**AWS Cloud Formation Docs |**

Part-1:

**Introduction to CloudFormation**

**What is Aws CloudFormation?**

* AWS CloudFormation is a service that helps you model and set up your resources so that you can spend less time managing those resources and more time focusing on your application that run in AWS.
* You create a template that describe all the AWS resources that you want ( like Amazon EC2 instance or Amazon RDS DB instances), and CloudFormation takes care of provisioning and configuring those resources for you.
* You don’t need to individually create and configure AWS resources and figure out what’s dependent on What: CloudFormation handles that.
* The following scenarios demonstrate how CloudFormation can help.
* Simplify infrastructure management.
* Quickly replicate your infrastructure
* Easily control and track changes to your infrastructure

**Define the AWS CloudFormation concepts?**

* When you use AWS CloudFormation, you work with templates and stack. You create template to describe your AWS resources and their properties.
* Whenever you create a stack, CloudFormation provisions the recourse that are described in your template.

What is AWS CloudFormation Template?

* A CloudFormation template is a JSON, or YAML formatted text file.
* You can save these files with any extension, such as. JSON, YAML, Template, or .txt
* CloudFormation uses these templates as a blueprint for building your AWS resources.
* For example, in a template, you can describe an Amazon EC2 Instances such as the instance type, the AMI ID, Block device mapping, and its amazon EC2 Key pair name.
* Whenever you create a stack , you also specify a template that CloudFormation uses to create whatever you described in the template.
* For example, if you created a stack with following template. CloudFormation provision an instance with an ami-0e42bfd2029a917a4 AMI ID, t2.micro instance type, test key pair name , and an Amazon EBS volume.

**What is AWS CloudFormation Stack?**

* When you use CloudFormation, you manage related resources as a single unit called a stack.
* You create ,update and delete a collection of resources by creating ,updating and deleting stack.
* All the resources in a stack are defined by the stack’s CloudFormation template.
* Suppose you created a template that includes an auto scaling group, Elastic Load Balancing Load and an Amazon Relational Database Services (Amazon RDS ), databases instance.
* To create those resources, you create a stack by submitting the template that you created and CloudFormation provisions all those resources for you.
* You can work with stacks, by using the CloudFormation console ,API, or AWS Cli.

**What is AWS CloudFormation Change sets?**

* If you need to make changes to the running resources in a stack , you update the stack.
* Before making changes to your resources ,you can generate a change set, which is a summary of your proposed changes.
* Change sets allow you to see how your changes might impact your running resources , especially for critical resources, before implementing them.
* For example, if you change the name of an Amazon RDS Database instance, CloudFormation will create a new database and delete the old one.
* You will lose the data in the old database unless you have already backed it up.
* If you generate a change set ,you will see that your change will cause your database to be replaced , and you will be able to plan according to before you update your stack.

Learn More: <https://docs.aws.amazon.com/AWSCloudFormation/latest/UserGuide/Welcome.html>

Part:2 (AWS Cloud Formation with LAB)

Introduction to YAML

What is YAML?

* YAML is a programming Language.
* Basically, it is used to write a configuration file.
* In AWS, it is also used to write a CloudFormation template.

What are the main components or topics used for AWS Cloud Formation?

* YAML Key value pairs.
* YAML Lists
* YAML Dictionary
* YAML list containing dictionaries.
* YAML lists containing.
* YAML Pipe
* YAML greater Than
* YAML comments

What IS YAML Key-Value Pairs?

* YAML documents will be full of key value pairs.
* Key and value are separated by colon.
* We must have a space after colon differentiating the