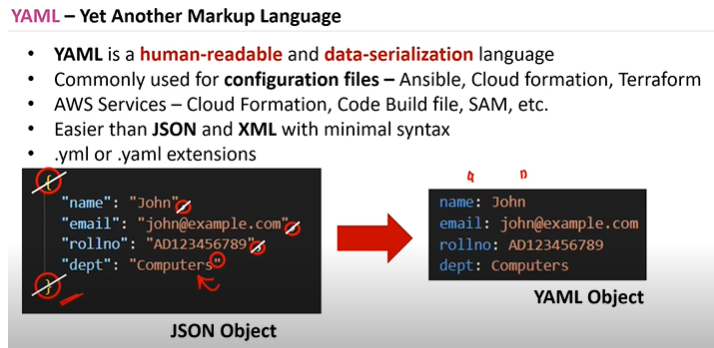
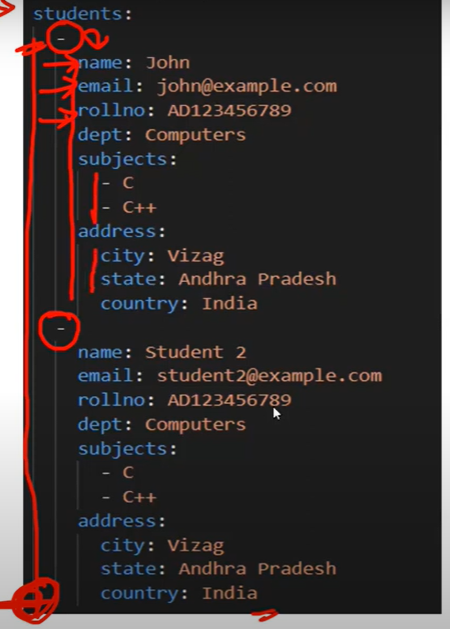
**Cloud formation**

**It’s a Infrastructure as a code which is only useful for AWS**

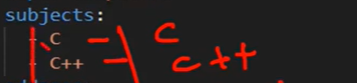
* **Cloudformation supports Yaml/json**
* **For multicloud we generally use terraform**
* **Terraform supports Yaml that too in Hashicorp format**
* **SAM is advanced version of Cloudformation**



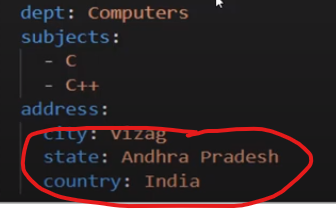
**For iam policies it wont support json**

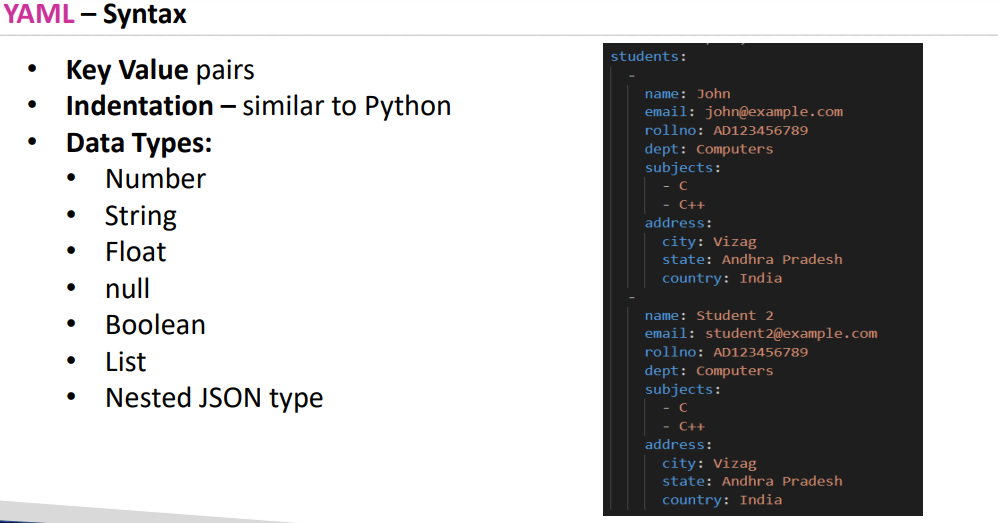
****

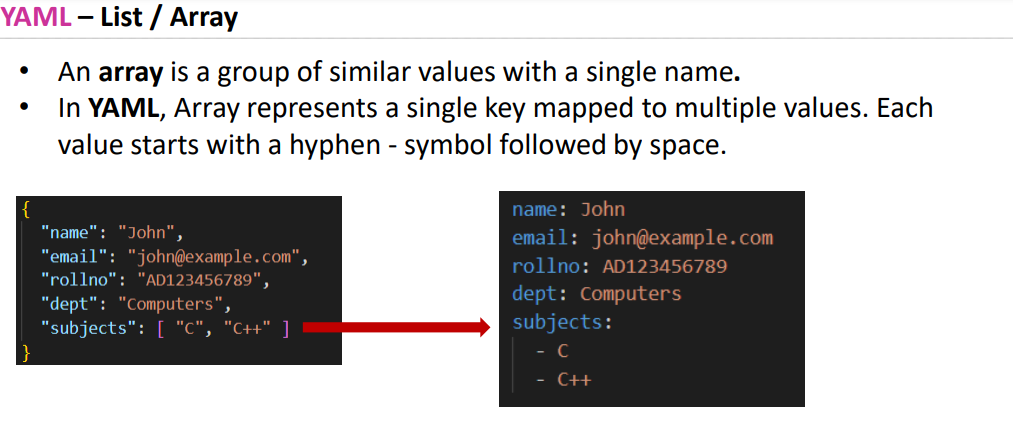
**It s called intendation**

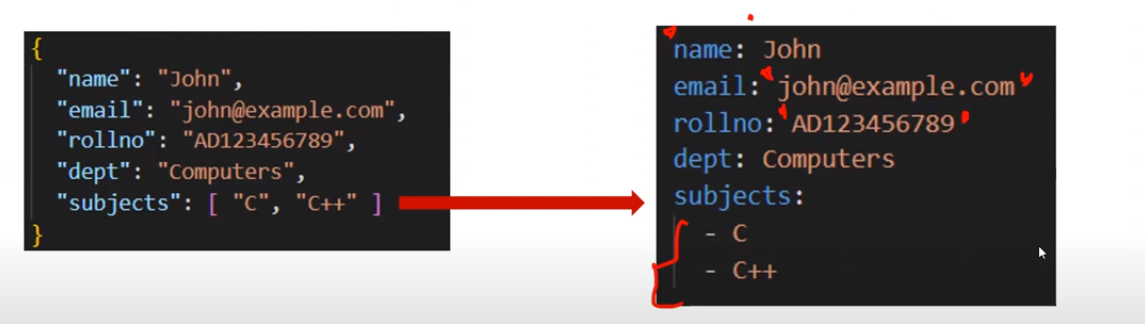
****

**This is list format in Yaml**

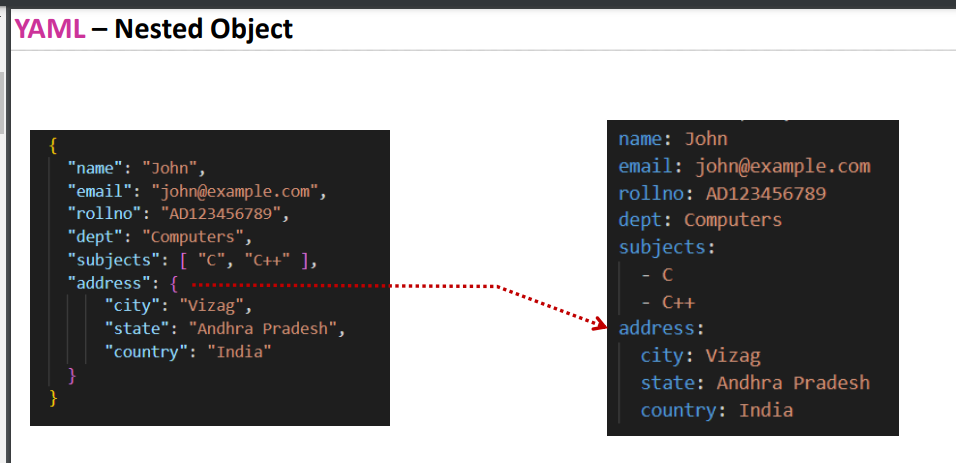
** This is nested json type**

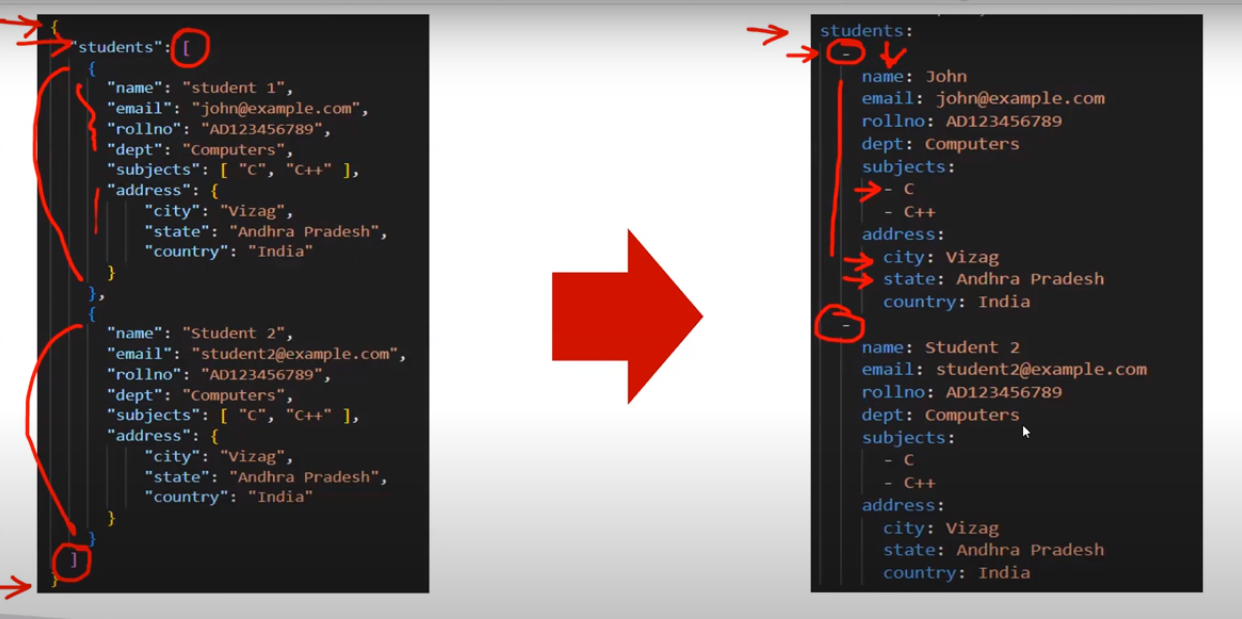
****

****

****

**For beutification purpose we can keep strings “ “ in yaml as well that too for write side values**

****

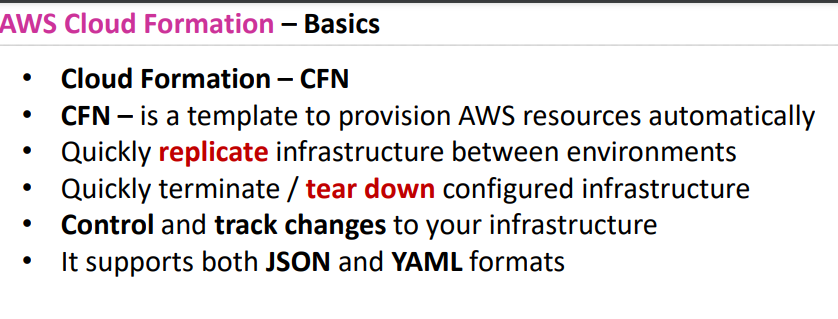
****

**Yaml is case sensitive**

* **Policies support json in aws**

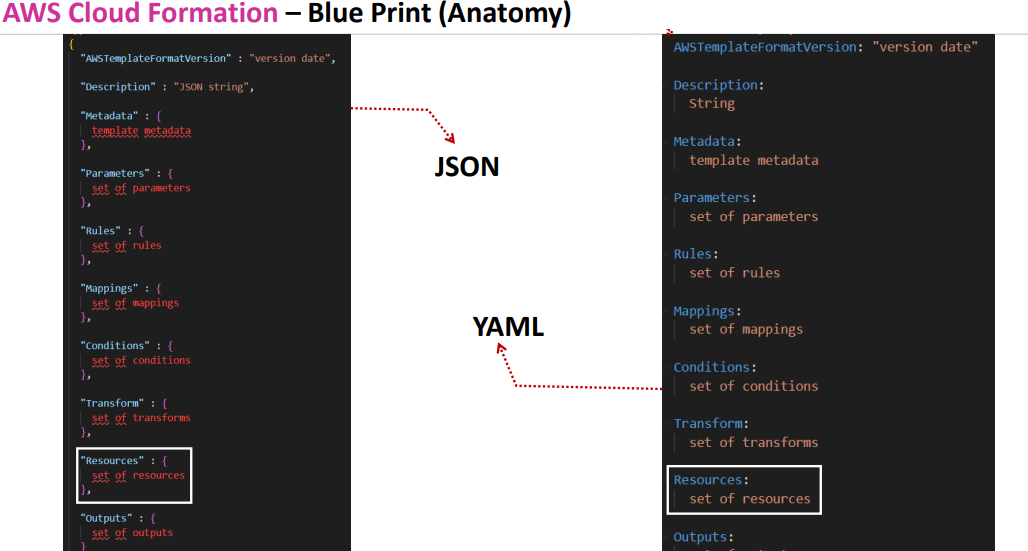
****

* **Manual setup will be time taking and there wont be any consistency**

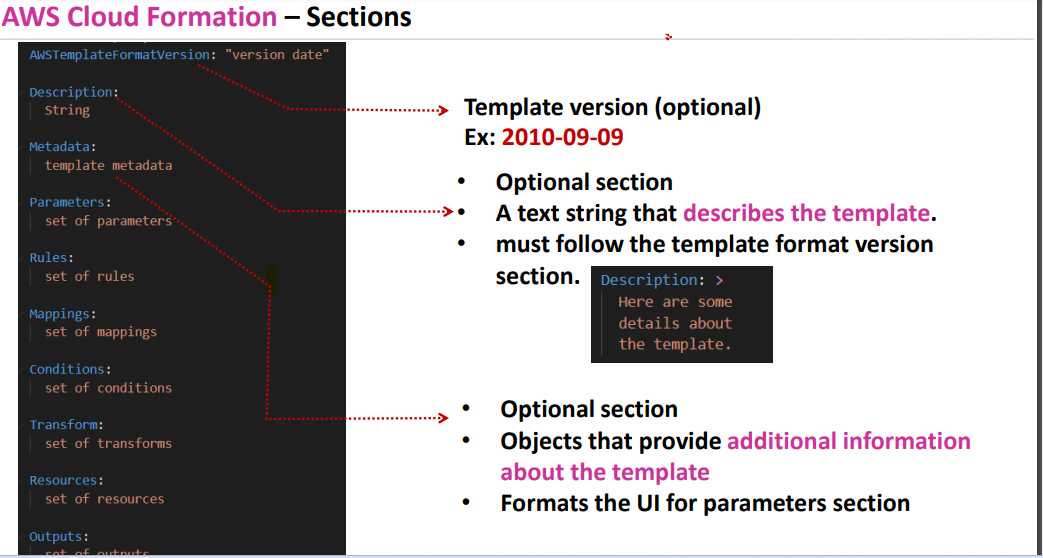
****

**CFN is a template for configuring resources**

* **For ex u r creating s3 bucket u will configure bucket name etc etc**
* **Control and track changes will be tracked easily as this code will be deployed in git so what ever is added in that particular stage in your projects that will be tracked**
* **For ex : you only added igw later after 20 days someone came and asked you to add nat gateway this kind of tracking will be maintained**

****

**Only resources section is important while creating resources in cloud formation or terraform what ever it might be rest all are optional**

****

**Description should always be placed below the aws templateformat version if not error throws**

**Meta data section is only useful when cloudformation created through console**

**In real time environment you have different purposes like prod , dev, testing**

**So you cant create different templates for all the three sections**

**For ex in production u don’t need too much of cpu and ram**

**For dev u need little less cpu when compare to testing so you will mention parameters using rules and conditions**

**Like if production team = create 2 c 8 ram**

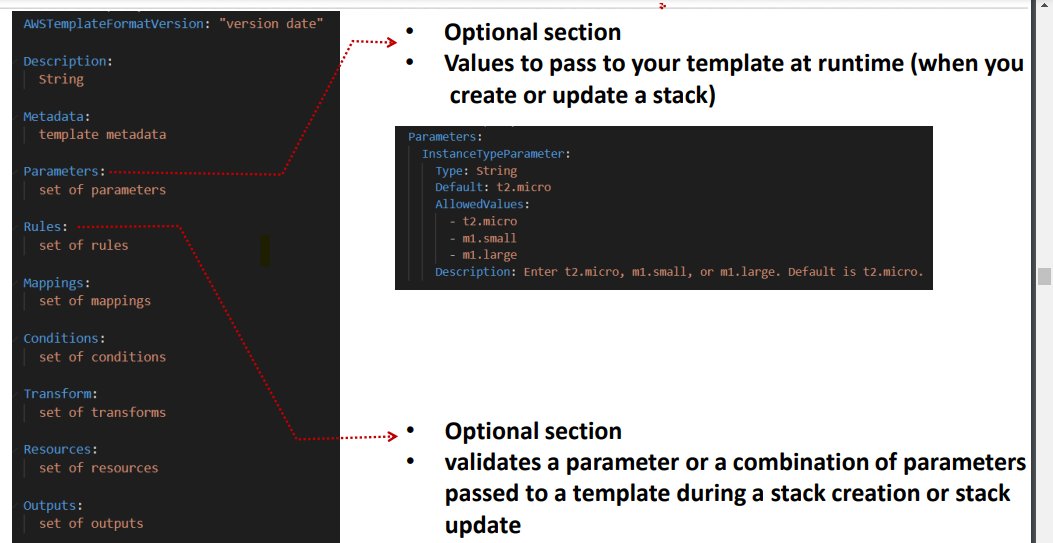
**Meta data should be there when ever there is a parameter**

* **What kind and what type**

****

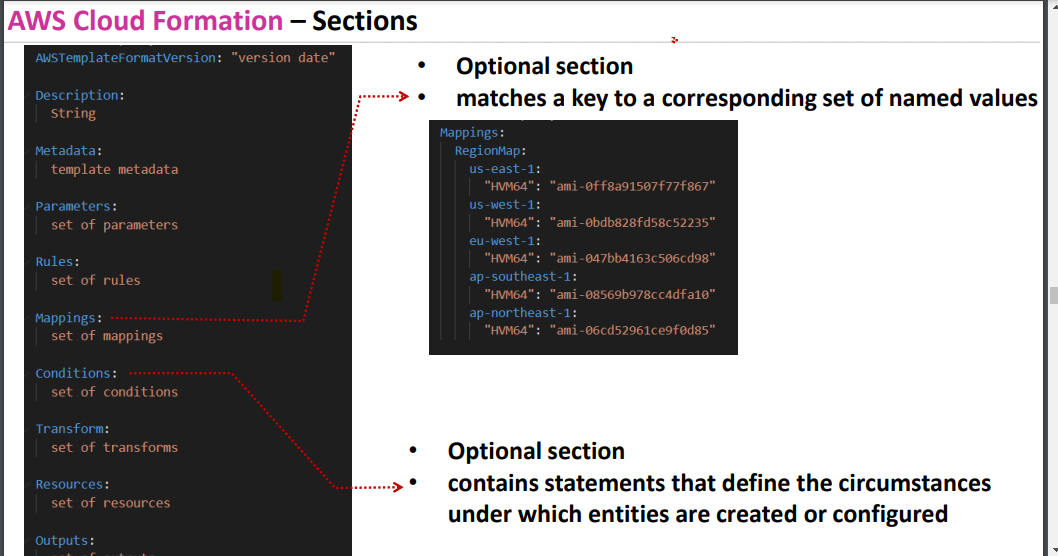
**Then cloud formations decides based on those conditions what to create**

**Conditions will be based on parameters**

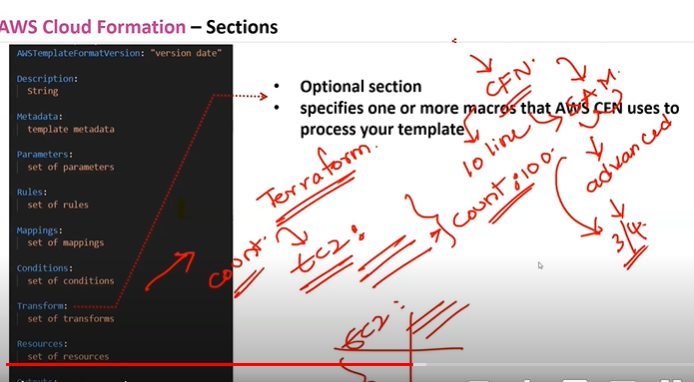
****

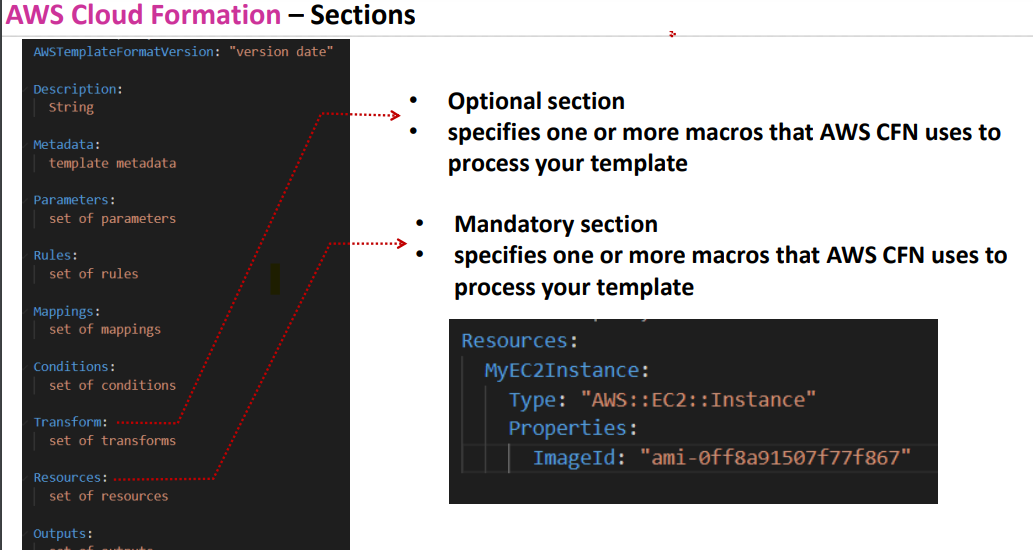
**Rules are optional its only useful for checking conditions or validating the parameters**

**.for ex for login or sign up its asks us to enter the email id and format like gmail.com or .in if its in that format then only it allows if not it throws the error**

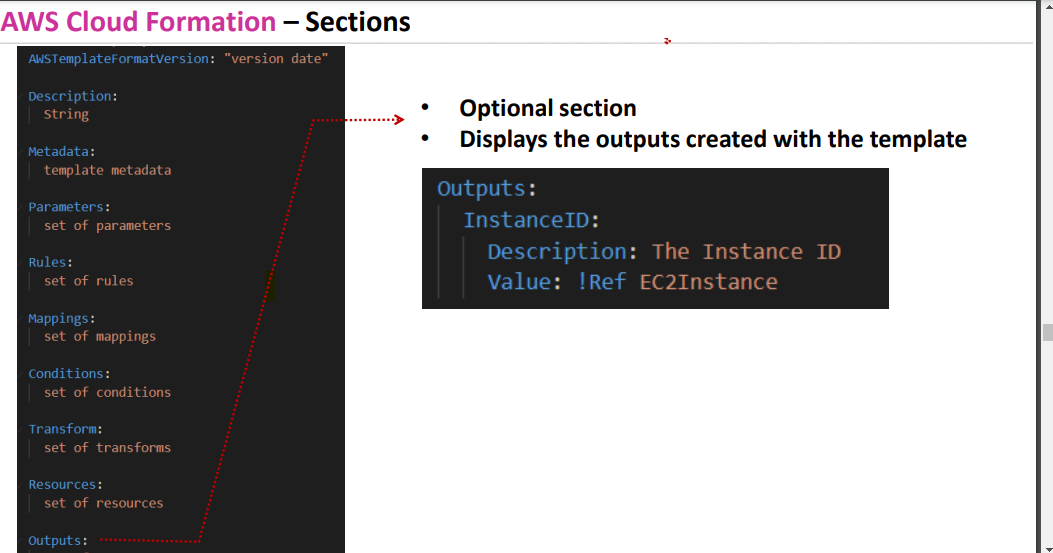
****

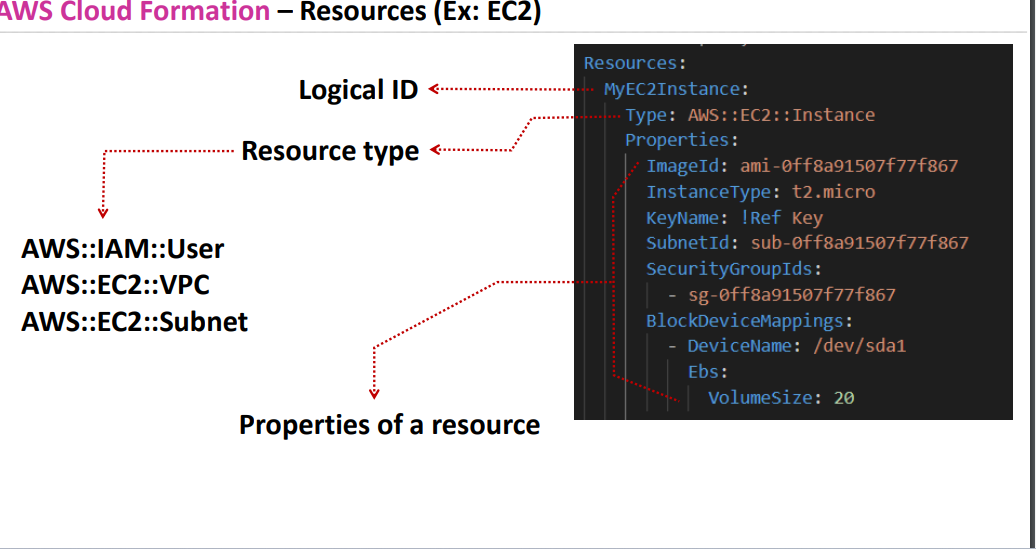
**Mappings ex : If u want to deploy an instance a region based ami id’s gets differed**

* **So in template if you give different id ‘s based on region when you enter the region it will map to that particular ami**
* **Resources, parameter, output**
* ****
* **In Sam advanced version of Cloud formation if cloudformation has 10 lines**
* **SAM ha 3-4 lines of codes**
* **Transform:if u want to create 10 resources using same configuration u will have count option in transform**
* **In terraform also u will have count option**
* **So No to write the code 100 times**

****

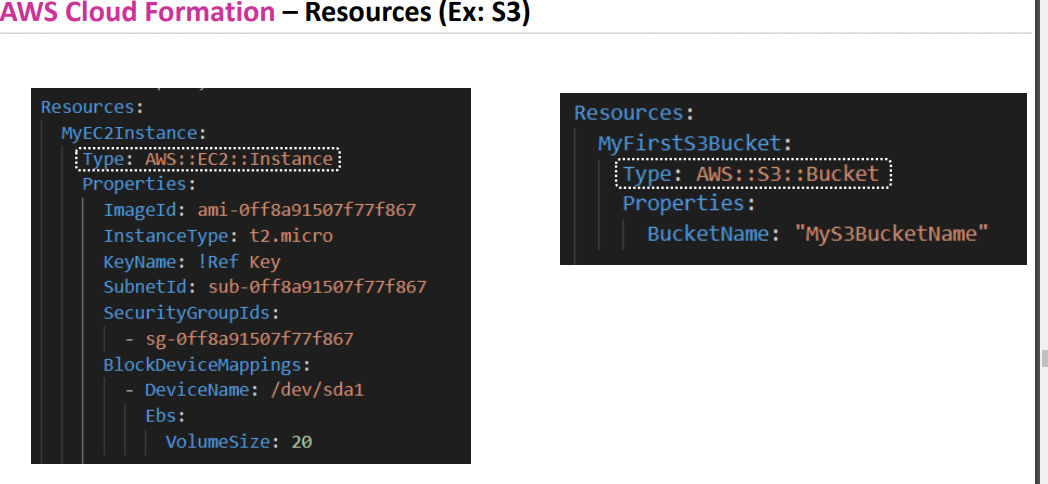
**Resource section is important**

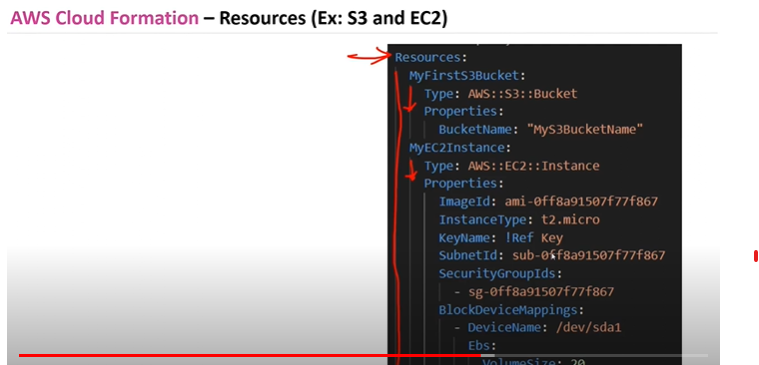
****

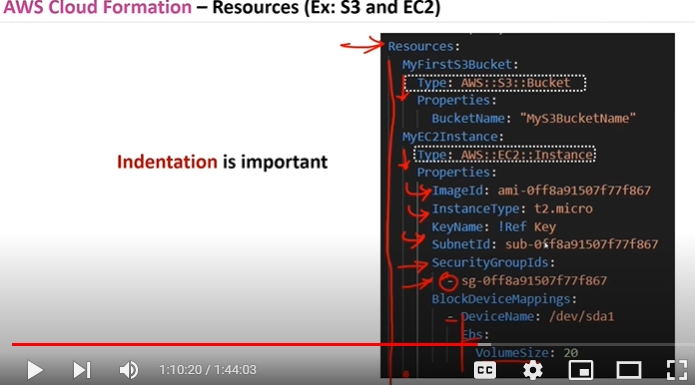
****

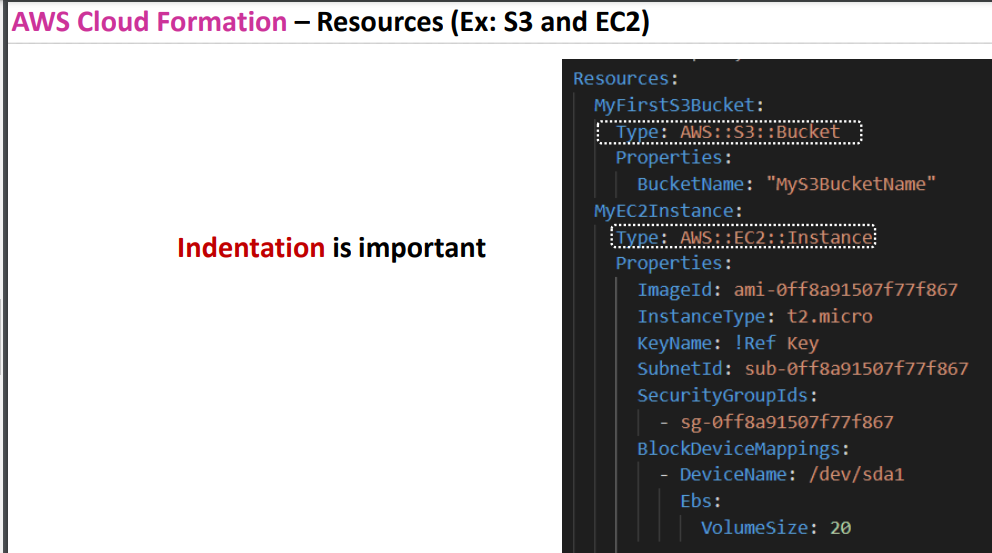
**Type and properties are imp**

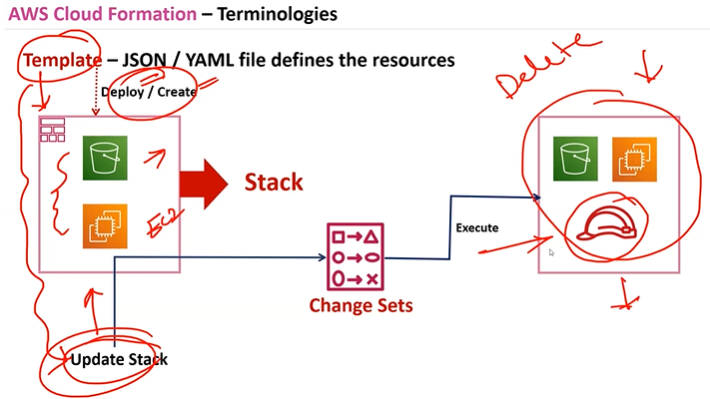
**Logical Id should be named as per ur convenience**

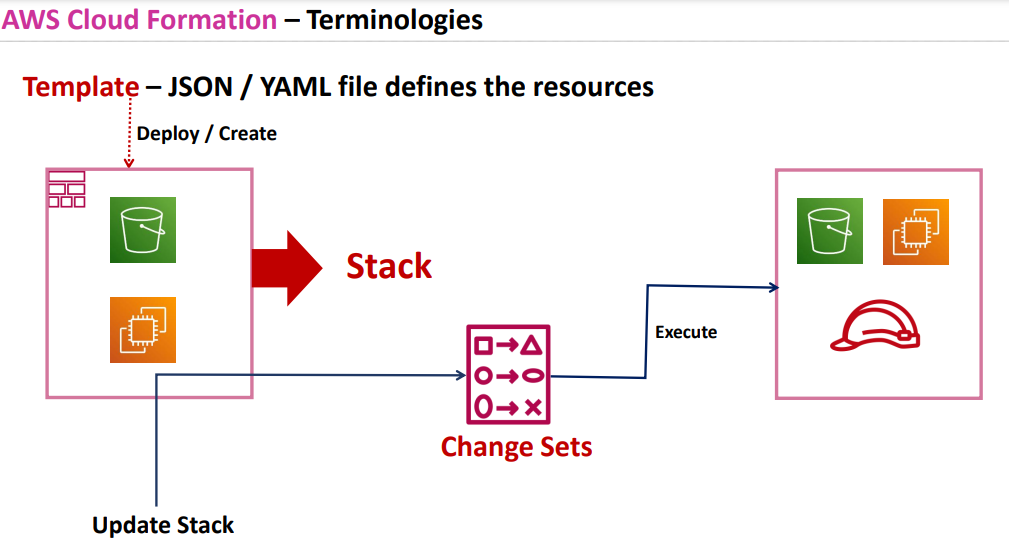
****

****

****

****

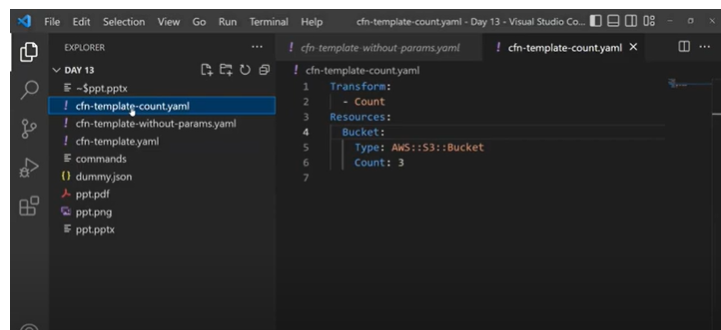
****

****

**From template you will either deploy or create the resources**

**Those are called stacks and when u update the stack that means changes in existing stacks will convert to change sets and then execute**

**Yaml extension should be added in visual studio for beautification**

****

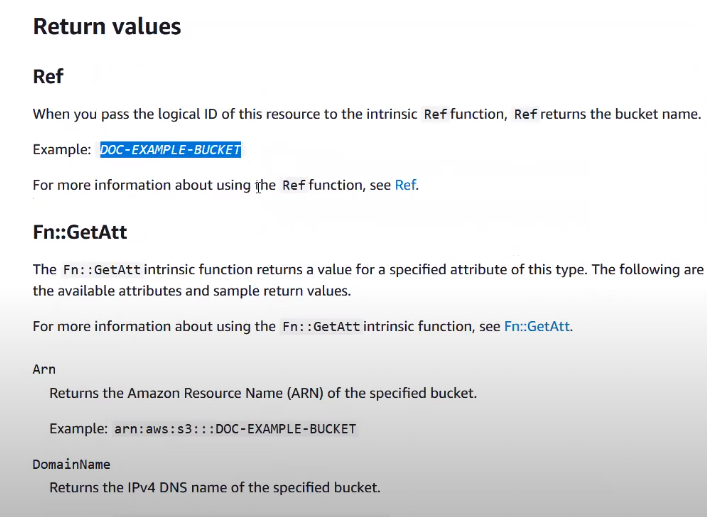
**Cfn-template yaml -with parameter**

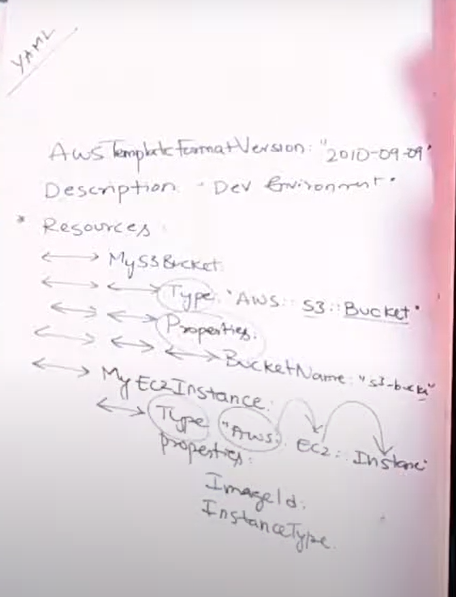
**Cfn-template-count.yaml- using count**

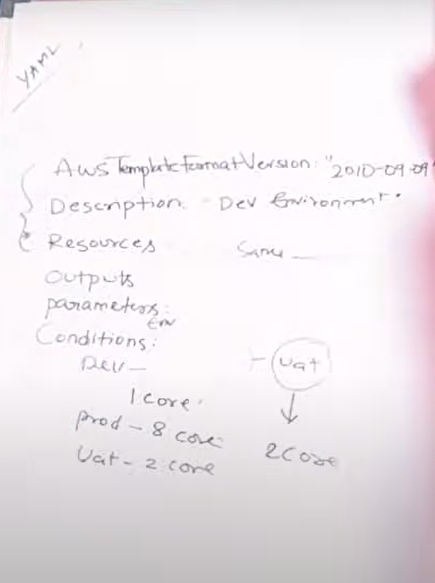
**Cfn template -without parameter.yaml**

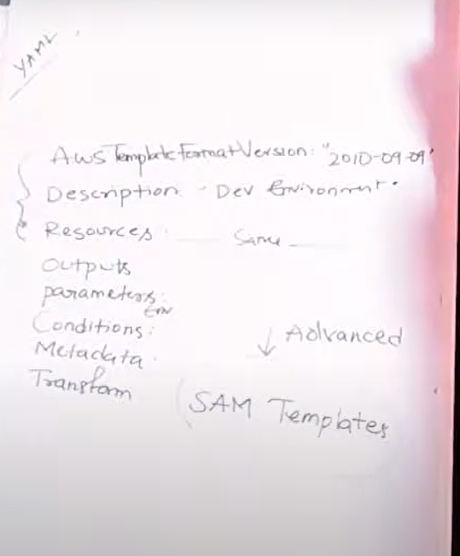
**After creating templates uploading those templates in stack**

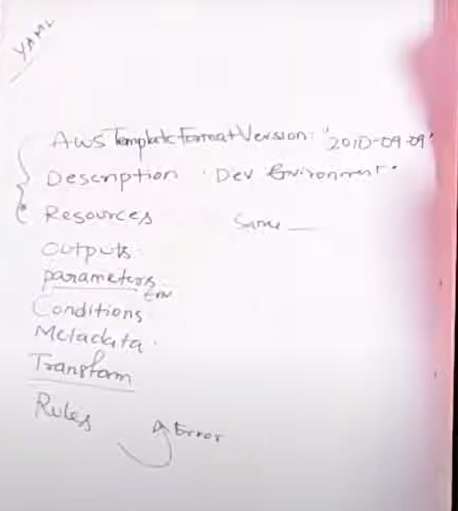
**Aws cloud formation template reference**

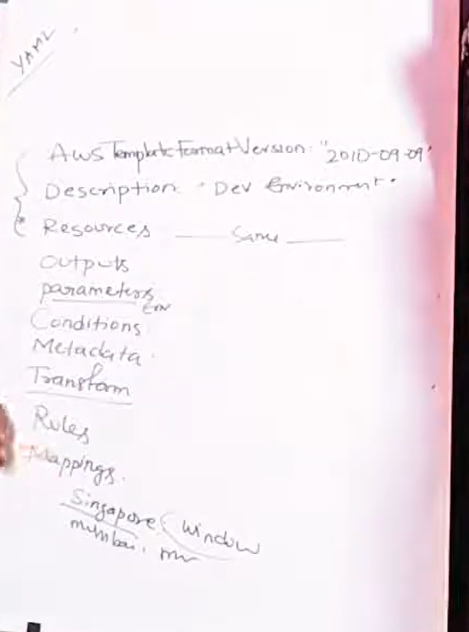
* **In template reference it shows both in yaml format and json format and also it gives info regarding whether that attribute is required or not**
* **In return values u will get reference documents of outputs**
* ****

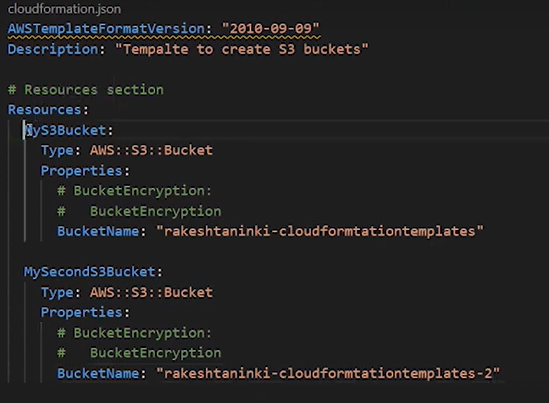
****

****

****

****

****

****