

Terraform Data Types

1. Primitive Types

a) string

Text value.

Example:

```
variable "name" {  
  type = string  
  default = "Anuja"  
}
```

b) number

Any numeric value.

Example:

```
variable "instance_count" {  
  type = number  
  default = 2  
}
```

c) bool

Boolean true/false.

Example:

```
variable "enable_monitoring" {  
  type = bool  
  default = true  
}
```

2. Complex Types

d) list(type)

Ordered same-type values.

Example:

```
variable "subnets" {
  type = list(string)
  default = ["subnet-1", "subnet-2", "subnet-3"]
}
```

e) map(type)

Key-value pairs of same type.

Example:

```
variable "tags" {
  type = map(string)
  default = {
    Name = "EC2"
    Environment = "dev"
  }
}
```

f) set(type)

Unique unordered values.

Example:

```
variable "ports" {
  type = set(number)
  default = [22, 80, 443]
}
```

3. Structural Types

g) object({})

Custom structured type.

Example:

```
variable "instance_config" {
  type = object({
```

```
instance_type = string
ami = string
volume_size = number
})

default = {
    instance_type = "t2.micro"
    ami = "ami-123456"
    volume_size = 20
}
}
```

h) tuple([])

Ordered values of different types.

Example:

```
variable "tuple_example" {
    type = tuple([string, number, bool])
    default = ["Anuja", 25, true]
}
```