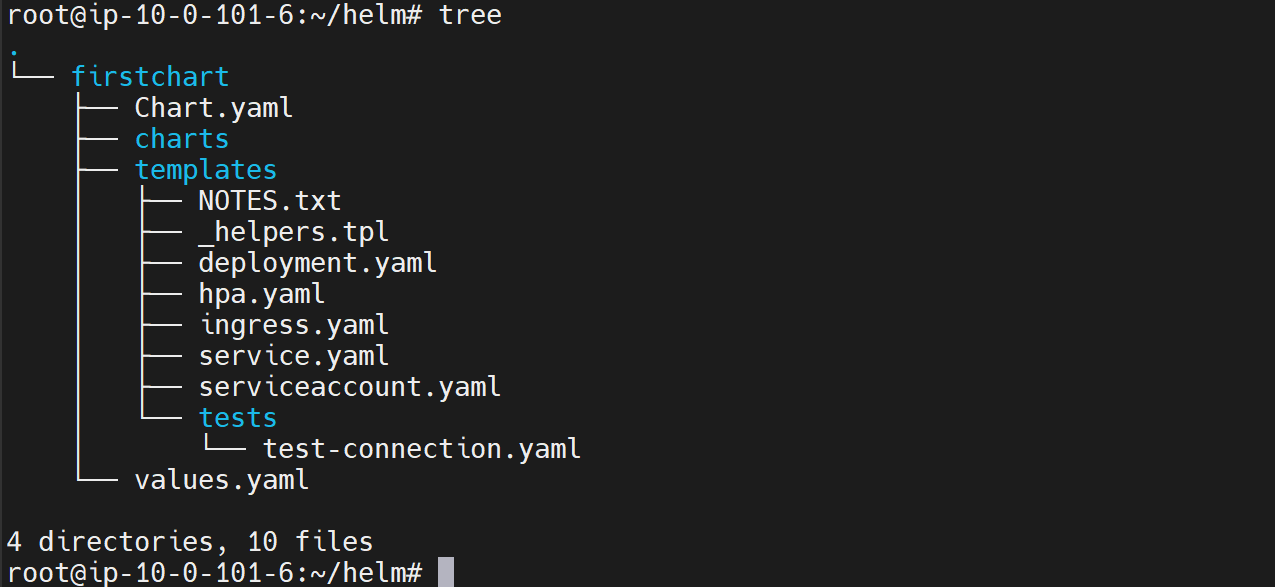
**50. Use Loops**

--- you will learn how to use loops in your template YAML files.

--- tree



--- cat values.yml

# Default values for firstchart.

# This is a YAML-formatted file.

# Declare variables to be passed into your templates.

my:

  flag: true

  values:

    - usq

    - india

    - mexico

    - brazil

    - uk

replicaCount: 1

image:

  repository: nginx

  pullPolicy: IfNotPresent

  # Overrides the image tag whose default is the chart appVersion.

  tag: ""

imagePullSecrets: []

nameOverride: ""

fullnameOverride: ""

--- cat deployment.yml

apiVersion: apps/v1

kind: Deployment

 {{ $myFLAG := .Values.my.flag}}

 {{- if $myFLAG }}

 {{"output of if" | nindent 2}}

 {{- else}}

 {{"output of else" | nindent 2}}

 {{- end}}

 {{- with .Values.my.values}}

 {{- toYaml . | nindent 2}}

 {{- end}}

**{{- range .Values.my.values}}**

**-{{ . | upper | quote | }}**

**{{- end}}**

spec:

  {{- if not .Values.autoscaling.enabled }}

  replicas: {{ .Values.replicaCount }}

  {{- end }}

--- **range** - Range is the looping construct in go templating. It is very similar to the python range. range will go through the list of values that we are providing here. So here this values element has this list. It will loop through that and give us one value at a time.

--- **. (dot)** - DOT represents the current value that is being written on that dot

**# Check the template of firstchart.**

--- helm template firstchart

# Source: firstchart/templates/deployment.yaml

apiVersion: apps/v1

kind: Deployment

  output of if

  - usq

  - india

  - mexico

  - brazil

  - uk

**-"USQ"**

**-"INDIA"**

**-"MEXICO"**

**-"BRAZIL"**

**-"UK"**

spec:

  replicas: 1

  selector:

    matchLabels:

      app.kubernetes.io/name: firstchart

      app.kubernetes.io/instance: release-name