Using Expressions and Functions

Ned Bellavance

HashiCorp Certified Instructor

nedinthecloud.com

Potential Improvements

Default tags and naming convention

Move startup script to a file

Make public DNS a full URL

Local Values

Local Values

Internal temporary values

Replace repeated values

Data transformation

Locals Syntax

main.tf

```
locals {
  key = value
}
```

Locals Syntax

main.tf

```
locals {
    instance_prefix = "globo"
    common_tags = {
        company      = "Globomantics"
        project      = var.project
        billing_code = var.billing_code
    }
}
```

```
locals {
  instance_prefix = "globo"
  common_tags = {
    company     = "Globomantics"
    project     = var.project
    billing_code = var.billing_code
  }
}
```

Terraform Locals Reference

```
local.<key>
local.instance_prefix
local.common_tags.company
```

Common Tags

Company (Globomantics)

Project

Environment

BillingCode

Variables to Create

Company

Project

Environment
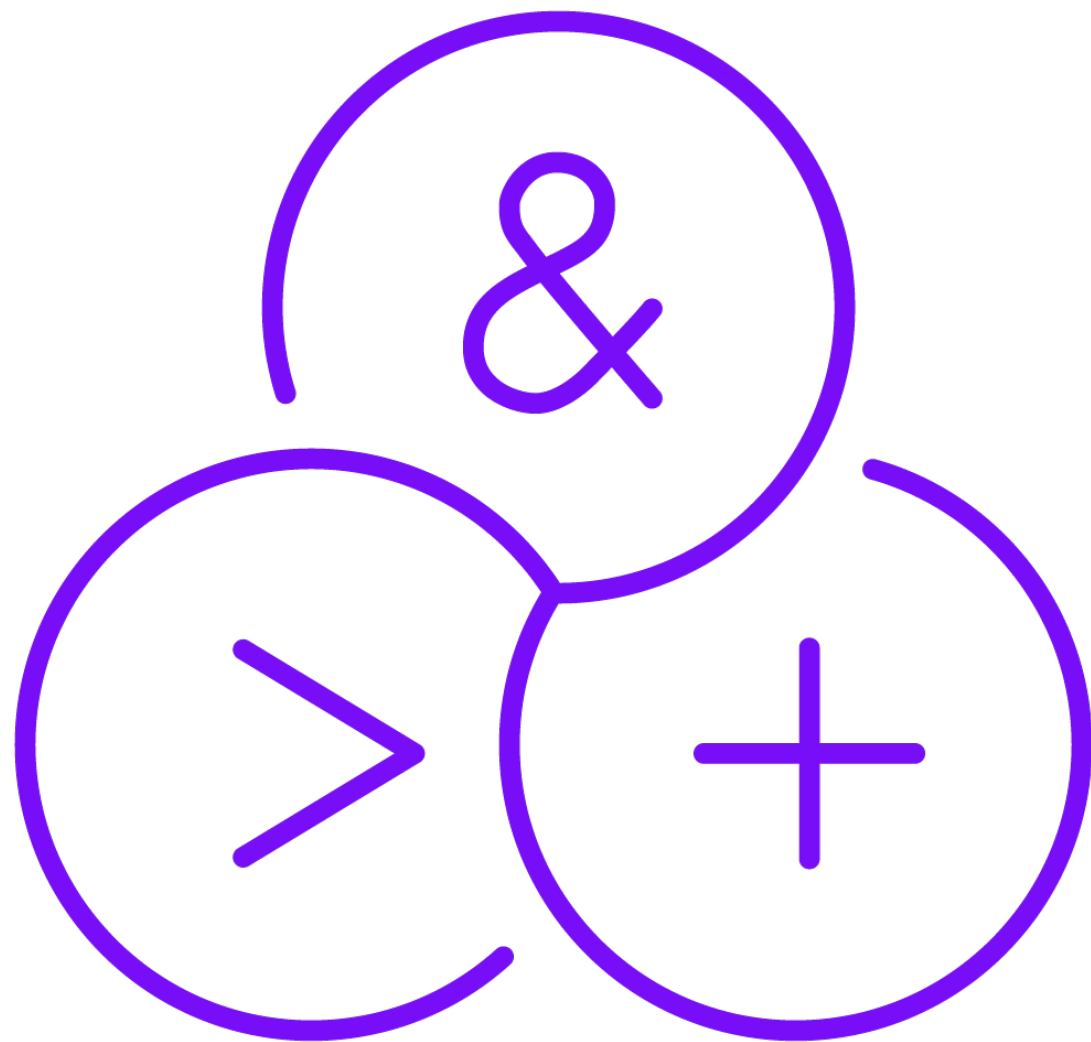
Billing code

Functions and Expressions

Terraform Expressions

Literal expressions

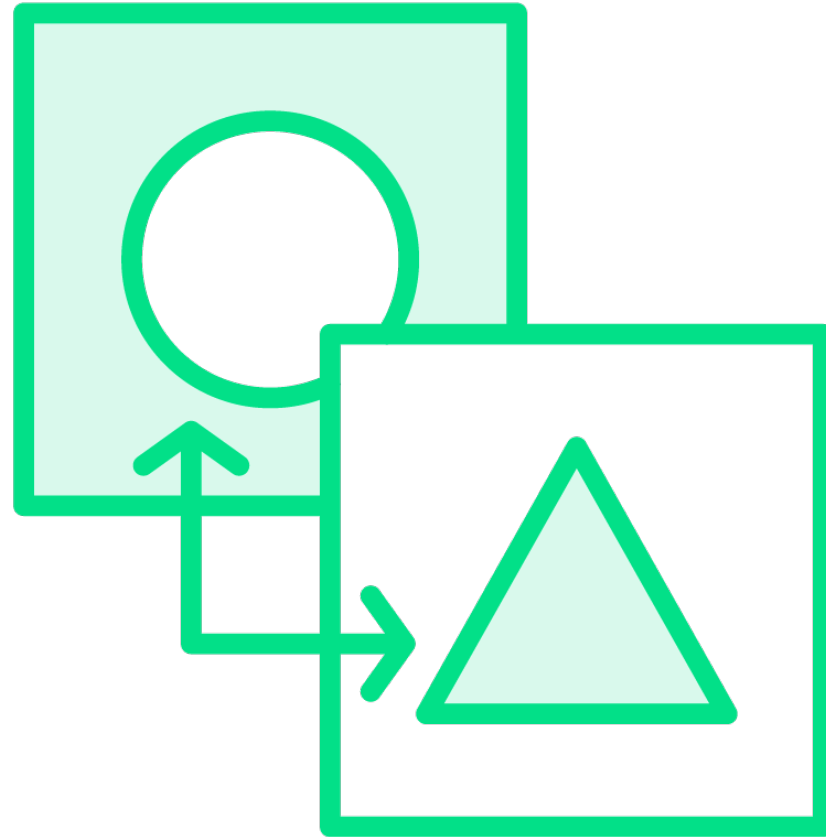Object or attribute references

Arithmetic and logical operators

Conditional expressions

For expressions

Terraform Functions

Built-in to Terraform

`func_name(arg1, arg2, arg3, …)`

Grouped by category

Functions to Use

```
# Startup script

templatefile(file_path, { map of variables })

# Consistent naming

lower(local.naming_prefix)

# Add tags to common tags

merge(local.common_tags, { map of additional tags })
```