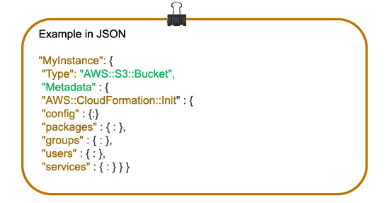
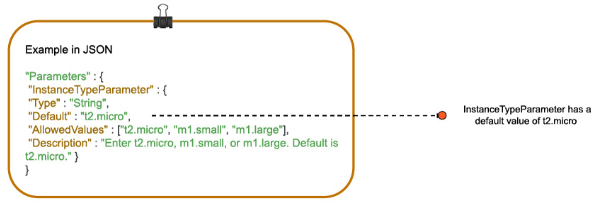
AWS CloudFormation Concepts

An AWS CloudFormation template is a formatted text file in JSON or YAML language that describes your AWS infrastructure. To create, view and modify templates, you can use AWS CloudFormation Designer or any text editor tool. An AWS CloudFormation template consists of below main objects:

1. Format version: Format version defines the capability of a template.
2. Description: Any comments about your template can be specified in the description.
3. Metadata: Metadata can be used in the template to provide further information using JSON or YAML objects.



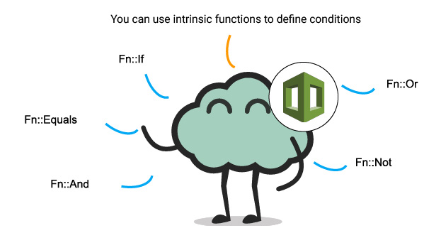
1. **Parameters:** Templates can be customized using parameters. Each time you create or update your stack, parameters help you give your template custom values at runtime.



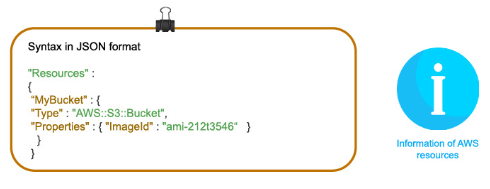
1. **Mappings:** Mapping enables you to map keys to a corresponding named value that you specify in a conditional parameter. Also, you can retrieve values in a map by using the “Fn:: FindInMap” intrinsic function.



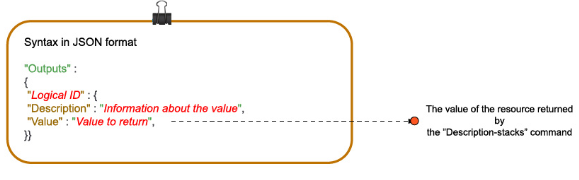
1. **Conditions:**In a template, conditions define whether certain resources are created or when resource properties are assigned to a value during stack creation or updating. Conditions can be used when you want to reuse the templates by creating resources in different contexts. You can use intrinsic functions to define conditions.



1. **Resources:** Using this section, you can declare the AWS resource that you want to create and specify in the stack, such as an Amazon S3 bucket or AWS Lambda.



1. **Output:** In a template, the output section describes the output values that you can import into other stacks or the values that are returned when you view your own stack properties. For example, for an S3 bucket name, you can declare an output and use the “Description-stacks” command from the AWS CloudFormation service to make the bucket name easier to find.



## Stacks in AWS CloudFormation

A collection of AWS resources is called a stack, and it can be managed in a single unit. CloudFormation’s template defines a stack in which the resources can be created, deleted or updated in a predictable way. A stack can have all the resources (web server, database, etc.) required to run a web application.

A [nested stack](https://docs.aws.amazon.com/AWSCloudFormation/latest/UserGuide/using-cfn-nested-stacks.html) results in a hierarchy of stacks. Using the CloudFormation stack resource, you can create a nested stack within another stack.

## CloudFormation Access Control

With IAM, CloudFormation can give users access control and ensure that only IAM users can create, update and delete stacks. On the user’s behalf, a service role allows AWS CloudFormation to make calls to resources in a stack. It is applied to all AWS CloudFormation users who attempt to update the stack. Here, you cannot include different users with different stack policies