



Terraform: HCL

Terraform : Deployment Automation

- Terraform uses its own configuration language, designed to allow concise descriptions of infrastructure.
- The main purpose of the Terraform language is declaring **resources**.
- A group of resources can be gathered into a **module**, which creates a larger unit of configuration.

Terraform : Deployment Automation

- The syntax of the Terraform language consists of only a few basic elements:
Arguments, Blocks, and Expressions

```
resource "aws_vpc" "main" {  
  cidr_block = var.base_cidr_block  
}
```

```
<BLOCK TYPE> "<BLOCK LABEL>" "<BLOCK LABEL>" {  
  # Block body  
  <IDENTIFIER> = <EXPRESSION> # Argument  
}
```

- *Blocks* are containers for other content and usually represent the configuration of some kind of object, like a resource. Blocks have a *block type*, can have zero or more *labels*, and have a *body* that contains any number of arguments and nested blocks. Most of Terraform's features are controlled by top-level blocks in a configuration file.

Terraform : Deployment Automation

- *Arguments* assign a value to a name. They appear within blocks.
- *Expressions* represent a value, either literally or by referencing and combining other values. They appear as values for arguments, or within other expressions.

Will see you in Next Lecture...

Thank you!

A close-up photograph of a hand holding a black marker, writing the words 'Thank you!' in a cursive script on a white surface. The hand is positioned on the right side of the frame, with the fingers gripping the marker. The text is written in a fluid, handwritten style.

See you in next lecture ...