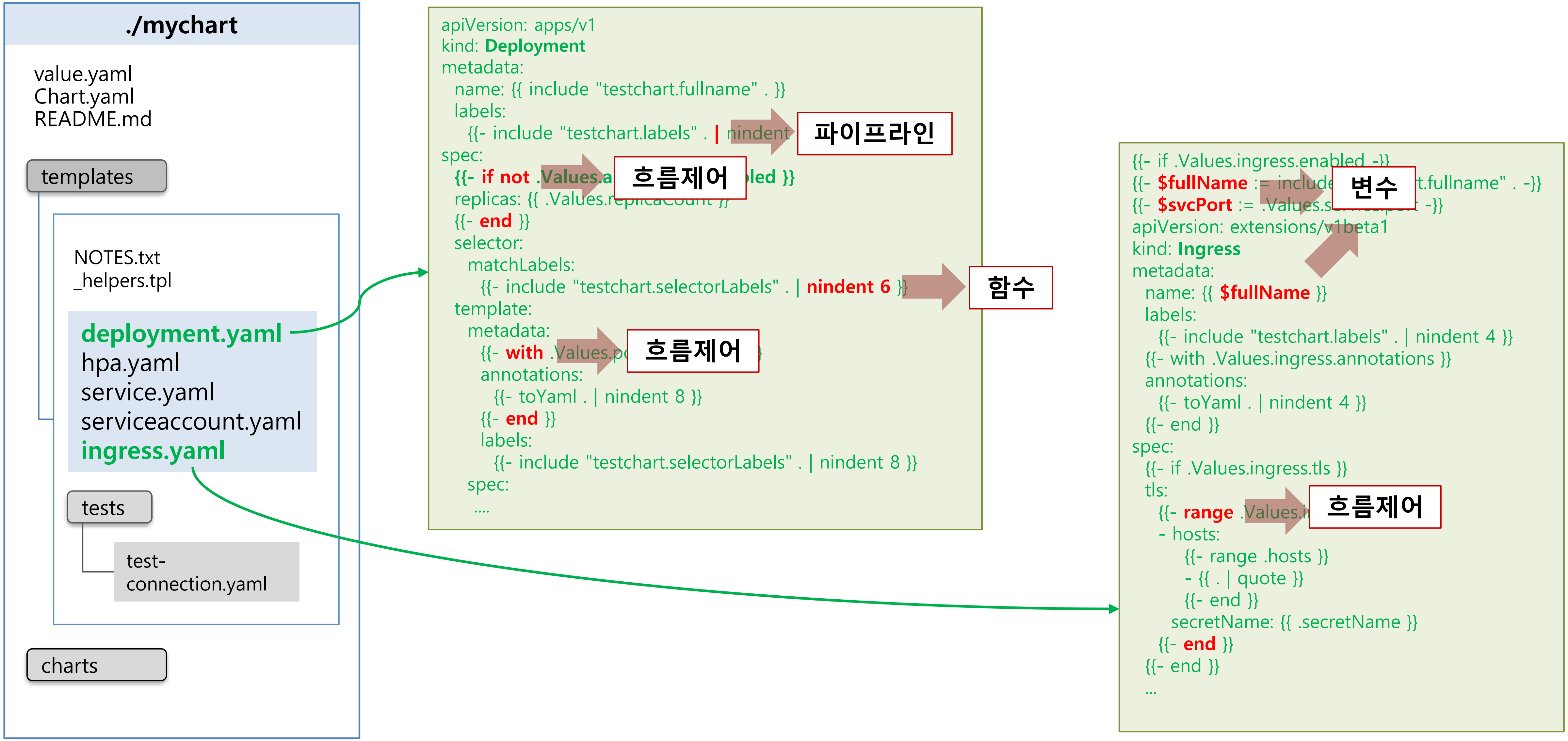
- 내 차트 만들기2 - template 만들기



./mychart

value.yaml
Chart.yaml
README.md

templates

NOTES.txt
_helpers.tpl

deployment.yaml
hpa.yaml
service.yaml
serviceaccount.yaml
ingress.yaml

tests

test-connection.yaml

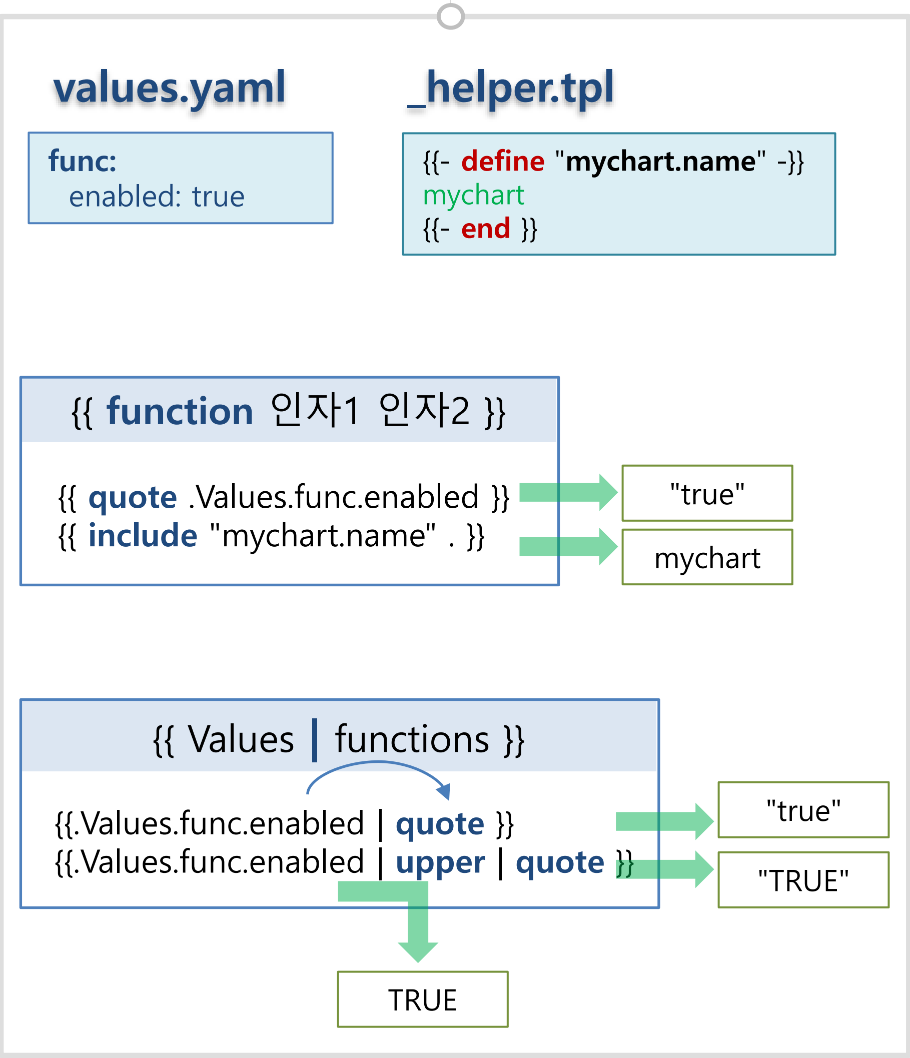
charts

```
apiVersion: apps/v1
kind: Deployment
metadata:
  name: {{ include "testchart.fullname" . }}
  labels:
    {{- include "testchart.labels" . | nindent
spec:
  {{- if not .Values.a                bled }}
  replicas: {{ .Values.replicaCount }}
  {{- end }}
  selector:
    matchLabels:
      {{- include "testchart.selectorLabels" . | nindent 6 }}
  template:
    metadata:
      {{- with .Values.p                 }
      annotations:
        {{- toYaml . | nindent 8 }}
      {{- end }}
      labels:
        {{- include "testchart.selectorLabels" . | nindent 8 }}
    spec:
     ....
```

```
{{- if .Values.ingress.enabled -}}
{{- $fullName := include            t.fullname" . -}}
{{- $svcPort := .Values.service.port -}}
apiVersion: extensions/v1beta1
kind: Ingress
metadata:
  name: {{ $fullName }}
  labels:
    {{- include "testchart.labels" . | nindent 4 }}
  {{- with .Values.ingress.annotations }}
  annotations:
    {{- toYaml . | nindent 4 }}
  {{- end }}
spec:
  {{- if .Values.ingress.tls }}
  tls:
    {{- range .Values.i
    - hosts:
        {{- range .hosts }}
        - {{ . | quote }}
        {{- end }}
      secretName: {{ .secretName }}
    {{- end }}
  {{- end }}
  ...
```

- 내 차트 만들기2 - 함수와 파이프라인 그리고 흐름제어

function & pipeline



values.yaml

```
func:
  enabled: true
```

_helper.tpl

```
{{- define "mychart.name" -}}
mychart
{{- end }}
```

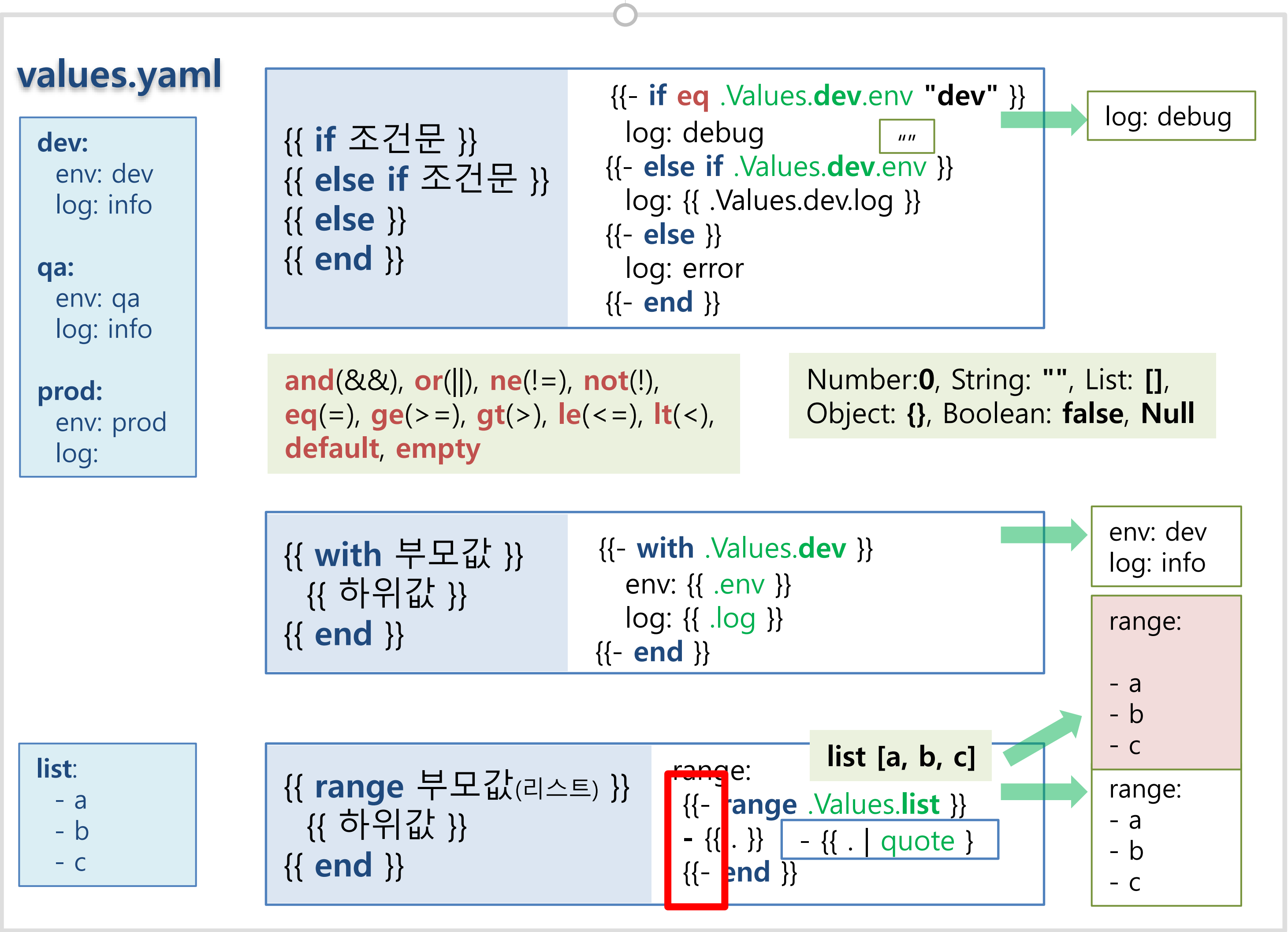
{{ function 인자1 인자2 }}

```
{{ quote .Values.func.enabled }}      → "true"
{{ include "mychart.name" . }}        → mychart
```

{{ Values | functions }}

```
{{.Values.func.enabled | quote }}             → "true"
{{.Values.func.enabled | upper | quote }}     → "TRUE"
                                              → TRUE
```

flow controll



values.yaml

```
dev:
  env: dev
  log: info

qa:
  env: qa
  log: info

prod:
  env: prod
  log:
```

```
list:
  - a
  - b
  - c
```

{{ if 조건문 }}
{{ else if 조건문 }}
{{ else }}
{{ end }}

```
{{- if eq .Values.dev.env "dev" }}
  log: debug                              ""
{{- else if .Values.dev.env }}
  log: {{ .Values.dev.log }}
{{- else }}
  log: error
{{- end }}
```

→ log: debug

and(&&), or(||), ne(!=), not(!), eq(=), ge(>=), gt(>), le(<=), lt(<), default, empty

Number:0, String: "", List: [], Object: {}, Boolean: false, Null

{{ with 부모값 }}
{{ 하위값 }}
{{ end }}

```
{{- with .Values.dev }}
  env: {{ .env }}
  log: {{ .log }}
{{- end }}
```

```
env: dev
log: info
```

{{ range 부모값(리스트) }}
{{ 하위값 }}
{{ end }}

list [a, b, c]

```
range:
{{- range .Values.list }}
- {{ . }}        - {{ . | quote }
{{- end }}
```

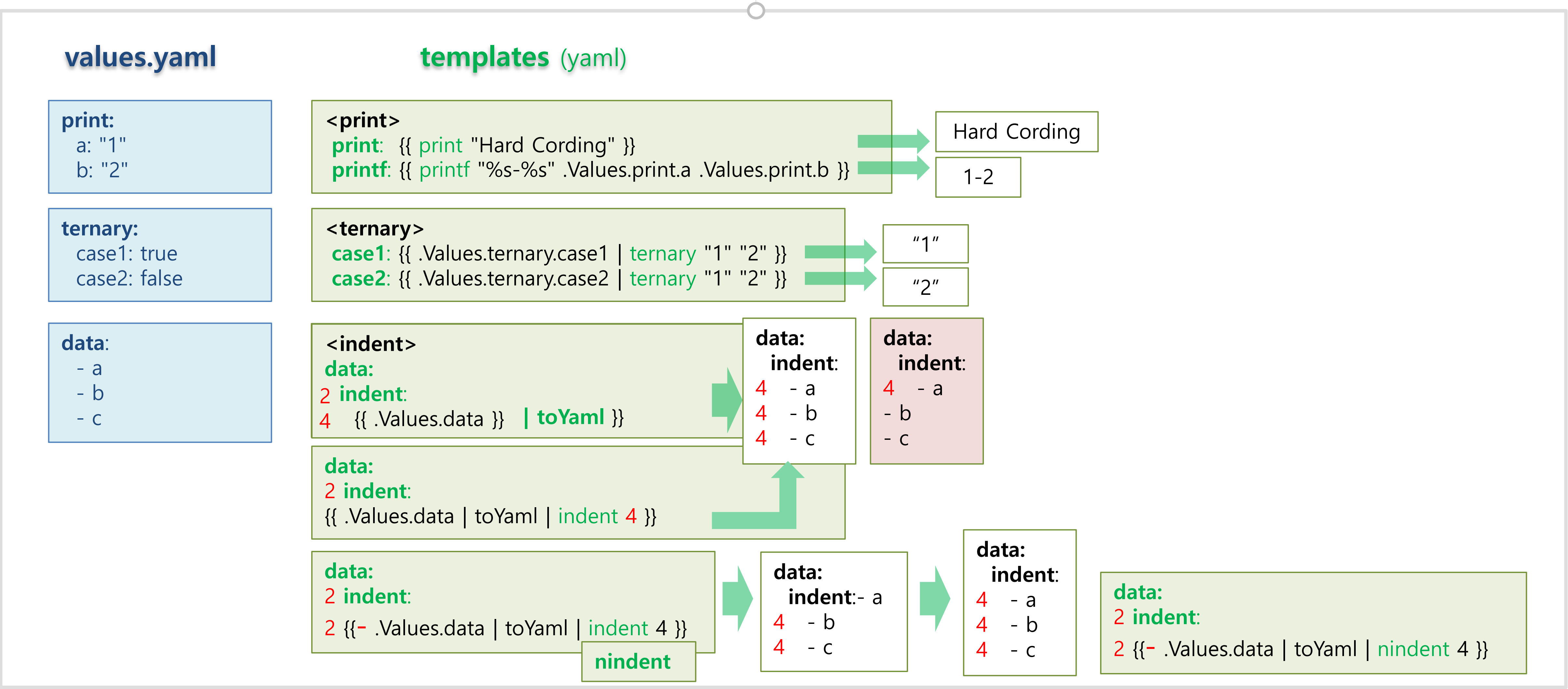
```
range:

- a
- b
- c
```

```
range:
- a
- b
- c
```

- 내 차트 만들기2 - 다양한 함수들1

various functions



values.yaml

templates (yaml)

```
print:
  a: "1"
  b: "2"
```

```
<print>
print:  {{ print "Hard Cording" }}
printf: {{ printf "%s-%s" .Values.print.a .Values.print.b }}
```

```
Hard Cording
```

```
1-2
```

```
ternary:
  case1: true
  case2: false
```

```
<ternary>
case1: {{ .Values.ternary.case1 | ternary "1" "2" }}
case2: {{ .Values.ternary.case2 | ternary "1" "2" }}
```

```
"1"
```

```
"2"
```

```
data:
  - a
  - b
  - c
```

```
<indent>
data:
2 indent:
4   {{ .Values.data }}   | toYaml }}
```

```
data:
  indent:
4   - a
4   - b
4   - c
```

```
data:
  indent:
4   - a
- b
- c
```

```
data:
2 indent:
{{ .Values.data | toYaml | indent 4 }}
```

```
data:
2 indent:
2 {{- .Values.data | toYaml | indent 4 }}
```

nindent

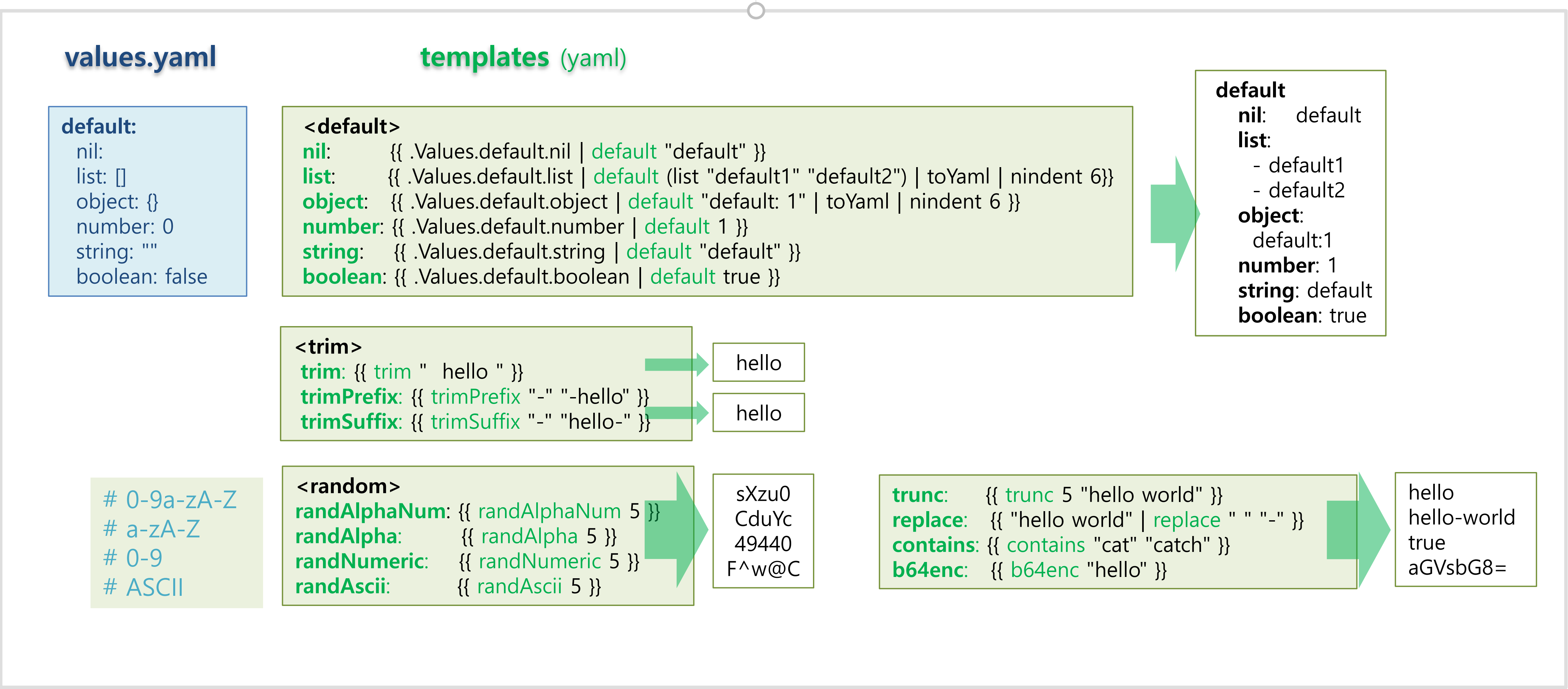
```
data:
  indent:- a
4   - b
4   - c
```

```
data:
  indent:
4   - a
4   - b
4   - c
```

```
data:
2 indent:
2 {{- .Values.data | toYaml | nindent 4 }}
```

- 내 차트 만들기2 - 다양한 함수들2

various functions



values.yaml

```
default:
  nil:
  list: []
  object: {}
  number: 0
  string: ""
  boolean: false
```

templates (yaml)

```
<default>
nil:      {{ .Values.default.nil | default "default" }}
list:     {{ .Values.default.list | default (list "default1" "default2") | toYaml | nindent 6}}
object:   {{ .Values.default.object | default "default: 1" | toYaml | nindent 6 }}
number: {{ .Values.default.number | default 1 }}
string:    {{ .Values.default.string | default "default" }}
boolean: {{ .Values.default.boolean | default true }}
```

```
default
  nil:     default
  list:
    - default1
    - default2
  object:
    default:1
  number: 1
  string: default
  boolean: true
```

```
<trim>
trim: {{ trim "   hello " }}
trimPrefix: {{ trimPrefix "-" "-hello" }}
trimSuffix: {{ trimSuffix "-" "hello-" }}
```

```
hello
```

```
hello
```

```
# 0-9a-zA-Z
# a-zA-Z
# 0-9
# ASCII
```

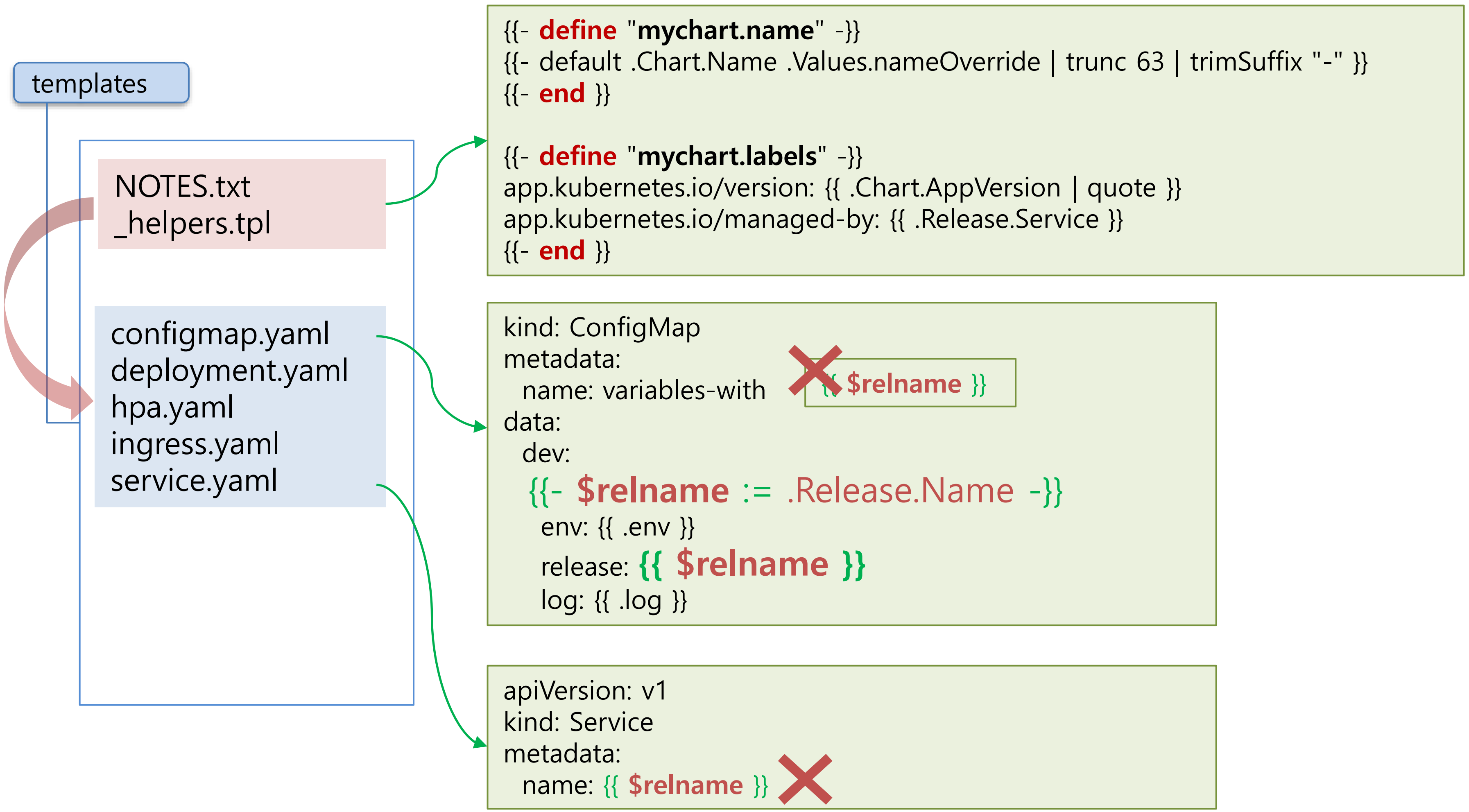
```
<random>
randAlphaNum: {{ randAlphaNum 5 }}
randAlpha:        {{ randAlpha 5 }}
randNumeric:    {{ randNumeric 5 }}
randAscii:         {{ randAscii 5 }}
```

```
sXzu0
CduYc
49440
F^w@C
```

```
trunc:      {{ trunc 5 "hello world" }}
replace:    {{ "hello world" | replace " " "-" }}
contains: {{ contains "cat" "catch" }}
b64enc:    {{ b64enc "hello" }}
```

```
hello
hello-world
true
aGVsbG8=
```

- 내 차트 만들기2 - 지역 변수 사용



templates

NOTES.txt
_helpers.tpl

configmap.yaml
deployment.yaml
hpa.yaml
ingress.yaml
service.yaml

```
{{- define "mychart.name" -}}
{{- default .Chart.Name .Values.nameOverride | trunc 63 | trimSuffix "-" }}
{{- end }}

{{- define "mychart.labels" -}}
app.kubernetes.io/version: {{ .Chart.AppVersion | quote }}
app.kubernetes.io/managed-by: {{ .Release.Service }}
{{- end }}
```

```
kind: ConfigMap
metadata:
  name: variables-with  {{ $relname }}
data:
  dev:
   {{- $relname := .Release.Name -}}
    env: {{ .env }}
    release: {{ $relname }}
    log: {{ .log }}
```

```
apiVersion: v1
kind: Service
metadata:
  name: {{ $relname }}
```

- 내 차트 만들기2 - 지역 변수 사용

with 내부에서 사용

values.yaml

```
dev:
  env: dev
  log: info

qa:
  env: qa
  log: info

prod:
  env: prod
  log:
```

templates (yaml)

```
apiVersion: v1
kind: ConfigMap
metadata:
  name: variables-with
data:
  dev:
  {{- with .Values.dev }}
    env: {{ .env }}
    release: {{ .Release.Name }}
    log: {{ .log }}
  {{- end }}
```

```
apiVersion: v1
kind: ConfigMap
metadata:
  name: variables-with
data:
  dev:
  {{- $relname := .Release.Name -}}
  {{- with .Values.dev }}
    env: {{ .env }}
    release: {{ $relname }}
    log: {{ .log }}
  {{- end }}
```

```
dev:
  env: dev
  release: mychart
  log: info
```

range와 함께 사용

values.yaml

```
list:
  - a
  - b
  - c
```

```
index:
  0: a
  1: b
  2: c
```

```
List<String> list = [a,b,c]
int index=0;

for (String value : list) {
  printf(index : value);
  index++;
};
```

templates (yaml)

```
apiVersion: v1
kind: ConfigMap
metadata:
  name: variables-range
data:
  index:
  {{- range $index, $value := .Values.list }}
    {{ $index }}: {{ $value }}
  {{- end }}
```

```
map:
  env: dev
  log: info
```

```
key-value:
  env: "dev"
  log: "info"
```

```
Map<String, String> map =
               [(env,dev), (log,info)]

for (String key : map.keySet()) {
  printf(key: "map.values()");
};
```

```
key-value:
{{- range $key, $value := .Values.map }}
  {{ $key }}: {{ $value | quote }}
{{- end }}
```