Cheatography

AWS CloudFormation Cheat Sheet

by forrestbrazeal via cheatography.com/62457/cs/16025/

Template Anatomy

Resources	Mappings
Description	Conditions
Metadata	Transform
Parameters	Outputs

A `Resources` section is required.

Intrinsic Functions Syntax

YAML Shorthand	`! Ref arg`
YAML	`Fn::Ref arg`
JSON	`{ " Fn: :Re f": arg }`

Common Intrinsic Functions

`!FindInMap [ Map, TopLev elKey, Second Lev elKe y ]`	Returns values of keys in 2-level map declared in Mappings section
`!GettAtt a.Arn`	Get `Arn` attribute of resource `a` in this stack
`!Impor tValue a`	Reference export `a` from another stack
`!Join [':',[ 'a' ,'b']]`	Produces `'a :b'`
`!Ref a`	Get value of parameter or resource `a` in this stack
`!Select ['1',[ 'a' ,'b']]`	Produces `'b '`
`!Split [ ':', 'a:b' ]`	Produces `[' a', 'b']`
`!Sub 'a-${b}'`	Inject the value of `b` into a string

You can't nest the shorthand YAML functions. You must do:
`Fn::Im por tValue: !Sub " ${a }-b "`
not `!Impor tValue !Sub " ${a }-b "`

Transforms

Common Pseudo Parameters

AWS::AccountId	12-digit AWS account
AWS::NoValue	Use in conditionals
AWS::Region	Deployment region
AWS::StackId	ARN of the current stack
AWS::StackName	Name of the current stack

Reference pseudo parameters just like regular parameters, e.g. `!Re f AWS::R egion`

Condition Functions

`!Equals ['a', 'b']`	false
`!And [!Equals ['a', 'a'], !Equals ['a', 'b']]`	false
`!Or [!Equals ['a', 'a'], !Equals ['a', 'b']]`	true
`!Not [!Equals ['a', 'a']]`	false
`!If [condi tion, 100, 10]`	100 if `itio` true, 10

Export Outputs

```
Outputs:
    MyV PCO utput:
        Value: !Ref VPCRes ource
        Export:
            Name: !Sub " ${A WS: :St ack Nam e}- -
VPC_ID"
```

- For each AWS account, Export names must be unique within a region.
- You can't create cross-stack references across regions.
- You can't delete a stack if another stack references one of its outputs.

```
'Fn::Transform':
   - Name: 'AWS:: Inc lude'
        Par ame ters:
            Loc ation: s3://b uck et/ sni ppe t.yml
```

Use an Include transform to reference a template snippet stored separately from the main CloudFormation template.

Custom Resources

Use the `AWS::C lou dFo rma tio n:: Cus tom Res ource` or `Custom ::S tring` resource type to define custom resources.
To create a custom resource, you need:
- **A template that includes a custom resource type.**
- **A custom resource provider** with a service token that the template developer uses.
During a stack operation, CloudFormation sends a request to a service token specified in the template, then waits for a response before proceeding.

Sponsored by **ApolloPad.com**
Everyone has a novel in them. Finish Yours!
https://apollopad.com