cloudformation:

- AWS **CloudFormation** is a service that helps you model and set up your AWS resources so that you can spend less time managing those resources and more time focusing on your applications that run in AWS.
- You create a **template** that describes all the AWS resources that you want (like Amazon EC2 instances or Amazon RDS DB instances), and CloudFormation takes care of provisioning and configuring those resources for you.

Template:

- A template is a declaration of the AWS resources that make up a stack. The template is stored as a text file whose format complies with the JavaScript Object Notation (JSON) or YAML standard.
- text file - YAML or JSON
- Cloudformation read template as an input.
- Dsecribe the state of the Infrastructure in this text files.

Stack:

- A stack is a collection of AWS resources that you can manage as a single unit.
- Cloudformation reads the template and create stack
- We update template and update stack to make changes.

Resources:

- Using this section, you can declare the AWS resource that you want to create and specify in the stack, such as an Amazon EC2, S3 bucket or AWS Lambda.

Example

Syntax in YAML format:

```
Resources:
    MyEC2Instance:
        Type: "AWS::EC2::Instance"
```

Resource type:

- The resource type identifies the type of resource that you are declaring. For example, AWS::EC2::Instance declares an EC2 instance.
- Resource type identifiers always take the following form

service-provider::service-name::data-type-name

Properties:

- Resource properties are additional options that you can specify for a resource.
- For example, for each EC2 instance, you must specify an Amazon Machine Image (AMI) ID for that instance. You declare the AMI ID as a property of the instance, as shown in the following example.

Syntax in YAML format:

```
Properties:
    ImageId: "ami-0ff8a91507f77f867"
    InstanceType: t2.micro
```

Parameters:

- AWS CloudFormation templates can contain parameters. Parameters can be used inside the CloudFormation template to refer to values that are provided at the time the CloudFormation template is used to create a new stack.
- This means that the same CloudFormation template can be used to create multiple stacks that have variations in the places where parameters are used

Parameters:

```
Parameters:
  InstanceTypeParameter:
    Type: String
    Default: t2.micro
```

Userdata:

- AWS userdata is the set of commands/data you can provide to a instance at launch time.
- User data scripts are executed as the root user, so there is no need to use sudo commands in the script.
- UserData must be Base64 encoded when passed from CloudFormation to EC2 instance. Use Fn::Base64 intrinsic function to encode the input string.

```
_UserData:
 Fn::Base64: |
   #!/bin/bash
   yum update -y
   yum install -y httpd php
   systemctl start httpd
   systemctl enable httpd
```