AWS CLOUDFORMATION

INFRASTRUCTURE AS CODE

AWS CloudFormation

Template

EC2

```yaml
001-Json-Yaml-Cfn >  !_mytemplate.yaml > YAML > ⊞ AWSTemplateFormatVersion
 1   AWSTemplateFormatVersion: "2010-09-09"
 2   Description: |
 3     This template launches an EC2 with a security group that enables SSH
 4   Resources:
 5     MyEc2:
 6       Type: AWS::EC2::Instance
 7       Properties:
 8         ImageId: ami-0022f774911c1d690
 9         InstanceType: t2.micro
10         KeyName: north-virginia # please use your own key file without extension
11         SecurityGroups:
12           - ssh
13         Tags:
14           - Key: Name
15             Value: First-Cfn-Instance
16   ---
```
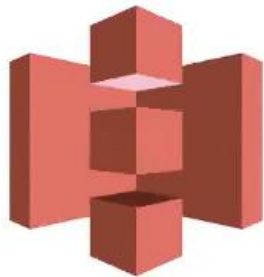
JSON

YAML

Introduction to CloudFormation

What is CloudFormation?

- Create-Delete-Modify one or few AWS resources

  - Easy ✅

- Hundreds ?

Introduction to CloudFormation

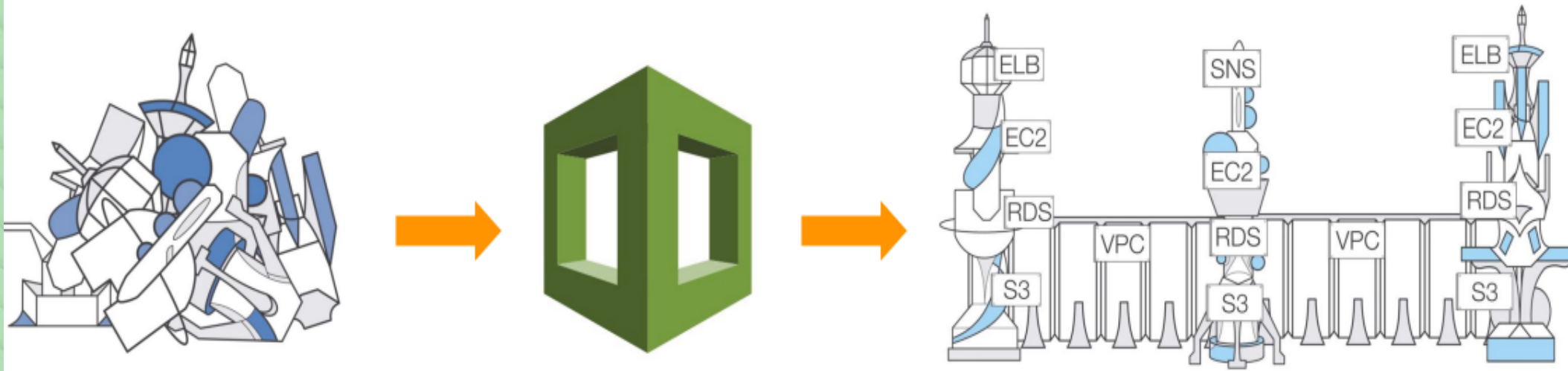## What is CloudFormation?

- CloudFormation is an AWS service which enables you to create, manage, configure, replicate and delete AWS resources easily and rapidly using templates.

Introduction to CloudFormation

What is CloudFormation?

- Provisions and manages stacks of AWS resources based on templates you create to model your infrastructure architecture.

Templates
What is a Template?

```
{
  "AWSTemplateFormatVersion" : "2010-09-09",
  "Description" : "A simple EC2 instance",
  "Resources" : {
    "MyEC2Instance" : {
      "Type" : "AWS::EC2::Instance",
      "Properties" : {
        "ImageId" : "ami-0ff8a91507f77f867",
        "InstanceType" : "t1.micro"
      }
    }
  }
}
```

- A template is a **JSON** or **YAML** formatted text file which you specify the AWS resources you want to create.

Templates
What is a Template?

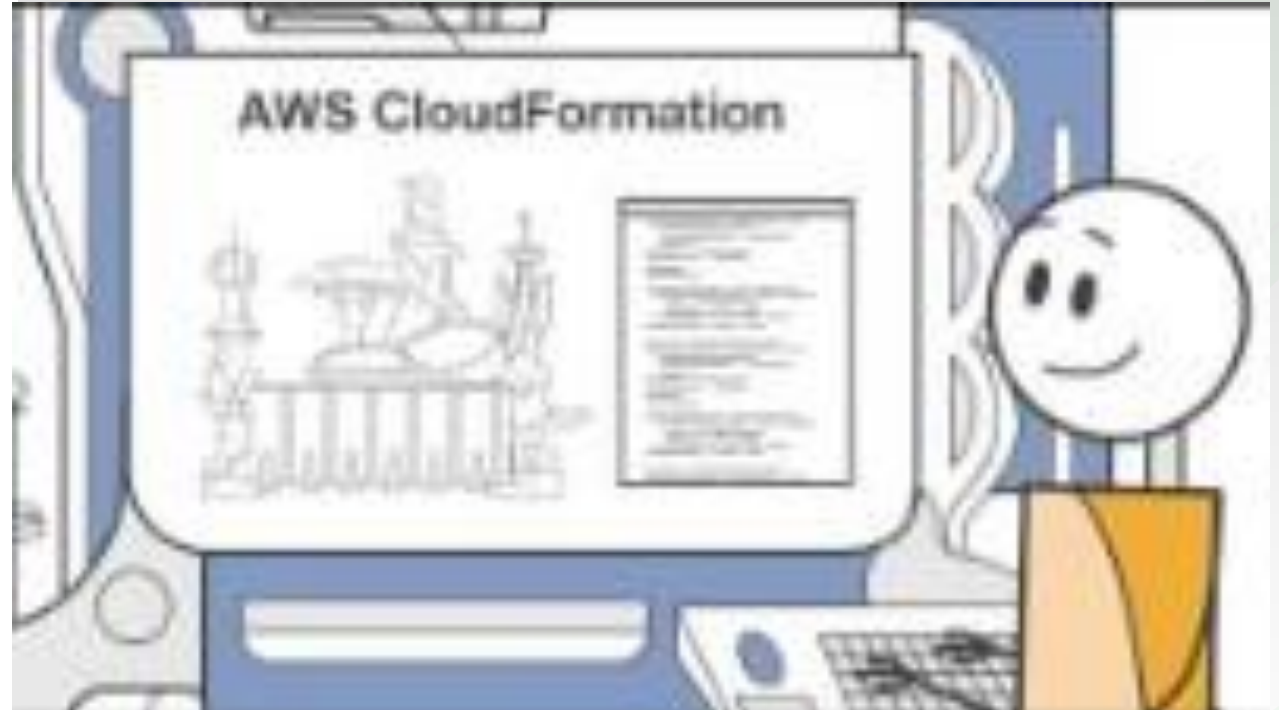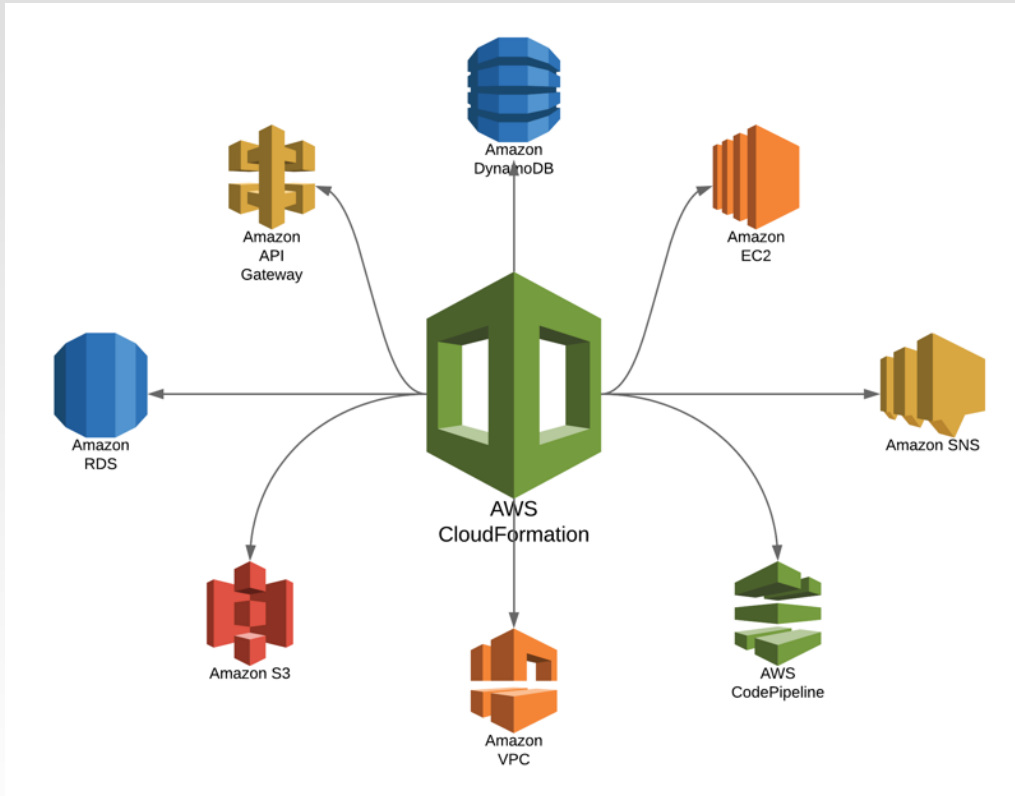- You create resources, define the parameters and configure your settings with templates.

- Templates can have the extensions `.json`, `.yaml`, `.yml`, `.template`, or `.txt`.

What is Cloudformation?

- CloudFormation is an AWS service which enables you to create, manage, configure, replicate and delete AWS resources easily and rapidly using with templates, formatted text files in JSON or YAML.

Stacks and Templates

Template
JSON/YAML formatted file
Parameter definiton
Resource creation
Configuration actions

CloudFormation
Framework
Stack creation
Stack updates
Error detection and rollback

Stack
Configured AWS resources
Comprehensive service support
Service event aware
Customizable
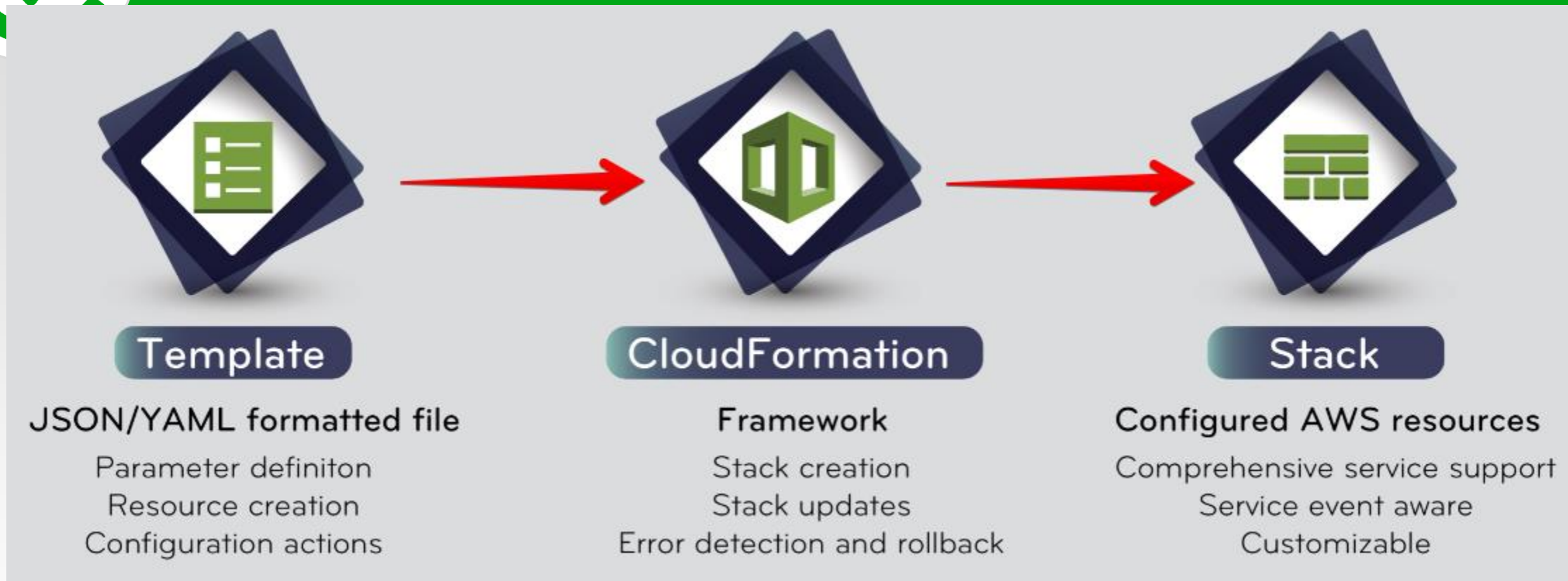
- **Stacks and templates** are the main components of AWS CloudFormation. A template is a JSON or YAML formatted text file which you specify the AWS resources you want to create. Templates can have the extensions .json, .yaml, .template, or .txt.

How Does AWS Cloudformation Work?

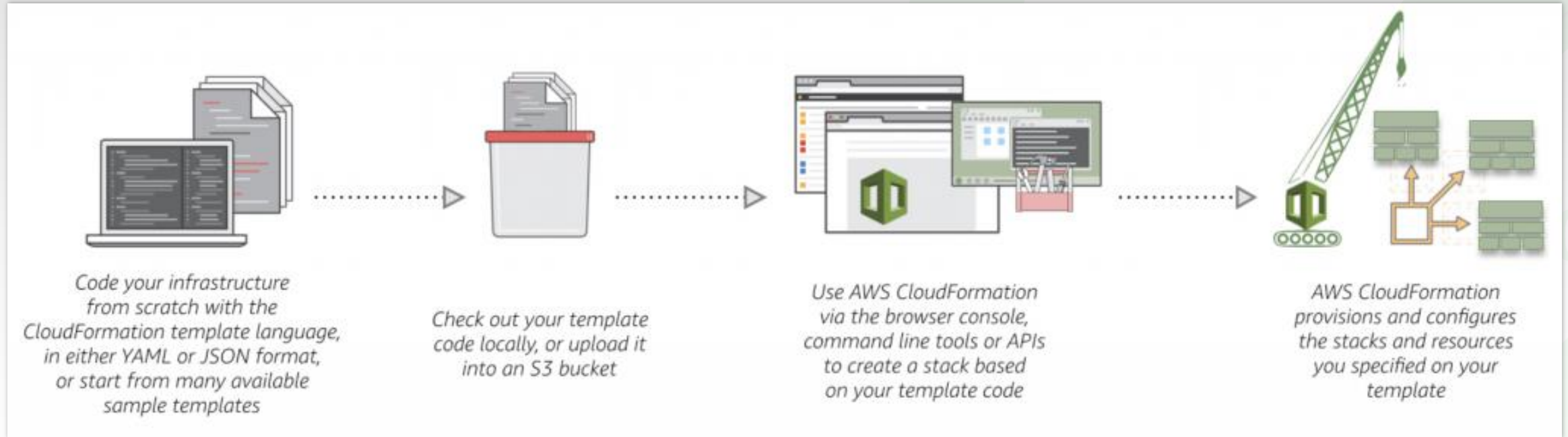Code your infrastructure from scratch with the CloudFormation template language, in either YAML or JSON format, or start from many available sample templates

Check out your template code locally, or upload it into an S3 bucket

Use AWS CloudFormation via the browser console, command line tools or APIs to create a stack based on your template code

AWS CloudFormation provisions and configures the stacks and resources you specified on your template

Accessing and Pricing

Accessing:

> AWS enables you to access CloudFormation via:

AWS Management Console: You can use Management Console to easily access CloudFormation like many other services.

AWS Command Line Interface: You can use also CLI to access CloudFormation. Please check this link if you need more information about AWS CloudFormation CLI.

CloudFormation API: AWS supports accessing CloudFormation via API, if you want to use. Please see AWS CloudFormation API Reference for information about making API requests and API actions.
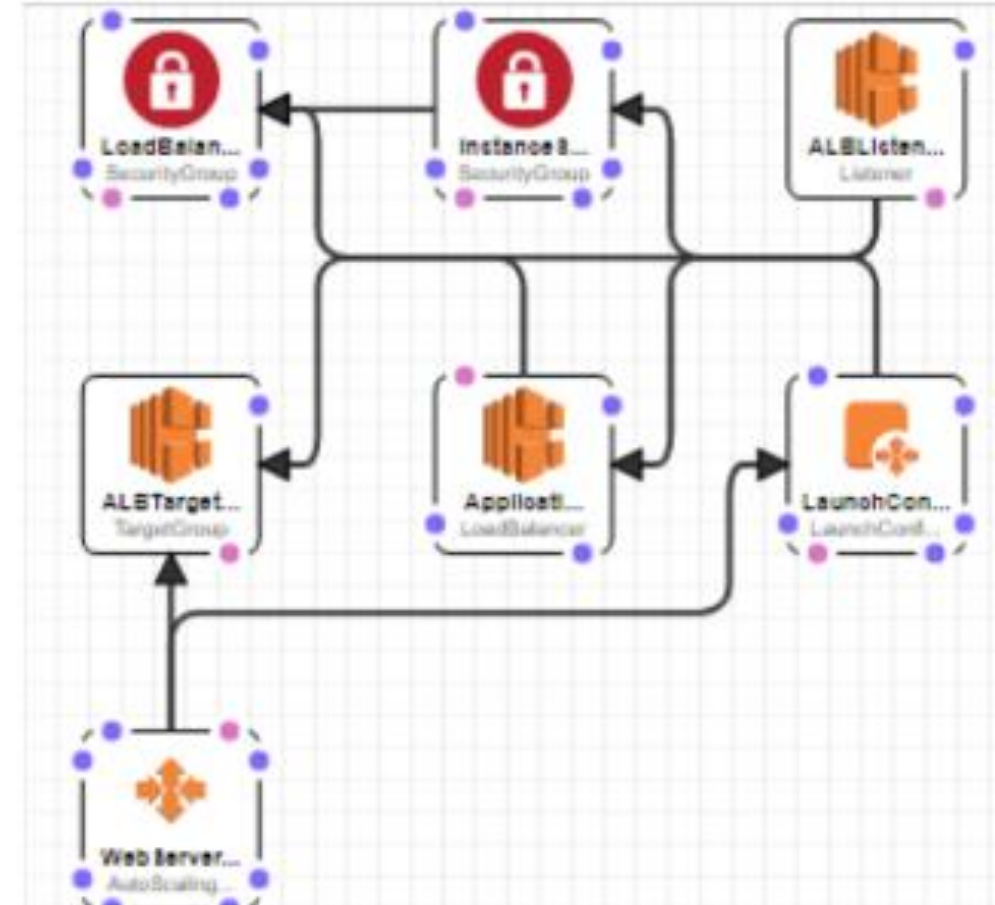
Pricing:

YOU PAY THE RESOURCES YOU USE

What is Stack?

- A Stack is a single unit composed ofo the AWS resources provisioned by Cloudformation:

TEMPLATES

- Template Version
- Description
- Metadata
- Parameters
- Mappings
- Conditions
- Transform
- Resources
- Outputs

YAML

What is YAML ?

```
# What does YAML mean?
YAML:
        - Y: YAML
        - A: Ain't
        - M: Markup
        - L: Language
```

.yml

YAML

What is YAML ?

✅ Its goal was another markup language, but YAML was filling the role of a data serialization language later, so it was called YAML Ain't Markup Language, which is a recursive acronym.

https://yaml.org/

Attributes of YAML

- ☑ YAML is human readable.

- ☑ YAML is portable.

- ☑ YAML works across multiple programming languages easily.

- ☑ YAML matches the native data of structures, to agile languages.

- ☑ YAML is consistent and is able to support generic tools.

- ☑ YAML supports one-pass processing,(when a programming language looks at YAML file, it only needs to go through it once to complete its task.)

- ☑ YAML is expressive easy to extend and adapt.

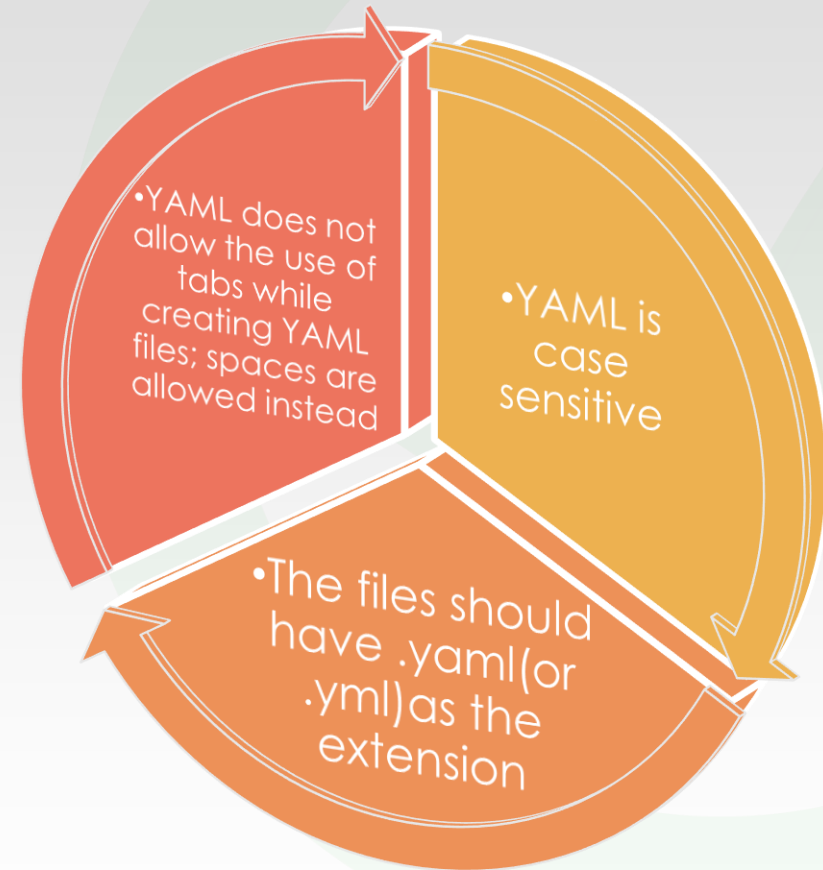- ☑ YAML should be easy to implement and use.

Basics of YAML

> When you are creating a file in YAML, you should remember the following basic rules

YAML does not allow the use of tabs. Spaces are used instead of tabs because tabs are not universally supported.

https://yaml.org/

- YAML does not allow the use of tabs while creating YAML files; spaces are allowed instead

- YAML is case sensitive

- The files should have .yaml(or .yml)as the extension

Basics of YAML

✓ The key-value is YAML's basic building block. The key is always a string. The value is a scalar so that it can be any data type.

https://yaml.org/

```
--- !clarkevans.com/^invoice
invoice: 34843
date   : 2001-01-23
bill-to: &id001
    given  : Chris
    family : Dumars
    address:
        lines: |
            458 Walkman Dr.
            Suite #292
        city    : Royal Oak
        state   : MI
        postal  : 48046
ship-to: *id001
product:
    - sku         : BL394D
      quantity    : 4
      description : Basketball
      price       : 450.00
    - sku         : BL4438H
      quantity    : 1
      description : Super Hoop
      price       : 2392.00
tax  : 251.42
total: 4443.52
comments: >
    Late afternoon is best.
    Backup contact is Nancy
    Billsmer @ 338-4338.
```

SCALAR

COLLECTIONS

MULTI-LINE COLLECTIONS

LISTS/DICTIONARIES

MULTI-LINE FORMATTING

JSON

What is JSON?

> **J**ava**S**cript **O**bject **N**otation

YAML

```
car:
    color: blue
    price: $30,000
```

JSON

```
{
    "car": {
        "color": "blue",
        "price": "$30,000"
    }
}
```

Color
Blue

Price
$30,000

JSON vs YAML

Data Serialization Languages

JSON	YAML	XML
JavaScript Object Notation	YAML Ain't Markup Language	eXtensible Markup Language
Data Interchange	Data Interchange	Markup Language
2002	2006	1996
Easy to read	Easier to read	A little complex
Fast	Fast	Slow
Map Structure	Map Structure	Tree Structure
.json	.yaml	.xml

- ☑ JSON Syntax:JSON is a subset of the JavaScript object notation syntax.
- ☑ JSON data stored in **name/value pairs**.
- ☑ JSON records separated by **commas.**
- ☑ JSON field names & strings are wrapped by **double quotes**.


```xml
<Servers>
    <Server>
        <name>Server1</name>
        <owner>John</owner>
        <created>12232012</created>
        <status>active</status>
    </Server>
</Servers>
```

```json
{
    Servers: [
        {
            name: Server1,
            owner: John,
            created: 12232012,
            status: active,
        }
    ]
}
```

```yaml
Servers:
    -    name: Server1
         owner: John
         created: 12232012
         status: active
```

- ☑ YAML Syntax:YAML stands for ain't markup language and is a superset of JSON – You Convert YAML to JSON
- ☑ **YAML files begin with '—',** marking the start of the document.
- ☑ YAML documents end with '…' but it's optional.
- ☑ YAML key value pairs are separated by **colon**.
- ☑ YAML lists begin with a hyphen.
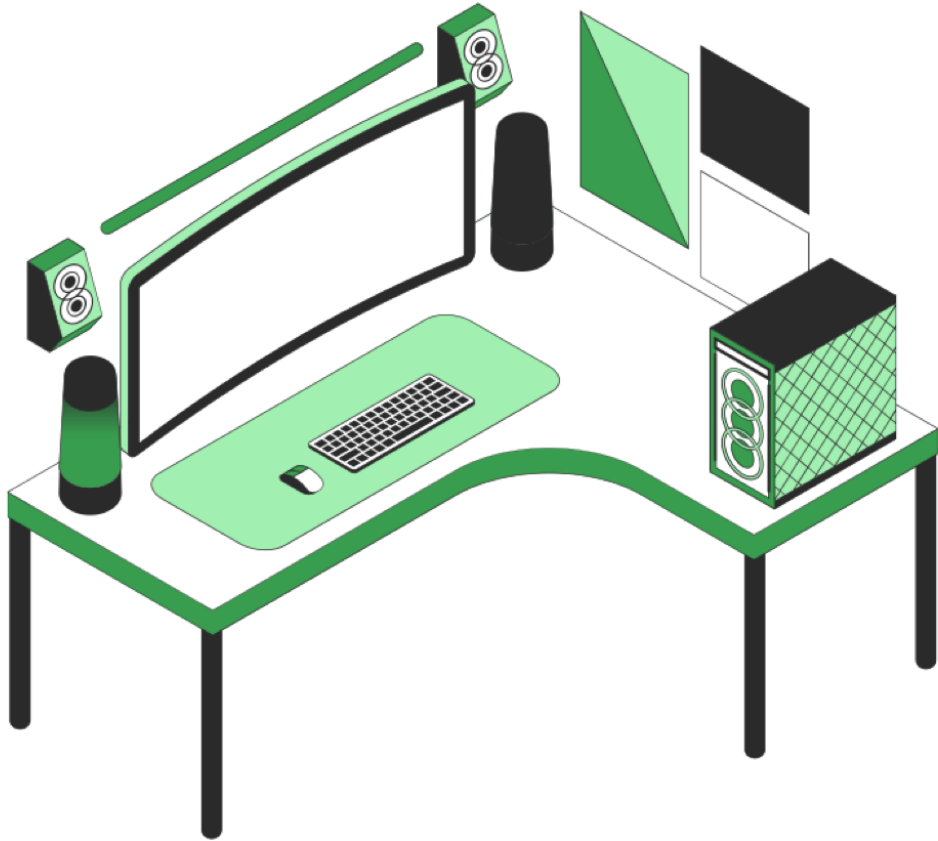
Do you have any questions?

Send it to us! We hope you learned something new.

TECHPROED