

# AWS CLOUD FORMATION

Infrastructure as a code(IaaC)

# What is AWS CloudFormation?

- AWS CloudFormation is a service that gives developers and businesses an easy way to create a collection of related AWS **resources** and provision them in an orderly and predictable fashion.

# What can developers now do with AWS CloudFormation that they could not before?

- AWS CloudFormation automates and simplifies the task of repeatedly and predictably creating groups of related resources that power your applications. Creating and interconnecting all resources your application needs to run is now as simple as creating a single EC2 or RDS instance.

# What new concepts does AWS CloudFormation introduce?

- AWS CloudFormation introduces two concepts: **The *template***, a JSON or YAML-format, text-based file that describes all the AWS resources you need to deploy to run your application and **the *stack***, the set of AWS resources that are created and managed as a single unit when AWS CloudFormation instantiates a template.
- If all resources is not created properly a **rollback** occurs and all the resources are deleted.
- If a successful stack is deleted, all resources are deleted.

# What are the elements of an AWS CloudFormation template?

- AWS CloudFormation templates are JSON or YAML-formatted text files that are comprised of five types of elements:
- An optional list of template parameters (input values supplied at stack creation time)
- An optional list of output values (e.g. the complete URL to a web application)
- An optional list of data tables used to lookup static configuration values (e.g., AMI names)
- **The list of AWS resources and their configuration values**
- A template file format version number

# Parts of Template Template

```
{
  "AWSTemplateFormatVersion" : "version date",
  "Description" : "Valid JSON strings up to 4K",
  "Parameters" : {
    set of parameters
  },
  "Mappings" : {
    set of mappings
  },
  "Resources" : {
    set of resources
  },
  "Conditions" : {
    set of conditions
  },
  "Outputs" : {
    set of outputs
  }
}
```

# Example Template : creates an instance

---

```
{
  "AWSTemplateFormatVersion": "2010-09-09",
  "Description": "Deploy a simple Amazon Linux Instance and allow SSH connectivity.",
  "Parameters": {
    "KeyName" : {
      "Description" : "EC2 Key Pair for SSH Access, you must have created these prior to running this.",
      "Type" : "String",
      "Default" : "cloudform-key"
    }
  },
  "Resources": {
    "SimpleInstance" : {
      "Type" : "AWS::EC2::Instance",
      "Properties" : {
        "KeyName" : { "Ref" : "KeyName" },
        "InstanceType" : "t2.micro" ,
        "ImageId" : "ami-8c1be5f6",

        "NetworkInterfaces" : [{
          "GroupSet" : [{ "Ref" : "SimpleInstanceSg" }],
          "AssociatePublicIpAddress" : "true",
          "DeviceIndex" : "0",
          "DeleteOnTermination" : "true",
          "SubnetId" : "subnet-a9bae785"
        }]
      }
    },
    "SimpleInstanceSg" : {
      "Type" : "AWS::EC2::SecurityGroup",

      "Properties" : {
        "GroupDescription" : "Enable SSH access via port 22",
        "VpcId": "vpc-7c5f6905",

        "SecurityGroupIngress" : [ {
          "IpProtocol" : "tcp",

          "FromPort" : "22",
          "ToPort" : "22",
          "CidrIp" : "0.0.0.0/0"
        } ]
      }
    }
  }
}
```