimpleList: -Item1

-Item2

-Item3

**You want to display multiple line in your browser but you want your browser to treat it as a single line #Use ">"**

aboutInSingleLine: >-

I am a CS student

Currently Learning DevOps

and Android Dev

**Same as**

about2: I am a CS student Currently Learning DevOps and Android Dev

**Remember Indentation (one space) is very important here**

aboutTwoMultipleLine: |-

I am a CS student

Currently Learning DevOps

and Android Dev

**Single Key but have two Values**

pairExample:

- job: Student

- job: Teacher

**Set**

SetExample:

?Hammad: Student

?Danish: Graduate

?Zubair: Freelancer

**Dictionary**

People:

-Hammad:

- Role: Student

- Age: 20

-Kunal:

-Role: Teacher

-Age: 78

**Reusing Properties**

Likes: &likes

- Fruites: Mango

- Food: Biryani

Person1:

-Name: Hammad

<<: \*likes